

THÈSE DE L'UNIVERSITÉ DE LYON

délivrée par

L'Université Claude Bernard Lyon 1

présentée et soutenue publiquement le 15 Décembre 2014

pour l'obtention du

DIPLÔME DE DOCTORAT

(arrêté du 7 août 2006)

spécialité informatique

par

Simon GAY

Mécanismes d'apprentissage développemental et intrinsèquement motivés en intelligence artificielle : étude des mécanismes d'intégration de l'espace environnemental.

Composition du jury

<i>Rapporteurs :</i>	Alain Dutech	HDR CR INRIA, LORIA, Nancy
	Mehdi Khamassi	HDR CR CNRS, ISIR, Paris
<i>Examineurs :</i>	Stéphane Doncieux	PU ISIR, UPMC Paris
	Pierre de Loor	PU LABSTIC, ENIB Brest
<i>Directeurs :</i>	Alain Mille	PU LIRIS, Lyon 1
	Olivier Georgeon	PhD LIRIS, Lyon 1

Mis en page avec la classe thloria.

Remerciements

Je tiens à remercier toutes les personnes sans qui cette thèse n'aurait pas pu être menée à son terme.

En premier lieu, je voudrais remercier mon directeur de thèse, Alain Mille, et mon co-encadrant, Olivier Georgeon, qui ont cru en moi, qui m'ont guidé, aidé et encouragé tout au long de cette thèse.

Je remercie également, Jong Wook Kim, James Marshall et Christian Wolf, qui m'ont apporté leur aide et leur savoir-faire dans la réalisation de travaux scientifiques. Mention spéciale également à Amélie Cordier, pour son aide précieuse et ses conseils avisés, en particulier dans la rédaction de ce manuscrit.

Je remercie nos stagiaires, Laurent Hu, Joseph Garnier et Martin Kodys, pour leur aide dans le développement de nos implémentations, mais aussi pour l'aide morale et la bonne humeur qu'ils ont apportées à l'équipe.

Un grand merci aux membres du laboratoire LIRIS, trop nombreux pour être cités, qui m'ont apporté aide et soutien tout au long de ma thèse.

Je remercie également les membres de mon jury, Alain Dutech, Mehdi Khamassi, Stéphane Doncieux et Pierre de Loor pour avoir accepté de participer à ma soutenance, et pour le temps qu'ils ont consacré à l'analyse et l'évaluation de mon travail de thèse.

Je remercie infiniment mes parents, ma sœur et son compagnon, mes grand-parents, tantes, oncles, cousins et cousines, ainsi que mes amis, pour leur soutien inconditionnel tout au long de ma thèse.

Table des matières

Table des figures	xi
-------------------	----

Introduction	1
--------------	---

I Problématique et état de l'art

Chapitre 1 Contexte	5
1.1 Cadre du travail de thèse	5
1.1.1 L'intelligence développementale	5
1.1.2 Une approche interactionnelle pour l'intelligence développementale : le projet IDEAL	6
1.1.3 Représentation de l'espace chez les êtres vivants	15
1.1.4 Cadre du travail de thèse	17
1.2 Question de recherche de la thèse	19
1.3 Sous-questions de recherche	20
Chapitre 2 État de l'art	21
2.1 Intelligence artificielle développementale	21
2.2 Apprentissage et exploitation des connaissances acquises par l'expérience . .	22
2.2.1 Apprentissage de connaissances basées sur l'expérience	22
2.2.2 Apprentissage de connaissances dans le but de générer des comporte- ments	23
2.3 Apprentissage et exploitation de modèles pour représenter l'espace	24

2.3.1	Espace péri-personnel	24
2.3.2	Espace extra-personnel	25
2.3.3	Permanence de l'objet	26
2.3.4	Modèles cognitifs de l'espace	26
2.4	État de l'art de la navigation basée sur des connaissances acquises par l'ex- périence	27
2.5	Conclusion	28

II Contributions

Chapitre 3	Présentation générale de la méthode d'élaboration des contribu- tions	33
3.1	Introduction aux contributions	33
3.2	Méthode de validation des mécanismes	34
3.2.1	Analyse des résultats et évaluation des mécanismes	34
3.2.2	Environnements de test	35
Chapitre 4	Modèle du Radical Interactionism (RI)	37
4.1	Formalisation du modèle d'interaction conforme à l'approche Radical Inter- actionism	37
4.2	Évolution de la formalisation du Radical Interactionism	39
4.2.1	Agnostic Agent Decision Process (AADP)	39
4.2.2	Enactive Markov Decision Process	40
4.3	Variations du modèle RI	42
4.3.1	Les variantes spatiales : Directional RI, Spatial RI (DRI, SRI)	42
4.3.2	Formalisation du SRI	44
4.3.3	Les variantes parallèles	44

Chapitre 5 Construction de structures pour la prise de connaissance de l'environnement	49
5.1 Construction d'une connaissance de l'environnement	51
5.2 Inspiration biologique et principes du mécanisme	51
5.2.1 Représentation des objets dans l'aire F5 du singe	51
5.2.2 Théorie des <i>Affordances</i>	53
5.2.3 Théorie des <i>Bundles</i>	54
5.3 Formalisation du mécanisme	54
5.3.1 Principe du mécanisme	54
5.3.2 Formalisation	56
5.3.3 Exemple pour illustrer le principe des signatures	60
5.3.4 Propriétés de relation entre les objets	63
5.4 Conclusion	64
Chapitre 6 Construction d'une structure caractérisant l'environnement	65
6.1 Inspiration biologique et principes du modèle	66
6.1.1 Gestion de l'espace observable	66
6.1.2 Reconnaissance et mise en mémoire des objets	66
6.1.3 Gestion de l'espace global	67
6.2 Formalisation du mécanisme de construction d'une structure caractérisant l'environnement	68
6.2.1 Gestion de l'espace observable	68
6.2.2 Effet mémoire	69
6.2.3 Reconnaissance des objets distants	70
6.2.4 Gestion de l'espace non-observable	71
6.2.5 Lieu d'interaction	71
6.2.6 Lieux composites	73
6.2.7 Signatures de lieu	74
6.2.8 Mise en mémoire des instances d'objets	75
6.2.9 Suivi des instances d'objets	77
6.2.10 Gestion de l'incertitude des instances d'objet	80
6.2.11 Identification des instances d'objet intégrées en mémoire	81
6.3 Conclusion	82
Chapitre 7 Exploitation d'une structure caractérisant l'environnement	83
7.1 Inspiration biologique	84
7.1.1 Espace péri-personnel	85

7.1.2	Espace lointain	88
7.2	Formalisation du mécanisme d'exploitation de la Mémoire Spatiale	88
7.2.1	Espace péri-personnel	88
7.2.2	Espace extra-personnel	102
7.3	Conclusion	110
Chapitre 8 Choix d'implémentation pour la validation		111
8.1	Indépendance entre l'agent et l'environnement	111
8.2	Mécanismes de construction de structures pour la prise de connaissance de l'environnement	111
8.3	Mécanisme de construction de la Mémoire Spatiale	112
8.4	Mécanismes d'exploitation de la Mémoire Spatiale	113
8.4.1	Espace péri-personnel	113
8.4.2	Espace extra-personnel	113

III Expérimentations

Chapitre 9 Expérimentation d'un dispositif de mémoire péri-personnelle		119
9.1	Description de l'environnement	119
9.1.1	Configuration de l'expérimentation	119
9.1.2	Simplifications de la prise en compte du contexte interactionnel	121
9.1.3	Détails de l'implémentation	122
9.1.4	Implémentation des signatures d'interaction	124
9.1.5	Implémentation du mécanisme de sélection	128
9.2	Expérimentation	133
9.2.1	Affichage des résultats	133
9.2.2	Étude du comportement sans mécanisme d'apprentissage	136
9.2.3	Étude du comportement avec un mécanisme d'apprentissage	146
9.2.4	Étude d'un modèle utilisant un système vestibulaire	161
9.3	Conclusion	165

**Chapitre 10 Expérimentation d'un dispositif de mémoire extra-personnelle :
étude des mécanismes d'exploitation de la mémoire** **167**

10.1	Description du système de test	167
10.2	Environnement simulé	169
10.3	Plate-forme robotique	171
10.3.1	Adaptation d'un système RI sur un robot	171
10.3.2	Conception d'un robot adapté à un système RI	172
10.4	Formation des signatures d'actes	175
10.4.1	Mécanisme d'apprentissage	176
10.4.2	Affichage des résultats	176
10.4.3	Résultats expérimentaux	177
10.4.4	Détection des instances d'objet	179
10.4.5	Effet prédictif des signatures d'interaction	183
10.4.6	Émergence de liens entre les actes	184
10.4.7	Utilisation des signatures pour compléter le contexte interactionnel	185
10.5	Test d'un mécanisme d'exploitation de la Mémoire Spatiale	188
10.5.1	Simplifications du système de test	188
10.5.2	Analyse des signatures des actes	191
10.5.3	"Interprétation" des éléments disposés dans l'environnement	192
10.5.4	Influence des coefficients d'influence des objets et de la mémoire spatiale	194
10.5.5	Attractivité des objets en fonction de leur contexte	195
10.5.6	Définition d'un objet à partir de son contexte et adaptation de la définition des objets à un changement environnemental	196
10.6	Conclusion	200

**Chapitre 11 Expérimentation d'un dispositif de mémoire extra-personnelle :
étude des mécanismes de construction d'une mémoire agnostique** **201**

11.1	Description du dispositif de test	201
11.2	Apprentissage des signatures	203
11.3	Expérimentations	207
11.3.1	Test de la <i>persistance de l'objet</i>	208
11.3.2	Influence des coefficients d'influence de la mémoire spatiale	210
11.3.3	Prise en compte de plusieurs objets non visibles	211
11.4	Conclusion	214

IV Synthèse, Discussion, Conclusion

Chapitre 12 Synthèse des résultats	219
Chapitre 13 Discussion	223
13.1 Critiques globales	223
13.2 Questions ouvertes	224
13.2.1 Éléments de réponse pour les questions posées par la recherche menée	224
13.2.2 Éléments de réponse aux problématiques abordées	226
13.3 Perspectives	227
13.3.1 Études complémentaires	227
13.3.2 Amélioration du mécanisme spatial	228
13.3.3 Amélioration du modèle RI	228
Chapitre 14 Conclusion	231
14.1 Table des symboles	234
Bibliographie	237

V ANNEXES

Annexe A : résultats détaillés des expérimentations	249
1 Expérimentations du chapitre 9	249
2 Expérimentations du chapitre 10	270
2.1 Système interactionnel complet	270
2.2 Système interactionnel simplifié	272
2.3 Signatures obtenues dans l'expérience des proies invisibles	273
3 Expérimentations du chapitre 11	274
3.1 Expérimentation avec une proie seule	274
3.2 Expérimentation avec une proie seule (à droite)	276
3.3 Expérimentation avec une proie et un bloc mur	279

Annexe B : Simplification algorithmique de la proximité globale	285
Annexe C : Algorithmes	287
1 algorithme général du modèle PSRI	287
2 Mise à jour de la mémoire spatio-séquentielle	287
3 reconstruction du path d'une interaction composite dans le cas d'un système implémentant l'hypothèse vestibulaire	288

Table des figures

1.1	Principe du mécanisme séquentiel IMOSHEM	9
1.2	Premières implémentations du système IMOSHEM	11
1.3	Implémentation sur un agent doté d'un système sensoriel à longue portée	12
1.4	Principe d'individuation	12
1.5	Architecture ECA (Enactive Cognitive Architecture)	14
1.6	Environnement utilisé pour tester l'architecture ECA	15
1.7	Modèle de l'espace de Previc	17
4.1	Formalisation du Radical Interactionism (RI)	38
4.2	Formalisation du Agnostic Agent Decision Process (AADP)	40
4.3	Seconde formalisation du AADP	41
4.4	Formalisation du Enactive Markov Decision Process (EMDP).	41
4.5	Directional RI et Spatial RI	43
4.6	Modèle du Parallel RI	45
5.1	Connexions entre les différents mécanismes	50
5.2	Aire cérébrale F5 dans le cerveau du singe	52
5.3	Objet qui afforde l'interaction <i>se cogner</i>	55
5.4	Objets pré et post-énaction des interactions primitives i_1 à i_6	61
5.5	Réversibilité des signatures	62
6.1	Principe de la gestion de l'espace non-observable	72
6.2	Illustration du principe de localisation des objets par les lieux composites	74
6.3	Détection multiple des instances d'objets	76
6.4	Signatures de présence	79
7.1	Affichage d'une interaction composite	89
7.2	Principe de mise à jour de la mémoire spatio-séquentielle	93
7.3	Illustration de l'effet mémoire lié aux interactions composites	94
7.4	Principe de la détection des instances d'objets	105
7.5	Proximité globale et variation de proximité	107

9.1	Configurations de l'expérimentation sur la mémoire peri-personnelle	120
9.2	Mémoire Spatiale	123
9.3	Un neurone formel	125
9.4	Implémentation des signatures d'interactions	127
9.5	Apprentissage des signatures	128
9.6	Affichage simplifié des signatures d'interaction	134
9.7	Affichage de la position des interactions composites dans l'espace	135
9.8	Affichage des <i>bundles</i> d'objets	136
9.9	Suivi du contenu de la mémoire spatiale	137
9.10	Séquences d'interactions utilisées par l'agent dans chaque contexte	138
9.11	Organigramme résumant le comportement de l'agent	139
9.12	Spécialisation des interactions composites	140
9.13	Suivi des objets par la mémoire spatio-séquentielle	141
9.14	Spécialisation des interactions (Simple Loop et Double Loop)	143
9.15	Suivi des objets par la mémoire spatio-séquentielle (voie sans issue)	144
9.16	Séquences d'interactions utilisées par l'agent (longueur 3)	145
9.17	Spécialisation des interactions (longueur 3)	146
9.18	Séquences d'interactions observées dans chaque contexte	148
9.19	Spécialisation des signatures	149
9.20	Suivi de la mémoire spatio-séquentielle	150
9.21	Séquences utilisées dans le contexte spatial <i>voie sans issue</i>	151
9.22	Contenu de la mémoire spatio-séquentielle (voie sans issue)	152
9.23	Séquences d'interactions obtenues dans les différents contextes spatiaux	154
9.24	Spécialisation des signatures	156
9.25	Suivi de la mémoire spatio-séquentielle dans le contexte spatial <i>virage à droite</i>	157
9.26	Séquences d'interaction utilisées dans le contexte spatial <i>voie sans issue</i>	159
9.27	Spécialisation des signatures	160
9.28	Suivi de la mémoire spatio-séquentielle	162
9.29	Spécialisation des interactions composites	164
10.1	Environnement simulé	170
10.2	Implémentation de nos mécanismes dans des robots	172
10.3	Le robot ErnestIRL	174
10.4	Traitement de l'image	175
10.5	Signatures des actes primaires	178
10.6	Exemples de signatures d'actes visuels	180
10.7	Déplacements implicitement définis par les signatures des actes visuels	181
10.8	Image de signatures par une séquence d'interactions	183
10.9	Prédiction du contexte interactionnel au pas suivant	184
10.10	Regroupement des actes désignés par une même signature	186

10.11	Maillage obtenu par regroupement des actes	187
10.12	Effet mémoire	188
10.13	Utilisation des signatures d'acte pour compléter le contexte interactionnel courant	189
10.14	Signatures des sept actes primaires	192
10.15	Proie masquée par une algue	193
10.16	Proie masquée par un mur	194
10.17	Proie masquée par un mur dans le cas du robot	195
10.18	Influence du coefficient γ sur le comportement de l'agent	196
10.19	Sélection d'une proie en fonction de son contexte	197
10.20	Signatures des actes <i>avancer d'un pas, se cogner</i> et <i>manger une proie</i>	197
10.21	Environnement de test pour l'expérience des proies invisibles	198
10.22	Approche des algues	199
10.23	signatures des actes <i>avancer, se cogner</i> et <i>manger</i>	199
11.1	Découpage des lieux primitifs	203
11.2	Signatures de lieu des lieux composites dotés d'un path de longueur 0 et 1	205
11.3	Affichage du contexte d'une instance d'objet	206
11.4	Exemple de mise à jour de la position d'un objet	207
11.5	Exemple de signatures de présence	208
11.6	Chemin utilisé par l'agent pour atteindre une proie	209
11.7	Position estimée de la proie par la mémoire spatiale	209
11.8	Trajectoire utilisée par l'agent pour contourner un bloc mur	210
11.9	Suivi de la proie par la mémoire spatiale	210
11.10	influence des coefficients	211
11.11	Chemin utilisé par l'agent pour atteindre une proie	212
11.12	Évolution de l'estimation de la position de la proie par la mémoire spatiale	213
11.13	Estimation de la position de la proie dans une configuration symétrique	214
11.14	Chemin utilisé par l'agent pour atteindre une proie en présence d'un mur	215
11.15	Évolution de l'estimation de la position de l'élément proie	216
11.16	Évolution de l'estimation de la position de l'élément mur	216

Introduction

La capacité à représenter l'espace qui nous entoure, et à mémoriser la position des éléments qui composent notre environnement dans l'espace est vitale pour les êtres vivants. Elle permet de localiser la source d'un danger, même si elle ne peut plus être perçue, et de s'orienter dans son environnement. Dans cette thèse, nous cherchons à implémenter cette capacité dans des agents artificiels dans le but de leur permettre d'interagir avec un environnement spatial.

L'implémentation d'une structure reflétant l'espace a soulevé un certain nombre de questions : comment définir les éléments qui composent l'environnement ? Comment localiser et suivre un objet dans l'espace ? Comment définir la structure de l'espace ? Comment générer des comportements adaptés à un contexte spatial ? Pour répondre à ces questions, nous nous sommes basés sur une approche dans laquelle l'agent interagit avec son environnement dans le but de l'expérimenter.

Dans la partie I de ce manuscrit, nous présenterons le cadre de notre travail, ainsi que les travaux issus des domaines de la psychologie et de la biologie qui ont orienté nos recherches. Cette partie posera également les limites de notre cadre d'étude. Nous présenterons brièvement le projet IDEAL, dans lequel s'inscrit cette thèse, et les limites observées, qui nous ont orienté vers la problématique de l'espace. La question de recherche de cette thèse, et les sous questions qui en découlent, sont décrites à la fin de ce premier chapitre. Un second chapitre présente un état de l'art des solutions existantes pour répondre aux différentes problématiques soulevées. Comme la plupart des travaux ne concernent qu'un nombre limité de problématiques, l'état de l'art est divisé en thèmes, chaque thème donnant l'état de l'art des solutions répondant à une problématique particulière.

Dans la partie II, nous introduisons les modèles théoriques que nous avons développés pour répondre aux différentes problématiques. Un premier chapitre présente le modèle théorique servant de base à notre système, ainsi que les variantes que nous avons utilisées au cours de nos expérimentations. Les chapitres suivants présentent les mécanismes que nous avons développés. Ces mécanismes sont interdépendants, mais répondent chacun à une problématique et des contraintes distinctes, ce qui rend possible leur étude séparée. Le premier mécanisme répond au problème de la définition des objets et leur reconnaissance dans l'environnement. Le second permet de définir l'espace environnant et de suivre et localiser les objets détectés. Enfin, le dernier mécanisme permet la génération de comportements adaptés à un contexte spatial formé par les mécanismes précédents.

La partie III présente les expérimentations que nous avons menées pour tester les différents mécanismes. Chaque expérimentation a été conçue pour mettre en évidence le fonctionnement de certains mécanismes sur des systèmes complets. Dans un premier chapitre de cette partie, nous analyserons l'émergence de la notion d'objet, et la génération de comportements issue d'un mécanisme reflétant la structure de l'espace proche, dans un environnement simple et discret. Dans un second chapitre, nous analyserons l'émergence de la notion d'objet dans un environnement plus complexe et continu, et le mécanisme d'exploitation, en se basant sur des structures partiellement codées en dur. Le dernier chapitre est consacré à l'analyse du mécanisme permettant de construire une structure caractérisant l'espace.

La dernière partie donne des pistes sur des évolutions possibles de nos mécanismes, dans le but de les implémenter sur des agents toujours plus complexes. Ces évolutions concernent plusieurs niveaux de nos systèmes.

Première partie

Problématique et état de l'art

Chapitre 1

Contexte

1.1 Cadre du travail de thèse

1.1.1 L'intelligence développementale

L'intelligence développementale est une approche relativement récente de l'intelligence artificielle, à l'intersection des domaines de la robotique, des neurosciences, de la psychologie, et des sciences cognitives. Les nombreux travaux dans ce domaine témoignent de l'engouement pour cette approche (synthèses de Lungarella et al. [77] et de Vernon et al. [130]). Dans ces travaux, un agent est doté de mécanismes d'apprentissage lui permettant de découvrir ses propres capacités sensorimotrices de façon autonome, en interagissant avec son environnement, en suivant une approche dite "par le bas" (ou *bottom-top*).

Le but de l'intelligence développementale est double [76] : d'une part, il consiste à mettre au point des systèmes capables d'apprendre et de s'adapter à leur environnement, afin de les rendre plus autonomes. D'autre part, il permet de tester et valider les modèles cognitifs issues de la biologie et de la psychologie : c'est ce que l'on appelle la *Méthodologie Synthétique* [104][101].

L'un des principes les plus importants de l'intelligence développementale est la notion d'*embodiment* (ou *incarnation*) [102] [16] [3] [63], c'est-à-dire qu'un agent existe et développe des processus cognitifs au travers de l'interaction entre un corps physique et un environnement. En effet, contrairement aux approches classiques, où un robot n'est considéré que comme un système qui exécute les commandes transmises par un système à base de règles, l'intelligence développementale propose que l'intelligence émerge du couplage entre l'agent et l'environnement. Cet apprentissage par l'expérience permet d'éviter ce que Blank et al. appellent le *biais anthropomorphique inhérent* [13]. Ce biais fait référence aux approches traditionnelles, qui cherchent à effectuer des tâches, et implémenter des mécanismes d'apprentissage, d'un point de vue humain. L'intelligence développementale permet d'éviter ce problème en donnant à un agent la possibilité de développer ses capacités d'actions sur l'environnement et ses propres buts par lui-même.

Le domaine de l'intelligence développementale soulève cependant de nombreuses questions sur la façon dont un agent peut d'une part générer des connaissances de son environnement basées

sur son expérience, et d'autre part exploiter ces connaissances pour générer des comportements. Le projet IDEAL, dans lequel s'inscrit cette thèse, propose des débuts de réponses à ces questions en proposant un mécanisme d'apprentissage basé uniquement sur l'interaction entre l'agent et l'environnement, et un principe de motivation indépendant de toute référence directe extérieure à l'agent, également basée sur l'interaction.

1.1.2 Une approche interactionnelle pour l'intelligence développementale : le projet IDEAL

Bases du projet IDEAL

Le projet IDEAL (Implementing DEvelopmentAl Learning) consiste à implémenter un mécanisme d'apprentissage développemental précoce dans un agent artificiel, dans le but d'étudier les hypothèses émergentistes et constructivistes de la cognition. L'approche IDEAL se différencie de l'hypothèse des symboles physiques de Newell et Simon [89]¹ En effet, si des mécanismes computationnels sont construits et utilisent des symboles pour générer des comportements intelligents, il ne leur est attribué aucune signification *a priori*, et il n'y a pas de règles qui interpréterait ces symboles. Le but est alors d'étudier comment l'agent attribue une sémantique à ces symboles en se basant sur son activité et sa propre expérience de son environnement. Comme l'étude de ces mécanismes repose sur l'organisation des comportements et l'analyse de connaissances émergentes, ils se rapprochent des théories de la cognition située [65][120], incarnée [129] et *enactive* [91][118].

L'approche IDEAL repose sur un modèle "expérience/résultat" plutôt que "observation/action". Les couples "expérience/résultat" sont appelés *interactions*. Considérer ainsi les interactions n'est pas seulement soutenu par des théories issues de la psychologie (Piaget [105], O'Regan [96] [97]), mais s'appuie également sur des résultats expérimentaux : une expérience de substitution sensorielle menée par Bach-y-Rita [8] a montré l'unicité cognitive de l'action et de la perception. Les volontaires portaient un dispositif qui restituait l'image captée par une caméra sous forme d'informations tactiles restituées par une matrice de vibreurs. Ces personnes ne parvenaient pas à appréhender les informations qu'elles percevaient, jusqu'à ce qu'elles soient autorisées à déplacer elles-même la caméra, montrant ainsi que la perception est étroitement liée à l'interaction entre un individu et son environnement. La notion d'interaction, telle que définie dans IDEAL, implémente la notion d'*action intentionnelle* de Engel et al. [34] : l'agent effectue une interaction dans le but de recevoir la perception qui en résulte. L'agent construit ainsi une *perception active* de son environnement [10].

Le concept d'*agnosticisme environnemental* [51] est proposé par Olivier Georgeon pour désigner une approche permettant de concevoir des systèmes autonomes dont les mécanismes d'apprentissage ne se basent pas sur des connaissances *a priori* de leur environnement. Un mécanisme d'apprentissage respectant parfaitement le concept de l'agnosticisme environnemental doit ainsi

1. Un système à symboles physiques possède les moyens nécessaires et suffisants pour une action intelligente générale.

pouvoir être implémenté sur n'importe quel système autonome, puisque indépendant de l'environnement, du système sensoriel et des actionneurs de celui-ci.

Le projet IDEAL a produit un modèle se rapprochant du principe d'agnosticisme environnemental basé sur la *motivation intrinsèque* [99] [13] [113] [30]. Dans ce cadre, un agent doté d'un système basé sur la motivation intrinsèque ne cherche pas à atteindre un but ou à effectuer une tâche dans le but d'obtenir une récompense attribuée par un mécanisme de supervision, comme c'est le cas en apprentissage par renforcement (e.g. [122]), mais interagira avec son environnement d'après un ensemble de principes innés qui lui sont propres. Ce modèle propose une forme de motivation intrinsèque, appelée *motivation Interactionnelle*. Cette motivation est basée sur des propriétés propres aux interactions, appelées *valeur de satisfaction*, qui caractérisent les interactions que l'agent "aime" effectuer et celle qu'il cherche à éviter, de façon innée. Associé à un mécanisme d'apprentissage, la motivation interactionnelle permet à un agent de démontrer à un observateur, à travers ses interactions avec l'environnement, qu'il donne un sens aux éléments que l'observateur distingue dans cet environnement. Un mécanisme d'apprentissage basé sur la motivation interactionnelle permet ainsi d'étudier comment une prise en compte de l'environnement émerge des interactions entre l'agent et son environnement.

Positionnement du projet IDEAL dans le domaine de l'Intelligence Artificielle

Le projet IDEAL se positionne dans le cadre de l'intelligence artificielle non symbolique. En effet, même si ses mécanismes utilisent des symboles, ceux-ci ne servent pas à caractériser des états du monde, mais des interactions entre l'agent et son environnement, sans aucune signification *a priori*, et auxquelles il ne peut être appliqué de règles liées à leur signification (règles sémantiques). Plus spécifiquement, nous pouvons situer cette approche dans le cadre des approches basées sur l'apprentissage par l'expérience, c'est-à-dire par le biais de l'interaction entre l'agent et son environnement. Cette forme d'apprentissage implique d'une part que le système de décision soit "incarné" dans un corps pouvant interagir avec l'environnement, et d'autre part, que la perception de l'agent est une construction interne issue d'interactions initiées volontairement par l'agent. Ces interactions sont initiées dans le but d'obtenir des informations sur l'environnement ou d'agir sur celui-ci. L'approche IDEAL se différencie ainsi des approches traditionnelles de l'apprentissage (apprentissage par renforcement, réseaux neuronaux...) par une forme d'inversion du cycle de décision, qui débute par une expérimentation de l'agent sur son environnement, plutôt que par une analyse de la perception issue de l'environnement. Cette inversion est soutenue par des théories issues de la psychologie et de la cognition [33][103], mais nécessite des expérimentations pour être validée. L'une des principales tâches de ce projet est donc de fournir des implémentations fonctionnelles des théories de la cognition situées [24] afin de les valider. Notre approche a pour but de permettre à un agent de générer des comportements lui permettant de satisfaire une forme de motivation qui lui est propre (ou *intrinsic motivation* [99]), et non d'effectuer une tâche ou atteindre un but prédéfini par un observateur extérieur. En effet, notre approche prévoit que l'agent ne puisse qu'expérimenter son environnement au travers de

ses interactions, sans jamais avoir accès à la notion d'état de l'environnement. La validation des modèles est basée sur l'étude du comportement des agents, puisqu'il n'y a pas de performances attendues ni de tâche prédéfinie *a priori*.

Un mécanisme de décision séquentiel proposé dans IDEAL : IMOSHEM

Le mécanisme d'apprentissage IMOSHEM (Intrinsically MOTivated SHEma Mechanism) [50] proposé par Olivier Georgeon repose sur la détection des régularités dans les interactions qu'un agent a avec son environnement. L'utilisation de la notion d'*interactions*, plutôt que d'actions et de perceptions, se base sur une hypothèse de Piaget, qui indique que la distinction entre le monde extérieur et le soi n'est pas inné, mais appris au travers de l'interaction : «*L'intelligence (et donc l'action de connaître) ne débute ni par la connaissance du moi, ni par celle des choses comme telles, mais par celle de leur interaction ; c'est en s'orientant simultanément vers les deux pôles de cette interaction qu'elle organise le monde en s'organisant elle-même*» (Piaget [105]). Ainsi, plutôt que de se baser sur un apprentissage de schémas de la forme [perception1,action,perception2] [32] appris au travers des actions successives de l'agent, on détecte des régularités dans l'enchaînement des interactions. L'algorithme est fondé sur un échange d'information entre l'agent et son environnement : à chaque cycle, le mécanisme choisit une interaction. L'agent effectue alors l'action associée, et l'environnement lui retourne la perception qui en résulte, validant ou invalidant l'attente de l'agent. L'interaction expérimentée est ensuite inscrite sur une *timeline*. L'observation d'un enchaînement répété de deux interactions va permettre l'apprentissage d'une séquence d'interactions appelé *scheme*, de la forme [interaction1, interaction2], où *interaction1* et *interaction2* peuvent être des interactions primitives (i.e. un couple {action ; perception} prédéfini) ou des *schemes* appris précédemment. Un *scheme* interactif ainsi formé peut alors être choisi au même titre qu'une interaction primitive, dont la valeur de satisfaction associée correspondant à la somme des valeurs des interactions primitives qui le composent. L'apprentissage de séquences d'interactions s'effectue ainsi de façon hiérarchique [121], les séquences devenant de plus en plus longues avec le temps.

Cet apprentissage hiérarchique permet à l'agent de construire des *schemes* interactifs qu'il peut énoncer de la même façon qu'une interaction primitive. Cette capacité à faire émerger de nouvelles interactions est appelée *auto-programmation* [45][125]. Ce terme d'auto-programmation caractérise le fait que l'agent devient capable de construire des comportements et de les exécuter automatiquement, comme des programmes. Le principe de l'auto-programmation implique que l'agent n'a pas de but prédéfini *a priori*. Il va, en revanche, définir par lui-même des comportements lui permettant de satisfaire sa motivation intrinsèque.

Le choix d'une interaction intention est basé sur sa valeur de satisfaction, pondérée par la certitude que cette interaction puisse être énoncée avec succès, en se basant sur les *schemes* interactifs. Le premier élément d'un *scheme* est appelé *contexte*, et le second, la *proposition*. Un *scheme* est également caractérisé par le nombre de succès et d'échecs, permettant de caractériser la certitude que sa proposition puisse être énoncée avec succès après énonciation de son contexte.

Lorsque le contexte d'un schème est énoncé, le schème propose sa proposition pour l'énoncé du cycle de décision suivant. À chaque cycle de décision, l'agent sélectionne parmi les interactions et schèmes interactifs proposés par les schèmes dont le contexte a été énoncé, celui dont la valeur de satisfaction pondérée par la certitude de succès est la plus élevée. L'agent énonce alors la séquence d'interaction définie par le schème jusqu'à ce qu'elle soit entièrement énoncée, ou jusqu'à ce que l'une des interactions de la séquence échoue. La Figure 1.1 résume le principe du mécanisme d'apprentissage et de sélection.

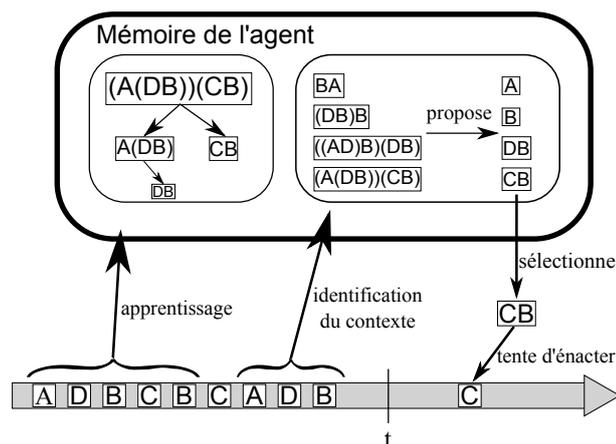


FIGURE 1.1 – Principe du mécanisme séquentiel IMOSHEM. Mécanisme d'apprentissage : sur la timeline, l'agent a pu observer à plusieurs reprises l'enchaînement des schèmes $A(DB)^3$ et CB . Il construit ainsi le schème $(A(DB))(CB)$. Mécanisme de sélection : En observant la *timeline*, un ensemble de schèmes voit leur contexte énoncé. Ceux-ci proposent leur proposition. L'agent sélectionne le schème dont la valeur de satisfaction, pondérée par la certitude de succès, est la plus élevée (ici, le schème CB). L'agent va alors énoncer consécutivement les interactions C , puis B si C est énoncée avec succès.

Ce mécanisme d'apprentissage permet à l'agent de naviguer dans son environnement en maximisant sa motivation intrinsèque. Il se différencie des approches classiques, comme le connexionnisme et le connectivisme, par le fait qu'il n'y a pas de but à atteindre ni de récompenses, et par le fait que l'apprentissage se fait en continu. On pourrait cependant remarquer que des interactions à forte valeur de satisfaction pourraient être considérées comme des buts à atteindre, et les séquences permettant de les atteindre comme une forme de rétro-propagation de récompense. Cependant, la multiplicité de ces "buts", puisqu'une interaction n'est pas liée à un lieu ou un élément unique de l'environnement, et le fait que la valeur de satisfaction soit interne à l'agent et non externe, comme dans le cas d'une récompense, différencient ce mécanisme de ces approches. Le fait que la taille des séquences d'interaction ne soit pas bornée différencie également ce mécanisme des approches par séquences markoviennes. L'un des principaux problèmes des approches markoviennes est d'établir quel nombre d'états minimums permet de déterminer l'état suivant, ce qui ne se pose pas dans le cas d'IMOSHEM : la longueur des schèmes dépend uniquement des

3. Le schème $A(DB)$ est constitué de l'interaction A et du schème DB , constitué des interactions D et B .

régularités observées.

Évaluation d'un système fondé sur le modèle IDEAL

L'évaluation des mécanismes proposés dans IDEAL est cependant problématique : le fait que le mécanisme n'ait pas de but précis ni de tâche à effectuer rend difficile la qualification des performances. Un ensemble de critères caractérisant les systèmes développementaux est proposé pour évaluer un système fondé sur le mécanisme IMOSHEM. Un premier ensemble de critères permet l'évaluation d'après des variables internes au système. On définit donc :

- Le principe d'*hédonisme objectif*, soit la capacité de l'agent à satisfaire au mieux sa motivation intrinsèque. Cependant, les interactions à forte valeur de satisfaction ne sont pas toujours possibles, ou peuvent conduire à des situations où seules des interactions peu satisfaisantes peuvent être utilisées. Le système ne peut donc pas se contenter de choisir les interactions les plus attractives, ce qui nous amène au critère suivant :

- Le principe d'*hédonisme tempéré*, c'est-à-dire la capacité à aller à l'encontre de sa motivation intrinsèque si cela peut conduire à une situation permettant d'effectuer des interactions à forte valeur de satisfaction.

Un second ensemble de critères permet d'évaluer le mécanisme du point de vue d'un observateur externe. Les trois premiers critères sont reconnus dans la robotique développementale. Les deux suivants sont inspirés par l'approche centrée sur les interactions.

1- Principe de catégorisation de la situation : l'agent peut catégoriser les différents aspects de sa situation, et adapter son comportement en fonction de ces catégories.

2- Principe de situation non ambiguë : l'agent peut différencier des situations générant le même état sensoriel (*perceptual aliasing* [28]).

3- principe de réadaptation gracieuse : l'agent peut s'adapter à de nouvelles situations en se basant sur des connaissances anciennes. Ces dernières ne sont jamais effacées, ce qui permet d'éviter le phénomène d'oubli catastrophique [36].

4- Principe de perception active : l'agent va chercher des informations sur son environnement par le biais de ses interactions.

5- Principe d'individuation : deux systèmes identiques peuvent conduire à des comportements différents.

L'observation de ces critères reposent sur une analyse temporelle du comportement de l'agent. L'évaluation du mécanisme doit donc passer par l'analyse de traces d'activité. Ces traces sont récupérées par un dispositif de collecte qui récupère des informations sur les interactions énoncées par l'agent, ainsi que les schémas impliqués dans ses décisions, sans interférer avec celui-ci.

Ces traces sont ensuite affichées⁴ pour permettre l'analyse du comportement de l'agent, mais aussi du contexte, de la satisfaction des interactions effectuées et de la longueur des interactions sélectionnées [49].

Exemples d'implémentation

Le mécanisme IMOSHEM a dans un premier temps été testé sur un agent simple évoluant dans un environnement discret (figure 1.2). L'agent, représenté par un triangle, peut interagir avec son environnement avec une liste de 10 interactions, soit cinq actions pouvant chacune conduire à deux perceptions différentes. Chaque interaction se voit attribuer une valeur de satisfactions. Dans cet exemple, l'agent "aime" *avancer* (satisfaction de 10), mais "déteste" *se cogner* dans un mur (satisfaction de -10). Il peut en outre tourner à droite ou à gauche de 90°, et se retrouver face à une case vide (interaction "neutre" de satisfaction 0), ou face à un bloc (interaction "négative" dotée d'une satisfaction de -5). Il dispose également d'"antennes" avec lesquelles il peut toucher les cases adjacentes, qui peuvent être vides (interaction "neutre") ou occupées par un bloc mur (interaction "faiblement négative").

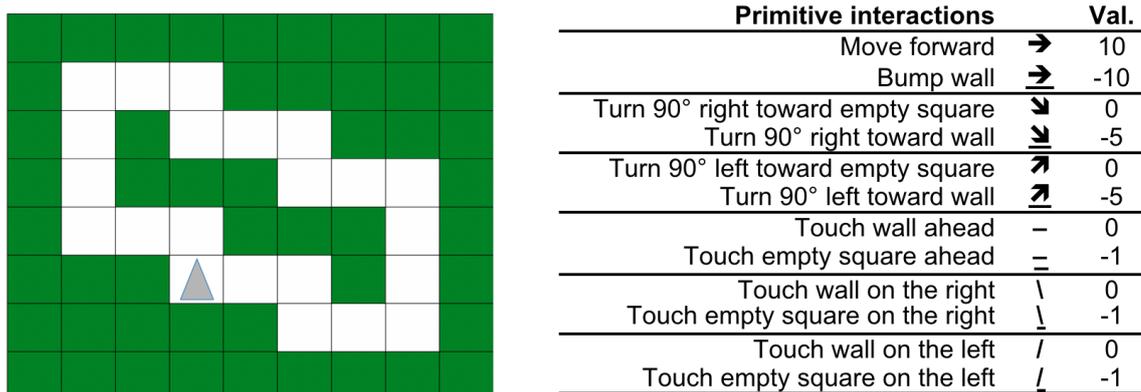


FIGURE 1.2 – Premières implémentations du système IMOSHEM. À gauche, l'agent (représenté par un triangle gris) dans son environnement. L'environnement dessine un circuit pour offrir des régularités séquentielles d'interaction. À droite, l'ensemble des interactions primitives que l'agent peut énoncer (avec leur valeur de satisfaction). La valeur de satisfaction caractérise la satisfaction qu'expérimente l'agent lorsqu'il énonce une interaction.

Certaines séquences qui émergent de ces interactions primitives montrent que l'agent a tendance à utiliser ses antennes pour "sonder" son environnement, même si ces interactions ont une valeur de satisfaction négative, avant d'avancer ou de tourner. Ce comportement illustre le principe d'hédonisme tempéré, et montre que l'agent attribue une fonction d'exploration aux interactions *toucher*.

Cependant, l'agent, placé dans un espace ouvert, a du mal à naviguer. Les séquences d'interaction se forment difficilement, faute de trouver des "repères" pour constituer des séquences

4. Les traces sont affichées à l'aide d'un outil appelé *Abstract-Lite*

d'interaction pertinentes. La première solution proposée consistait à doter l'agent d'un système sensoriel à longue portée, sensible à un élément particulier de l'environnement. L'agent a ainsi été doté de deux "yeux" sensibles au mouvement apparent de cet élément [44]. Ce système sensoriel peut fournir des informations sur l'apparition, la disparition de l'objet dans le champ visuel de chaque œil, ainsi que le rapprochement ou l'éloignement de l'objet par rapport à l'agent. Plutôt que d'attribuer une valeur de satisfaction à chaque combinaison possible d'interactions prenant en compte ces informations, on ajoute des valeurs additionnelles aux interactions déjà présentes. Ainsi, en calculant la valeur de satisfaction d'une interaction à partir d'un nombre limité de règles, on rend possible l'implémentation du mécanisme lorsque le nombre d'interaction (que ce soit en augmentant le nombre de capteurs ou d'actions) est important. La Figure 1.3 montre un exemple d'implémentation d'un système utilisant un système sensoriel à longue portée.

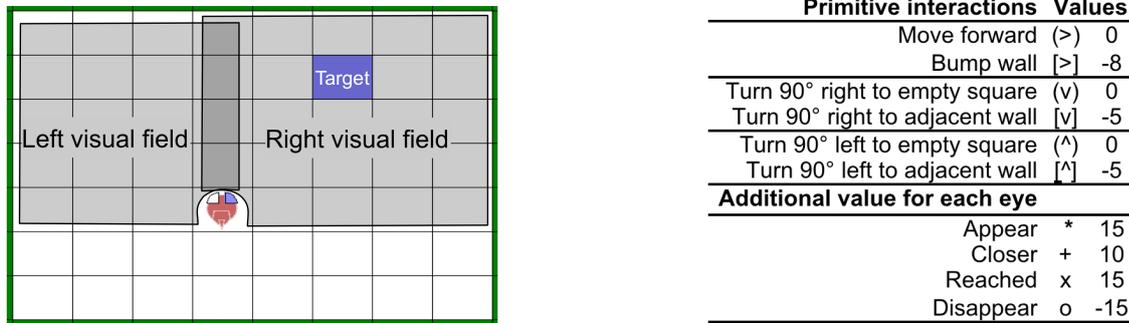


FIGURE 1.3 – Implémentation sur un agent doté d'un système sensoriel à longue portée. L'agent dispose de deux "yeux" simples, capable de détecter l'apparition, le rapprochement ou la disparition d'un objet saillant dans le champ visuel de chaque œil. Ce système sensoriel ajoute une valeur de satisfaction aux interactions qui contiennent ces informations additionnelles

On définit un ensemble de valeurs additionnelles pour les stimuli issus du système sensoriel, des valeurs positives pour l'apparition ou le rapprochement de l'objet, négatives pour la disparition et l'éloignement. L'agent montre alors une certaine capacité à naviguer dans son environnement. Poussé par sa motivation interactionnelle, il finit par apprendre des séquences lui permettant de se rapprocher et finalement d'atteindre l'objet. On notera l'émergence de deux comportements d'approches distincts, ce qui illustre le principe d'individuation (Figure 1.4).

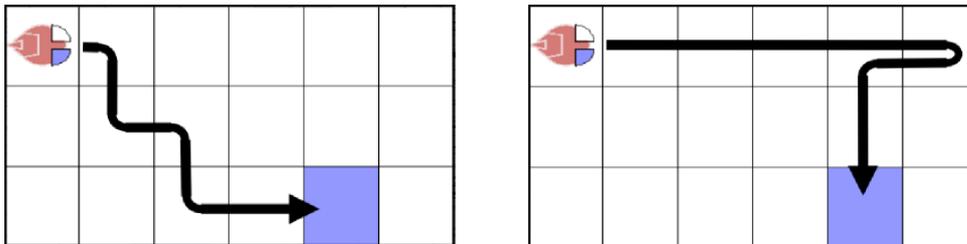


FIGURE 1.4 – Principe d'individuation : deux agents identiques peuvent générer des stratégies différentes, en fonction de leur expérience avec l'environnement.

Une mémoire locale de l'espace

Cependant, la capacité de l'agent à naviguer dans l'espace avec un mécanisme basé uniquement sur les régularités séquentielles reste limitée. Il ne peut pas, par exemple, apprendre les régularités spatiales, c'est-à-dire qu'il ne peut pas intégrer le fait que plusieurs séquences, comme tourner trois fois à gauche de 90° et tourner une fois à droite de 90° , peuvent mener à une situation identique. La notion de persistance de l'objet ne peut pas émerger. L'agent cesse de suivre les éléments de son environnement quand ils sont hors de portée de son système sensoriel [42].

L'émergence de la persistance de l'objet et l'organisation des comportements dans l'espace sont des questions fondamentales étudiées dans cette thèse. Afin d'apporter des réponses à ces questions, nous nous sommes inspiré des êtres vivants. En effet, la plupart des êtres vivants disposent d'une région du cerveau leur permettant de constituer un modèle de leur environnement [26]. Nous faisons référence au tectum (ou colliculus chez les mammifères)[82] qui intègre les stimuli visuels et auditifs, ou encore l'hippocampe [93], contenant notamment les cellules de lieu et d'orientation. Ces régions permettent aux êtres vivants de se localiser et de déterminer la position des éléments qui les entourent.

En se basant sur ces observations, nous avons développé des mécanismes destinés à compléter le mécanisme IMOSHEM. Ces mécanismes ont abouti à un modèle, que nous appelons *Mémoire Spatiale Locale*, constituée de deux éléments. Le premier, appelé *Mémoire Persistante*, permet de construire une représentation des objets sous forme d'associations d'interactions, appelés *bundles*. On appelle *bundle* une association d'interactions, formées lorsqu'il y a co-occurrence de plusieurs interactions en une même position de l'espace. Le second élément, la *mémoire spatiale* elle-même, permet de localiser les interactions et les *bundles* sur une représentation interne de l'environnement en référentiel égocentrique, et de les suivre même lorsqu'ils sont hors de portée du système sensoriel. L'association du mécanisme IMOSHEM et de la Mémoire Spatiale Locale forme une architecture appelée ECA [48] (Enactive Cognitive Architecture, Figure 1.5).

La mémoire spatiale locale n'a pas pour but de cartographier l'environnement, comme ce serait le cas avec des algorithmes de type SLAM (e.g. [9]), ni de planifier des trajectoires avec des algorithmes de *path planning* (e.g. [1]), mais de fournir un contexte "amélioré" à l'agent et lui permettre de générer des comportements adaptés à son environnement. La mémoire spatiale constitue une mémoire à court terme qui ne conserve les éléments que sur de courtes périodes.

Les *bundles* permettent de construire une représentation des "objets" au sens donné par David Hume [64], c'est-à-dire la somme de leurs propriétés, sans qu'aucune connaissance innée, ni aucune propriété "ontologique", intrinsèquement liée à l'objet, ne soit nécessaire. Un *bundle* est formé lorsqu'il y a co-occurrence de deux (ou plus) interactions en un même point de l'environnement.

Cette architecture a été implémentée sur une nouvelle version de l'agent (Figure 1.6). L'agent peut se déplacer librement dans l'environnement sans être contraint à une grille, et dispose d'un système sensoriel plus précis. La mémoire spatiale implémentée est entièrement codée en dur, dans le but de valider cette structure avant de développer les mécanismes permettant de la construire.

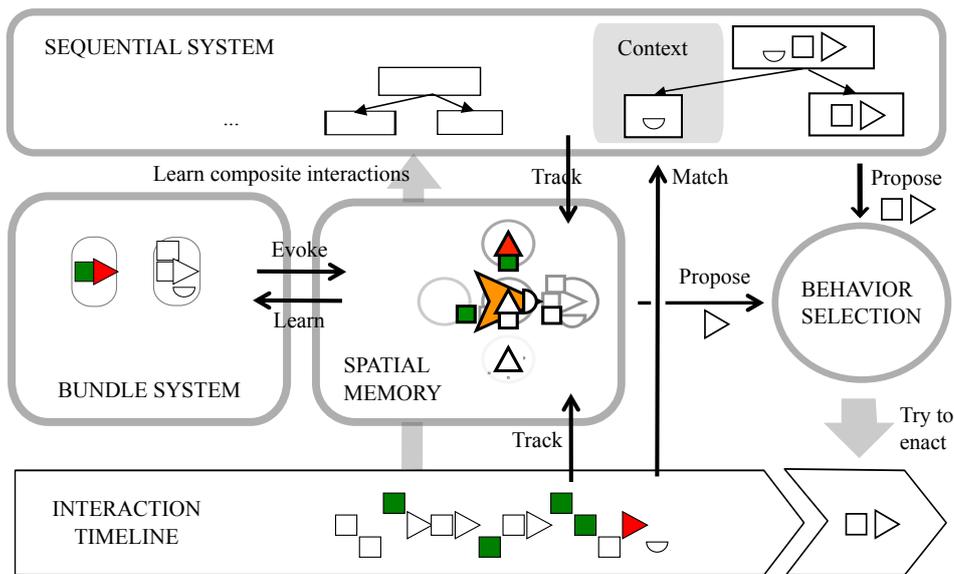


FIGURE 1.5 – Architecture ECA (Enactive Cognitive Architecture). En haut, le système séquentiel basé sur IMOSHÉM. La mémoire spatiale (au centre) conserve les interactions précédemment effectuées en référentiel égo-centré. Les occurrences d’interactions placées à une même position sont associées sous la forme de *bundles*, conservées par le *Bundle System*. Les *bundles* caractérisent les interactions permettant d’interagir avec un même type d’objet. Ces *bundles* sont utilisés pour compléter le contenu de la mémoire spatiale. Le système de décision utilise les informations issues des systèmes séquentiel et spatial pour sélectionner l’interaction suivante. Une description détaillée de l’architecture ECA est disponible dans un article d’Olivier Georgeon [48]

La structure sensorielle de l’agent, permettant de localiser les informations sensorielles dans l’espace, ainsi que les déplacements provoqués par les interactions, sont prédéfinis.

Pendant les essais, l’agent a montré une capacité à associer les interactions et les stimuli, attribuant une attractivité aux stimuli basés sur les possibilités d’interaction, et ainsi, une sémantique aux éléments qui l’entourent. Il a aussi montré qu’il disposait d’une persistance de l’objet. Il était en effet capable de réagir à des objets situés en dehors de son champ de perception. On observe aussi que les séquences d’interaction sont plus courtes. Une explication serait qu’en espace ouvert, les régularités séquentielles d’interactions sont moins évidentes à définir : il y a par exemple plusieurs façons de rejoindre un point de l’espace.

Afin de développer les mécanismes permettant la construction d’une mémoire spatiale sans se baser sur des préconceptions, nous nous sommes intéressé à la façon dont les êtres vivants intègrent et encodent l’espace qui les entoure.

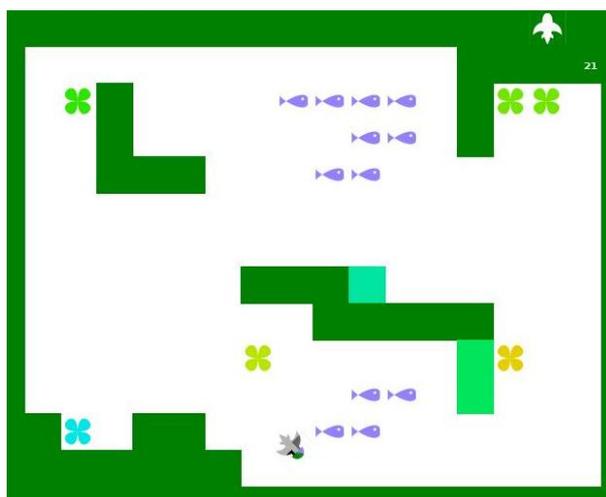


FIGURE 1.6 – Environnement utilisé pour tester l’architecture ECA. L’environnement est continu et contient différents objets colorés doté de propriétés diverses visuelles et tactiles.

1.1.3 Représentation de l’espace chez les êtres vivants

Nous nous inspirons de la façon dont les êtres vivants intègrent l’espace qui les entourent pour concevoir nos modèles spatiaux. Nous limiterons cette introduction aux cerveaux des vertébrés. En effet, malgré leurs différences, on observe de nombreuses structures similaires ou au fonctionnement analogue. Chez les vertébrés, l’espace intervient dans différentes régions fonctionnelles du cerveau décrivant chacune une portion de l’espace entourant l’agent, et définies pour une utilisation donnée. Des observations menées sur des patients souffrant de lésions cérébrales [58] [27] ont confirmé la multiplicité des régions de l’espace traitées par le cerveau : certains sujets présentaient en effet une hémignégligence dans un espace (proche ou lointain), mais pas dans l’autre. Plusieurs modèles permettant de définir ces espaces ont été proposés dans la littérature, et sont présentés ci-dessous :

Théorie du Body Schema

La théorie du *Body Schema* consiste en une représentation mentale qu’un agent a de son corps, de l’espace qu’il occupe et de l’espace qu’il peut atteindre au travers de ses actions. Décrit pour la première fois en 1911 par les neurologistes Henry Head et Gordon M. Holmes [59] comme un modèle de la posture qui organise et localise les informations sensorielles de façon cohérentes, le *Body Schema* est considéré aujourd’hui comme l’ensemble des propriétés spatiales du corps d’un agent. La séparation entre la connaissance du corps de l’agent et l’espace proche n’est pourtant pas clairement définie : ces deux espaces sont en effet interdépendants, et leur séparation même est sujette à débat [20]. Le *Body Schema* est le sujet de nombreuses études récentes, notamment dans le domaine de la robotique développementale [62], la prise de conscience du fonctionnement de son propre corps étant en effet primordiale pour pouvoir agir et interagir avec son environnement.

Modèle de l'espace de Cutting et Vishton (1995)

Cutting et Vishton [29] ont défini un découpage de l'espace en trois régions en référentiel égocentré :

- l'espace personnel : il s'agit de l'espace que l'on peut atteindre sans se déplacer, et donc à portée de main. Les auteurs estiment que cette région a une portée de 2 mètres environ.
- l'espace d'action : espace que l'on peut atteindre en effectuant un court déplacement. Il est défini comme l'espace dans lequel on peut se déplacer rapidement ou lancer un objet. La portée de cette région est estimée à environ 30 mètres chez l'homme.
- l'espace éloigné : espace global qui prend en compte la totalité de l'espace couvert par le champ visuel.

Ces régions sont emboîtées les unes dans les autres. On peut cependant reprocher à ce modèle de ne tenir compte que du système visuel.

Modèle de Previc (1998)

Previc [108] propose un découpage de l'espace en quatre régions (Figure 1.7). La première, appelée espace péri-personnel, est défini comme l'espace proche directement atteignable, et dans lequel on peut manipuler les éléments qui composent notre environnement. Cette région est similaire à l'espace personnel défini par Cutting et Vishton. Cependant, Previc ajoute un aspect sensorimoteur, puisque l'espace péri-personnel est défini comme l'espace avec lequel on peut interagir. De plus, la gestion de cet espace prend en compte des stimuli issus de nombreuses modalités sensorielles : le système visuel, tactile, vestibulaire et proprioceptif, permettant de connaître la posture de notre corps dans l'espace. Les trois autres régions concernent l'espace situé au-delà de l'espace péri-personnel, c'est-à-dire hors de portée des actions. Cet espace, appelé espace extra-personnel, est divisé en trois régions :

- l'espace extra-personnel focal : région entourant le point de l'espace visé par le système visuel (chez les êtres vivants dotés de fovéa), utilisé pour la reconnaissance visuelle d'objets.
- l'espace extra-personnel d'action : espace entourant l'individu, servant à calculer les déplacements pour localiser et rejoindre des éléments dans l'espace. Sa portée est estimée à environ 30 mètres chez l'homme par l'auteur.
- l'espace extra-personnel ambiant : espace lointain servant de référentiel pour l'orientation dans l'espace.

Navigation dans l'espace

Les mammifères disposent d'une structure cérébrale, l'hippocampe, contenant des neurones appelés cellules de position et cellule d'orientation [93][124]. Les premières s'activent lorsque l'animal se trouve à une certaine position dans son environnement (en référentiel géocentré), et les secondes, à l'orientation de la tête. L'ensemble des cellules de lieu forment une carte cognitive [94] aidant à la navigation dans l'espace sur de grandes distances, et sont impliquées dans les mécanismes de la mémoire pour attribuer des contextes spatiaux aux souvenirs [116]. De par

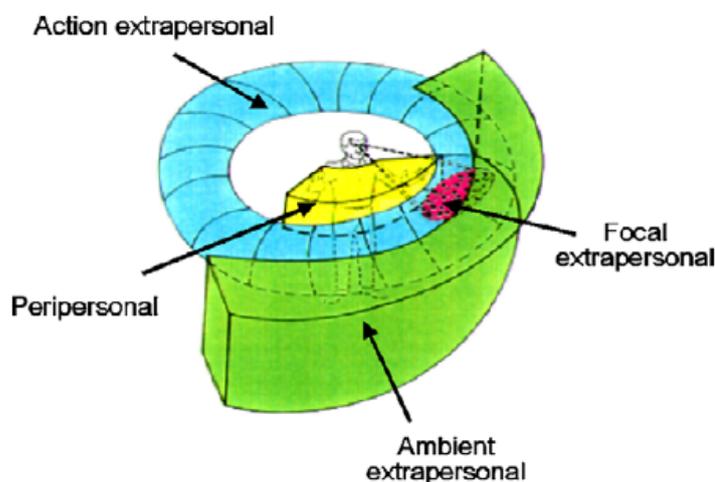


FIGURE 1.7 – Modèle de l'espace de Previc [108]. L'espace est divisé en quatre parties : l'espace péri-personnel, l'espace extra-personnel d'action, l'espace extra-personnel ambiant et l'espace extra-personnel focal).

les capacités de navigation qu'il confère, l'hippocampe a été une source d'inspirations dans le domaine de la robotique (review par Trullier et al. [126]).

1.1.4 Cadre du travail de thèse

Comme le montre la section 1.1.3, les êtres vivants caractérisent l'espace qui les entourent en définissant un large éventail de régions fonctionnelles. Dans cette thèse, nous limiterons notre étude à certaines de ces régions que nous estimons nécessaires et suffisantes pour permettre à un agent de naviguer dans un environnement ouvert ; nos agents étant en effet inspirés par des êtres vivants très simples ne nécessitant pas forcément l'émergence de l'ensemble de ces régions pour survivre.

Nous définissons un modèle simplifié composé des deux régions suivantes :

- l'espace péri-personnel, caractérisant l'espace dans lequel l'agent peut interagir avec son environnement. Contrairement à l'espace péri-personnel défini par Previc, nous prenons en compte l'espace qui peut être atteint par un "court" déplacement, que nous considérons comme faisant partie de l'interaction.

- l'espace extra-personnel d'action, qui définit l'espace qui entoure l'agent.

Notons que ces régions ne sont pas limitées à l'espace couvert par le système sensoriel, mais permettent de caractériser l'ensemble de l'espace pouvant être interagi par l'agent.

Nos agents étant très simples, nous ne cherchons pas à étudier le *Body Schema* : en effet, nos agents sont indéformables, et, de ce fait, l'agent n'a pas besoin de connaître la structure de son corps. Cependant, nos résultats montrent que les structures apprises par l'agent reflètent en partie l'espace occupée par le corps de l'agent. Notre modèle ne définit pas l'espace extra-personnel focal, puisque nos agents ne disposent pas de fovéa.

Notre étude concerne principalement la façon dont l'agent apprend à définir les éléments présents dans son environnement, et à générer des comportements adaptés à un contexte spatial. Nous ne cherchons donc pas à implémenter des mécanismes de navigation à longue distance. De ce fait, notre modèle ne définit pas d'espace extra-personnel ambiant, et n'implémente pas un modèle de l'hippocampe. Des espèces animales rudimentaires peuvent en effet survivre sans sembler être capable de planifier des navigations complexes.

1.2 Question de recherche de la thèse

Les travaux et observations présentés précédemment montrent l'importance d'utiliser des mécanismes additionnels pour permettre à un agent artificiel de découvrir et exploiter les régularités spatiales offertes par son environnement, afin qu'il puisse générer des comportements dans un espace ouvert. Les problématiques abordées dans cette thèse peuvent se résumer à la question de recherche suivante :

Comment un agent artificiel peut-il découvrir, organiser et exploiter des régularités spatio-temporelles issues du couplage entre l'agent et son environnement pour exhiber des comportements "intelligents" ?

Plus spécifiquement, nous cherchons à doter un agent de la possibilité de construire une représentation interne de son contexte spatio-temporel lui permettant de prédire quelles interactions avec l'environnement sont possibles, basé sur un retour sensoriel limité (c'est-à-dire insuffisant à lui seul pour générer de telles prédictions). Comme nous sommes dans le contexte d'une étude intégrant la motivation intrinsèque, nous ne cherchons pas à générer une modélisation explicite de l'environnement ni de l'agent lui-même. Nous cherchons à construire un contexte "augmenté" à l'aide des informations obtenues précédemment, et suffisant pour permettre une exploitation par l'agent dans l'unique but de générer des comportements "intelligents" adaptés au contexte environnemental et satisfaisant sa motivation intrinsèque. Par comportement intelligent, nous définissons le comportement d'un agent dont les propriétés intrinsèques donne à un observateur extérieur l'impression que l'agent est intelligent (de façon subjective), et qu'il attribue un sens aux éléments qui composent son environnement par la façon qu'il a d'interagir avec ces éléments.

Nous définissons le couplage agent-environnement comme l'ensemble des interactions possibles entre l'agent et son environnement, au sens défini par l'approche interactionnelle proposée par le projet IDEAL décrit en section 1.1.2. Cette approche cherche à implémenter les hypothèses d'épistémologie constructiviste défini par Piaget, qui stipulent que c'est par l'organisation des interactions qu'un agent attribue un sens au monde qui l'entoure. Afin de tester ces hypothèses, nos agents interagissent avec leur environnement au travers d'interactions implémentant la notion de schème sensorimoteur. Une régularité de ce couplage est définie comme une association entre un contexte spatial et/ou temporel de l'environnement et une interaction qui peut être effectuée dans ce contexte. Étant donné que l'agent intègre son environnement par le biais des interactions qu'il effectue, le contexte sera lui-même un ensemble d'interactions. Une régularité se définit donc comme une association entre des interactions ou groupes d'interaction. L'apprentissage de ces régularités permettra ainsi à un agent d'organiser ses interactions, et donc, de donner un sens à son environnement.

La construction d'un contexte spatio-temporel consiste à stocker et organiser les régularités observées par l'agent de telle sorte que des régularités non observables par le biais des interac-

tions énoncées par l'agent (i.e. sa "perception" de l'environnement) puissent être détectées. Ce contexte permet ainsi d'augmenter l'expérience que l'agent a de son environnement en exploitant les informations portées par les régularités évoquées, et lui permet de prédire l'énonciation future de certaines de ses interactions. La construction du contexte repose ainsi sur l'apprentissage d'associations entre des régularités spatio-temporelles. L'exploitation du contexte revient à définir l'ensemble des interactions que l'agent peut effectuer pour permettre au système de sélection de choisir la ou les interactions suivantes de façon à satisfaire son système motivationnel à court et/ou long terme. Les mécanismes permettant la construction de ce contexte doivent se baser sur un minimum de préconceptions afin de respecter au mieux le principe d'agnosticisme environnemental. Les préconceptions utilisées devront être soutenues par des observations biologiques.

1.3 Sous-questions de recherche

Le développement des mécanismes d'apprentissage ont fait émerger un ensemble de sous-problèmes :

- **Quelles sont les propriétés nécessaires et suffisantes de l'environnement pour permettre la construction d'un contexte spatio-temporel ?** La spécification de l'environnement est une sous-question de la spécification du couplage agent/environnement. L'environnement doit satisfaire un certain nombre de propriétés pour permettre l'existence de régularités. C'est notamment vrai pour les régularités spatiales, qui ne peuvent exister que si les propriétés géométriques de l'espace sont invariantes en tout point. La définition de ces limites permet de définir dans quel cadre nos mécanismes peuvent être utilisés.

- **Quelles informations sont nécessaires et suffisantes pour permettre à un agent d'interagir avec son environnement en se basant sur le couplage entre l'agent et l'environnement ?** Nous reprenons l'idée du couplage sensori-moteur : l'agent doit organiser ses connaissances dans l'unique but de générer des comportements. Il faut éviter l'apprentissage d'informations non pertinentes.

- **L'agent a-t-il besoin d'une organisation topographique de l'espace pour interagir avec son contexte spatial ?** La plupart des êtres vivants disposent de structures cérébrales présentant une structure représentant une topographie déformée de l'espace. Cette topographie est-elle nécessaire ou n'est-ce qu'un moyen de simplifier le traitement des informations ?

- **Comment permettre un apprentissage efficace des régularités spatio-séquentielles ?**

Le comportement de l'agent est conditionné par les régularités et associations de régularités qu'il a appris. Nous développons l'idée qu'il faut un mécanisme qui pousse l'agent à agir dans le but d'apprendre des régularités plutôt que pour suivre sa motivation intrinsèque.

- **Comment analyser le comportement de l'agent ?** Comme nous ne définissons pas de but à atteindre ni de tâche à accomplir, nous devons trouver des méthodologies d'évaluation du comportement de nos agents.

Chapitre 2

État de l'art

Ce chapitre présente une synthèse des travaux en relation avec le travail de thèse. Les mécanismes que nous proposons répondent à un ensemble de problématiques, étudiées séparément dans les travaux cités. L'état de l'art est structuré par problématiques : les travaux menés dans le domaine de l'intelligence artificielle développementale dont les axes de recherche sont similaires aux nôtres, les travaux portant sur l'apprentissage et l'exploitation de connaissances acquises par l'expérience, puis les modèles destinés à représenter l'espace, et les travaux portant sur la navigation d'un agent dans son environnement à partir de connaissances acquises par l'expérience.

2.1 Intelligence artificielle développementale

Cette section s'intéresse aux travaux en intelligence artificielle développementale portant sur l'étude de mécanismes intrinsèquement motivés et dotés d'une forme de perception active.

Oudeyer, Kaplan et Hafner [99] ont défini un mécanisme d'apprentissage développemental autonome basé sur le principe de la motivation intrinsèque. Ce principe pousse l'agent à effectuer des actions dans le but d'en apprendre les conséquences, en portant plus d'intérêt aux actions dont l'apprentissage est le plus rapide. Cependant, cette approche se limite à l'apprentissage des actions de l'agent, et ne porte pas sur la construction de connaissances spatiales d'objets de l'environnement.

Nguyen et al. [90] et Ivaldi et al. [67] proposent une approche dans laquelle un robot apprend à définir et reconnaître des objets, poussé par un mécanisme d'apprentissage intrinsèquement motivé, implémentant une forme de curiosité artificielle. Le robot apprend à reconnaître les objets de façon active, en les manipulant et en observant les propriétés qui en résultent. Cette approche se focalise également sur l'apprentissage de nouvelles connaissances et ne porte pas sur la génération de nouveaux comportements basés sur ces connaissances.

Blank et al. [13] proposent un mécanisme d'apprentissage hiérarchique, où chaque niveau définit une abstraction des informations issues du niveau inférieur, et apprend à prédire ces informations. Le système est intrinsèquement motivé, et cherche à atteindre des états de l'environnement où les informations sont prédictibles et permettent un apprentissage rapide. L'agent

définit alors des *protoplans* à effectuer et des *protogoals* à atteindre pour satisfaire cette motivation. Cette motivation est ainsi comparable à une forme de curiosité artificielle. Cependant, ce mécanisme n'organise pas spatialement les connaissances acquises et le comportement de l'agent.

2.2 Apprentissage et exploitation des connaissances acquises par l'expérience

Cette section s'intéresse aux travaux centrés sur l'apprentissage et l'exploitation de connaissances sur les objets présents dans l'environnement d'un agent artificiel, ainsi que leur exploitation par cet agent pour générer des comportements. Une première partie reprend les approches dont le but consiste seulement à extraire des connaissances de haut niveau à partir des affordances découvertes. La seconde partie commente (de façon non exhaustive) les travaux destinés à exploiter les affordances dans le but de générer des comportements.

2.2.1 Apprentissage de connaissances basées sur l'expérience

Dans ces modèles, les connaissances acquises ne sont pas exploités pour générer des comportements. Les approches listées dans cette section ont pour but de classifier des objets à partir de la façon dont un agent peut les percevoir et/ou interagir avec eux.

Alexander Maye et Andreas K. Engel [83] proposent une approche dans laquelle un robot apprend à catégoriser des objets à l'aide d'une séquence d'interactions. Les objets réagissent en effet différemment aux actions du robot. L'agent apprend à définir des séquences d'interactions lui permettant de reconnaître les objets qui lui sont présentés. Cependant, les déplacements de l'agent dans l'espace sont limités à une translation. L'agent ne peut donc pas découvrir et exploiter de régularités spatiales.

Tucker Hermans, James M. Rehg et Aaron Bobick [60] proposent un mécanisme d'apprentissage des affordances basé sur la reconnaissance des propriétés visuelles des objets. Ce mécanisme centre automatiquement les objets et ne peut donc pas être utilisé pour localiser les objets dans l'espace.

Ignasi Cos-Aguilera, Lola Cañamero et Gillian Hayes [25] utilisent une carte SOM (Self Organizing Map) pour segmenter les objets. Leur approche repose sur l'apprentissage, par un robot (simulé) muni d'une pince, des actions permettant de saisir des objets de différentes largeurs. Le robot navigue aléatoirement dans son environnement. Lorsqu'il est proche d'un objet, le système de décision centre automatiquement le robot face à l'objet, qui peut tester un mouvement pour le saisir. Comme la segmentation des objets est basée sur une carte SOM, il n'est pas nécessaire de définir *a priori* des classes d'objets. Cependant, l'agent ne génère aucune information spatiale sur les objets qu'il définit.

L'approche proposée par S. Griffith et al. [56] utilise les propriétés acoustiques et proprioceptives pour reconnaître les objets : un robot humanoïde manipule des objets sous un jet d'eau et analyse le bruit et les informations proprioceptives. S. Griffith et al. [55] proposent une

approche, similaire à la précédente, où l'interaction avec les objets permet de catégoriser les objets. Un robot utilise une balle qu'il laisse tomber sur l'objet à étudier. Le comportement de la balle permet de savoir si l'objet est un récipient ou non. Le résultat est appris et prédit à l'aide des propriétés visuelles de l'objet. Dans ces deux approches, l'aspect interaction est très important puisque, d'une part, les informations acoustiques et proprioceptives, et d'autre part, les déplacements de la balle, sont liées aux actions du robot sur les objets. Ces approches ne considèrent pas le problème de l'émergence des comportements, et ne cherchent pas à extraire des informations spatiales des interactions.

Akgün et al. [2] proposent une approche dans laquelle un robot humanoïde (virtuel) apprend à identifier des objets d'après les conséquences des actions du robot sur ces objets. L'apprentissage est qualifié de non-supervisé et l'agent ne connaît pas *a priori* le nombre d'objets. Une carte SOM permet de définir de façon autonome des classes d'objets pour lesquels les actions de l'agent sur ces objets ont des conséquences similaires. Ce modèle ne considère pas le problème de l'émergence des comportements, ni de l'apprentissage des propriétés spatiales des objets et des actions de l'agent.

Dans ces modèles, les connaissances acquises ne sont pas exploitées pour générer des comportements. En cela, ces approches diffèrent de l'approche sensorimotrice, les connaissances acquises par l'expérience doivent être d'une part construites dans un but précis, c'est-à-dire leur utilisation pour interagir et percevoir l'environnement, et d'autre part, être utilisées dans la génération de comportements afin d'éprouver et valider ces connaissances.

2.2.2 Apprentissage de connaissances dans le but de générer des comportements

Les modèles présentés dans cette section permettent l'apprentissage de connaissances exploitables par l'expérience, dans le but d'être réutilisées pour générer des comportements. Ils se rapprochent ainsi de nos questions de recherche.

Rolf Pfeifer and Christian Scheier [104] proposent une approche dans laquelle un robot apprend à reconnaître les objets en fonction de la possibilité de les manipuler (les porter ou les pousser) en fonction de la perception qu'il en a. Le robot se déplace aléatoirement dans l'espace jusqu'à ce qu'il perçoive un objet. L'agent génère implicitement des modèles fonctionnels des éléments composant son environnement. Cette segmentation lui permet de reconnaître ces objets par la façon dont il les perçoit, et de savoir comment interagir avec un élément situé devant lui sans qu'il ne soit nécessaire de générer une modélisation interne de ces objets. Le mécanisme permettant l'apprentissage et la reconnaissance des objets reste cependant limitée à l'espace immédiat de l'agent, et ne porte pas sur la génération de comportements spatiaux adaptés à l'environnement lointain.

Luis Montesano et Manuel Lopes [87] proposent un modèle permettant à un robot de déterminer s'il peut saisir un objet, en reconnaissant descripteurs visuels caractéristiques des objets qu'il peut saisir. La connaissance des affordances ne permet pas de générer des comportements,

mais les informations apprises sont directement exploitables par l'agent, et, de ce fait, peuvent contribuer à la décision des actions, et à la validation des connaissances acquises.

Dans l'approche proposée par Uğur et al. [128], l'agent apprend à définir la *traversabilité* d'un objet, c'est-à-dire la possibilité d'avancer, selon une certaine trajectoire, en fonction de ce que l'agent perçoit de son environnement. Les connaissances acquises sont utilisées pour naviguer dans un environnement encombré. Ce mécanisme apprend à prédire la possibilité d'avancer d'après des informations visuelles. Cependant, les structures apprises ne sont pas utilisées pour localiser les objets dans l'espace, ce qui limite la portée du mécanisme de décision : l'agent peut se retrouver bloqué dans une situation où aucune des trajectoires qu'il connaît ne lui permet d'avancer, puisqu'il ne perçoit cette situation qu'au dernier moment.

José Baleia et al. [11] proposent une approche similaire. Ces auteurs utilisent un système constitué d'un bras permettant de "sonder" l'environnement situé devant le robot, lorsque la traversabilité ne peut être définie avec une certitude suffisante. Le robot est ainsi doté d'une forme de perception active lui permettant de tester ses connaissances. Cependant, l'utilisation de cette perception active est fondée sur des préconceptions.

Ces modèles permettent l'apprentissage de connaissances exploitables par l'expérience. Cependant, ils ne permettent pas d'intégrer l'environnement environnant et, de ce fait, ne sont pas utilisables pour la navigation dans un environnement ouvert.

2.3 Apprentissage et exploitation de modèles pour représenter l'espace

La représentation de l'espace d'un agent est un élément essentiel dans la génération de comportements dans des environnements ouverts. Cette section résume les formalisations et modèles décrivant la représentation de l'espace, ainsi que les implémentations chez des agents artificiels. La grande majorité de ces travaux concernent l'étude d'un seul type d'espace. Les travaux sont regroupés en fonction de l'espace concerné : espace péri-personnel, espace extra-personnel, navigation dans l'espace lointain. Nous listerons également les travaux portant sur l'étude des propriétés d'une représentation de l'espace, telle la permanence de l'objet, et les modèles cognitifs de l'espace.

2.3.1 Espace péri-personnel

Dans le domaine de la robotique développementale, on retrouve un grand nombre de travaux sur l'étude de l'émergence du *Body Schema*, celui-ci étant considéré comme essentiel pour permettre à un agent d'interagir avec son environnement [62]. Une partie de ces travaux s'étend également à l'espace directement atteignable par l'agent, et permettent la génération d'une structure caractérisant l'espace autour de l'agent. Ces travaux restent cependant limités à l'espace proche de l'agent et ne permettent pas de générer des comportements spatiaux.

Dans l'approche proposée par R. Detry et al. [31], le système détermine les positions de l'espace à partir desquelles un bras robotique peut saisir un objet, à partir d'informations visuelles. Ce système permet ainsi à un agent de saisir un objet sans présupposés sur cet objet. Notons cependant que les positions sont définies en simulant un modèle corporel prédéfini du robot. Cette approche ne génère pas des comportements dans l'espace.

Certains travaux concernent la façon dont les multiples modalités sensorielles (visuelles, tactiles et proprioceptives) peuvent encoder l'espace. Fuke, Ogino et Asada [38][37], et Chinellato et al. [23] proposent des modèles permettant de générer un lien implicite entre une position de l'espace visuel et les positions atteintes par le bras d'un robot simulé. Le premier modèle permet, en outre, de localiser les stimuli tactiles dans cette représentation, y compris les stimuli situés en dehors de l'espace visuel de l'agent. Ces approches permettent de générer un modèle de l'espace qui ne repose pas sur un référentiel cartésien, mais ne peuvent pas être utilisées pour générer des comportements impliquant des déplacements dans l'espace.

D'autres modèles concernent l'apprentissage et l'exploitation de modèles permettant de caractériser l'espace proche et son contenu dans le but d'agir sur l'environnement. Pierce et Kuipper [106] proposent une approche permettant à un agent de construire une représentation de son propre système sensoriel dans l'espace, utilisé pour définir les déplacements associés aux commandes motrices. Cette représentation se base sur les similarités des valeurs fournies par les capteurs. Le modèle généré est ensuite utilisée pour la navigation de l'agent dans son environnement, notamment pour éviter les collisions. Dans notre approche, les stimuli sensoriels sont positionnés les uns par rapport aux autres par les interactions qui permettent de passer d'un stimulus à un autre. Il n'est donc plus nécessaire de définir les déplacements de l'agent puisque la structure caractérisant l'espace est déjà basée sur ces informations.

2.3.2 Espace extra-personnel

Contrairement à l'espace péri-personnel, peu d'études ont été menées sur l'espace extra-personnel d'action. Je regroupe ici les méthodes utilisées pour permettre à un agent de représenter l'espace qui l'entoure, et de naviguer dans son environnement. Nous nous limiterons aux approches basées sur la localisation des objets en référentiel égo-centré.

Kawamura et al. [70] proposent le modèle de l'*Ego-Sphère*. Ce modèle consiste à projeter des points d'intérêt sur une sphère centrée sur l'agent. Les points d'intérêt sont localisés en coordonnées polaires sur la sphère. La position d'un robot peut ainsi être caractérisée par un ensemble de points d'intérêt, et les déplacements par des variations de coordonnées. L'agent se déplace pour rapprocher la position de ces points vers l'état observable d'un point de l'environnement défini comme la position à atteindre. L'agent prend ainsi en compte l'ensemble de l'environnement qu'il perçoit pour définir sa position dans l'environnement. Cependant, la distance des points d'intérêt n'est pas définie, ce qui ne permet pas de localisation d'un objet dans l'espace.

La *neural map* proposée par Lagoudakis et Maida [75] définit une mémoire égo-centrée de portée limitée, qui génère une carte des *stimuli* issus du système sensoriel de l'agent, et donne la

direction d'un but à atteindre, quelle que soit sa distance. La mémoire est constituée de nœuds, positions élémentaires sur cette carte, associés à des positions de l'espace égocentré, et liés par des liens. Ces liens permettent, d'une part, de déplacer les informations des stimuli d'un nœuds vers un autre lorsque l'agent se déplace, et d'autre part, de calculer les déplacements vers un but. La *neural map* nécessite de pré-définir les liens entre les nœuds. Il est également nécessaire de donner une signification *a priori* aux *stimuli* : dans l'implémentation proposée par l'auteur, les *stimuli* sont définis par des télémètres donnant la distance et la position des obstacles à éviter. Cette approche diffère également de l'approche développementale par le fait qu'un but doit être défini.

2.3.3 Permanence de l'objet

Chez les êtres vivants, la permanence de l'objet est la compréhension que les objets détectés dans l'environnement continuent d'exister même lorsque l'on ne peut pas interagir directement avec eux. Cette notion a été définie pour la première fois par Piaget en 1954 [105]. La persistance de l'objet est primordiale dans la compréhension de l'espace puisque l'agent ne peut le percevoir dans sa globalité. Les travaux suivants portent sur l'étude et l'implémentation de mécanismes permettant d'attribuer cette capacité à un agent artificiel.

Mareshal et al. [80] proposent un modèle prédictif qui apprend à prédire l'image d'un objet sur une rétine artificielle en fonction des positions passées de l'image de cet objet sur la rétine. Ces prédictions permettent de suivre un objet et d'estimer sa position lorsque l'objet est masqué ("effet tunnel"). Notons toutefois que le modèle est très simple : la projection de l'objet a toujours la même taille et se déplace à vitesse constante sur la rétine. Ce modèle ne peut donc pas être utilisé pour suivre un objet dans l'espace. On peut noter que ces auteurs estiment que la persistance de l'objet repose sur deux mécanismes distincts, pour reconnaître les objets et pour les suivre dans l'espace. Cette distinction se retrouve dans nos mécanismes de mémoire spatiale.

Hande Çelikkanat, Erol Şahin et Sinan Kalkan [21] proposent une approche basée sur le *Slow Feature Analysis*. Cette méthode extrait d'un signal sensoriel les informations qui varient le moins tout en étant porteuses d'informations. Cette approche permet d'estimer la position d'un objet en mouvement, même lorsqu'il est masqué, et détecte le changement d'objets. Cependant, les expérimentations sont limitées à des décors fixes dans lesquels l'objet à suivre se déplace. Il n'est pas possible d'extraire d'autres informations sur l'objet mobile que sa position sur l'image. Le fait que le mécanisme n'interagisse pas avec l'environnement empêche l'émergence d'une forme de compréhension des éléments présents dans l'environnement. Ce mécanisme ne peut donc pas être utilisé pour générer des comportements spatiaux.

2.3.4 Modèles cognitifs de l'espace

Un modèle mental de l'espace consiste en une structure permettant à un agent d'organiser ses connaissances dans l'espace et d'effectuer des simulations de son comportement. Jona Boedinghaus et al. [14], proposent un modèle dans lequel l'espace est représenté par une matrice.

On donne un ensemble de relation spatiales entre des objets (à droite de, à gauche de...). Le système, basé sur l'architecture cognitive ACT-R [4], détermine la position relative des objets dans le modèle mental. Ce modèle reste cependant limité à l'organisation d'éléments dans l'espace à partir d'un ensemble de relations donné, et ne permet pas l'exploitation de cette organisation dans le but de générer des comportements.

Erdem Erdemir et al. [35] proposent une approche dans laquelle un robot humanoïde (constitué d'un torse muni de deux bras) apprend à définir les mouvements qu'il peut effectuer en présence d'objets pouvant faire obstacle à ses mouvements. Le robot peut effectuer des simulations de comportements permettant d'atteindre une certaine position avec ses bras, à partir des affordances détectées dans l'espace proche, ce qui lui permet de simuler différentes stratégies avant d'effectuer le mouvement. Cette approche nécessite de découvrir les points de collision dans l'espace pour définir une carte des affordances, ce qui implique que les objets soient statiques par rapport à l'agent. Ce mécanisme ne peut donc pas être utilisé pour générer des déplacements de l'agent dans l'espace.

2.4 État de l'art de la navigation basée sur des connaissances acquises par l'expérience

Cette section regroupe les mécanismes permettant à un agent de naviguer dans son environnement, fondés sur des modèles bio-inspirés et/ou utilisant des connaissances acquises par l'expérience.

Un grand nombre de travaux [126] proposent des mécanismes de planification des déplacements dans l'espace, inspirés de l'*hippocampe* des mammifères. On peut citer les mécanismes de navigation bio-inspirés de M. J. Mataric [81] et de Sheynikhovich et al. [114], et les travaux menés sur le robot-rat *Psikharpax* [85],[19]. Ces approches se basent sur l'utilisation de *cellules de lieu* et *d'orientations*, inspirées des neurones de l'hippocampe. Chaque cellule de lieu est associée à un endroit de l'environnement, reconnaissable par un ensemble de repères (visuels, tactiles...) que l'agent peut percevoir. Des liens sont définis entre les cellules de lieu lorsqu'il est possible de passer d'un endroit à un autre de l'environnement, définissant ainsi une carte topologique de l'environnement. Cette carte permet la planification des déplacements dans l'espace. À chaque cellule de lieu peut être attribuée une certaine attractivité, liée aux expériences de l'agent dans cet environnement. Ces approches permettent ainsi d'établir une carte des différents lieux visités par l'agent et de leur attribuer des propriétés liées à leur expérience dans ces lieux. Cependant, elles ne permettent pas de caractériser l'espace entourant l'agent et les éléments détectés. Ces approches ne permettent donc pas de générer des comportements dans un nouveau contexte environnemental.

Dongshin Kim et al. [72] proposent une approche dans laquelle un robot mobile apprend à définir la possibilité d'avancer sur un terrain naturel (i.e. contenant des obstacles variés). Le robot catégorise des portions de son environnement en deux types : les surfaces traversables et

les surfaces non-traversables, ce qui permet de réduire la complexité avec laquelle il appréhende son environnement. L'espace est divisé en régions dont la traversabilité est définie en fonction de ses propriétés visuelles. Les déplacements sont ensuite générés par un algorithme de *path planning*, ce qui nécessite de prédéfinir un but à atteindre. On peut également noter que la carte générée par le robot est topographique, et nécessite de prédéfinir la structure de l'espace.

Jefferson Provost, Benjamin J. Kuipers et Risto Miikkulainen [109] proposent une approche basée sur l'apprentissage par renforcement dans laquelle l'agent apprend un ensemble de prototypes de perceptions, à l'aide d'une carte SOM. Le système de contrôle utilise un contrôleur de type *trajectory following* pour emmener l'agent à proximité de la position caractérisée par le prototype de perception. Un contrôleur *Hill-climbing* permet ensuite à l'agent d'atteindre la position correspondant au prototype de perception. Ce système permet à l'agent de naviguer par étapes, et donc de réduire la complexité du problème de l'apprentissage par renforcement. Cependant, il est toujours nécessaire de définir le but à atteindre.

La méthode des champs de potentiel (potential field method) [5] [71] [73] est devenue très populaire de par sa simplicité. La position du but à atteindre génère une forte attractivité sur l'agent, et les obstacles, une répulsion. On peut ainsi calculer en tout point de l'espace un vecteur force à appliquer à l'agent. Un algorithme de *path planning* calcule, à partir de ces forces, le chemin le plus court et le plus sûr pour atteindre le but. Le mécanisme proposé dans cette thèse s'inspire fortement de cette méthode. Cependant, la méthode des champs de potentiel nécessite de définir les propriétés des objets *a priori* (attraction ou répulsion), ainsi qu'un but à atteindre. Il nécessite également de connaître l'ensemble des éléments présents dans l'environnement, ce qui n'est pas envisageable sur un environnement réel.

Dejanira Araiza-Illan et Tony J. Dodd [6] proposent un mécanisme de *path planning* permettant à un agent d'atteindre une position de l'espace en évitant des prédateurs. Ce système prend ainsi en considération plusieurs éléments de son environnement, dont certains sont dynamiques. Cependant, ces éléments doivent être prédéfinis.

2.5 Conclusion

Comme le montre ce chapitre, les principales problématiques de la prise en compte de l'espace par un agent ont été abordées dans la littérature. Cependant, ces problématiques ont été abordées séparément. Ainsi, les mécanismes proposés qui utilisent des connaissances acquises par l'expérience ne permettent pas de générer des comportements spatiaux, et ceux dont le but est d'étudier des mécanismes de navigation utilisent des connaissances prédéfinies par le concepteur de l'agent. Nous considérons que ces deux aspects sont liés et ne peuvent pas être développés séparément. Nous pensons en effet que les comportements générés par l'agent doivent se baser uniquement sur des connaissances de l'environnement qu'il a lui-même généré par le biais de son expérience, afin que ces informations reflètent au plus près le couplage entre l'agent et l'environnement, et donne à l'agent la possibilité de s'adapter à des changements environnementaux.

TABLE 2.1 – Tableau récapitulatif de l'état de l'art. Nous mettons en évidences les différentes propriétés présentées dans ce chapitre. Chaque ligne regroupe les travaux d'une même section. Les mécanismes proposés dans IDEAL sont affichés en bas du tableau.

	Intrinsèquement motivé	Intégration autonome des objets	intégration autonome de l'espace proche	intégration autonome de l'espace lointain	émergence de comportements	exploitation des connaissances	apprentissage du <i>Body Schema</i>
Oudeyer et al. [99]	*						
Nguyen et al. [90]	*	*					
Ivaldi et al. [67]	*	*					
Blank et al. [13]	*					*	
Approches basées sur l'apprentissage des objets [83], [60], [25], [56], [55],[2]		*					
Rolf et al. [104]		*				*	~
Luis et al. [87]		*				~	
Uğur et al. [128]		**	~			*	
José et al. [11]		*	~			*	
Detry et al. [31]		~	*			*	
Fuke et al.[38][37]			*				*
Chinellato et al. [23]			*				*
Pierce et al. [106]			*		~	*	*
Kawamura et al. [70]				~	*		
Lagoudakis et al. [75]			~		*		
Mécanismes inspirés de l'hippocampe [81],[114], [85], [19]				*	*	*	
Kim et al. [72]		*	~	~	*	*	
Provost et al. [109]		~			*		
Méthode des <i>Potential Fields</i> [5], [71], [73]					*		
Araiza-Illan et al. [6]					*		
ECA [48]	*	*	~	~	*	*	
Mécanisme intégrant l'espace péri-personnel (Chapitre 9)	*	*	*		~	*	~
Système PSRI (Chapitre 10)	*	*	*	~	*	*	~
Système PRI (Chapitre 11)	*	*	*	*	*	*	~

Deuxième partie

Contributions

Chapitre 3

Présentation générale de la méthode d'élaboration des contributions

3.1 Introduction aux contributions

L'objectif de cette thèse est de dépasser les limites observées avec le mécanisme séquentiel IMOSHEM (Chapitre 1). Je propose un modèle permettant à un agent artificiel de naviguer dans un environnement ouvert et continu. Ce modèle permet de construire et d'exploiter un contexte spatial construit au travers des interactions entre l'agent et son environnement.

Nous adoptons un modèle cognitif implémentant des théories de la psychologie qui implique que l'action et la perception ne sont pas séparées *a priori*, appelé *Radical Interactionism* (RI) [43]. Le modèle RI permet de considérer un agent indépendant de tout présumé sur l'environnement : contrairement aux modèles classiques (par exemple, le modèle POMDP [7]), il ne se base pas sur la notion d'état de l'environnement, mais uniquement sur la notion d'interaction de l'agent avec son environnement. L'agent ne cherche pas à atteindre un but ou à effectuer une tâche prédéfinie, mais est dirigé par une motivation intrinsèque, associée à ses interactions.

L'implémentation du modèle RI nous a conduit à repenser la conception d'un robot basé sur ce modèle. En effet, les robots nécessitent des systèmes sensoriels et moteurs "équilibrés" pour respecter le principe de *balance écologique*, et ainsi permettre une implémentation de l'approche sensorimotrice du RI. Le terme *balance écologique* a été proposé par Pfeifer [100] pour caractériser le fait que les possibilités offertes par les capteurs et les actionneurs doivent être équilibrées pour permettre une approche sensorimotrice. Il est plus facile de construire un robot à partir de son système interactionnel plutôt que d'adapter le système interactionnel à un système sensoriel et moteur existant. Les agents robotiques utilisés pour nos expérimentations sont développés selon cette approche.

Je propose un système permettant à un agent artificiel de générer dans un premier temps des modèles implicites des éléments qui composent son environnement, en se basant sur ses interactions. Le système permet ensuite de construire des structures caractérisant l'espace proche (péri-personnel) et lointain (extra-personnel d'action). Les mécanismes permettant de construire

les structures caractérisant ces deux espaces sont très similaires, et non exclusifs, ce qui rend leur utilisation simultanée possible. De plus, le mécanisme lié à l'espace lointain utilise les connaissances acquises par le système lié à l'espace proche, et utilise des principes similaires, montrant que ces deux systèmes sont fortement liés.

Ces mécanismes permettent la construction d'une connaissance de l'espace et des objets en accord avec l'approche *Radical Interactionism*. Je me suis efforcé de supprimer les préconceptions à la fois sur les objets présents dans l'environnement, et sur la structure de l'espace. Dans la littérature, la question de l'émergence de la notion d'objets à partir des affordances a été étudiée. Cependant, cette connaissance n'est utilisée que dans le but de déterminer les actions immédiatement faisables. Nous proposons que l'agent utilise d'une part cette connaissance émergente sur les objets pour définir son environnement en termes d'interactions futures, et d'autre part pour mémoriser et suivre les interactions potentiellement énonçables.

3.2 Méthode de validation des mécanismes

Les propriétés des algorithmes mis en œuvre pour implémenter les différents mécanismes de mon système ont été testées au cours de trois expérimentations. Chacune d'elle met en évidence le fonctionnement d'un ensemble de processus sur un système fonctionnel, en supposant, lorsque c'est nécessaire, le fonctionnement des autres. Les trois expérimentations permettent de couvrir l'ensemble des processus.

3.2.1 Analyse des résultats et évaluation des mécanismes

Les mécanismes sont évalués en utilisant les critères décrits au chapitre 1. Cependant, on ne cherche pas seulement à évaluer les comportements de l'agent. Nous analysons également les connaissances apprises par l'agent. Ces informations nous permettent en effet de décrire comment l'agent définit son environnement au travers de ses interactions. L'implémentation du premier processus (voir chapitre 5) permet d'observer, d'un point de vue extérieur, les modèles implicites des objets construits par l'agent et leurs propriétés associées en observant les groupes d'interactions qu'il forme. Dans le cas d'un agent évoluant dans un environnement discret, il est possible de définir des modèles théoriques des objets auxquels nous pourrions comparer les modèles générés par l'agent. En revanche, dans le cas d'un agent évoluant dans un environnement continu, et plus encore, dans le cas d'un robot, l'étude des modèles ne pourra être que qualitative. Il sera en revanche possible, dans les deux cas, d'effectuer des comparaisons entre les modèles émergents des objets liés aux différentes interactions.

Étant donné la nature parallèle et bi-dimensionnelle (nos agents évoluent dans des environnements à deux dimensions) des informations spatiales que nous voulons étudier, nous ne pourrions pas nous baser sur les outils utilisés pour le mécanisme IMOSHEM pour afficher les traces de nos agents. Dans les chapitres 9 à 11, les traces et les structures apprises par les agents seront affichées d'une façon spécifique à chaque expérimentation, pour faciliter la lecture et l'analyse

des résultats.

3.2.2 Environnements de test

Les expérimentations ont été menées dans deux environnements de tests. Le premier est un environnement simulé. Le second est une plate-forme robotique, utilisée pour tester nos mécanismes dans un environnement bruité.

Pour les expérimentations en environnement simulé, nous nous sommes basés sur une version modifiée de l'environnement *Vacuum* [112]. Cette version permet les tests dans un environnement "continu", bien que l'agent puisse être restreint à une grille pour les expérimentations en environnement discret. Le robot est une plate-forme développée à partir de *Lego Mindstorm* pour implémenter le système interactionnel de l'agent. Le système de décision est identique à celui utilisé pour le système simulé. Le robot contient un programme lui permettant d'exécuter les interactions issues du système de décision, et constitue ainsi une interface entre le système de décision et l'environnement réel.

Chapitre 4

Modèle du Radical Interactionism (RI)

Notre étude se situe dans le contexte d'une approche interactionnaliste. Au cours de mon travail de thèse, nous avons abouti au modèle *Interactionnisme Radical* (ou *Radical Interactionism*, RI), que nous présentons dans ce chapitre. Le *Radical Interactionism* se présente comme une nouvelle approche pour modéliser l'interaction d'un agent avec son environnement. Cette approche se base sur l'utilisation de schémas sensorimoteurs, en accord avec les théories de la cognition qui stipulent que l'action et la perception ne sont pas dissociables (i.e. Piaget [105], O'Regan [96]).

4.1 Formalisation du modèle d'interaction conforme à l'approche Radical Interactionism

Selon l'approche *Radical Interactionism* (RI), le monde interactionnel d'un agent se compose d'un ensemble I d'interactions primitives i . Une *interaction primitive* est une association indissociable entre une action de l'agent et la perception qui résulte de l'interaction entre l'agent et son environnement, formant un schéma sensorimoteur. L'ensemble des interactions I forme ce que Buhrmann et al. appellent *l'environnement sensorimoteur* [17] de l'agent. La perception de l'environnement n'est donc pas passive, puisqu'elle implique une action volontaire de l'agent pour obtenir des informations sur l'environnement. À chaque interaction primitive correspond une *valeur de satisfaction* s_i , orientant le comportement de l'agent dans son processus interactionnel. Cette valeur de satisfaction caractérise les interactions que l'agent va "aimer" exécuter et celles qui lui sont préjudiciables. La motivation de l'agent ne dépend donc pas des états de l'environnement, mais de la valeur de satisfaction des interactions qu'il énonce, ce qui implémente une forme de *motivation intrinsèque*, que nous appelons *motivation interactionnelle*. Ce principe implémente une forme de *motivation autotélique* [117] : l'agent effectue une interaction sans autre but que la satisfaction qu'elle procure. Nous emploierons le terme *énoncer*, de l'anglais *to enact*, pour désigner la tentative d'exécution d'une interaction, en référence à la théorie de la *contingence sensorimotrice* [98].

Définitions :

Définition 1. *Interaction primitive* : une interaction primitive i est une association indissociable entre une action que l'agent peut effectuer sur l'environnement par le biais de ses actionneurs, et une perception de l'environnement qui résulte de l'action de l'agent sur son environnement. Par exemple, sentir par le toucher : pour être pertinente, l'information tactile ne doit pas être séparée du geste qui lui a donné naissance.

Définition 2. *Interaction intention* : interaction primitive sélectionnée par l'agent au début d'un cycle de décision.

Définition 3. *Interaction énoncée* : interaction primitive issue du couplage entre l'agent et son environnement, effectivement obtenue à la fin d'un cycle de décision.

Le principe du modèle est le suivant : au début d'un cycle de décision t , l'agent sélectionne l'interaction de l'ensemble I qu'il veut effectuer, que l'on appelle interaction intention i_t . À la fin du cycle t , il constate l'interaction qui a réellement eu lieu, que l'on appelle interaction énoncée e_t . Si l'interaction énoncée est bien l'interaction intention, c'est-à-dire si $i_t = e_t$, alors l'énoncé de i_t est un succès. Dans le cas contraire, l'énoncé est un échec. Le cycle de décision est illustré par la Figure 4.1. Le système de sélection apprend à prédire le résultat de l'énoncé des interactions, de façon à ne sélectionner que les interactions qui devraient être énoncées avec succès.

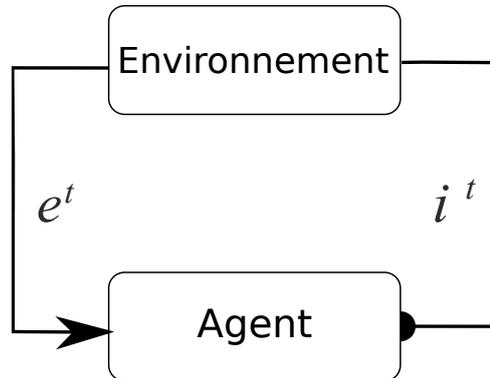


FIGURE 4.1 – Formalisation du Radical Interactionism (RI). Au cycle de décision t , l'agent sélectionne et teste une interaction i_t . Il constate l'interaction qui a réellement été énoncée e_t . Le cycle de décision est un succès si $i_t = e_t$ et un échec sinon. L'agent capitalise la satisfaction associée à l'énoncé de e_t .

Dans le cas d'un échec, l'interaction e_t sera considérée comme une interaction *alternative* de i_t . Par exemple, si l'agent tente d'énoncer l'interaction *avancer d'un pas*, mais que l'interaction *se cogner* est effectivement énoncée, alors l'interaction *se cogner* est une interaction alternative de *avancer d'un pas*. On définit les interactions *alternatives* et le *groupe alternatif* \mathcal{A}_i d'une interaction i comme :

Définition 4. *Interaction alternative* : une interaction j est dite alternative de l'interaction i si énoncer i peut conduire à l'énoncé effective de j .

Définition 5. *Groupe alternatif* : le groupe alternatif \mathcal{A}_i d'une interaction i est l'ensemble des interactions alternatives de i .

Le groupe alternatif peut se formaliser par :

$$\mathcal{A}_i = \{j_k\} / \forall k, i \neq j_k, \{\exists t \in \mathbb{N} / (i = i_t) \wedge (j_k = e_t)\} \quad (4.1)$$

Par définition, le groupe alternatif d'une interaction est appris et complété par l'agent à chaque fois que l'énaction de cette interaction échoue. Notons que le succès d'une interaction i implique l'échec de ses interactions alternatives, puisque celles-ci n'ont pas été énoncées à la place de i . Il est également possible que deux interactions soient mutuellement alternatives. Dans ce cas, on parlera d'interactions *opposées* :

Définition 6. *Interactions opposées* : deux interactions i_1 et i_2 sont dites opposées si et seulement si $i_1 \in \mathcal{A}_{i_2}$ et $i_2 \in \mathcal{A}_{i_1}$.

Cette modélisation basée sur les interactions présente trois différences principales par rapport aux modèles Markoviens traditionnels :

- le système se base sur un ensemble d'interactions primitives plutôt que sur des actions et des interactions séparées. En effet, conformément aux théories de Piaget, la séparation entre l'agent et l'environnement, et entre l'action et la perception n'est pas définie *a priori*, mais apprise au travers de l'énaction des interactions.
- le cycle de décision commence par l'interaction intention plutôt qu'avec la perception de l'environnement. Ainsi, la perception est une conséquence de l'action, et non une perception de l'état du monde avant l'action de l'agent [103]. L'agent "perçoit" son environnement de façon active, en l'expérimentant par le biais de ses interactions.
- la motivation interactionnelle de l'agent se base sur la valeur de satisfaction des interactions, que l'agent expérimente lorsqu'il énonce une interaction, et non sur les états de l'environnement. Elle se distingue ainsi de la notion de récompense intrinsèque (e.g. [115]), issue de l'apprentissage par renforcement. Le mécanisme de sélection ne dépend ainsi que des interactions possibles avec l'environnement, et non de l'environnement.

4.2 Évolution de la formalisation du Radical Interactionism

La formalisation du modèle RI s'est raffinée tout au long de ma thèse. Je présente ici les principales étapes qui ont conduit de nos premières formalisations (Georgeon et al. [47]) jusqu'au modèle RI tel que formalisé précédemment.

4.2.1 Agnostic Agent Decision Process (AADP)

Cette formalisation s'inspire des *Partially Observable Markov Decision Process* (POMDP) [7] [68], et considère les actions et les perceptions séparément. Cependant, les caractéristiques du modèle RI sont déjà présentes : l'agent sélectionne, au début d'un cycle de décision, un couple

(action, perception), bien que seule l'action se répercute dans l'environnement. La perception est utilisée pour construire le couple (action, perception) réellement observé. La valeur de satisfaction est déterminée par l'action effectuée et par la perception qui en résulte. On peut cependant noter que dans ce modèle, l'agent ne débute pas avec un ensemble d'interactions, mais avec un ensemble d'actions. Lorsque l'agent effectue une action, la perception qui en résulte lui permet d'apprendre une interaction. L'utilisation des interactions remplace peu à peu l'utilisation des actions, quand les schémas appris par l'agent lui permettent d'organiser des comportements lui permettant de satisfaire sa motivation interactionnelle. Cette formalisation est illustrée par la Figure 4.2.

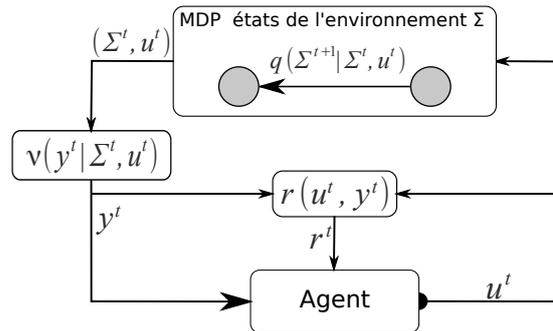


FIGURE 4.2 – Formalisation du Agnostic Agent Decision Process (AADP). Au début d'un cycle de décision t , l'agent sélectionne une interaction primitive intention $i_t = (u_t, y_t)$, avec u et y respectivement l'action et la perception qui composent i . Si l'agent ne connaît pas d'interactions, il peut sélectionner une action u_t seule pour apprendre les perceptions qui peuvent en résulter. L'interaction éactée est $e_t = (u_t, y'_t)$ où u_t est l'action réalisée, issue de l'action intention $i_t = (u_t, y_t)$, et y'_t la perception effectivement observée. Le cycle de décision est un succès si $i_t = e_t$, et un échec sinon. À la fin du cycle, l'agent reçoit la valeur de satisfaction associée à l'interaction éactée (u_t, y_t) .

Pour mettre en évidence les interactions, nous avons modélisé une interface afin d'isoler le mécanisme de décision de l'agent, basé sur des interactions, du système actionneurs/capteurs de l'agent, qui utilise des actions et des perceptions (Figure 4.3). Cette interface permet la séparation de l'action constituant l'interaction intention et reconstitue l'interaction éactée avec la perception qui en résulte. L'interface constitue ainsi le "corps" de l'agent (réel ou simulé), qui agit et perçoit l'environnement, indépendamment du système de décision. Cette formalisation a l'avantage de refléter une implémentation complète du AADP dans un système "réel" en associant de manière séparée le mécanisme de décision et le système physique servant d'interface avec l'environnement. La valeur de satisfaction est désormais portée par l'interaction éactée e_t .

4.2.2 Enactive Markov Decision Process

Le corps de l'agent peut être considéré comme une partie du système agent-environnement. Il est donc possible d'intégrer l'interface du AADP dans l'environnement. La formalisation obtenue

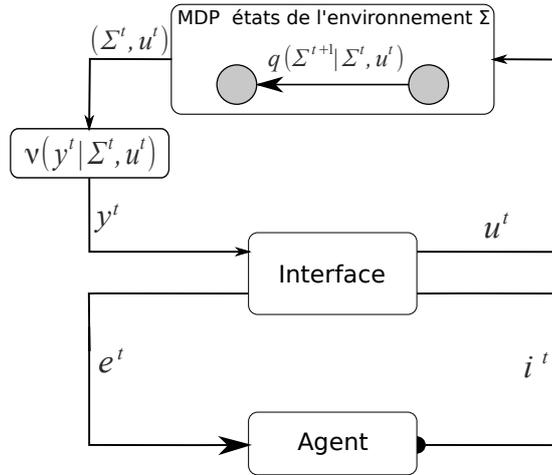


FIGURE 4.3 – Seconde formalisation du AADP. Une interface connecte le système décisionnel de l'agent, basé sur les interactions, et le système "physique" (le *corps* de l'agent situé dans l'environnement), constitué d'actionneurs et de capteurs. Au début du cycle de décision t , l'agent sélectionne une interaction intention $i_t = (u_t, y_t)$. L'interface transmet l'action u_t constituant i_t pour commander les actionneurs. Elle construit, à la fin du cycle de décision, l'interaction énoncée $e_t = (u_t, y_t)$ en tenant compte de l'observation y_t . Notons la disparition de la valeur de satisfaction, celle-ci étant portée par e_t .

(Figure 4.4) fait disparaître toute référence aux actions et aux perceptions de la formalisation. Le modèle ainsi défini a été appelé *Interactional Motivation Decision Process* (IMDP), en raison de sa motivation interactionnelle, puis *Enactive Markov Decision Process* (EMDP) [48].

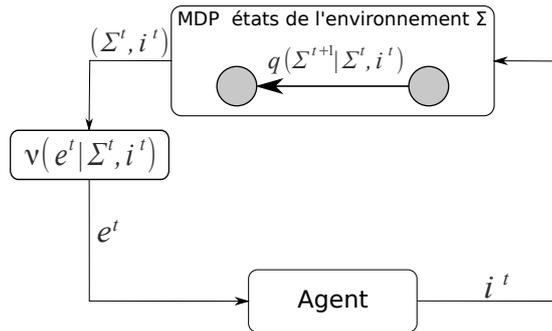


FIGURE 4.4 – Formalisation du Enactive Markov Decision Process (EMDP).

Cependant, comme le modèle ainsi formalisé ne se base pas sur les états de l'environnement, il n'est plus pertinent de mentionner le modèle MDP pour décrire l'environnement. Le système de décision de l'agent est en effet indépendant de l'environnement, puisqu'il n'a pas accès aux états de l'environnement. En marquant cette indépendance vis-à-vis de l'environnement, le modèle

respecte le principe d'*agnosticisme environnemental*. L'agent interagit avec l'environnement de façon transparente par le biais des interactions énoncées. La formalisation ainsi épurée du EMDP définit le modèle du *Radical Interactionism* tel que décrit précédemment (Figure 4.1).

Le modèle *Radical Interactionism* a été testé avec succès dans des environnements offrant des régularités séquentielles (cf. chapitre 1). Cependant, dans un environnement ouvert offrant des régularités spatiales, un agent doté d'un système de décision basé sur le modèle RI ne parvient pas à organiser efficacement ses comportements en tenant compte de ces régularités. Il ne peut pas, par exemple, se rendre compte que deux séquences d'interactions différentes, comme tourner trois fois à gauche de 90°, et tourner une fois à droite de 90°, peuvent conduire au même état spatial du système agent-environnement. Il ne peut également pas suivre dans l'espace un élément qui échappe à son système sensoriel [42]. Ces limitations nous ont emmené à proposer des variations du modèle RI.

4.3 Variations du modèle RI

Nos investigations sur la modélisation de l'espace nous ont conduits à proposer des variantes du modèle RI. Nous proposons deux types de variantes : les premières ajoutent des informations spatiales aux interactions énoncées et permettent le suivi des interactions précédemment énoncées. Ces variantes se basent sur des heuristiques inspirées des êtres vivants, qui disposent de structures cérébrales leur permettant d'intégrer l'espace dans des structures cognitives. L'apprentissage n'est donc plus seulement ontogénétique, mais partiellement phylogénétique, puisque l'agent dispose de structures que l'on pourrait supposer construites au cours de l'évolution. Le second type de variante n'ajoute pas d'heuristique, mais permet la prise en compte d'interactions énoncées en "parallèle". Ces variantes s'appuient sur l'hypothèse qu'un être vivant, lorsqu'il interagit avec son environnement, reçoit un grand nombre de stimuli sensoriels indépendants de l'interaction qui l'aident à caractériser sa situation, et qu'en conséquence, un grand nombre d'interactions peuvent être énoncées simultanément.

4.3.1 Les variantes spatiales : Directional RI, Spatial RI (DRI, SRI)

Nous proposons deux variantes spatiales, le *Directional RI* (DRI) [40] et le *Spatial RI* (SRI). Ces variantes ajoutent des informations spatiales sur les interactions énoncées et permettent de suivre les interactions précédemment énoncées, dans le but d'aider l'agent à construire un contexte spatial exploitable pour générer des comportements dans un espace ouvert. Ces versions reposent sur une hypothèse que nous appelons "hypothèse de l'espace" :

Hypothèse 1. *Hypothèse de l'espace : l'agent interagit avec un environnement doté d'une structure spatiale cartésienne à N dimensions. Les propriétés géométriques de cet environnement sont invariantes en tout point de l'espace.*

L'idée, derrière ces variantes du modèle RI, est qu'un agent navigant dans l'espace dispose d'organes sensoriels lui permettant d'une part de localiser l'origine des interactions énoncées dans

l'espace, lorsqu'elles sont liées à la présence d'un élément situé à une certaine position de l'environnement, et d'autre part, d'estimer ses déplacements dans l'espace. Les êtres vivants disposent en effet de nombreux sens leur permettant de définir la position d'un élément dans l'espace, ainsi que leurs propres déplacements. On pense notamment à la convergence visuelle, ou au flux optique pour la première hypothèse, et au système vestibulaire des vertébrés pour la seconde. Certains êtres vivants disposent également de structures cérébrales pré-câblées qui maintiennent une correspondance avec la structure de l'espace en référentiel égocentré. Les vertébrés, notamment, disposent d'une structure qui génère une correspondance avec l'espace visuel, appelée *tectum* (également appelé *colliculus* chez les mammifères) [95].

Le modèle *Directional RI* (DRI, Figure 4.5.a) ajoute des informations sur la direction de l'interaction énéactée (par rapport à l'agent), et sur le mouvement angulaire produit par l'éneaction de l'interaction énéactée en référentiel égocentré. On peut noter que le modèle DRI ne fournit pas la notion de distance et ne permet pas de mesurer le rapprochement des objets. Le modèle *Spatial RI* (SRI, Figure 4.5.b) donne, quant à lui, des informations sur la position de l'interaction énéactée dans l'espace et le déplacement produit par l'éneaction de l'interaction énéactée, en référentiel égocentré.

Formellement, les modèles DRI et SRI sont très proches du modèle RI : à la fin du cycle de décision t , l'agent constate l'interaction énéactée e_t , mais également la position p_t de e_t dans l'espace (c'est-à-dire son angle ou ses coordonnées dans l'espace), et le mouvement angulaire θ_t (dans le cas d'un DRI) ou le mouvement τ_t (dans le cas d'un SRI) de l'environnement par rapport à l'agent (en référentiel égocentré), produit par l'éneaction de e_t .

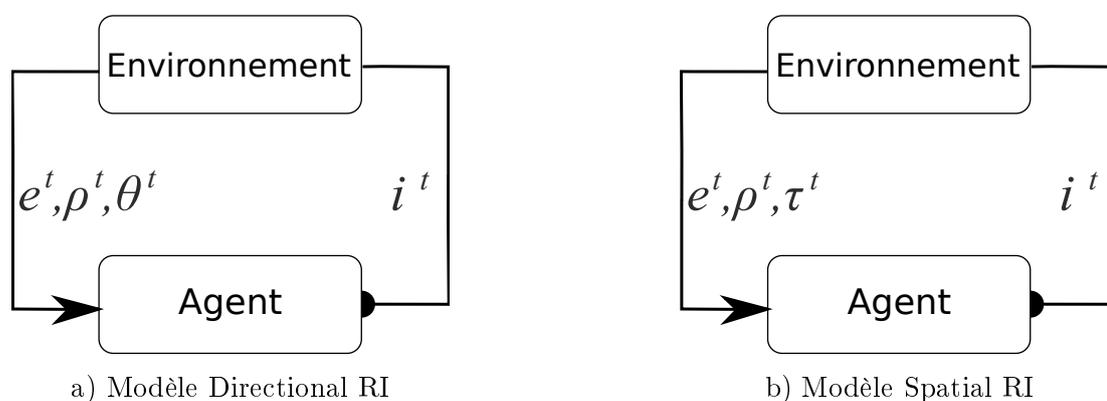


FIGURE 4.5 – À gauche, le Directional RI, à droite le Spatial RI. Dans ces deux systèmes, l'agent est informé de la position p où l'interaction énéactée e_t est localisée dans l'espace, sous la forme d'une position angulaire dans le cas du DRI et sous la forme d'une position de l'espace dans le cas du SRI. Dans le cas du DRI, l'agent reçoit également l'angle de rotation θ de l'agent. Dans le cas du SRI, l'agent est informé de la transformation spatiale τ provoquée par l'éneaction de e_t .

Les systèmes DRI et SRI intègrent également une structure permettant de suivre et localiser les interactions énéactées dans l'espace. Cette structure est appelée *Mémoire Spatiale*.

Définition 7. *Mémoire spatiale* : la mémoire spatiale est une structure permettant de stocker, suivre et localiser les interactions énoncées aux cycles de décision précédents, dans un référentiel égocentré.

4.3.2 Formalisation du SRI

Je propose une formalisation du SRI aussi générale que possible. Cette formalisation s'applique également au modèle DRI, en remplaçant la transformation spatiale τ par un mouvement angulaire θ .

On définit l'ensemble P des positions p de l'espace dans lesquelles une interaction peut être énoncées. On définit l'ensemble P' des positions p de l'espace intégrable par la mémoire spatiale avec $P \subset P'$. Nous appelons *acte* une interaction placée dans l'espace :

Définition 8. *Acte* : un acte a est une interaction localisée dans l'espace. L'acte est défini par un couple (i, p) avec $i \in I$ une interaction primitive et $p \in P'$ la position à laquelle i est localisée.

L'ensemble des actes que l'agent peut énoncer est défini par l'ensemble $A \subseteq I \times P$. L'ensemble des actes pouvant être stockés par la mémoire spatiale est défini par l'ensemble $A' \subseteq I \times P'$.

La mémoire spatiale peut être formalisée comme une liste de couples (a, o) de $A' \times \mathbb{N}$, avec $a = (i, p)$ un acte énoncé à la fin d'un précédent cycle de décision, p la position courante de a et o le nombre de cycles de décision effectués depuis l'énoncé de a . Les éléments sont retirés de la mémoire spatiale après un certain nombre de cycles (o supérieur à un certain seuil). Notons que dans un environnement bruité, l'âge o d'une interaction énoncée permet de définir le niveau de fiabilité d'un élément de la mémoire spatiale.

Une transformation τ_t renseigne l'agent sur le mouvement résultant de l'énoncé d'une interaction (une translation et/ou une rotation, par exemple). Cette information est utilisée pour définir les mouvements des éléments de la mémoire spatiale. On note T l'ensemble des transformations géométriques que l'agent peut observer. Une transformation τ peut se formaliser sous la forme d'une fonction de transformation de la mémoire spatiale, définie par :

$$\begin{aligned} \tau : P' &\rightarrow P' \cup \{\emptyset\} \\ p &\mapsto p' \vee \emptyset \end{aligned}$$

Ainsi, mettre à jour les éléments de la mémoire spatiale revient à mettre à jour les positions de ces éléments en utilisant le mouvement observé. Notons que la mise à jour d'une position peut mener à une position en dehors de l'espace intégré par la mémoire spatiale. On note \emptyset une position non définie par la mémoire spatiale.

4.3.3 Les variantes parallèles

Mes investigations sur les systèmes de décision basés sur le modèle RI en environnement ouvert m'ont conduit à penser qu'une seule interaction énoncée n'était pas suffisante pour permettre

à un agent de construire un contexte spatial exploitable de son environnement. J'en ai donc conclu qu'il fallait tenir compte de plusieurs interactions énoncées en parallèle. D'un point de vue biologique, cette hypothèse semble valide : les êtres vivants reçoivent en effet de nombreux stimuli sensoriels de façon simultanée. Il est donc raisonnable de penser que des interactions puissent être énoncées simultanément. Cette parallélisation permet d'échapper à l'explosion combinatoire de l'ensemble d'interaction I que l'on obtiendrait si l'on devait attribuer une interaction pour chaque combinaison de perceptions sensorielles.

Formellement, le modèle Parallel RI (PRI) est très proche du RI : il est composé d'un ensemble I d'interactions primitives i . Le principe est qu'au début d'un cycle de décision t , l'agent tente d'énoncer une interaction intention i_t . L'énonciation de cette interaction conduit à l'énonciation d'un ensemble d'interactions primitives énoncées $\{e_k\}_t$. Nous appellerons cet ensemble, *ensemble énoncé* E_t . La satisfaction liée à l'énonciation de l'ensemble E_t est définie comme la somme des valeurs de satisfactions des interactions énoncées. Le cycle de décision est illustré par la Figure 4.6.

Définition 9. *Ensemble énoncé : on appelle Ensemble énoncé E_t l'ensemble des interactions énoncées à la fin du cycle de décision t .*

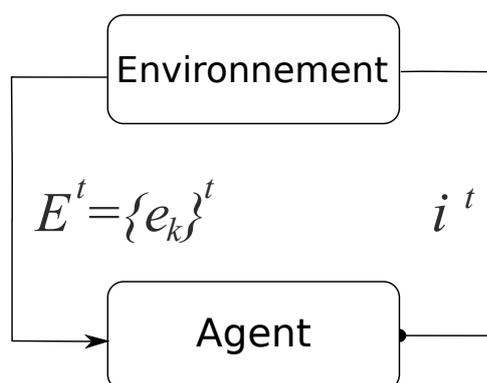


FIGURE 4.6 – Modèle du Parallel RI. Ce modèle est similaire au modèle RI. Cependant, l'environnement retourne un ensemble d'interactions énoncées, appelé ensemble énoncé E_t . La satisfaction de l'agent est la somme des valeurs de satisfaction des interactions énoncées.

Il faut considérer le fait que les interactions de l'ensemble énoncé sont toutes issues de l'énonciation d'une même interaction intention. Je pars donc du principe que certaines interactions sont des conséquences de l'énonciation d'une autre interaction. Dans le modèle que nous avons étudié, nous nous sommes limités à des interactions ne pouvant être la conséquence que d'au plus une interaction. Les interactions qui dépendent d'une autre interaction sont appelées *interactions secondaires*, et ajoutent des informations à une *interaction primaire* dont elles dépendent.

Définition 10. *Interaction primaire : une interaction primaire est une interaction qui peut être énoncée seule.*

Définition 11. *Interaction secondaire : une interaction secondaire est une interaction qui ne*

peut être énoncée que dans le cas où l'interaction primaire dont elle dépend est énoncée avec succès. Cette interaction primaire est appelée interaction associée.

Les interactions secondaires peuvent être considérées comme des stimuli générés par le système sensoriel de l'agent lorsque celui-ci énonce une interaction, qui ajoutent des informations à l'interaction énoncée. Une interaction secondaire ne peut pas être considérée comme une simple perception, car le stimulus qui la caractérise est une conséquence de l'énoncé de l'interaction primaire énoncée. On peut considérer que les interactions secondaires sont des interactions opportunistes, qui peuvent être énoncées de façon non intentionnelle. Un ensemble énoncé ne contient qu'une seule interaction primaire : celle qui a donné naissance aux interactions secondaires énoncées. Énoncer une interaction secondaire revient à énoncer son interaction primaire associée. En effet, l'énoncé d'une interaction secondaire repose sur l'énoncé de son interaction primaire associée.

Le type d'interaction (primaire ou secondaire) ainsi que les associations peuvent être appris de la façon suivante :

Une interaction i est primaire si $\forall t \in \mathbb{N}, \cap_t \{E_t/i \in E_t\} = i$, c'est-à-dire si l'énoncé de i n'est pas lié à l'énoncé d'une autre interaction.

Une interaction i est secondaire si $\forall t \in \mathbb{N}, (\cap_t \{E_t/i \in E_t\}) - \{i\} \neq \emptyset$, c'est-à-dire que tout contexte interactionnel E_t contenant i contient également une autre interaction différente de i . Les interactions associées d'une interaction secondaire i sont définies par l'ensemble :

$(\cap_t \{E_t/i \in E_t\}) - \{i\}$. Tant que cet ensemble contient plus d'un élément, on considérera que cette interaction est associée à l'énoncé simultanée des interactions de cet ensemble. Notons qu'une interaction est considérée comme secondaire jusqu'à ce que la première condition soit vraie. Cependant, pour permettre l'apprentissage des interactions associées, on considérera qu'une interaction ne satisfaisant pas la première condition est une interaction primaire si elle est mutuellement associée avec toutes les interactions de l'ensemble $(\cap_t \{E_t/i \in E_t\}) - \{i\}$.

Il faut étendre la définition des interactions alternatives pour tenir compte des interactions secondaires. Nous proposons ainsi la définition étendue :

Définition 12. *Interaction alternative (définition étendue au PRI) : une interaction alternative j est dite alternative de l'interaction i si énoncer i peut conduire à l'énoncé effective de j , et si i et j n'ont jamais été énoncées simultanément.*

Notons que cette définition est valable pour le modèle RI. La formalisation du groupe alternatif A_i d'une interaction, définie dans l'équation (4.1), est également étendue pour tenir compte des interactions secondaires :

$$A_i = \{j_k\} / \forall k, i \neq j_k, (\exists t \in \mathbb{N} / (i = i_t) \wedge (i \notin E_t) \wedge (j_k \in E_t)) \wedge (\nexists t \in \mathbb{N} / (i \in E_t) \wedge (j_k \in E_t)) \quad (4.2)$$

La définition du résultat de l'énoncé d'une interaction est également modifiée : le résultat ne dépend plus de l'interaction intention i_t , mais de la présence ou l'absence d'interactions dans

l'ensemble E_t . Il faut en effet tenir compte du fait qu'une interaction secondaire ne peut être énoncée que si l'interaction primaire associée est énoncée avec succès. Nous n'attribuons plus un résultat au cycle de décision, mais à chaque interaction de l'ensemble I . Le résultat d'une interaction est défini par :

- Une interaction i est énoncée avec succès au pas t si $i \in E_t$.
- Une interaction i est un échec si $i \notin E_t$, et il existe une interaction i' opposée (cf. définition 6) à i telle que $i' \in E_t$ ou si l'interaction associée à i est énoncée.
- Dans tout autre cas, le résultat n'est pas défini.

Il est bien entendu possible de combiner les modèles spatiaux et parallèles. Un modèle spatial et parallèle est appelé Parallel Spatial Radical Interactionism (PSRI). Dans ce modèle, l'ensemble énoncé est constitué des actes énoncés : chaque interaction énoncée est localisée à la position où elle a été observée. Notons également que plusieurs instances d'une même interaction peuvent être observées à des positions différentes.

La première expérimentation, décrite au chapitre 9, utilise un système implémentant le modèle RI. La seconde expérimentation (Chapitre 10) utilise un système basé sur le modèle PSRI. L'utilisation d'un environnement ouvert a en effet nécessité l'utilisation d'interactions visuelles secondaires. Les heuristiques spatiales permettent quant à elles, de se focaliser sur l'aspect exploitation de la mémoire spatiale plutôt que sur sa construction. La troisième expérimentation (Chapitre 11) se focalise sur la construction d'une mémoire spatiale sans se baser sur des préconceptions, et sera ainsi basée sur le modèle PRI. Pour des raisons de simplifications, le système de test utilise certains aspects du modèle PSRI.

Chapitre 5

Construction de structures pour la prise de connaissance de l'environnement

Nous proposons un modèle permettant à un agent *environnementalement agnostique* de prendre connaissance des éléments constituant son environnement, en donnant un sens à ces éléments. Cette prise de connaissance se matérialise par la construction d'une structure permettant de caractériser implicitement l'environnement de l'agent ainsi que sa structure spatiale, et de lui permettre de générer des comportements adaptés au contexte pour satisfaire au mieux sa motivation interactionnelle.

Nous décomposons le problème général en trois sous-problèmes distincts, liés à trois sous-mécanismes du modèle général, que nous développerons séparément. Ces sous-problèmes correspondent à trois mécanismes nécessaires pour fournir à un agent un contexte exploitable lui permettant de définir le ou les comportements les plus adaptés pour satisfaire sa motivation interactionnelle à court et moyen terme :

1. mécanisme de construction d'une structure de connaissance des constituants de l'environnement, c'est-à-dire la façon dont un agent définit et apprend à reconnaître les éléments de son environnement en termes de possibilité d'interaction. Ce mécanisme est appelé *mécanisme de construction de structures pour la prise de connaissance de l'environnement*.

2. mécanisme de construction d'une structure intégrant l'environnement. Cette structure fournit à l'agent un contexte exploitable contenant des informations complémentaires au retour interactionnel issu du couplage agent-environnement. Ce mécanisme est appelé *mécanisme de construction de la mémoire spatiale*.

3. mécanisme d'exploitation de la structure construite par le second mécanisme, c'est-à-dire un mécanisme permettant d'extraire des informations exploitables des premier et second mécanismes pour générer des comportements conformes à la motivation intrinsèque de l'agent. Ce mécanisme est appelé *mécanisme d'exploitation de la mémoire spatiale*.

Il faut noter que ces mécanismes sont interdépendants. En effet, notre but est de permettre à

un agent d'énacter des comportements, et non de générer un modèle explicite de l'environnement. Il faut donc chercher, pour chaque mécanisme, les informations minimales à fournir aux autres mécanismes. Les contraintes liées à chaque mécanisme sont ainsi influencées par les informations fournies par les autres. La Figure 5.1 montre les connexions entre les différents mécanismes dans le modèle que nous proposons.

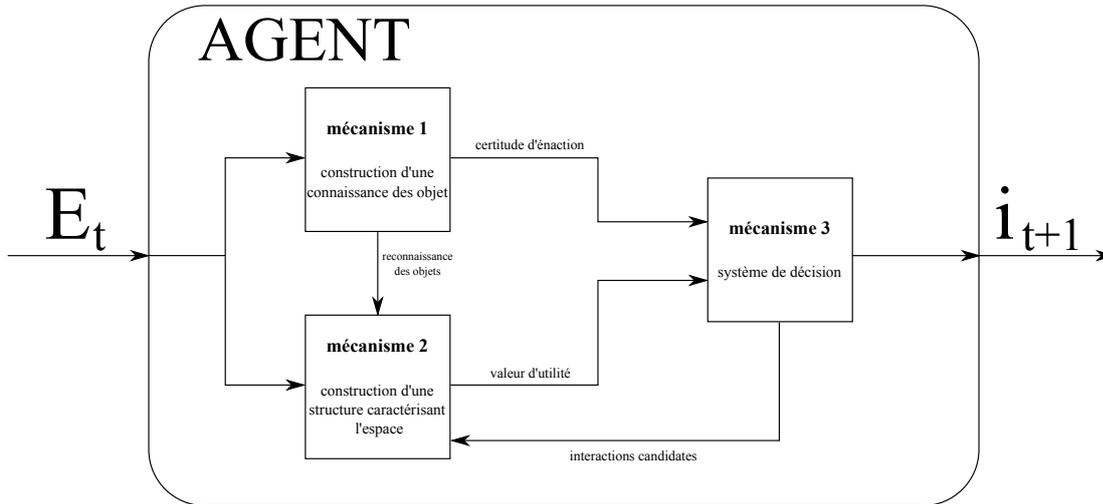


FIGURE 5.1 – Connexions entre les différents mécanismes, dans le modèle PRI (Parallel Radical Interactionism). À la fin du cycle de décision t , l'agent reçoit l'ensemble énéacté E_t . Cet ensemble est utilisé pour intégrer une connaissance des objets que l'agent peut expérimenter (mécanisme 1), et pour compléter et mettre à jour une structure reflétant l'espace environnant, construite et maintenue par le mécanisme 2. Notons que dans le cas du modèle PSRI, cette structure est partiellement ou totalement codée en dur. Le système de décision (mécanisme 3), reçoit du mécanisme 1 les interactions qu'il peut énéacter dans le contexte actuel. Les interactions candidates sont envoyées vers le mécanisme 2 qui détermine une fonction d'utilité pour chaque candidate, en fonction de leurs effets sur l'environnement. Le système de décision désigne ensuite l'interaction intention i_{t+1} du cycle de décision suivant.

Les différents mécanismes ont été mis en œuvre et testés au cours de trois expérimentations. La première met en évidence le fonctionnement des mécanismes 1 et 3 sans utiliser le mécanisme 2. Le système de décision défini par le mécanisme 3 permet une gestion de l'espace limité à l'espace proche de l'agent. La seconde expérimentation utilise une version codée en dur du mécanisme 2 afin d'analyser plus spécifiquement le fonctionnement du mécanisme 3 lorsqu'il peut exploiter les informations fournies par une structure construite par le mécanisme 2. Nous utilisons un mécanisme de décision différent de celui utilisé dans la première expérimentation afin d'étudier plus spécifiquement comment le mécanisme 3 peut exploiter les informations issues du mécanisme 2. Cependant, nous verrons que ces deux versions du mécanisme 3 ne sont pas incompatibles. La troisième expérimentation met en évidence la construction d'une structure reflétant l'environnement qui entoure l'agent, basée sur un minimum de préconceptions, et les

propriétés qui émergent de cette structure. Nous réutiliserons les informations issues du mécanisme 1, apprises au cours de la seconde expérimentation, et une version modifiée du mécanisme 3, adaptée au système implémentant le mécanisme 2.

5.1 Construction d'une connaissance de l'environnement

Un agent basé sur le principe du *Radical Interactionism* (RI) n'expérimente son environnement qu'en termes d'interactions. Les structures de connaissances de l'environnement construites par l'agent ne pourront donc se baser que sur des interactions. Je propose un mécanisme permettant à un agent de construire sa propre "compréhension" des objets composant son environnement à partir de ses interactions. Nous employons ici le terme *objet* pour définir des structures matérielles ou des propriétés physiques définies par les interactions qu'elles permettent, que l'agent doit apprendre à détecter et à reconnaître, et auxquelles il doit attribuer un sens par la façon dont il peut interagir avec ces objets.

5.2 Inspiration biologique et principes du mécanisme de construction de structures pour la prise de connaissance de l'environnement

Les principes de conception de ce mécanisme sont inspirés par des observations biologiques publiées dans la littérature, mais également par des théories de la cognition.

5.2.1 Représentation des objets dans l'aire F5 du singe

Une expérience a été menée par Murata et al. [88] sur des singes, dans le but d'étudier comment les neurones du cortex prémoteur ventral, et plus précisément de la région F5 (Figure 5.2), réagissent lorsque le singe est mis en présence d'objets de différentes formes. La région F5 est connue pour son implication dans le contrôle des mouvements de la main et du visage [111][74]. Cette région contient également les neurones miroirs qui s'activent quand l'animal effectue un geste, mais également quand il voit un autre animal ou un expérimentateur effectuer ce geste [110]. Durant cette expérience, des objets de différentes formes sont présentés consécutivement à un singe, pendant que l'activité de certains de ses neurones de la région F5 est enregistrée. Différents protocoles sont mis en œuvre : dans le premier, le singe doit appuyer sur un bouton pour illuminer l'objet, puis saisir l'objet. Dans le second, la lumière qui éclaire l'objet s'éteint et le singe doit à nouveau saisir l'objet dans le noir. Dans le troisième, le singe doit seulement fixer l'objet.

Les observations montrent que les neurones de cette aire sont liés à la façon de saisir un objet. En effet, certains neurones de cette région ne s'activent que lorsqu'un objet d'une certaine forme et d'une certaine taille est présenté à l'animal. Les objets testés pouvaient être regroupés en plusieurs groupes activant un même neurone : le cube, le cylindre et le cône formant le premier

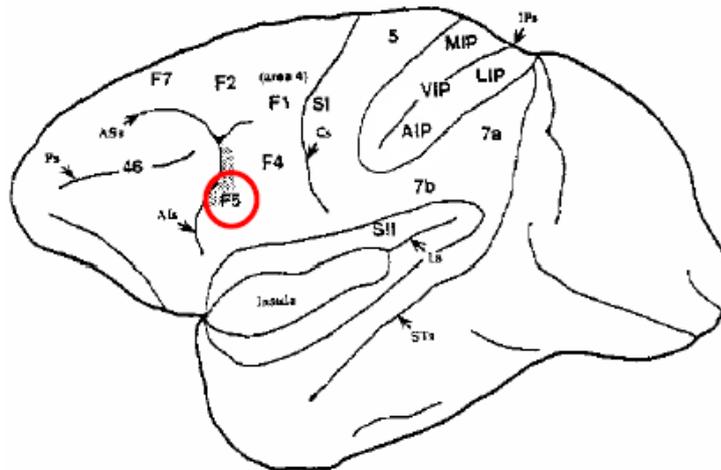


FIGURE 5.2 – Aire cérébrale F5 dans le cerveau du singe. Les neurones de cette aire s'activent lorsque l'animal effectue un geste vers un but précis, où lorsqu'un autre singe ou un expérimentateur effectue ce même geste. Adapté de [15].

groupe, la sphère un second, et le tore un troisième. Ceci montre que les neurones sont sensibles à la façon de saisir l'objet, liés à sa forme globale et sa taille, plutôt qu'à sa forme exacte (la sphère, par exemple, doit être saisie de façon à l'empêcher de rouler). Autre particularité de ces neurones, ils s'activent quand on présente un objet et ce, que le singe effectue le geste ou se contente de fixer l'objet. Notons aussi que le neurone reste actif même lorsque les lumières s'éteignent, et ne se désactive que lorsque l'animal peut constater l'absence de l'objet. Nous pouvons donc supposer que les objets sont notamment représentés par les interactions qui permettent d'interagir avec lui, ici la façon de saisir un objet. On peut également supposer que les objets sont mémorisés en relation avec le mouvement qui permet d'interagir avec eux.

Une autre expérience, menée par Gallese et al. [39], consistait à mesurer l'activité des neurones miroirs de la région F5 du cerveau de singes, lorsque l'animal effectue un geste pour saisir un objet et lorsqu'il observe l'expérimentateur saisir l'objet avec un même geste. Ces études montrent que les neurones miroirs s'activent de façon similaire si le singe effectue le geste de préhension ou si ce geste est effectué par l'expérimentateur. Elles montrent également que la décharge du neurone est beaucoup plus faible si l'expérimentateur effectue le geste en l'absence de l'objet ou si l'expérimentateur déplace l'objet avec une tige. Ceci pourrait indiquer que les neurones miroirs pourraient réagir lorsque l'agent perçoit non pas le mouvement d'un autre agent, mais l'interaction entre cet autre agent avec l'environnement. Cette reconnaissance peut se baser sur la reconnaissance du contexte permettant d'effectuer le mouvement : dans l'expérience décrite précédemment, le contexte permettant la préhension d'un objet nécessite la présence d'un objet dont la forme permette la préhension, et la présence de la main d'un agent (animal ou expérimentateur) à proximité de l'objet. L'absence de l'un ou l'autre de ces éléments provoque une réponse affaiblie, voire inexistante, de la part des neurones miroirs. L'utilisation d'un mécanisme de reconnaissance du contexte évite de devoir reconnaître un autre agent et le geste qu'il effectue,

mais se base uniquement sur la configuration spatiale qui permettrait d'effectuer le mouvement.

De ces observations, nous choisissons d'extraire les principes suivants :

- les objets peuvent être définis par l'interaction que l'agent peut effectuer avec eux. Cette propriété permet de réduire la complexité du monde expérimenté par l'agent : en effet, en ne considérant que les interactions qui peuvent être énoncées, on n'a pas besoin de représenter tous les types d'objets qui composent l'environnement.

- la possibilité d'énoncer une interaction implique la présence de l'objet qui permet d'énoncer cette interaction. Tout comme les neurones de la région F5 vus précédemment, le fait de garder active l'interaction jusqu'à ce qu'on puisse constater la disparition de l'objet permet de mémoriser la présence de cet objet, même lorsqu'un agent ne peut plus le percevoir. Nous définissons ainsi un effet mémoire qui ne nécessite pas de conserver un modèle de l'objet.

- Les objets distants peuvent être reconnus en reconnaissant la configuration qui permet d'énoncer une interaction dans l'espace. La notion d'objet, telle que définie ci-dessus, peut devenir indépendante de sa position si l'on ne cherche pas à énoncer l'interaction qu'il *afforde* (voir ci-dessous). Cette propriété sera utilisée par les mécanismes de construction (Chapitre 6) et d'exploitation (Chapitre 7) de la mémoire spatiale pour reconnaître des objets distants de la position permettant d'énoncer les interactions affordées.

5.2.2 Théorie des *Affordances*

Nous pouvons mettre en parallèle les expériences décrites précédemment avec la *théorie des affordances* de J.J. Gibson [53]. Une *affordance* est définie comme une possibilité d'interaction proposée par l'environnement à un agent. En effet, dans ces expériences, chaque objet révèle qu'il peut être saisi par un certain mouvement, et, de ce fait, *afforde* ce mouvement.

Les premières formalisations des affordances proposaient qu'un objet afforde une action indépendamment de l'agent [127]. Cependant, si un agent ne peut pas effectuer une action, alors il lui est inutile de définir cette affordance.

Notre modèle se rapproche donc plutôt des définitions sur les affordances données par Stoffregen [119] et Chemero [22], pour qui une affordance est définie par le couplage agent-environnement, plutôt qu'une propriété de l'agent ou de l'environnement seul. Nous emploierons le terme *affordance* pour caractériser une possibilité d'énonciation d'une interaction offerte à l'agent par l'environnement.

Dans notre approche, les interactions et les affordances sont étroitement liées. Cependant, nous considérons que c'est l'interaction qui définit l'affordance, et non l'affordance qui propose une interaction. En effet, la structure de connaissance d'un agent basé sur le modèle RI débute par un ensemble d'interactions définies *a priori*. Les affordances, ainsi que la capacité de l'agent à les détecter, reposent sur cet ensemble initial d'interactions.

Le fait de se baser sur des interactions rend indépendant le système agent-environnement

de toute notion extérieure d'objet : les modèles d'objets construits par un agent, basés sur ses interactions, seront différents de ceux d'un observateur extérieur. De même, deux agents présentant des différences, que ce soit par leur ensemble d'interaction initial ou par des variations physiques, construiront des modèles d'objets différents, car les affordances de l'environnement ne seront pas les mêmes.

5.2.3 Théorie des *Bundles*

Le modèle que nous proposons implémente la notion de *bundle* définie par David Hume [64]. La théorie des *bundles* stipule qu'un objet est simplement la somme de ses propriétés physiques et d'usage (c'est-à-dire la façon dont on les utilise), sans qu'il ne soit nécessaire de faire intervenir une propriété intrinsèque à l'objet (l'"*essence*"). En effet, les objets tels que nous les définissons sont définis comme des configurations spatiales d'éléments de l'environnement permettant l'événement d'une interaction.

Notre approche diverge cependant par le fait que les associations de propriétés ne sont pas explicites : un objet (au sens élément physique de l'environnement), afforde un certain nombre d'interactions, et peut donc se décomposer en un ensemble de "sous-objets", caractérisés par les différentes interactions affordées par l'objet. Ces interactions, et donc les objets qu'elles définissent, ne sont pas associées explicitement et restent indépendantes. Cependant, un objet physique associe implicitement les interactions qu'il afforde, puisque sa présence permet l'événement d'un groupe d'interactions, formant ainsi un *bundle*. Notons que la notion de *propriété d'usage* précède ici la notion d'objet physique.

5.3 Formalisation du mécanisme de construction de structures pour la prise de connaissance de l'environnement

5.3.1 Principe du mécanisme

Le mécanisme d'apprentissage des objets est inspiré de la théorie des affordances définies par J. J. Gibson, en ce sens où les objets sont définis comme des structures de l'environnement qui affordent des interactions. Notre modèle se base sur l'hypothèse que l'environnement offre des régularités spatiales, c'est-à-dire des configurations spatiales favorables à l'événement d'une interaction. Nous posons ainsi l'hypothèse suivante :

Hypothèse 2. *le résultat (succès ou échec) de l'événement d'une interaction primitive d'un agent est lié à la présence ou l'absence d'une certaine configuration spatiale d'éléments ou de propriétés de l'environnement de cet agent.*

Ainsi, on définit un lien entre une interaction et une certaine configuration qui afforde cette interaction. Nous appellerons objet O_i une configuration physique de l'environnement qui afforde une interaction i .

Définition 13. *Objet : un Objet est une configuration d'éléments ou de propriétés de l'environnement, minimale et suffisante, qui afforde l'interaction qui le définit.*

Par *élément* et *propriété*, nous entendons tout élément physique de l'environnement, existant indépendamment de l'agent. La configuration est dite *minimale*, car ne contient que les éléments susceptibles d'influencer, par leur présence ou leur absence, le résultat d'une interaction. Nous utilisons le terme *afforder* dans le sens donné par Gibson, c'est-à-dire qui permet d'effectuer une interaction. Par exemple, une interaction *se cogner* est affordée par la présence d'un élément solide de l'environnement, situé devant l'agent. La "frontière" de l'objet est définie par la région de l'espace dans laquelle un élément solide peut provoquer l'événement de l'interaction *se cogner*, qui dépend de la forme et de la taille de l'agent. La figure 5.3 illustre cet exemple.

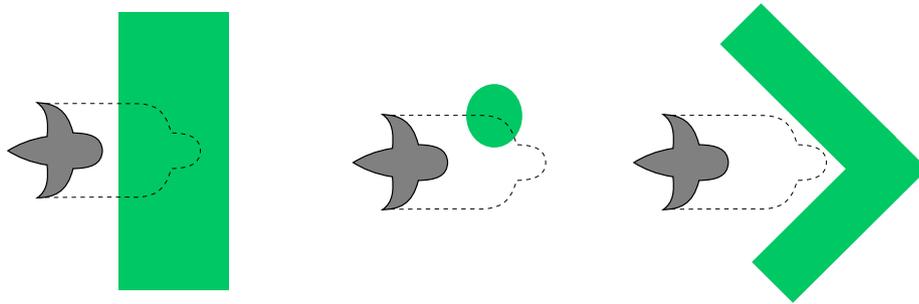


FIGURE 5.3 – Objet qui afforde l'interaction *se cogner*, qui consiste à avancer d'un pas, et être arrêté avant de terminer ce pas. La figure représente l'agent (en gris) dans trois configurations spatiales, telles qu'un observateur extérieur pourrait les voir. Les éléments solides sont représentés en vert. À gauche et au milieu, deux configurations qui affordent l'interaction *se cogner*. À droite, une configuration qui n'afforde pas cette interaction. L'objet qui afforde *se cogner* est défini par la *présence d'un élément solide dans une région (en pointillé) devant l'agent*, indépendamment de la taille et de la forme des éléments solides.

Pour satisfaire cette définition, nous proposons de poser les propriétés suivantes à respecter par les objets d'interaction :

Propriété 1. *un objet est défini par une interaction, qui dépend du couplage agent/environnement, et non de l'agent seul ou de l'environnement seul.*

Propriété 2. *un objet n'est pas accessible à l'agent. L'agent ne peut pas connaître la nature ontologique d'un objet, mais connaît l'objet seulement au travers de ses interactions.*

Propriété 3. *un objet est défini à partir du moment où l'interaction qu'il afforde est définie.*

Cette relation entre interactions et éléments physiques permet notamment de caractériser l'environnement en termes d'interactions. Elle permet également de prédire la faisabilité d'une interaction en détectant la présence de l'objet qui l'afforde. La connaissance de l'environnement peut donc se baser sur la formation d'une relation entre les interactions et les objets tels qu'ils sont détectés par l'agent par le biais de ses interactions.

Le mécanisme proposé est basé sur le principe suivant : un agent basé sur le principe du Radical Interactionism "expérimente" son environnement par le biais des interactions qu'il énonce. La ou les interactions énoncées renseignent l'agent sur certaines propriétés de l'environnement. Nous définissons l'hypothèse suivante :

Hypothèse 3. *l'énoncé d'une interaction primitive informe sur la présence d'une certaine configuration spatiale d'éléments ou de propriétés de l'environnement dans le contexte de l'agent. Cette configuration est le "résultat" de l'énoncé de l'interaction dans l'environnement.*

L'ensemble des interactions énoncées par l'agent caractérise ainsi une certaine configuration de l'environnement. Notons que l'énoncé d'une interaction i ne fournit aucune information sur l'objet que l'agent intègre ainsi. Cependant, le contexte obtenu peut, dans certains cas, "contenir" l'objet qui afforde une certaine interaction j . Ainsi, l'énoncé de l'interaction i peut permettre la détection de l'objet qui afforde j , et ainsi, caractériser la possibilité de l'énoncer au pas suivant.

Le mécanisme repose donc sur la possibilité de définir, pour chaque interaction, le ou les contextes minimaux qui incluent l'objet qui afforde cette interaction. Ceci revient à définir un ensemble d'interactions dont l'énoncé caractérise la présence de cet objet. Nous établissons ainsi un modèle des objets basé sur un ensemble de connexions entre les interactions. Ce modèle ne repose donc pas sur la notion d'objets physiques.

5.3.2 Formalisation

Objets

Nous avons défini un objet d'interaction comme une configuration spatiale d'éléments ou de propriétés de l'environnement qui afforde une interaction. Nous notons O_i un objet qui afforde une interaction i , que nous appelons objet *pré-énoncé* de i . Nous notons \hat{O}_i un objet qui résulte de l'énoncé d'une interaction i , que nous appelons objet *post-énoncé* de i . Il faut souligner que les objets post-énoncé constituent l'ensemble des objets que l'agent peut "expérimenter" : l'énoncé d'une interaction donne implicitement la "présence" de l'objet post-énoncé correspondant. Il faut rappeler que les objets, pré et post-énoncé, sont définis implicitement par les interactions.

Définition 14. *Objet pré-énoncé : un objet pré-énoncé O_i est une configuration d'éléments physiques ou de propriétés de l'environnement qui afforde l'interaction i qui le définit.*

Définition 15. *Objet post-énoncé : un objet post-énoncé \hat{O}_i est une configuration d'éléments physiques ou de propriétés de l'environnement qui résulte de l'énoncé de l'interaction i qui le définit.*

Par exemple, une interaction *pousser un objet déplaçable* sera affordée par la présence d'un élément déplaçable devant l'agent, qui constituera ainsi l'objet pré-énoncé de cette interaction. Si l'agent énonce cette interaction, l'élément se retrouvera à une nouvelle position. L'objet post-énoncé de l'interaction sera ainsi l'élément déplaçable initialement considéré, à une certaine

position de l'espace, et potentiellement endommagé par le déplacement. La présence de cet objet post-énaction est caractérisé par le succès de l'interaction *pousser un objet déplaçable*.

Le mécanisme d'apprentissage peut se formaliser ainsi : l'agent apprend à reconnaître un objet pré-énaction O_i qui afforde i en déterminant des ensembles d'objets post-énaction \hat{O}_{j_k} résultant de l'énaction des interactions j_k , qui contiennent O_i . Soit :

$$\{j_k\} / O_i \subseteq \cup_k \hat{O}_{j_k} \quad (5.1)$$

Notons que l'objet qui afforde une interaction i est inclus dans l'ensemble des objets résultant de l'énaction de l'ensemble d'interactions $\{j_k\}$, constituant l'objet expérimenté. Ainsi, l'agent peut, dans le cas d'une inclusion stricte, ne pas pouvoir définir et détecter l'objet qui afforde l'une de ses interactions, mais seulement une approximation qui inclut cet objet.

Il faut rappeler que l'agent ne peut pas percevoir directement les objets, mais les expérimenter au travers de ses interactions. Les objets étant liés par définition aux interactions, nous pouvons éliminer toute référence aux objets physiques de l'équation (5.1) pour obtenir une condition basée uniquement sur des interactions. En effet, un objet post-énaction \hat{O}_i est considéré comme présent si l'interaction i est énéctée avec succès, donc si $i \in E_t$.

Un objet pré-énaction O_i est considéré comme présent au cycle de décision t si i est énécté avec succès au cycle de décision $t+1$. On peut donc définir les conditions suivantes :

$$\{j_k\} / \begin{cases} (\{j_k\} \subseteq E_t \wedge i = i_{t+1}) \Rightarrow i \in E_{t+1} \\ (\forall j_k, \exists i \in \mathcal{A}_{j_k} / i \in E_t \wedge i = i_{t+1}) \Rightarrow i \notin E_{t+1} \end{cases} \quad (5.2)$$

La première condition implique que si toutes les interactions du groupe sont énéctées, alors l'objet est présent et l'interaction peut être énéctée avec succès. La seconde implique que si toutes les interactions du groupe échouent, l'objet est absent et l'énaction de l'interaction sera un échec.

Nous pouvons ajouter à la description d'un objet pré-énaction O_i l'ensemble des interactions j_l dont l'objet post-énaction \hat{O}_{j_l} est incompatible avec l'objet O_i , c'est-à-dire que tout ou partie du contexte qui afforde i et du contexte défini par l'énaction de j_l sont mutuellement exclusifs. De ce fait, l'énaction d'une interaction j_l implique la non-présence de O_i . La description de ces interactions peut s'avérer utile, notamment si il n'existe pas d'interactions permettant de détecter la présence de l'objet O_i , mais qu'il existe des interactions permettant de détecter son absence. La définition d'un objet peut être étendue aux interactions permettant d'expérimenter l'absence d'un objet :

$$\{j_l\} / \begin{cases} (\forall j_l, \exists i \in \mathcal{A}_{j_l} / i \in E_t \wedge i = i_{t+1}) \Rightarrow i \in E_{t+1} \\ (\{j_l\} \subseteq E_t \wedge i = i_{t+1}) \Rightarrow i \notin E_{t+1} \end{cases} \quad (5.3)$$

Signature d'interaction

Nous proposons la notion de *signature d'interaction* pour qualifier une structure regroupant les ensembles permettant de détecter l'objet qui afforde une interaction. La signature est définie

par :

Définition 16. *Signature d'une interaction : nous appelons signature d'une interaction i , notée S_i , une structure caractérisant le ou les ensembles d'interactions dont le succès caractérise la présence (ou l'absence) de l'objet qui afforde i .*

En conservant les notations des conditions (5.2) et (5.3), une signature peut s'écrire sous la forme :

$$S_i = (\{j_k\} ; \{j_l\}) \quad (5.4)$$

Où $\{j_k\}$ est l'ensemble d'interactions permettant de détecter la présence de l'objet qui afforde i et $\{j_l\}$ est l'ensemble des interactions permettant de détecter l'absence de cet objet. Notons qu'une signature permet de définir un objet du point de vue de l'agent, c'est-à-dire sous la forme de l'ensemble des interactions qui permettent de détecter cet objet (en l'expérimentant) avant d'interagir avec lui. On peut également noter que la signature définit l'objet depuis la position de l'espace à partir de laquelle l'interaction affordée par cet objet peut être énoncée. Cette position définit le "centre" de l'objet. Une interaction peut en effet être affordée par une configuration d'éléments éloignées spatialement de l'agent. Cependant, seule la position de l'espace à partir de laquelle une interaction peut être énoncée est nécessaire à l'agent.

Définition 17. *Centre d'un objet : le centre d'un objet est la position de l'espace depuis laquelle l'interaction qu'il afforde peut être énoncée.*

La signature d'une interaction i ne peut être définie que si l'objet O_i peut être détecté par l'agent par le biais de ses interactions. Dans le cas contraire, il n'est pas possible de prédire le résultat de l'interaction i . La signature d'une interaction affordée par un objet qui échappe au système interactionnel de l'agent est appelée *signature vide*.

Définition 18. *Signature vide : une signature vide est la signature d'une interaction dont le résultat ne peut pas être prédit d'après le contexte interactionnel.*

Propriété 4. *Une signature vide est la signature d'une interaction dont l'objet ne peut être détecté.*

Certitude d'une interaction

La définition (5.2) ne permet de prédire le résultat de l'énoncé d'une interaction que dans le cas où toutes les interactions caractérisant la présence ou l'absence d'un objet sont énoncées. Il faut donc définir, en cas de détection partielle d'un objet, la certitude que l'objet soit présent (ou absent). Les degrés de certitude de présence et d'absence d'un objet, qui définissent également le degré de certitude de succès ou d'échec d'une interaction, sont définis par deux fonctions, que nous appelons fonction de certitude de succès $c_{success}$ et fonction de certitude d'échec $c_{failure}$, qui comparent la signature S_i d'une interaction i et l'ensemble énoncé E_t . Ces fonctions sont définies par :

$$c_{success} : I \times \mathcal{P}(I) \rightarrow [0; 1]$$

$$c_{failure} : I \times \mathcal{P}(I) \rightarrow [0; 1]$$

avec une certitude absolue de présence d'un objet qui afforde une interaction i , et donc le succès de l'énaction de i , si $c_{success}(i, E_t) = 1$ et une certitude absolue d'absence si $c_{failure}(i, E_t) = 1$. Ces fonctions permettent de définir la certitude de succès ou d'échec d'une interaction à partir d'un contexte interactionnel ne caractérisant que partiellement la présence de l'objet qui l'afforde.

Nous pouvons caractériser la prédiction d'une interaction en fonction des certitudes de succès et d'échec :

Définition 19. Une interaction i est dite énaactable si $c_{success}(i, E_t) > \mu$, et comme non énaactable si $c_{failure}(i, E_t) > \mu$, avec μ le seuil de fiabilité, $\mu \in [0, 1]$.

Plusieurs interactions peuvent être affordées par un même contexte d'éléments de l'environnement, localisé à une même position de l'espace. Ces ensembles d'interactions peuvent être détectés lorsque leurs signatures sont identiques. De tels ensembles, associés à un même objet, forment des *bundles*, au sens donné par David Hume, c'est-à-dire un ensemble de propriétés qui définissent un objet. Cependant, notre mécanisme ne construit pas explicitement ces *bundles*. En effet, comme deux interactions disposant de signatures identiques seront considérées comme énaatables ou non énaatables simultanément, il n'est pas nécessaire de les associer explicitement.

Apprentissage des signatures d'interaction

L'apprentissage doit permettre le renforcement des signatures, par l'expérience, de façon à pouvoir évaluer des certitudes de succès et d'échec pertinentes. L'apprentissage de la signature d'une interaction s'effectue à chaque fois que cette interaction se termine comme un succès ou un échec. Cet apprentissage se fait de manière supervisée, en comparant les prédictions de certitude de succès et d'échec obtenues au début du cycle de décision avec le résultat réellement énaqué de l'interaction considérée. Le système d'apprentissage doit satisfaire la condition suivante, permettant de garantir la pertinence des certitudes :

$$\lim_{t \rightarrow +\infty} (c_{success}(i, E_{t-1}) - c_{failure}(i, E_{t-1})) - res(i, t) = 0 \quad (5.5)$$

pour toute interaction $i \in I$ dont la signature n'est pas vide, avec $res(i, t) = 1$ si i est un succès au pas t et $res(i, t) = -1$ si i est un échec. On ne prendra pas en compte les cas où i n'est pas énaquée.

Réversibilité des signatures

Le principe de la signature d'interaction doit également lui permettre d'être réversible : en effet, si une interaction est considérée comme énaactable (ou non-énaactable), indépendamment du contenu du contexte interactionnel, alors l'objet qui afforde cette interaction est considéré comme

TABLE 5.1 – Liste des interactions primitives. Le symbole de chaque interaction est celui qui sera utilisé dans les figures de ce chapitre

- □	i_1 toucher un espace vide devant.
- ■	i_2 toucher un obstacle devant.
- ▷	i_3 avancer d'une case dans l'environnement.
- ►	i_4 se cogner dans un obstacle.
- ◁	i_5 tourner à droite face à un espace vide.
- ◄	i_6 tourner à droite face à un obstacle.

présent (respectivement absent). La signature doit ainsi permettre de compléter le contexte interactionnel en ajoutant les interactions qui auraient permis la détection de l'objet, c'est-à-dire les interactions qui composent cette signature. Bien entendu, en cas de conflit avec le contexte interactionnel, seules les interactions réellement énoncées sont prises en compte, et permettent éventuellement de corriger la signature en conflit. On définit la fonction $c_{reverse}$, permettant de compléter le contexte interactionnel, comme une fonction donnant la certitude qu'une interaction j ait pu être énoncée au pas t , en considérant la certitude de succès $c_{success}^t$ et d'échec $c_{failure}^t$ d'une interaction i .

Cette fonction doit tenir compte du contexte courant. En effet, si une interaction peut être abordée par deux contextes distincts, un contexte partiel E_t peut aider à définir lequel des deux a la plus grande certitude d'être présent. Il est alors possible de définir un contexte augmenté en ajoutant au contexte interactionnel E_t les interactions j pour lesquelles la certitude ainsi définie est élevée.

L'utilisation d'un contexte augmenté peut s'avérer utile lorsque le contexte interactionnel est limité. Dans la première expérimentation, décrite au chapitre 9, l'agent n'énonce qu'une seule interaction primitive par cycle de décision. La possibilité de compléter le contexte, grâce à la mémoire spatio-séquentielle, permet à l'agent de caractériser sa situation malgré un retour interactionnel limité.

5.3.3 Exemple pour illustrer le principe des signatures

Supposons un agent dans un environnement discret, implémentant le système RI décrit au chapitre 4, et disposant d'un ensemble d'interactions primitives I contenant les interactions définies dans le tableau 5.1.

Nous pouvons définir les objets pré et post-énoncé des interactions *toucher* : l'énoncé de l'interaction *toucher un espace vide* est abordé, comme on peut s'en douter, par une case vide devant l'agent, et l'énoncé de l'interaction *toucher un obstacle*, par la présence d'une case pleine.

L'interaction *avancer* est abordé par la présence d'une case vide devant l'agent. Une fois cette interaction énoncée, l'agent s'est déplacé et la case vide se retrouve dans la même case que l'agent (on supposera que cette interaction ne tient pas compte de la case initialement occupée

par l'agent). L'interaction *se cogner* est affordée par la présence d'une case pleine devant l'agent. Après éaction de cette interaction, l'agent ne s'est pas déplacé, et la case pleine est toujours devant l'agent.

L'interaction *tourner à droite face à un espace vide* implique la présence d'une case vide à droite de l'agent. Une fois la rotation effectuée, cette case vide se retrouve devant l'agent. Il en est de même avec l'interaction *tourner face à un mur*, avec des cases pleines. La Figure 5.4 résume les objets associés aux interactions.

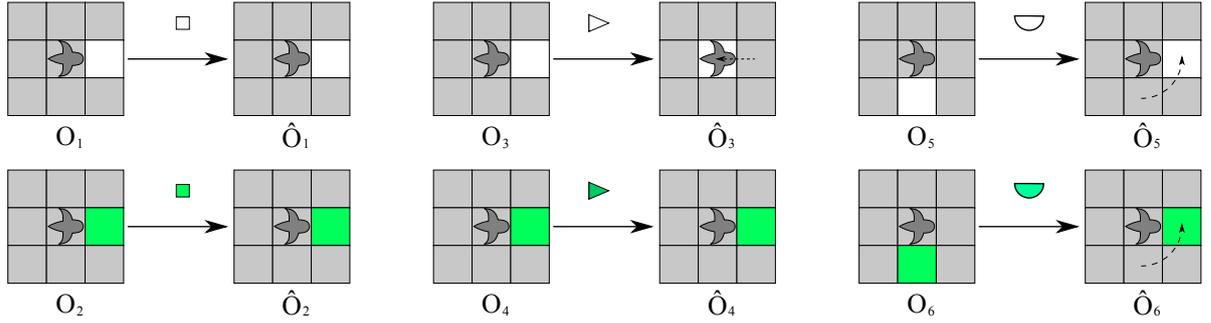


FIGURE 5.4 – Objets pré et post-énaction des interactions primitives i_1 à i_6 . Le contenu des cases définissant les objets est représenté. Les cases blanches correspondent à un espace vide, les cases vertes à une case pleine. Le contenu des cases grisées n'est pas pertinent, car ces cases ne font pas partie des objets. Il faut préciser que les objets sont représentés de la façon dont un observateur externe peut représenter ces objets d'après le contenu de l'environnement qu'il observe. L'agent, quant à lui, intègre ces objets uniquement par ses interactions.

D'un point de vue extérieur, nous pouvons définir intuitivement les inclusions suivantes :

$$O_1 \in \hat{O}_1, O_1 \in \hat{O}_5, O_3 \in \hat{O}_1, O_3 \in \hat{O}_5.$$

$$O_2 \in \hat{O}_2, O_2 \in \hat{O}_4, O_2 \in \hat{O}_6, O_4 \in \hat{O}_2, O_4 \in \hat{O}_4, O_4 \in \hat{O}_6.$$

Ainsi, il est possible de détecter l'objet qui éactionne l'interaction *avancer d'un pas* par l'éaction de i_1 ou de i_5 . La signature de cette interaction est donc $\{i_1, i_5\}$ (nous ne prenons pas en compte, dans cet exemple, les interactions permettant de détecter l'absence d'un objet). De même, on peut définir les signatures des interactions i_1 à i_4 :

$$S_{i_1} = (\{i_1\}, \{i_5\}), S_{i_3} = (\{i_1\}, \{i_5\}),$$

$$S_{i_2} = (\{i_1\}, \{i_4\}, \{i_6\}), S_{i_4} = (\{i_1\}, \{i_4\}, \{i_6\})$$

Nous pouvons remarquer que les signatures S_{i_1} et S_{i_3} sont identiques, de même que les signatures S_{i_2} et S_{i_4} , caractérisant le fait que ces signatures désignent un même objet. Ainsi, les interactions i_1 et i_3 , et les interactions i_2 et i_4 seront toujours reconnues comme potentiellement éactionnables simultanément en présence, respectivement, d'une case vide et d'une case pleine devant l'agent. Ces interactions forment ainsi des *bundles* au sens donné par Hume, sans qu'il ne soit nécessaire de définir explicitement le lien entre ces interactions. Les signatures des interactions i_5

et i_6 ne peuvent pas être définies avec les interactions spécifiées précédemment, car les objets O_5 et O_6 ne sont contenus dans aucun des objets post-énaction définis par les interactions primitives. Leur signature est donc vide.

La réversibilité des signatures permet de compléter le contexte expérimenté par l'agent. Supposons qu'à un certain cycle de décision t , l'agent vienne d'énacter l'interaction *avancer d'un pas*. Le contexte contient donc l'interaction i_3 . Ce contexte n'apporte malheureusement aucune information sur la possibilité d'énacter une interaction avec succès. On considère cependant que l'agent dispose d'un mécanisme lui permettant d'affirmer, avec une certaine certitude, que l'interaction *toucher un espace vide* est énable. Ceci implique que l'objet O_1 est considéré comme présent dans l'environnement. Comme nous l'avons vu, cette interaction dispose d'une signature contenant les interactions i_1 et i_5 , dont les objets post-énaction intègrent l'objet O_1 .

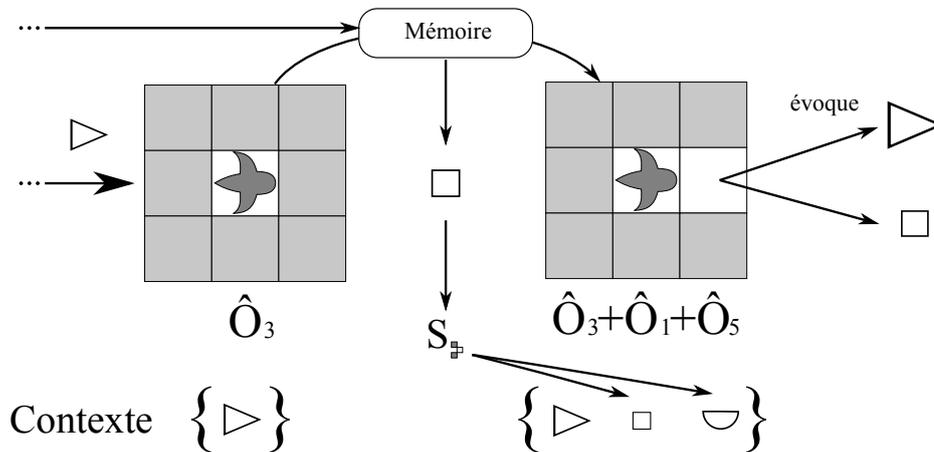


FIGURE 5.5 – Réversibilité des signatures : l'agent vient d'énacter l'interaction *avancer*. Le contexte ne contient pas assez d'informations pour prédire le résultat des interactions. Cependant, l'agent dispose d'une structure lui permettant de considérer la certitude que l'énaction de l'interaction *toucher devant* soit un succès. Ceci implique la présence de l'objet O_1 , caractérisé par la signature $S_{i_1} = (\{i_1\}, \{i_5\})$. Les interactions i_1 et i_5 peuvent être considérées comme faisant partie du contexte, puisque les objets \hat{O}_1 et \hat{O}_5 ont une forte certitude d'être présent. Le nouveau contexte permet de prédire le résultat des interactions *toucher un espace vide devant* et *avancer*.

On peut donc compléter le contexte en ajoutant les interactions i_1 et i_5 , puisque ces interactions caractérisent la présence de O_1 . Le contexte ainsi augmenté contient les interactions $\{i_3, i_1, i_5\}$. Ce contexte permet de prédire l'énaction des interactions i_1 et i_3 , permettant, par exemple, à l'agent d'avancer une seconde fois sans avoir besoin d'expérimenter son environnement pour acquérir des informations sur son contenu (Figure 5.5).

5.3.4 Propriétés de relation entre les objets

La définition des objets permet de définir un ensemble de propriétés sur la certitude de présence et d'absence des interactions d'une signature. Ces propriétés apparaissent implicitement par le biais des signatures d'interaction et des fonctions de certitude de succès et d'échec.

Inclusion

L'objet O_{i_1} permettant l'énaction d'une interaction i_1 est inclus dans l'objet O_{i_2} , permettant l'énaction de l'interaction i_2 , si et seulement si O_{i_2} affine i_1 . L'inclusion peut également se traduire par : $S_{i_1} \in S_{i_2}$. Cela signifie que si un objet d'interaction O_i est présent dans l'environnement, alors toutes les interactions dont l'objet est inclus dans O_i sont énonçables. De même, si l'objet d'interaction O_{i_2} est absent, alors O_{i_1} sera absent, et i_1 sera non énonçable.

Propriété 5. *si i_1 est incluse dans i_2 , alors si i_2 est énonçable, i_1 sera énonçable. Si i_1 est non énonçable, i_2 sera non énonçable.*

Identité

L'objet O_{i_1} permettant l'énaction d'une interaction i_1 est identique à l'objet O_{i_2} , permettant l'énaction de l'interaction i_2 , si et seulement si O_{i_1} affine i_2 et O_{i_2} affine i_1 . L'identité peut également se traduire par : $S_{i_1} = S_{i_2}$. Cela signifie que si une interaction i est énonçable à un cycle de décision t , alors toutes les interactions identiques sont également énonçables.

Propriété 6. *si i_1 et i_2 sont identiques, alors si i_1 est énonçable, i_2 sera énonçable (et inversement).*

Similarité

Deux objets O_{i_1} et O_{i_2} sont dits similaires s'ils sont définis par un même élément ou ensemble d'éléments de l'environnement, mais localisés à des positions différentes de l'environnement (en référentiel égocentré). Soient O_{i_1} et O_{i_2} deux objets qui affinent respectivement les interactions i_1 et i_2 . O_{i_1} et O_{i_2} sont dit similaires si à un cycle de décision t , i_1 est énonçable, et il existe une séquence d'interactions s de longueur n éventuellement nulle, telle que si l'agent énonce s (dont on supposera le succès), alors i_2 sera énonçable au cycle de décision $t+n$. L'inverse est également vrai. Notons que si s a une longueur nulle, alors O_{i_1} et O_{i_2} sont identiques, car cela implique que les i_1 et i_2 soient énonçables simultanément.

Il est possible par la suite de définir des classes d'objets en regroupant les objets similaires. Notons que de telles classes seront elles aussi implicites, puisque l'agent n'utilise pas directement ces associations.

5.4 Conclusion

Nous avons défini dans ce chapitre un mécanisme permettant à l'agent de construire une structure pour détecter et reconnaître les objets associés à ses interactions. Cette structure lui permet de définir les interactions qu'il peut ou ne peut pas effectuer dans un contexte donné, et peut également compléter ce contexte à partir d'interactions considérées comme énonçables par d'autres mécanismes, comme ceux définis par les mécanismes de construction (cf chapitre 6) et d'exploitation (cf chapitre 7) de la mémoire spatiale.

Chapitre 6

Construction d'une structure caractérisant l'environnement

Les expérimentations du mécanisme permettant la prise en compte de l'espace péri-personnel (que nous présenterons au chapitre 9) nous ont conduit à penser qu'un mécanisme additionnel devait être ajouté pour permettre à un agent de tenir compte d'éléments situés à distance. Les expérimentations présentées au chapitre 10 utilisent une structure codée en dur pour caractériser la structure topographique de l'espace. Les structures cérébrales des êtres vivants dont nous nous inspirons, comme le tectum optique [92], ne couvrent que l'espace rétinien, c'est-à-dire, l'espace visuel. Le tectum est en quelque sorte "pré-câblé" pour respecter la topographie de la rétine, mais on ne voit pas ce qui pourrait expliquer le pré-câblage d'un espace en dehors du champ visuel. Le mécanisme proposé dans ce chapitre permet à un agent de construire une structure lui permettant dans un premier temps d'intégrer la présence d'objet dans l'espace atteignable par son système d'interaction et de continuer à les suivre et les localiser même lorsqu'ils échappent à son système interactionnel.

Ce mécanisme est divisé en deux parties, considérant chacune une région de l'espace environnant :

- l'espace "*observable*", c'est-à-dire l'ensemble des positions dans laquelle l'agent peut intégrer une interaction éactée. Le mécanisme qui traite cet espace sert principalement à intégrer spatialement les objets dans l'espace couvert par le système interactionnel de l'agent.

- l'espace "*global*", c'est-à-dire l'ensemble de l'espace entourant l'agent (en référentiel égocentré). Cet espace comprend l'ensemble de l'espace que l'agent peut intégrer. Le mécanisme qui traite de cet espace doit construire une structure intégrant les éléments de l'espace environnant et permettant le suivi des objets intégrés dans l'espace observable. Notons que l'espace non observable, c'est-à-dire la partie de l'espace global ne faisant pas partie de l'espace observable, ne définit pas de positions, car celles-ci ne pourraient pas être expérimentées par l'agent.

6.1 Inspiration biologique et principes du modèle

6.1.1 Gestion de l'espace observable

Nous nous inspirons du tectum optique des vertébrés (appelé colliculus chez les mammifères). Cette structure permet de caractériser l'espace visuel. Chez les poissons et les batraciens, le tectum optique joue un rôle central dans la vision et la représentation spatiale de l'environnement [92]. Le tectum est "pré-câblé" pour respecter la topographie de l'espace rétinien, et le nerf optique peut se régénérer en respectant cette topographie s'il est endommagé.

Stephen Grossberg et al. ont proposé un modèle prédictif du colliculus [57] dans lequel le mécanisme apprend à générer des saccades visuelles vers un point du champ visuel, et à prédire la position des *stimuli* visuels avant d'effectuer une saccade. Ce système apprend à associer des positions de l'espace avec le mouvement qui permet d'observer ces positions. Il génère implicitement des liens entre les positions de l'espace rétinien. Nous cherchons à implémenter un mécanisme similaire pour générer des liens entre les positions de l'espace observable.

Les observations sur les signatures d'interaction obtenues dans la seconde expérimentation (Chapitre 10) ont en effet montré que la structure de l'espace observable pouvait être apprise par le biais des signatures des interactions, en reliant des points de l'espace par le mouvement qui permet de passer d'une position à une autre, sans que la topographie de l'espace soit nécessaire. L'utilisation d'une structure intégrant la topographie de l'espace observable permet de simplifier la recherche des objets lointains, en autorisant l'utilisation de transformations géométriques, même approximatives : les expérimentations menées avec le robot ont en effet montré que le système de décision fonctionne même avec une faible précision dans les déplacements.

6.1.2 Reconnaissance et mise en mémoire des objets

Chez les mammifères, la reconnaissance des objets et leur localisation dans l'espace sont attribués aux lobes pariétaux postérieurs et aux lobes temporaux. Les lobes pariétaux sont associés à la reconnaissance de la position des objets dans l'espace, ainsi qu'à la façon d'interagir avec eux. Les lobes temporaux sont quant à eux associés à la reconnaissance iconique des objets, c'est-à-dire à la reconnaissance visuelle des objets indépendamment de leurs propriétés. Cependant, une étude menée sur certains patients souffrant de lésions aux lobes temporaux [69] montre qu'ils souffrent d'héminégligence spatiale, ce qui tendrait à montrer que la reconnaissance des objets et leur localisation dans l'espace pourraient être liés. Cette relation est intégrée dans nos modèles puisque la reconnaissance des éléments du contexte spatial est basée à la fois sur les signatures des interactions et sur la possibilité de simuler des déplacements dans l'espace.

Ces modèles permettent de reconnaître les objets distants à partir de leur signature. Certaines observations de l'expérience sur l'exploitation de la mémoire spatiale (présentée au chapitre 10) ont montré qu'il est possible de "déplacer" les signatures dans l'espace, d'après une séquence d'interactions. Un objet distant peut ainsi être reconnu si il existe au moins une séquence d'interactions qui permet de l'atteindre. Sa position peut alors être définie comme la plus courte

séquence d'interactions permettant de l'atteindre.

On implémente ainsi la notion d'*espace représentatif* défini par Poincaré, pour qui localiser un objet « *signifie simplement que nous nous représentons les mouvements qu'il faut faire pour atteindre cet objet; et qu'on ne nous dise pas que pour se représenter ces mouvements, [...], la notion d'espace [géométrique] doit, [...], préexister. Quand je dis que nous nous représentons ces mouvements, je veux seulement dire que nous nous représentons les sensations musculaires qui les accompagnent et qui n'ont aucun caractère géométrique, qui par conséquent n'impliquent nullement la préexistence de la notion d'espace [géométrique].* » [107].

La mise en mémoire des objets nécessite dans un premier temps de définir leurs positions sous une forme qui permettra leur suivi et leur exploitation. L'étude des mécanismes d'exploitation d'une structure reflétant l'espace (seconde expérimentation) a mis en évidence que seules deux informations étaient nécessaires pour intégrer la position d'un objet, et permettre son suivi :

- une estimation de sa distance, afin d'en définir l'influence sur le comportement de l'agent,
- l'existence d'une interaction qui permet de s'en rapprocher.

Nous proposons de caractériser dans un premier temps des régions de l'espace définies par des propriétés similaires, c'est-à-dire les régions dont une même interaction permet un rapprochement maximal et situé à une distance comprise dans un certain intervalle.

Étant donné que l'on peut détecter et localiser un objet lointain à l'aide d'une séquence d'interactions, il est possible de définir dans quelle région l'objet se trouve : en effet, la première interaction de la séquence est l'interaction qui permet de se rapprocher le plus de l'objet, puisque cette séquence est considérée comme la plus courte. La distance est estimée par la longueur de cette séquence : la distance est donc définie en fonction du nombre d'interactions nécessaires pour interagir directement avec l'objet. Notons qu'en cas d'utilisation d'une structure topographique inspirée du tectum optique, on peut simplifier l'estimation de la distance en prenant la distance géométrique de l'objet sur cette structure, et déterminer les interactions qui permettent de s'en rapprocher en mesurant les variations de distances produites par ces interactions.

6.1.3 Gestion de l'espace global

Le tectum optique des vertébrés couvre l'espace visuel des animaux. Peu d'entre eux disposent d'un champ de vision de 360°. Une autre structure est donc nécessaire pour gérer et suivre des objets en dehors du champ visuel.

Nous supposons que l'ensemble de l'espace global peut être expérimenté en effectuant une séquence courte d'interactions. Chez les êtres vivants, il suffit, par exemple, de tourner la tête et/ou une partie du corps pour observer une région initialement hors de portée du système interactionnel. De même que nous avons défini des régions de l'espace observable, nous pouvons définir des régions de l'espace non-observable comme des régions de l'espace observable "déplacées" d'une interaction ou d'une séquence d'interaction. Un intérêt de définir de telles régions est qu'il est possible de suivre les objets qu'elles contiennent tant qu'on suit la séquence d'inter-

action qui les caractérise. Cependant, il n'est pas possible de suivre ces régions sur de longues périodes.

Pour pallier cette limite, je propose un mécanisme reposant sur l'apprentissage de liens entre les régions d'espace observable "déplacées" et non plus sur les positions de l'espace observable. Nous avons vu précédemment qu'un objet pouvait être localisé dans une région de l'espace observable, et donc dans l'espace global, s'il est déplacé par rapport à l'agent. Tout comme les signatures d'interaction peuvent lier des interactions, des liens sont introduits entre les régions de l'espace global. L'utilisation de ces liens permet de suivre des objets jusqu'à ce que la fiabilité de l'estimation de sa position soit jugée insuffisante.

6.2 Formalisation du mécanisme de construction d'une structure caractérisant l'environnement

6.2.1 Gestion de l'espace observable

Un mécanisme est proposé pour permettre à un agent d'organiser de façon fonctionnelle l'espace avec lequel il peut interagir. Par organiser, il faut comprendre apprendre la structure qui lie les positions de l'espace, et générer une structure exploitable par le mécanisme de décision reflétant le contenu de l'espace. Nous définissons une *position* comme un point de l'espace en référentiel égocentré. Dans un modèle RI (ou PRI), chaque interaction primitive est associée à une position, bien que cette position soit inconnue. Dans le cas du modèle SRI, les interactions sont associées explicitement aux positions, pour former des *actes* (cf. chapitre 4). Notons que dans le cadre d'un modèle Spatial RI (SRI), l'apprentissage des propriétés spatiales de l'espace observable est inutile, puisque les positions des interactions et les mouvements dans l'espace sont pré-définis. Le mécanisme d'apprentissage est basé sur l'hypothèse suivante :

Hypothèse 4. *Unicité de la position d'une interaction : l'énaction d'une interaction i caractérise la présence de son objet post-interaction \hat{O}_i en une position non-explicite mais unique dans l'espace en référentiel égocentré.*

Construire la structure de l'espace revient ainsi à déterminer la position relative des interactions en définissant des liens entre ces interactions. Nous exploitons certaines propriétés des interactions et des signatures : nous savons que l'énaction d'une interaction est affordée par son objet pré-énaction O_i , et génère un objet post-énaction \hat{O}_i . Une interaction i produit donc une transformation de l'objet pré-énaction vers l'objet post-énaction. Cette transformation peut impliquer des modifications physiques de l'objet ou un déplacement de celui-ci par rapport à l'agent, soit parce que l'objet a été déplacé, soit parce que l'agent s'est déplacé dans l'environnement. Nous pouvons donc considérer que i produit un *déplacement* de l'objet O_i vers l'objet \hat{O}_i , en référentiel égocentré. Le mécanisme construit ainsi la structure de l'espace en établissant des liens entre les interactions. Chaque lien est caractérisé par l'interaction produisant le déplacement. Nous ne tiendrons compte que des déplacements liés aux interactions primaires : le

déplacement lié aux interactions secondaires étant le même que celui de leur interaction primaire associée.

Nous proposons trois méthodes pour définir ces liens entre les interactions. Les deux premières utilisent les signatures des interactions, la troisième se base sur la *similarité* (cf chapitre 5) des objets.

La première méthode exploite le déplacement de l'objet pré-énaction vers l'objet post-énaction provoqué par l'énaction d'une interaction. La position d'un objet post-énaction \hat{O}_i est caractérisée par l'interaction i . En revanche, la position de l'objet pré-énaction O_i est inconnue. On peut en avoir une estimation par le biais de la signature de i . En effet, la signature S_i désigne un ensemble d'interactions $\{j_k\}$ pour lesquelles l'ensemble des objets post-énaction contient O_i . Or, les interactions j_k caractérisent la position de leur objet post-énaction respectif. Nous pouvons alors considérer que l'interaction i (ou de son interaction associée, dans le cas d'une interaction secondaire) produit un mouvement des positions définies par les interactions j_k vers la position définie par i . Notons que ce mouvement correspond au déplacement par rapport à l'agent de l'objet provoqué par l'énaction de cette interaction, indépendamment du mouvement de l'agent dans son environnement. Le déplacement provoqué par l'énaction d'une interaction peut ainsi être local, par exemple si l'interaction ne fait que déplacer l'objet sans provoquer de déplacement de l'agent.

La seconde méthode se base sur le raisonnement inverse : si une interaction i fait partie de la signature d'une interaction j , alors l'énaction de j génère un déplacement de l'objet \hat{O}_i vers \hat{O}_j . Nous pouvons donc établir un lien entre une interaction i et les interactions j pour lesquelles $i \in S_j$. Notons que chaque lien définit le déplacement provoqué par l'interaction j (ou de son interaction associée, dans le cas d'une interaction secondaire).

La troisième méthode exploite la *similarité* des objets définie par les *interactions composites* (voir chapitre 7). Une interaction composite φ permet de caractériser la présence d'un objet caractérisé par une interaction, appelée *interaction finale*, mais localisé à une position différente, caractérisée par une séquence d'interaction appelée *path*. L'objet pré-énaction d'une interaction composite est similaire à l'objet post-énaction de son interaction finale. Ainsi, il est possible d'établir un lien entre les interactions signées par la signature S_φ de l'interaction composite, et celles signées par la signature de son interaction finale. Le déplacement est associé à l'énaction de la séquence d'interaction de l'interaction composite. Notons que cette méthode est la moins précise. Il n'est en effet pas possible de définir les déplacements entre chaque couple d'interaction, mais seulement un déplacement d'un groupe d'interactions vers un autre groupe d'interactions.

6.2.2 Effet mémoire

Il est possible d'exploiter l'effet prédictif des signatures pour compléter le contexte interactionnel : si une interaction, prédite comme un succès avec une forte certitude, n'a pas été énoncée (sans avoir échoué), alors il est possible de considérer cette interaction comme énoncée. En effet, l'objet qui afforde cette interaction peut se soustraire au système interactionnel de l'agent. Par

exemple, un objet offrant une interaction visuelle peut disparaître derrière un élément opaque, empêchant l'interaction d'être énoncée. Cependant, l'interaction, prédite comme un succès, implique la présence de cet objet, même si l'agent ne peut plus le percevoir. Cet effet mémoire peut s'avérer utile pour reconnaître un objet dont certains aspects peuvent échapper au système interactionnel de l'agent.

L'expérience sur l'exploitation de la mémoire spatiale (présentée au chapitre 10, section 4.7) montre que la mémoire spatiale peut conserver la présence d'un objet qui se retrouve masqué. Cet effet a également pu être mis en évidence dans l'expérience sur la construction de la mémoire spatiale (présentée au chapitre 11), bien qu'il n'ait pas été exploité par notre mécanisme.

6.2.3 Reconnaissance des objets distants

Nous proposons un mécanisme permettant d'intégrer spatialement les objets qu'il reconnaît dans l'environnement. Le mécanisme proposé est différent selon qu'on soit dans le cadre d'un modèle RI (ou PRI) ou d'un modèle SRI. Dans le premier cas, nous mettons à profit les liens entre les positions de l'espace caractérisées par les interactions. Nous pouvons ainsi définir les transformations de la mémoire spatiale sous la forme de séquences d'interactions. Nous ne tenons compte que des interactions les plus courtes permettant de lier deux interactions. Dans le second cas, on peut appliquer des transformations géométriques à la mémoire spatiale.

On note τ une transformation de la mémoire spatiale, et T l'ensemble des transformations possibles. Dans notre modèle, une transformation spatiale est équivalente à la position de l'espace qu'elle permet d'atteindre. On note $E_{\tau,t}$ le contexte interactionnel obtenu en appliquant la transformation τ au contexte interactionnel E_t . Pour toute transformation $\tau \in T$, une instance de l'objet offrant une interaction i est considérée comme *présente en τ* avec une certitude de $c_{success}(i, E_{\tau,t})$. On définit ainsi la certitude de présence d'un objet offrant l'interaction i en toute position de l'espace que l'agent peut intégrer.

Définition 20. *Instance d'objet* : nous appelons instance d'objet ω_i offrant une interaction i une configuration spatiale d'éléments ou de propriétés de l'environnement qui, si déplacée en référentiel égo-centré par une certaine transformation τ , offre i .

Définition 21. *Position d'une instance d'objet* : nous appelons Position d'une instance d'objet ω_i la transformation τ permettant à cette instance d'offrir l'interaction i .

La possibilité de calculer la certitude de présence des instances d'objets en toute "position" de l'espace permet d'établir une carte implicite des affordances offertes par l'environnement de l'agent. Cependant, cette "carte" n'est pas directement exploitable : la position des instances d'objets ne tient pas compte de la possibilité d'effectuer les transformations permettant de les atteindre. En effet, nous ne cherchons pas à développer un mécanisme de *path planning*.

6.2.4 Gestion de l'espace non-observable

La gestion de l'espace non observable consiste à utiliser les informations issues de l'espace observable pour détecter et intégrer les instances d'objets et caractériser leurs positions, même quand celles-ci échappent au système interactionnel de l'agent. Afin de permettre le suivi des instances d'objets, un mécanisme permet de localiser et mettre à jour la position des instances dans l'espace global.

Nous nous basons sur l'hypothèse que toute région de l'espace global est accessible au système interactionnel de l'agent en effectuant une séquence d'interactions. Cela signifie par exemple que l'espace situé derrière l'agent peut être observé si celui-ci tourne sur lui-même.

Hypothèse 5. *l'ensemble de l'espace environnant peut être interagi en effectuant une séquence "courte" d'interaction.*

Il est ainsi possible de définir des positions dans l'espace non observable comme des structures constituées d'une position de l'espace observable et d'une séquence d'interactions qui lie cette position à la position de l'espace observable. De par ce principe, il devient possible de localiser une instance d'objet dans l'espace global en considérant sa position comme un mouvement (une séquence d'interactions) qu'il faudrait effectuer pour pouvoir détecter cette instance dans l'espace observable. La Figure 6.1 illustre ce principe.

6.2.5 Lieu d'interaction

Si l'on appliquait le principe décrit dans la section précédente (6.2.4), le nombre de positions obtenues serait trop important, et un long apprentissage serait nécessaire pour apprendre à prédire la présence d'une interaction sur ces positions, même en supposant qu'il soit possible de fusionner ces "positions composites" (par analogie avec les interactions composites), lorsqu'elles signent une même position de l'espace. Il est donc proposé de réduire le nombre de positions nécessaires pour caractériser l'espace global : les résultats de l'expérience sur l'exploitation de la mémoire spatiale (présentée au chapitre 10) ont montré qu'il n'est pas nécessaire de définir la position des objets de façon précise. Nous ne considérons non pas des positions ponctuelles, mais des régions de l'espace, plus ou moins grandes, que nous appelons *lieu*. Un lieu est défini comme une portion quelconque de l'espace observable, regroupant les positions qui s'y trouvent, et pouvant être défini arbitrairement. Notons que les lieux peuvent se chevaucher. Nous verrons plus tard l'utilité de cette propriété. La sélection de l'ensemble L des lieux utilisés par la mémoire spatiale nécessite de respecter certaines contraintes, dont notamment d'apporter des informations utiles au mécanisme de sélection.

Nos études sur le développement d'un mécanisme de sélection exploitant une mémoire spatiale ont montré qu'un tel mécanisme n'a besoin que de la distance d'une instance d'objet, et de connaître les interactions qui permettent de s'en rapprocher. Ceci permet de regrouper les positions de l'espace observable qui partagent les mêmes caractéristiques, pour définir des lieux permettant d'extraire les deux informations requises par le mécanisme de sélection de la mémoire

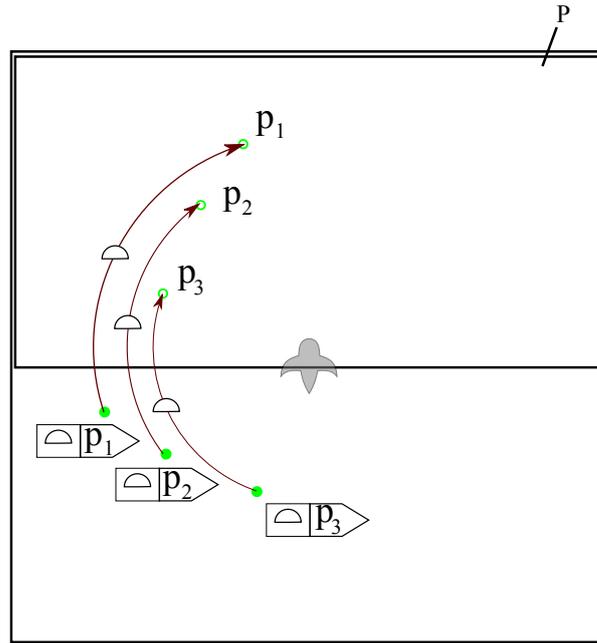


FIGURE 6.1 – Principe de la gestion de l'espace non-observable : soient trois positions de l'espace non observable (disques verts). Ces positions ne peuvent pas être directement expérimentées par l'agent. Cependant, si l'agent effectue une rotation à gauche (\triangleleft), ces positions seront localisées respectivement en p_1 , p_2 et p_3 (cercles verts) de l'espace observable (P). Les trois positions de l'espace non-observable peuvent donc être caractérisées par $[tourner\ à\ gauche][p_1]$, $[tourner\ à\ gauche][p_2]$ et $[tourner\ à\ gauche][p_3]$. Notons qu'une position de l'espace non observable peut être caractérisée par plusieurs séquences d'interactions.

spatiale, à savoir la distance et l'interaction permettant un rapprochement maximal. Un lieu est donc défini comme :

Définition 22. *Lieu* : un lieu $l \in L$ est un ensemble de positions de l'espace observable dont tous les points sont équidistants de l'agent et pour lesquels une même interaction permet un rapprochement maximal.

Propriété 7. *pour toute position τ de l'espace (accessible en appliquant la transformation τ), il existe un lieu $l \in L$ tel que $\tau \in l$.*

Un lieu peut, dans cette approche, se noter sous la forme d'un couple $l = (i, d)$, où i est l'interaction provoquant un rapprochement maximal de ce lieu, et d sa distance par rapport à l'agent.

La manière d'intégrer l'espace diffère si l'on utilise le modèle RI ou le modèle SRI. Dans le premier cas, on se base sur les séquences les plus courtes permettant de joindre toute position de l'espace observable. Deux positions font parties d'un même lieu si les séquences qui permettent de les atteindre ont une même longueur (c'est-à-dire qu'elles sont considérées comme équidistants de l'agent), et si elles commencent par une même interaction. En effet, les séquences caractérisent

le "chemin" le plus court pour atteindre les positions de l'espace. Le premier élément est donc une interaction qui permet de se rapprocher le plus de cette position.

L'utilisation d'une structure topographique, dans le modèle SRI, évite de devoir chercher de telles séquences. Pour chaque position, l'interaction la plus "efficace" est définie en comparant la distance géométrique par rapport au centre du référentiel (équivalente à la position actuelle de l'agent), et la distance de cette position une fois déplacée par les interactions. Nous pouvons ainsi déterminer l'interaction la plus efficace pour se rapprocher de cette position. Nous regroupons ensuite les positions dont la distance par rapport à l'agent est comprise dans un certain intervalle. Ces intervalles sont définis arbitrairement. Cependant, si ceux-ci sont trop importants, l'agent risque de ne pas pouvoir mesurer le rapprochement des instances d'objets. Trop petits, le nombre de lieux augmente, ce qui augmentera la durée d'apprentissage.

Notons que, comme les distances sont définies d'après la position à partir de laquelle une instance d'objet affiche une interaction, et non la position de l'objet "physique", les lieux sont définis indépendamment des interactions. Nous procédons ainsi à une abstraction de la notion de position dans l'espace, qui ne repose pas sur des affordances.

6.2.6 Lieux composites

Les lieux permettent ainsi un découpage fonctionnel de l'espace observable. En utilisant l'hypothèse H5, nous pouvons couvrir l'ensemble de l'espace global. Les *lieux composites* sont définis comme des structures constituées d'un lieu primitif et d'un *path* composé d'une séquence d'interactions. Pour comprendre la notion de lieu composite, prenons un exemple : supposons qu'un objet soit situé derrière l'agent, en dehors de son système interactionnel. L'agent effectue ensuite un quart de tour et voit l'objet dans un lieu l' . Ainsi, avant l'événement de l'interaction tourner, l'objet était dans un lieu composite l caractérisant une région de l'espace global "séparé" de l' par l'interaction tourner d'un quart de tour. Le lieu l ne peut être défini, puisque hors du système sensoriel de l'agent. Nous pouvons toutefois considérer le lieu l comme un lieu composite, constitué d'un lieu final $l_f = l'$ et d'une séquence d'interactions, que nous appelons *path* φ_p , qui permet d'y accéder. La Figure 6.2 illustre cet exemple. Un lieu composite se formalise ainsi :

$$l = ([i_1, \dots, i_n]; l_f), n \in \mathbb{N}$$

$$\varphi_p = [i_1, \dots, i_n]$$

Un lieu composite caractérise la portion de l'espace caractérisée par son lieu final, qui aurait été "déplacé" (par rapport à l'agent), par une séquence d'interactions. Le lieu composite peut donc caractériser une portion de l'espace global, et est indépendant de l'espace observable. Notons qu'un lieu est un lieu composite dont le path a une longueur nulle. On note L' l'ensemble des lieux composites possibles.

Nous pouvons noter certaines propriétés des lieux composites :

Propriété 8. *Un lieu composite caractérise implicitement une portion de l'espace global, puisqu'il ne définit pas de positions.*

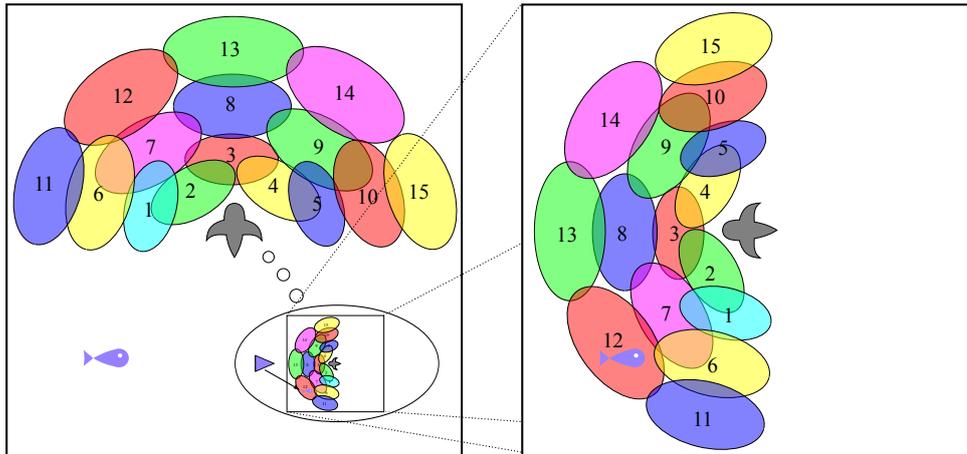


FIGURE 6.2 – Illustration du principe de localisation des objets par les lieux composites. À gauche : l'espace interactionnel de l'agent comprend un ensemble de lieux. Un objet, situé derrière l'agent, échappe à son système interactionnel. Cependant, si l'agent tourne à gauche de 90° , alors l'objet peut être localisé dans un lieu de l'espace observable, ici, le lieu l_{12} . La position de cet objet pouvait initialement être caractérisée par le lieu composite $[tourner \ à \ gauche \ de \ 90^\circ][l_{12}]$. Notons qu'un lieu composite désigne l'interaction, ou la séquence d'interactions, permettant de détecter à nouveau l'objet.

Propriété 9. *Une position de l'espace global peut être incluse dans plus d'un lieu composite.*

6.2.7 Signatures de lieu

Afin de détecter la présence d'une instance d'objet dans un lieu composite, il faut définir les positions de l'espace observable qui en font partie. Dans le cas d'un lieu primitif, ces positions correspondent aux positions de l'espace qui définissent ce lieu. Dans le cas d'un lieu composite doté d'un *path* de longueur non nulle, il faut tenir compte des déplacements par rapport aux positions du lieu final. Une position τ de l'espace est considérée comme faisant partie d'un lieu composite l si :

Propriété 10. *une position τ de l'espace observable fait partie d'un lieu composite l , composé d'un path φ_p , et d'un lieu final l_f , s'il existe une position τ' de l'espace observable telle que τ et τ' soient liées par φ_p , et τ' appartienne à l_f .*

Il est donc possible de définir l'appartenance d'une position à un lieu composite l si le *path* φ_p de ce lieu composite permet de lier cette position à une position appartenant au *lieu final* l_f de l . Ceci n'est cependant pas toujours possible : le *path* peut en effet déplacer une instance d'objet en dehors de l'espace observable, ce qui rend impossible le recours aux liens qui relient les positions par des interactions. L'appartenance des positions devra donc être apprise par l'expérience.

Plus généralement, une position τ de l'espace observable fait partie d'un lieu composite l si chaque fois qu'une instance d'objet est détectée en τ , et après avoir énéacté le *path* de l , une

instance du même type d'objet est présente dans l'une des positions de l'espace appartenant au lieu final l_f de l .

Un apprentissage basé sur un système de signatures, similaire au mécanisme de signatures d'interactions, permet d'apprendre des positions de l'espace observable appartenant à un lieu composite. Nous définissons la *signature de lieu* :

Définition 23. *Signature de lieu : une signature de lieu d'un lieu composite l est l'ensemble des positions τ de l'espace observable considérées comme appartenant à la région caractérisée par l .*

Propriété 11. *La signature de lieu d'un lieu primitif l est composé des positions τ de l'espace qui composent l .*

Nous définissons un contexte $T_{i,t}$ pour chaque type d'objet (c'est-à-dire pour chaque interaction primitive i). Un contexte $T_{i,t}$ regroupe toutes les positions de l'espace observable où a été détecté une instance de l'objet considéré. Nous considérons qu'une instance est présente en une position τ si la certitude de présence d'une instance de l'objet affordant i est supérieure à un certain seuil. Formellement, une signature de lieu S_l doit satisfaire les conditions suivantes :

$$S_l = \{\tau_k\} / \left\{ \begin{array}{l} (\exists \tau_k \in S_l, i \in I / \tau_k \in T_{i,t} \wedge \varphi_p \text{ success}) \Rightarrow \exists \tau' \in S_{l_f} / \tau' \in T_{i,t+n} \\ (\nexists \tau_k \in S_l, i \in I / \tau_k \in T_{i,t} \wedge \varphi_p \text{ success}) \Rightarrow \nexists \tau' \in S_{l_f} / \tau' \in T_{i,t+n} \end{array} \right. \quad (6.1)$$

avec n longueur du path φ_p , τ' une position appartenant au lieu final de l .

Notons qu'il est préférable, pour l'apprentissage des signatures de lieu, de donner une plus grande influence aux instances d'objets les moins nombreuses : en effet, on réduit les chances que deux instances soient prises pour un seul et même objet.

Comme pour les signatures d'interaction, nous définissons une fonction définissant la certitude qu'une position $\tau \in T$ appartienne au lieu l , notée c_{in} , et une fonction définissant la certitude qu'une position $\tau \in T$ n'appartienne pas au lieu l , notée c_{out} . Ces fonctions sont caractérisées par :

$$c_{in} : L' \times T \rightarrow [0;1]$$

$$c_{out} : L' \times I \rightarrow [0;1]$$

Une position τ fait partie d'un lieu composite avec une certitude absolue si $c_{in}(l, \tau) = 1$ et une certitude absolue que τ n'en fasse pas partie si $c_{out}(l, \tau) = 1$.

6.2.8 Mise en mémoire des instances d'objets

Lorsqu'une instance d'objet est détectée en une position τ , cette instance est intégrée sous la forme de l'ensemble des lieux composites contenant τ , c'est-à-dire pour lesquels $c_{in}(l, \tau)$ est supérieur à un certain seuil. Nous définissons ainsi une façon de caractériser une position de l'espace basée sur les lieux composites. La région est définie par l'intersection des lieux composites

considérés. Ainsi, plus les lieux composites seront nombreux, plus la région sera petite, et donc, plus la position de l'instance d'objet sera précise.

Cette méthode de stockage des instances d'objets ne tient plus compte des signatures d'interactions ni des objets. Une instance n'est alors plus définie que comme une région de l'espace à partir de laquelle une interaction peut être énoncée.

Détecter les instances d'objets peut poser certains problèmes : en effet, une même configuration d'éléments de l'environnement peut être détectée comme un ensemble d'instances d'objets. C'est surtout vrai pour les objets avec lesquels l'agent peut interagir depuis un nombre illimité de positions. On pensera notamment aux objets cylindriques et aux murs de grandes longueurs (Figure 6.3), avec lesquels l'agent peut interagir à partir de nombreuses positions. Il est bien entendu impossible de stocker et suivre toutes les instances d'objets offertes par une configuration d'éléments. Il nous faut donc discerner les plus pertinentes.

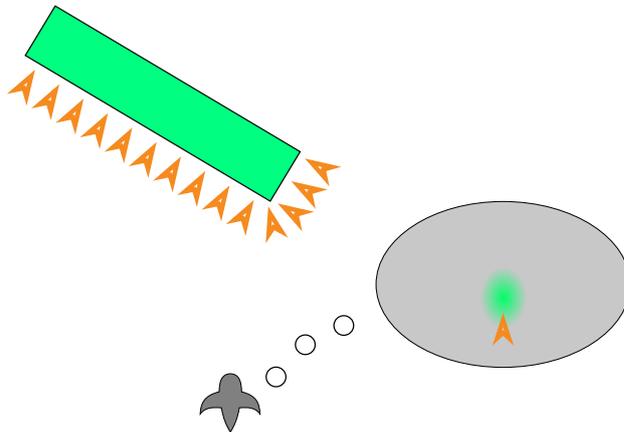


FIGURE 6.3 – Détection multiple des instances d'objets : un élément de l'environnement (ici, un mur) est présenté à l'agent. Celui-ci reconnaît l'élément qui afforde l'interaction *se cogner*, dont la signature est affichée à droite. Cependant, cette signature peut s'appliquer à un nombre illimité de positions de l'espace (les flèches oranges montrent quelques exemples), évoquant un nombre lui aussi illimité d'instances d'objet. Il est donc nécessaire de n'en garder que les plus pertinentes.

Nous estimons la pertinence d'une instance d'objet à partir de sa certitude et de sa proximité avec l'agent. La certitude permet d'indiquer à partir de quelle position de l'espace l'agent a le plus de chance d'interagir avec l'objet. La proximité permet d'indiquer les positions que l'agent a le plus de chance d'atteindre dans un avenir proche. Le cas d'un mur de grande largeur illustre parfaitement ce principe : un agent peut interagir avec un mur depuis n'importe quelle position située à une certaine distance du mur et orientée vers le mur, puisqu'une translation parallèle au mur ne change pas le contexte. Il est cependant judicieux de tenir compte de l'instance la plus proche de l'agent, car celle-ci est celle que l'agent peut atteindre le plus rapidement. Notons

qu'à chaque déplacement, le point le plus proche se déplace : le mur est donc un objet qui peut se "déplacer" lorsque l'agent évolue dans une certaine direction. La pertinence d'une instance d'objet est estimée par : $\frac{c_{success}(i, E_{\tau, t})}{d(\tau)}$, où τ est la position de l'instance d'objet et $d(\tau)$ la distance de cette instance objet. Cette distance correspond à la longueur de la séquence d'interaction qui caractérise la position de l'instance d'objet⁵.

Nous cherchons à présent les instances d'interactions les plus pertinentes pour constituer une liste des objets que la mémoire va intégrer et permettre de suivre. Les instances sont sélectionnées suivant deux règles :

-un lieu composite ne peut définir qu'une seule instance d'un certain type d'objet. En effet, comme chaque lieu composite caractérise une certaine région de l'espace sans distinction des points qui la composent, il ne sera pas possible de dissocier deux instances d'un même d'objet. Comme ces instances seront considérées comme un même objet, on veillera à ne considérer que la plus pertinente d'entre elles.

-la position d'une instance d'objet doit être caractérisée par au moins un lieu. Il est en effet impossible de suivre une instance d'objet si on ne peut pas la localiser.

Nous adoptons la démarche suivante pour définir les instances d'objet qui seront stockées en mémoire et les lieux composites qui caractériseront leur position :

-On détecte dans un premier temps l'instance d'objet la plus pertinente de chaque lieu⁶.

-Une fois les maxima de pertinence identifiés pour chaque lieu, on élimine les lieux couvrant plus d'un maximum.

-Quand il ne reste plus que des lieux couvrant au plus un maximum de pertinence, on supprime, parmi les maxima non couverts par un lieu, celui qui a la plus petite pertinence. On recommence jusqu'à ce que chaque maximum de pertinence soit couvert par au moins un lieu. Les maxima retenus sont les instances qui seront stockées en mémoire.

Ainsi, on s'assure que chaque instance d'objet ne soit définie que par un ensemble de lieux composites qui lui soient propres. Chaque instance d'objet retenue se voit attribuer la liste des lieux qui ne couvrent que cette instance d'objet. Cette liste permet de caractériser la position de l'instance d'objet, en indiquant les interactions (ou séquences d'interactions dans le cas d'un lieu composite) permettant de s'en rapprocher pour interagir directement.

6.2.9 Suivi des instances d'objets

Deux mécanismes pour suivre les instances d'objets intégrés en mémoire sont proposés. Ces deux mécanismes sont utilisés simultanément, et diffèrent par la durée et la fiabilité du suivi qu'ils permettent.

5. ou, dans le cas du modèle SRI, par la distance géométrique par rapport au centre du référentiel

6. Dans le cas du modèle SRI, pour lequel les positions sont organisées topographiquement, je propose une première sélection en ne considérant que les maxima locaux de pertinence. Ceci permet de réduire le nombre d'instances d'objets et donc, de réduire les calculs nécessaires pour définir les instances d'objet à retenir.

Le premier mécanisme exploite l'aspect séquentiel des lieux composites : les lieux composites peuvent être mis à jour en supprimant la première interaction de leur path si celui-ci est énéacté. Nous appelons *lieu suivant* d'un lieu composite l le lieu obtenu après une telle mise à jour :

Définition 24. *Lieu suivant* : Un lieu est dit suivant d'un lieu composite l si il est composé du même lieu final et si son path correspond au path de l moins la première interaction.

Ainsi, le lieu *suivant* du lieu composite $l = [j, i] [l_1]$ est le lieu $l' = [i] [l_1]$. La mise à jour est possible si l'interaction j est énéactée avec succès. Les lieux composites obtenus caractérisent une nouvelle région de l'espace. Ainsi, en mettant à jour les lieux qui caractérisent la position d'une instance d'objet, les lieux obtenus caractérisent la région dans laquelle se situe cette instance après énéaction de la dernière interaction énéactée. Il faut noter que ce mécanisme de mise à jour réduit le nombre de lieux de la liste à chaque cycle de décision, puisque les lieux composites ne commençant pas par la dernière interaction énéactée sont éliminés. Ainsi, la précision de la position de cette instance, caractérisée par l'intersection des lieux, diminue à chaque pas. Cependant, la certitude de présence de l'instance dans les lieux mis à jour ne diminue pas. Bien entendu, la durée du suivi des instances d'objets est limitée par la longueur maximale du path des lieux composites.

Afin de contourner ces limitations, un second mécanisme permet d'évoquer de nouveaux lieux composites. Ce mécanisme se base sur le principe suivant : chaque position peut être définie par une liste des lieux qui la contiennent, y compris les positions de l'espace non observable. En effet, les lieux composites, par définition, peuvent définir des régions de l'espace non observable. Une position de l'espace non observable (non définie par l'agent) appartient donc à un ensemble de lieux composites. Les listes de lieux composites caractérisant la position des instances d'objet intégrées en mémoire constituent ainsi un contexte indiquant les régions de l'espace occupées par des instances d'objets. Ce contexte peut être utilisé pour prédire la détection d'une instance d'objet dans un lieu composite, si l'une des instances intégrées en mémoire se retrouve dans l'espace observable. La Figure 6.4 illustre ce principe.

Nous définissons un second type de signatures liées aux lieux composites : les *signatures de présence*. Cette signature caractérise la certitude de présence d'une instance d'objet intégrée en mémoire dans un lieu l , à partir de l'ensemble de lieux définissant la position estimée de cette instance. La présence peut être testée en énéactant les interactions du path de l : si le contenu de la mémoire spatiale caractérise la présence d'une instance d'objet par un ensemble de lieux $\{l_k\}$, et si après énéaction des interactions du path de l , une instance du même objet est présente dans le lieu final de l , alors, en supposant que l'instance est la même, on peut considérer qu'avant d'énéacter les interactions du path de l , l'instance d'objet était dans la région caractérisée par l . La région caractérisée par l'ensemble $\{l_k\}$ fait donc partie de la région qui caractérise l .

Tout comme pour les signatures d'interaction et de lieu, nous devons définir un contexte permettant l'apprentissage des signatures. La position de chaque instance d'objet est caractérisée par un ensemble des lieux. Ainsi, chaque instance d'objet définit une région de l'espace plus ou moins vaste dans laquelle l'instance peut se trouver. C'est cette région, sous la forme de listes de

lieux composites, que nous considérons comme contexte.

À chaque instance d'objet intégrée en mémoire est associé un contexte défini par la liste des lieux composites qui caractérisent sa position.

Définition 25. *Position d'une instance d'objet :* la position d'une instance d'objet ω est définie par l'intersection des régions associées aux lieux qui contiennent cette instance d'objet.

Définition 26. *Contexte associé à une instance d'objet :* nous notons $\Omega_{\omega,t}$ l'ensemble des lieux qui caractérisent la position d'une instance d'objet ω à un cycle de décision t , formant le contexte associé à ω .

Nous considérons les contextes de chaque instance comme indépendants : en effet, la position d'une instance d'objet est définie par l'intersection d'un ensemble de lieux. Chaque instance caractérise donc une région de l'espace différente à partir des lieux qui lui sont attribués.

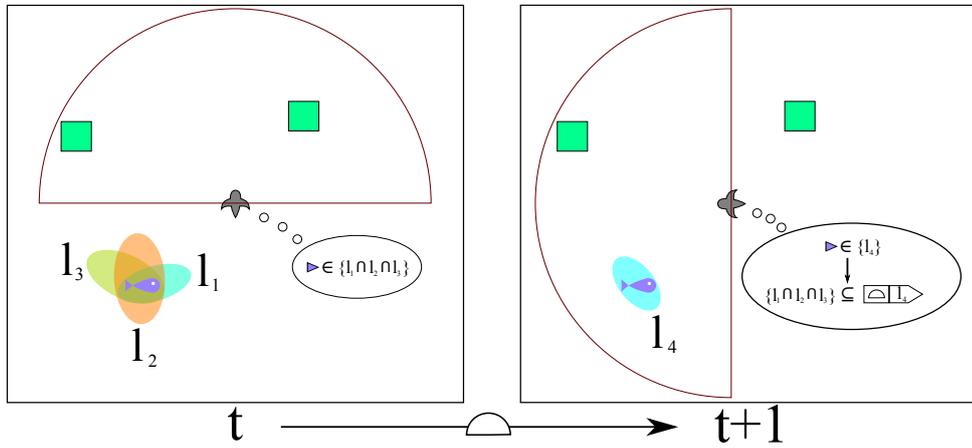


FIGURE 6.4 – Signatures de présence. Au cycle de décision t , la mémoire intègre une instance de l'objet affordant l'interaction *manger* (\blacktriangleright), localisée par les lieux composites l_1 , l_2 et l_3 . Au cycle de décision $t+1$, l'agent énonce l'interaction *tourner à gauche de 90°* (\triangleleft). Une instance de l'objet affordant l'interaction *manger* est détectée au lieu primitif l_4 . Le contexte $\Omega_{manger_1,t} = \{l_1, l_2, l_3\}$ permet donc de caractériser la région de l'espace définie par le lieu composite $[tourner \ à \ gauche \ de \ 90^\circ][l_4]$.

Nous définissons ensuite l'activation d'un lieu :

Définition 27. *Activation d'un lieu :* un lieu est dit activé par un d'objet O_i affordant une interaction i s'il contient une instance de cet objet. Un lieu est dit inactif pour un objet O_i s'il ne contient aucune instance de cet objet.

Notons qu'un lieu peut être activé par plusieurs instances de différents objets. Formellement, un lieu l est actif pour un objet affordant l'interaction i si :

$$\exists \omega_i / l \in \Omega_{\omega_i,t} \quad (6.2)$$

La signature de présence $S_{\pi,l}$ d'un lieu composite $l = (\varphi_p; l_f)$, avec φ_p le path de longueur n , est défini par les conditions suivantes :

$$S_{\pi,l} = \{l_k\} / \{\forall t / \varphi_p \text{ success at } t+n\}, \left\{ \begin{array}{l} \exists \omega_i / \{l_k\} \subseteq \Omega_{\omega_i,t} \Rightarrow \exists \omega_i / l_f \in \Omega_{\omega_i,t+n} \\ (\nexists \omega_i / \{l_k\} \cup \Omega_{\omega_i,t} \neq \emptyset) \Rightarrow \nexists \omega_i / l_f \in \Omega_{\omega_i,t+n} \end{array} \right. \quad (6.3)$$

Notons que la signature de présence d'un lieu doit être incluse dans le contexte d'une instance d'objet : en effet, une région est définie par l'intersection d'un ensemble de lieux. La région de l'instance d'objet sera donc incluse dans celle définie par la signature.

Comme pour les signatures d'interaction, nous définissons une fonction de certitude de présence et une fonction de certitude d'absence d'une instance d'objet définie par un contexte $\Omega_{\omega_i,t}$ dans un lieu composite l . Ces fonctions, notées respectivement c_{pres} et c_{abs} , sont définies par :

$$c_{pres} : L' \times \mathcal{P}(L') \rightarrow [0; 1]$$

$$c_{abs} : L' \times \mathcal{P}(L') \rightarrow [0; 1]$$

Ces fonctions caractérisent la certitude de présence et d'absence d'une instance ω_i d'objet dans un lieu l , avec une certitude absolue de présence si $c_{pres}(l, \Omega_{\omega_i,t}) = 1$ et une certitude absolue d'absence si $c_{abs}(l, \Omega_{\omega_i,t}) = 1$. Cette certitude permet d'ajouter des lieux composites à l'ensemble des lieux caractérisant une instance d'objet suivie, lorsque ceux-ci sont considérés comme actifs avec une certitude suffisante (i.e. supérieure à un seuil). Notons que, contrairement au suivi par mise à jour des lieux composites, la certitude de présence d'une instance dans un ensemble de lieux peut décroître à chaque pas si la certitude de présence de l'instance dans le nouveau lieu n'est pas absolue.

Il est inutile d'ajouter un lieu composite si sa première interaction du *path* n'est pas éactée par l'agent au pas suivant : le lieu ajouté sera supprimé de la liste et n'apportera, au cycle de décision $t+1$, aucune information supplémentaire. Pour des raisons de simplifications, nous proposons de n'ajouter un lieu composite l considéré comme actif à un cycle de décision t que si et seulement si l'interaction éactée au cycle $t+1$ est identique à la première interaction du *path* du lieu l . Le lieu ajouté sera bien entendu le lieu l auquel on aura retiré le premier élément de son *path*. En conséquence, le mécanisme d'apprentissage des signatures de déplacement ne tient donc compte que des lieux composites dont le *path* a une longueur supérieure ou égale à 1.

6.2.10 Gestion de l'incertitude des instances d'objet

À chaque mise à jour de la mémoire spatiale, les erreurs dans le suivi des instances d'objets s'accumulent. Il est donc important de définir la certitude qu'une instance d'objet soit dans la région de l'espace caractérisée par un lieu composite. La certitude qu'un lieu composite l

caractérise la position d'une instance d'objet ω est notée $c_{l,\omega}$. Lorsqu'une instance d'objet est détectée et intégrée en mémoire, la certitude que sa présence soit caractérisée par un lieu l est équivalente à la certitude que la position τ_ω qu'elle occupe fasse partie de l . Soit :

$$c_{l,\omega}^0 = c_{in}(l, \tau_\omega) \quad (6.4)$$

Avec c_{in} la fonction qui estime la certitude avec laquelle une position τ fait partie d'un lieu composite l . La mise à jour des lieux composites par la première méthode n'affecte pas la certitude initiale. En effet, le remplacement d'un lieu composite par le lieu *suivant* n'est soumis à aucune incertitude. On peut toutefois noter que le déplacement de l'agent peut entraîner des incertitudes (notamment dans le cas d'un agent robotique, dont les déplacements sont bruités). La certitude est donc mise à jour par :

$$c_{l',\omega}^t = \alpha \cdot c_{l,\omega}^{t-1} \quad (6.5)$$

avec $\alpha \in [0; 1]$ un coefficient caractérisant l'incertitude générée par les déplacements de l'agent, et l' le lieu *suivant* de l .

La seconde méthode apporte plus d'incertitude. En effet, la mise à jour est effectuée par des estimations liées à des signatures de déplacement. Il faut donc tenir compte de la certitude de présence observée lorsque le lieu composite est évoqué. Soit $c_{max,\omega}$ la certitude maximale des lieux composites de l'ensemble Ω_ω caractérisant la présence de l'instance ω , la certitude de présence d'un lieu composite l évoqué est définie par :

$$c_{l,\omega}^0 = c_{max} \cdot c_{pres}(l, \Omega_\omega) \quad (6.6)$$

On peut donc constater que si l'instance ne peut plus être détectée dans l'espace observable, la certitude des lieux qui caractérisent sa position ne peut que décroître.

6.2.11 Identification des instances d'objet intégrées en mémoire

Lorsque une instance d'objet revient dans l'espace observable, elle peut être détectée comme une nouvelle instance, qui formera alors un doublon. Je propose un mécanisme permettant de reconnaître les instances d'objets déjà intégrées en mémoire lorsqu'elles sont à nouveau interagies. Ce mécanisme compare la position de l'instance d'objet nouvellement détecté avec la position estimée des instances du même objet stockées en mémoire. Le principe est le suivant : s'il existe une instance intégrée en mémoire pour laquelle la région définie par les lieux composites qui lui sont associés couvrent la position de la nouvelle instance d'objet, alors cette instance est considérée comme étant la même que celle nouvellement détectée. Dans le cas où plusieurs instances recouvrent cette position, seule celle qui dispose du plus grand nombre de lieux contenant la nouvelle instance d'objet sera considérée comme étant cette instance d'objet nouvellement détectée. En cas d'égalité, le choix se basera sur la certitude de présence des instances d'objets. L'instance

existante reconnue est supprimée et remplacée par la nouvelle, afin de se baser sur la position la plus récente. Notons cependant qu'il est impossible de garantir que la nouvelle instance est bien la même que celle mémorisée.

Nous définissons une règle supplémentaire permettant de se débarrasser plus rapidement des instances d'objets "anciennes" ou trop peu fiables. Après identification des nouvelles instances d'objets, s'il reste d'une part des nouvelles instances d'objet non identifiées, et d'autre part des instances du même objet intégrées en mémoire dont la certitude sur sa position est considérée comme faible, alors on peut considérer que cette nouvelle instance a une certaine probabilité d'être l'instance mémorisée, ce qui nous incite à supprimer cette instance.

Ce mécanisme permet en premier lieu d'éviter les doublons en supprimant les instances reconnues comme étant les mêmes. D'autre part, il permet une mise à jour en continu de la mémoire, en ne prenant en compte que les informations les plus récentes. En effet, la position d'un objet devient de moins en moins sûre et précise à chaque pas de l'agent. Cette mise à jour permet de redéfinir de façon précise la position de cet objet.

Étant donné que l'incertitude de la position occupée par une instance d'objet croît à chaque cycle de décision, des erreurs peuvent apparaître lors de cette mise à jour, notamment si la position estimée d'une instance d'objet est trop éloignée de sa position réelle, ou si la position estimée se retrouve proche de la position d'une autre instance d'un même type d'objet. Cependant, ce genre d'erreur n'est pas préjudiciable à l'agent, puisque seule compte la position des affordances proposées par les instances d'objet. Le suivi d'une instance particulière d'un objet n'est pour le moment pas étudié dans nos travaux.

6.3 Conclusion

Dans ce chapitre, nous avons défini un mécanisme permettant à l'agent de construire une structure lui permettant d'intégrer, localiser et suivre les instances des objets qui affordent ses interactions. Cette structure, appelée Mémoire Spatiale, permet de compléter le contexte interactionnel de l'agent, et fournit à l'agent des informations pertinentes sur la position des instances d'objet qui l'entourent, lui permettant de se rapprocher ou de s'éloigner de ces instances d'objet. Les informations fournies par cette structure sont exploitées par le mécanisme d'exploitation de la mémoire spatiale, décrit dans le chapitre suivant (Chapitre 7).

Chapitre 7

Exploitation d'une structure caractérisant l'environnement

Ce chapitre présente l'exploitation d'une structure informationnelle, telle que décrite dans les chapitres précédents, pour permettre l'émergence de comportements adaptés pour un agent plongé dans un espace environnemental donné. Les structures informationnelles proposées permettent l'intégration des éléments de l'environnement avec lesquels l'agent peut interagir : les instances d'objets. Ces instances d'objets sont intégrées par un mécanisme d'apprentissage décrit dans les chapitres précédents. Dans le cadre de l'étude du mécanisme permettant l'émergence d'un comportement adapté à la prise en compte de ces "objets", nous avons utilisé une version codée en dur de la mémoire spatiale dont le comportement est équivalent à une mémoire spatiale qui aurait été apprise :

Hypothèse 6. *l'agent dispose d'une mémoire spatiale lui permettant de localiser et suivre les interactions énoncées. Cette mémoire peut fournir toute information sur la position des interactions énoncées nécessaire au mécanisme de décision.*

Deux mécanismes sont étudiés pour exploiter une structure intégrant l'environnement. Ces deux mécanismes sont conçus pour pouvoir fonctionner simultanément.

Le premier mécanisme génère une structure informationnelle caractérisant l'espace de proximité. Par proximité, il faut comprendre que ce mécanisme ne prend en compte que les objets que l'agent peut atteindre en énonçant une courte séquence d'interactions, la longueur maximale de ces séquences définissant la portée de cette structure. Cependant, ce mécanisme ne nécessite que peu de calculs pour engendrer des comportements. Son utilisation est donc recommandée pour générer des comportements "automatiques" permettant d'interagir rapidement et avec précision avec les objets proches, ce qui peut être utile dans un environnement dynamique, ou nécessitant une forte réactivité de l'agent.

Le second mécanisme prend en compte des objets sur une plus grande distance. Ce mécanisme estime les interactions permettant de se rapprocher ou s'éloigner des objets distants, et permet d'engendrer des comportements adaptés aux objets qui entourent l'agent et à leur localisation

relative à l'agent.

Comme nous l'avons vu au chapitre 1, cette distinction entre espace proche et espace lointain est observée chez les êtres vivants (dont l'homme). La gestion de l'espace est différenciée entre espace péri-personnel (proche) et espace extra-personnel. Les mécanismes permettant de gérer l'espace proche et l'espace lointain ont été appelés respectivement mécanisme de l'espace péri-personnel et de l'espace extra-personnel, et seront présentés séparément dans ce chapitre.

Ce chapitre est divisé en deux parties : la première partie présente les observations biologiques dont nous nous inspirons pour développer nos mécanismes, et notamment les propriétés observées chez les êtres vivants pour gérer l'espace péri-personnel et extra-personnel. La seconde partie propose une formalisation des mécanismes que nous proposons, permettant d'intégrer l'espace péri-personnel et l'espace lointain. Les propriétés de ces deux mécanismes sont présentées suivant le même ordre que les propriétés décrites dans la première partie.

7.1 Inspiration biologique

Chez les êtres vivants, la prise en compte de l'espace fait intervenir différentes régions fonctionnelles. Des études menées chez des patients souffrant de lésions cérébrales ont montré que ces régions correspondent à des aires différentes dans le cerveau. Des études menées par Berti et Fransinetti [12] ont montré qu'une de leur patiente souffrait d'héminégligence dans l'espace lointain, mais pas dans l'espace péri-personnel. L'exercice consiste à barrer des lignes sur une feuille située devant soi, et des lignes disposées sur un tableau distant à l'aide d'un pointeur lumineux. La patiente a omis de barrer les lignes situées de son côté gauche sur le tableau, mais a barré toutes les lignes tracées sur la feuille. En revanche, en utilisant un bâton plutôt qu'un pointeur, l'héminégligence disparaît. En effet, la manipulation du bâton étend l'espace péri-personnel jusqu'à couvrir le tableau. Cette étude montre qu'il y a bien une séparation fonctionnelle entre l'espace péri-personnel et l'espace lointain, mais également que la frontière entre ces deux espaces est dynamique et s'adapte aux possibilités d'interactions entre un agent et son environnement.

Cette séparation fonctionnelle et dynamique de l'espace a également été étudiée chez le singe [79]. Ces études montrent que l'espace péri-personnel prend en compte l'utilisation d'un outil, à condition que celui-ci soit utilisé de manière active par l'animal : si le bras portant l'outil est manipulé par un expérimentateur, l'outil n'est pas pris en compte dans l'espace péri-personnel de l'animal. L'espace péri-personnel peut également s'étendre au reflet dans un miroir, ou une image des membres sur un écran [66]. Ceci montre que la notion d'espace péri-personnel est fortement liée aux interactions entre l'agent et son environnement, et à la façon dont il l'expérimente, et non à une information issue d'une mesure perceptive simple (comme une mesure de distance, par exemple).

Si un obstacle empêche d'atteindre un objet situé dans l'espace péri-personnel, alors la région inaccessible n'est plus considérée comme faisant partie de l'espace péri-personnel par certains neurones miroir, même si l'objet est perceptible [18] [78]. L'espace péri-personnel est ainsi lié aux possibilités d'interactions et donc à leur accessibilité : en effet, un agent ne peut pas interagir

avec une région non accessible.

Nous définissons deux mécanismes distincts pour représenter l'espace péri-personnel et l'espace lointain. L'espace péri-personnel correspond à l'espace que l'agent peut atteindre en effectuant une séquence courte d'interactions (de 1 à 4 dans nos implémentations présentées au chapitre 9). L'espace lointain correspond à l'espace entourant l'agent nécessitant plus d'interactions pour l'atteindre. Notons que l'espace lointain englobe l'espace péri-personnel. Le mécanisme gérant l'espace péri-personnel n'est donc pas forcément nécessaire. Cependant, les expérimentations montrent que son usage est conseillé pour améliorer la facilité avec laquelle l'agent se rapproche des objets proches.

7.1.1 Espace péri-personnel

Ce premier mécanisme doit permettre à un agent de tenir compte des objets "proches" de lui, c'est-à-dire pour lesquels une courte séquence d'interactions permet de les atteindre et d'interagir avec eux. Ce mécanisme est inspiré d'une expérience effectuée chez le singe sur l'encodage des mouvements dans l'espace [54].

Cette étude a montré que les neurones du cortex précentral sont associés à des postures complexes du corps, impliquant notamment la position du bras et de la tête. Lorsque l'un de ces neurones est stimulé électriquement, le singe prend la pose correspondante, indépendamment de la pose initiale, et la garde tant que la stimulation est présente. En testant plusieurs de ces neurones et en comparant la position finale de la main par rapport au corps, il est apparu que leur répartition forme une représentation topographique de l'espace atteint par la main. Cette étude semble donc montrer qu'une position dans l'espace péri-personnel est représentée par le geste qui permet de l'atteindre. La représentation de l'espace décrite ici ne concerne que l'espace péri-personnel : la position est donnée en fonction des angles des articulations des membres du singe. Cependant, en prenant pour principe que l'intégration d'un déplacement est équivalent à une position, ce principe est appliqué à l'espace environnant, atteignable en effectuant des séquences d'interactions courtes.

Le principe de cette mémoire spatiale à courte portée s'inspire des observations décrites dans les expériences menées par Murata et al. [88] (Chapitre 5) et par Graziano et al. [54] (décrite ci-dessus) : on définit une position comme une région de l'espace qui peut être atteinte en effectuant une certaine séquence d'interactions, et une instance d'objet comme une configuration spatiale d'éléments de l'environnement qui afforde une interaction, indépendamment de sa position relative à l'agent. Le fait qu'une instance d'objet soit caractérisée par l'interaction qu'elle afforde, indépendamment de la séquence d'interaction permettant de l'atteindre, fait qu'elle est liée implicitement à l'objet qui afforde cette interaction : la position, caractérisée par la séquence d'interactions permettant d'atteindre l'instance d'objet, ne modifie que la façon dont l'agent expérimente l'objet (depuis la position initiale), et non l'objet lui-même.

Ce mécanisme implémente très fidèlement la notion d'espace représentatif de Poincaré [107]. En effet, nous représentons une position de l'espace comme une séquence d'interactions permet-

tant de l'atteindre, et non comme un ensemble de coordonnées ou de régions d'espace dotées de propriétés reflétant leur position.

Notons qu'une même région de l'espace peut être atteinte par plusieurs séquences d'interactions. Cette redondance est utile et est exploitée par notre mécanisme de suivi des objets en mémoire. Il est ainsi possible de caractériser et localiser l'ensemble des éléments de l'espace proche de l'agent à l'aide d'une liste de séquences d'interaction, sans faire appel à une représentation explicite de l'espace. La portée de la mémoire spatiale dépend bien sûr de la longueur maximale des séquences d'interactions utilisées. L'espace proche de l'agent se résume ainsi à l'ensemble des séquences d'interactions qu'il peut énoncer.

Mécanisme de sélection

L'étude de Murata et al. [88] a montré que les neurones du cortex prémoteur ventral s'activent lorsqu'un geste est possible, et ce, que l'animal effectue le geste ou non. Il est donc raisonnable de penser que ces neurones intègrent l'ensemble des gestes qu'un agent peut effectuer dans le contexte environnemental courant. Notre mécanisme repose sur ce principe : à chaque cycle de décision, le mécanisme de détection des objets (décrit au chapitre 5) détermine l'ensemble des interactions que l'agent peut énoncer, en incluant, bien sûr, les séquences d'interactions abordées par les objets situés dans l'espace péri-personnel. Le mécanisme de décision sélectionne alors l'interaction possible la plus profitable à l'agent, en se référant à sa motivation interactionnelle. Notons que ce mécanisme de décision rudimentaire conditionne l'agent comme purement réactif à son contexte. Les implémentations de ce mécanisme ont montré qu'il est suffisant dans un environnement simple adapté pour un agent sans perception lointaine.

Perception active

La perception active consiste à interagir avec son environnement pour l'explorer et récupérer l'information d'exploration. La perception active implique qu'un agent élabore dans un premier temps l'information dont il a besoin, et dans un second temps, qu'il énonce l'interaction permettant d'obtenir cette information.

La perception active est cohérente avec la théorie de Piaget selon laquelle il n'y a pas de séparation *a priori* entre l'action et la perception. Une perception est donc construite à partir de l'exécution d'un schéma sensorimoteur de l'agent dans le but d'obtenir une information sur le système agent-environnement. Cette information est liée non seulement à la perception, mais également à l'action qui lui a donné naissance.

L'information manquante peut être définie grâce aux signatures d'interaction (Chapitre 5). En effet, nous avons vu que l'on pouvait considérer les interactions et les objets qui les abordent comme équivalents. Les signatures d'interaction établissent un lien entre les objets qui abordent les interactions et la façon dont on peut expérimenter leur présence. On peut donc définir l'information manquante d'une interaction intentionnelle comme l'énonciation des interactions permettant la détection de l'objet qui aborde l'interaction intentionnelle. Ainsi, l'intention précède la perception.

Définir comment obtenir une information manquante revient à déterminer, pour chaque information observable, l'interaction qui permet de l'obtenir. Les mammifères disposent d'une structure cérébrale, le colliculus, qui gère les saccades oculaires, mais est également responsable des mouvements de la tête et du tronc nécessaires à l'exécution des saccades oculaires [82]. Le colliculus crée un lien entre les positions de l'espace et les mouvements des yeux, de la tête et du tronc permettant d'observer cette position. Notons que le colliculus tient compte des positions de l'espace dans le référentiel de la tête et non des yeux, ce qui implique la capacité de pouvoir changer de référentiel. Nous nous inspirons de cette structure simple pour implémenter un mécanisme associant les informations du contexte de l'agent et les interactions qui permettent de les obtenir.

Le fait d'utiliser des interactions dans le but d'obtenir des informations sur l'environnement se rapproche de la notion d'actions sensorielles épistémiques de Gatti [41], définie comme une action « *dans laquelle l'agent cognitif structure sa propre action sensorielle dans le but de recevoir de l'environnement une sensation structurée qui porte de l'information* »⁷. Certaines interactions de l'agent peuvent ainsi être énoncées dans le but de recueillir des informations sur l'environnement, plutôt que de satisfaire la motivation interactionnelle de l'agent.

Hypothèse vestibulaire

La plupart des êtres vivants disposent d'organes sensoriels leur permettant de mesurer leurs déplacements dans l'espace. On pense notamment au système vestibulaire des vertébrés [61], qui mesure les translations et les rotations. D'autres animaux plus simples, comme les insectes, disposent de poils sensitifs leur permettant de mesurer la vitesse relative de l'air. Le fait que la plupart des êtres vivants disposent de tels sens semble indiquer qu'un tel système fournit un avantage décisif qui a conduit à sa généralisation par le biais de l'évolution.

L'intérêt d'un tel organe sensoriel repose sur le fait que l'espace théorique des interactions possibles d'un être vivant a plus de dimensions que l'espace physique. En effet, deux interactions de l'agent avec son environnement peuvent produire un même mouvement dans l'espace.

Nous ne pouvons pas exploiter directement les informations issues d'un système vestibulaire : celui-ci étant différent d'un individu à l'autre, il n'est pas possible de supposer que les stimuli sensoriels et leurs liens avec les déplacements dans l'espace soient connus *a priori*. Cependant, nous pouvons supposer que chaque interaction produit un unique stimulus vestibulaire. En effet, l'énoncé réussi d'une interaction implique que le mouvement qu'elle génère a été effectué. Ainsi, même si il n'est pas possible de comparer *a priori* des stimuli vestibulaires différents, il est possible de regrouper des interactions produisant un même mouvement.

7. Traduit de l'anglais : « *in which the cognitive agent structures her own sensorial action in order to receive from the environment a feedback structured sensation that carries information* » [41]

7.1.2 Espace lointain

Le mécanisme d'exploitation de l'espace lointain permet à l'agent de tenir compte des objets situés dans son environnement global, et hors de portée de l'espace péri-personnel. Ce mécanisme exploite les informations fournies par la structure construite par le mécanisme de construction de la mémoire spatiale (décrit au chapitre 6), c'est-à-dire la structure de l'espace et le contenu de l'environnement courant. Il s'appuie sur l'intuition suivante : afin d'énacter une interaction, il est nécessaire de se rapprocher d'un objet qui afforde cette interaction. Se rapprocher d'un objet affordant une interaction dotée d'une forte valeur de satisfaction est donc un événement positif pour l'agent au regard de sa motivation interactionnelle. De même, se rapprocher d'un objet affordant une interaction dotée d'une valeur de satisfaction négative sera un événement négatif.

Ce mécanisme repose ainsi sur l'observation des variations de l'environnement interagi produites par l'énaction des interactions, de façon à mesurer le rapprochement ou l'éloignement des affordances du point de vue de l'agent. Ce rapprochement permet d'attribuer aux interactions une valeur d'utilité, indépendante de leur valeur de satisfaction, caractérisant leur contribution à l'énaction future d'autres interactions. Un tel système permet de tenir compte de la motivation interactionnelle de l'agent à moyen terme, en tenant compte des affordances distantes comme des interactions potentiellement énableables dans un avenir proche.

7.2 Formalisation du mécanisme d'exploitation de la Mémoire Spatiale

7.2.1 Espace péri-personnel

Le mécanisme gérant l'espace péri-personnel repose sur la possibilité de reconnaître un objet à une position accessible en énantant une séquence courte d'interactions. Le principe est similaire à celui des signatures d'interactions décrit au chapitre 5. Cette section formalise les notions utilisées par ce système : interactions composites, mémoire spatio-séquentielle et système de décision.

Les interactions composites

Une *interaction composite* est une structure basée sur une séquence d'interactions permettant de caractériser un objet à une certaine position de l'espace.

Définition 28. *Interaction composite* : une *interaction composite* φ est une séquence de n interactions primitives constituée d'une séquence d'interactions finale φ_f , caractérisant l'objet associé à l'interaction composite, et d'un path φ_p (les k premiers éléments de la séquence), éventuellement vide, caractérisant la position de cet objet :

$$\begin{aligned} \varphi &= [i_1, \dots, i_n], \quad n \in \mathbb{N} \\ \varphi_p &= [i_1, \dots, i_{k-1}], \quad \varphi_f = [i_k, \dots, i_n], \quad k \in [1; n] \end{aligned} \tag{7.1}$$

On note Φ l'ensemble des interactions composites possibles. La valeur de satisfaction d'une interaction composite est définie comme la somme des valeurs de satisfaction des interactions primitives qui la composent.

La séquence finale caractérise l'objet associé à l'interaction composite. Le *path* définit le "chemin" à parcourir pour atteindre cet objet, et ainsi permettre l'énaction de la séquence finale. Le *path* caractérise ainsi la position de l'espace à partir de laquelle l'interaction finale peut être énoncée. Notons que cette position est définie de manière implicite : le *path* permet en effet d'atteindre une position inconnue de l'espace, mais cette information reste exploitable car le *path* décrit comment l'atteindre. Le mécanisme de la mémoire spatiale péri-personnelle se passe donc de représentation topographique de l'espace, et implémente la notion d'*espace représentatif* défini par Poincaré [107].

Notons que dans le cas du modèle PRI, le path ne peut être constitué que d'interactions primaires. En effet, seul le mouvement provoqué par l'interaction primaire d'un ensemble énoncé E_t est nécessaire pour définir le déplacement nécessaire pour atteindre un objet.

Dans ce chapitre, ainsi que dans les expérimentations, nous afficherons les interactions composites comme indiqué dans la Figure 7.1.



FIGURE 7.1 – Affichage d'une interaction composite. Cette interaction composite caractérise l'objet qui afforde la séquence d'interaction $[i_k, \dots, i_n]$, localisée à une position atteignable en énonçant la séquence $[i_1, \dots, i_{k-1}]$.

Il faut noter que l'objet associé à l'interaction composite est défini uniquement par la séquence finale, car seul le résultat de l'énaction de la séquence finale est pris en compte ; le path indique seulement la position de l'objet, indépendamment de son accessibilité réelle. Nous définissons ainsi la propriété suivante :

Propriété 12. *deux interactions composites composées d'une même séquence finale caractérisent un même type d'objet, localisé à deux positions différentes en espace égocentré. Les objets qui caractérisent ces deux interactions composites sont des objets similaires.*

Les interactions composites peuvent être manipulées et énoncées par l'agent comme de simples interactions. Une interaction composite peut ainsi être intentée par l'agent. Celui-ci exécute alors l'ensemble de la séquence qui compose l'interaction composite. Le résultat de l'énaction d'une interaction composite est défini par un succès si la séquence a été énoncée avec succès, et comme un échec si l'une des interactions de la séquence finale a échoué. En revanche, si une interaction du path a échoué, l'interaction composite est considérée comme abandonnée. En effet, seule la

séquence finale caractérise l'objet qui afforde l'interaction composite. Un échec d'une interaction primitive du path indique seulement que l'objet n'était pas accessible par le chemin défini par le path. Tout comme les interactions primitives, les interactions composites possèdent leur propre signature d'interaction, que l'agent doit apprendre en expérimentant ces interactions. Notons qu'une interaction primitive est équivalente à une interaction composite de longueur 1, dont le path a une longueur nulle.

Nous définissons un ensemble de propriétés concernant les interactions composites :

Propriété 13. *une interaction composite $\varphi_2 = [i_{\varphi_2,1}, \dots, i_{\varphi_2,n}]$ est dite inscrite dans une interaction composite $\varphi_1 = [\varphi_{1p}; \varphi_{1f}]$, avec $\varphi_{1p} = [i_{\varphi_1,1}, \dots, i_{\varphi_1,k}]$ si $n \leq k$ et si $\forall a \in [1; n], i_{\varphi_2,a} = i_{\varphi_1,a}$*

Propriété 14. *une interaction composite $\varphi_2 = [\varphi_{2p}; \varphi_{2f}]$ est dite suivante d'une interaction composite $\varphi_1 = [\varphi_{1p}; \varphi_{1f}]$, si $\varphi_{1p} = [i_1, i_2, \dots, i_n]$ et $\varphi_{2p} = [i_2, \dots, i_n]$, c'est-à-dire que le path de φ_2 est identique au path de φ_1 moins le premier élément, et si $\varphi_{2f} = \varphi_{1f}$.*

On définit ensuite l'énectabilité d'une interaction :

Définition 29. *Énectabilité d'une interaction primitive : une interaction primitive est dite énectable si elle peut être énectée avec succès avec une certitude de succès suffisante.*

Définition 30. *Énectabilité d'une interaction composite : une interaction composite est définie comme énectable si la séquence finale peut être énectée avec succès avec une certitude de succès suffisante et si le path est énectable.*

Définition 31. *Possibilité d'une interaction composite : une interaction composite est définie comme possible si la séquence finale peut être énectée avec succès avec une certitude de succès suffisante, en supposant le path énectable.*

L'énectabilité du path d'une interaction composite est déterminé de façon récursive, à l'aide des interactions *inscrites* dans cette interaction.

Les notions d'*énectabilité* et de *possibilité* impliquent des propriétés très différentes. Ainsi, la possibilité d'une interaction caractérise la présence de l'objet qui énecte sa séquence finale, mais pas l'accessibilité de cet objet. Il n'est en effet pas nécessaire de savoir si un objet est accessible pour caractériser sa présence dans l'environnement de l'agent. L'énectabilité permet de préciser l'accessibilité de l'objet. Cette information est en effet requise dans le cas où l'agent doit énecter une interaction composite. Une interaction possible mais non énectable ne pourra donc pas être proposée par le système de décision.

De même qu'il est possible de caractériser la présence d'un objet, on peut caractériser son absence. L'absence d'un objet peut être caractérisée par une interaction composite dont la séquence d'interaction finale est prédite comme un échec :

Propriété 15. *Impossibilité d'une interaction composite : une interaction composite est définie comme impossible si la séquence finale n'est pas énectable, avec une certitude d'échec suffisante, en supposant le path énectable.*

Si l'accessibilité d'un objet est nécessaire pour énoncer l'interaction composite qui le caractérise, elle est également nécessaire pour tester une interaction composite. En effet, seul le résultat de la séquence finale est pris en compte. Il est donc nécessaire que le path soit énonçable :

Propriété 16. *on définit une interaction composite comme testable si son path est énonçable.*

La *testabilité* d'une interaction permet de garantir, avec une certitude considérée comme suffisante, que l'énonciation d'une interaction composite se terminera par un succès ou par un échec, permettant son apprentissage. Notons que par définition, une interaction composite dont le path a une longueur nulle est toujours *testable*.

Apprentissage des interactions composites

L'apprentissage des interactions composites repose sur l'observation de la séquence des n dernières interactions énoncées au cycle de décision t , avec $n \in \mathbb{N}$ la longueur maximale des interactions composites que l'agent peut apprendre. Notons que cette limite caractérise la portée de la mémoire spatiale, et donc les limites de l'espace péri-personnel de l'agent. Cet apprentissage est hiérarchique, les interactions composites apprises par l'agent ont une longueur croissante au cours du temps. En effet, l'apprentissage des interactions composites de grande longueur nécessite de connaître les interactions composites plus courtes qui la composent.

L'apprentissage des interactions composites est basé sur le principe suivant : soit Ξ_t la séquence des n dernières interactions énoncées aux cycles de décision $t-n$ à t . Soit Φ_t l'ensemble des interactions composites connues de l'agent au cycle de décision t , $\Phi_t \in \Phi$. À chaque cycle de décision, l'agent apprend toutes les interactions composites inscrites dans Ξ_t qui ne sont pas dans Φ_t . Cependant, il convient de limiter cet apprentissage pour garantir un apprentissage hiérarchique des interactions composites, en ajoutant les conditions suivantes :

- une interaction composite φ peut être ajoutée à Φ_{t+1} s'il existe une sous séquence *inscrite* dans son path φ_p qui dispose d'une fiabilité suffisante ou dont la signature d'interaction est vide.
- une interaction composite φ peut être ajoutée à Φ_{t+1} s'il existe une sous séquence *suivante* qui dispose d'une fiabilité suffisante ou dont la signature d'interaction est vide.

La première condition assure qu'il est possible de définir la testabilité de l'interaction composite avec une fiabilité suffisante. La seconde assure que la mise à jour de l'interaction composite, décrite dans la section suivante, conduit à une interaction composite fiable, condition nécessaire pour limiter les incohérences lorsque l'on cherche à améliorer le contexte interactionnel par ajout de signatures. Notons que dans le cas de signatures vides, on ajoute des interactions composites dont la testabilité ne peut pas être définie. Cependant, ces interactions composites permettent de suivre des objets qui échappent au système interactionnel de l'agent.

Principe de la Mémoire Spatio-Séquentielle (MSS)

Le principe de ce mécanisme est basé sur le fait qu'une interaction composite reconnue comme possible permet de localiser un objet par la séquence d'interactions primitives nécessaires pour

l'atteindre. En effet, si un agent effectue la première interaction du chemin d'une interaction composite considérée comme possible (i.e. l'objet associé est défini comme présent), on peut "retirer" cette interaction du chemin de l'interaction composite initiale, puisqu'elle a été parcourue par l'agent. On obtient alors une nouvelle interaction composite caractérisant la présence du même objet, puisque la séquence finale est la même, mais localisé à une position différente par rapport à l'agent. Cette nouvelle position est caractérisée par le nouveau path. La faisabilité de cette nouvelle interaction composite implique que l'objet qui l'afforde est présent, même dans le cas où le contexte interactionnel ne permet plus de le percevoir.

Nous définissons une fonction de mise à jour m des interactions composites, définie par :

$$m : \Phi \times I \rightarrow \Phi \cup \emptyset$$

avec φ' l'interaction composite *suivante* de φ , si celle-ci est définie et si le premier élément du path de φ est i .

Le mécanisme de la mémoire spatiale gère la liste des interactions considérées comme *possibles*, à laquelle est appliquée une fonction de mise à jour permettant de suivre les objets qui affordent les interactions composites. Une seconde liste est utilisée pour suivre les interactions composites considérées comme *impossibles*. En effet, une interaction composite impossible indique que la séquence qui la compose n'est pas énonçable. Cette information est utile pour définir si le path d'une interaction composite est énonçable. Nous définissons ainsi la mémoire spatiale comme une structure informationnelle constituée de deux listes : \mathcal{T} (True), la liste des interactions composites considérées comme *possible*, et \mathcal{F} (False) la liste des interactions considérées comme *impossibles*. Le mécanisme de mise à jour de la mémoire est illustré dans la Figure 7.2.

On notera que le suivi d'un objet n'est possible que si sa présence est caractérisée par une interaction composite dont le path commence par la prochaine interaction énoncée. Dans ce cas, on observe un effet mémoire permettant de conserver et suivre les objets qui entourent l'agent. Cet effet se traduit par deux propriétés, qui dépendent des signatures des interactions composites :

- si une interaction composite obtenue après mise à jour dispose d'une signature non vide, c'est-à-dire que l'objet qui l'afforde peut être expérimenté, alors les interactions qui composent cette signature peuvent être utilisées pour compléter le contexte interactionnel. Dans le cas d'une interaction composite considérée comme possible, l'objet qui afforde cette interaction est considéré comme présent, ce qui implique le succès des interactions caractérisant la présence de l'objet. Dans le cas d'une interaction composite considérée comme impossible, l'objet est considéré comme absent, ce qui implique l'échec des interactions caractérisant la présence de cet objet. Il faut cependant tenir compte des certitudes de présence ou d'absence de chaque interaction. Ce contexte augmenté peut évoquer de nouvelles interactions composites, caractérisant le même objet, mais atteignable par des chemins différents, qui permettront par la suite de continuer le suivi de l'objet.

- si une interaction composite obtenue après mise à jour dispose d'une signature vide, c'est-à-dire que l'objet qui l'afforde échappe au système interactionnel de l'agent, alors cette interaction

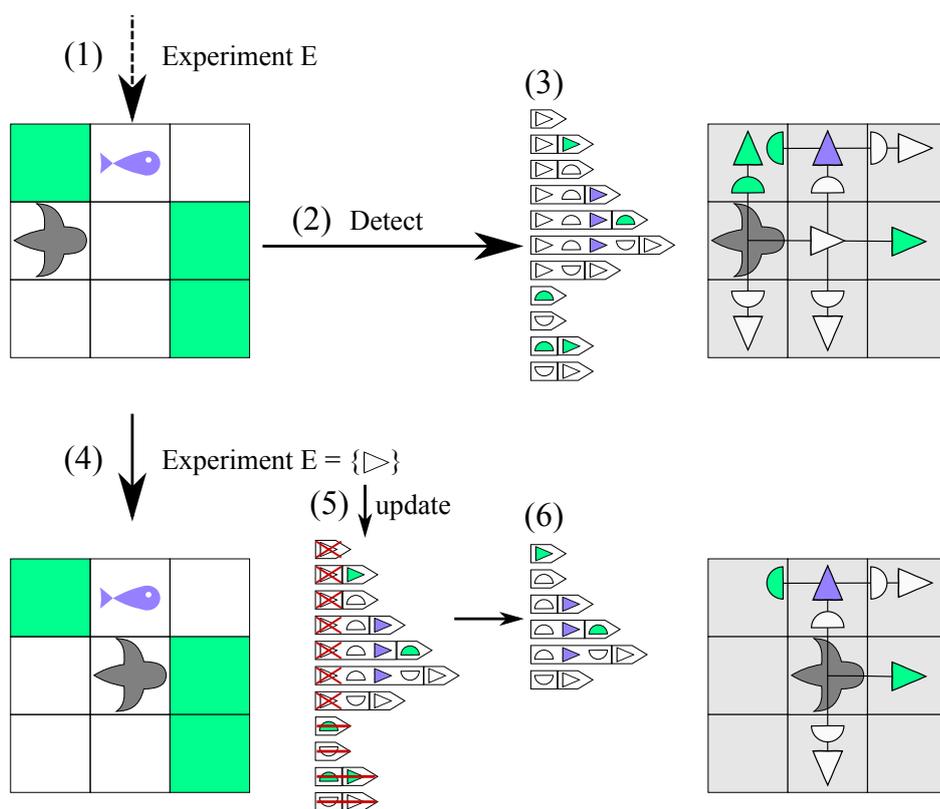


FIGURE 7.2 – Principe de mise à jour de la mémoire spatio-séquentielle : à la fin du cycle de décision t (1), le contexte interactionnel E_t permet à l'agent de détecter un ensemble d'interactions composites pouvant être effectuées (2). L'ensemble de ces interactions permet de caractériser les éléments qui entourent l'agent tout en donnant l'ensemble des interactions *possibles* (3). Notons que l'agent n'a pas accès à la "reconstruction" de l'environnement. Au cycle suivant (4), l'agent expérimente l'interaction *avancer d'un pas* (\triangleright). Le mécanisme de mise à jour (5) génère un nouvel ensemble d'interactions composites en supprimant le premier élément des interactions composites qui commencent par l'interaction énoncée, et en éliminant les interactions composites qui ne commencent pas par l'interaction énoncée. Cet ensemble permet de caractériser les éléments qui entourent l'agent (6), avant même de procéder à une nouvelle détection d'interactions composites.

composite renseigne l'agent sur la présence de cet objet et la façon de l'atteindre. En effet, l'existence de cette interaction composite implique l'existence de l'objet qui l'afforde. Si l'interaction composite est considérée comme énonçable ou non énonçable, alors l'objet qui afforde cette interaction composite peut être considéré comme présent (ou absent), même si celui-ci ne peut pas être détecté. Il ne sera cependant pas possible de suivre cet objet si l'agent n'effectue pas le path de cette interaction composite. Le principe permettant de compléter la mémoire spatio-séquentielle est illustré par la Figure 7.3 .

Ce mécanisme de mise à jour permet également d'effectuer des simulations : on peut en effet

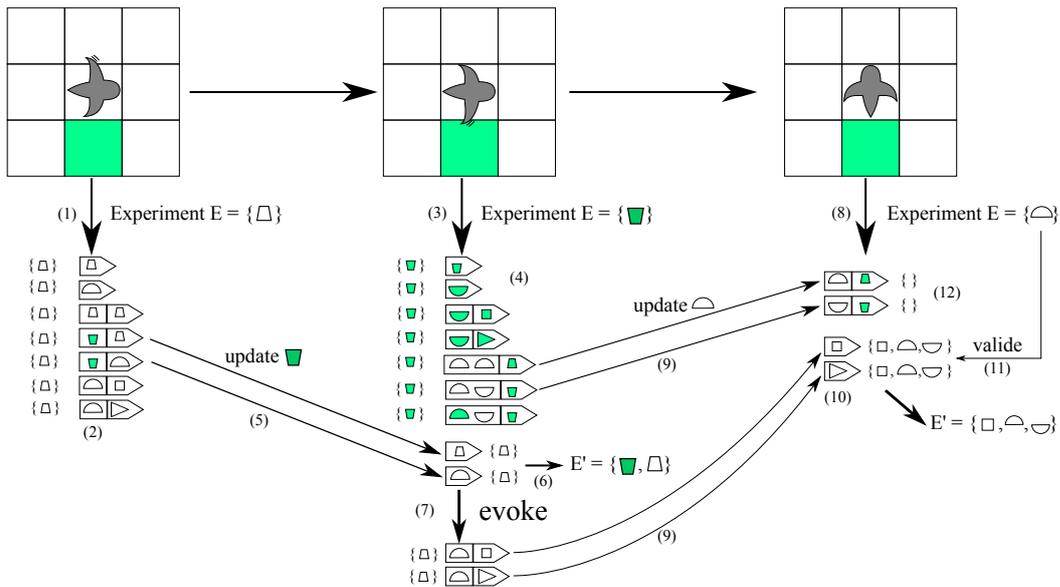


FIGURE 7.3 – Illustration de l'effet mémoire lié aux interactions composites. Seule une partie de la mémoire \mathcal{T} (True) est affichée. Au cycle de décision t , l'agent touche un espace vide à sa gauche (1). Le contexte obtenu permet de définir un ensemble d'interactions possibles (2) (les signatures sont affichées à gauche des interactions). Au cycle suivant (3), l'agent touche un mur à sa droite (■). Un nouvel ensemble d'interactions est considéré comme possible (4). Au pas précédent, deux interactions composites commençaient par l'interaction *toucher un mur à droite*. Ces deux interactions sont mises à jour (5). Les signatures de ces interactions contiennent l'interaction *toucher un espace vide à gauche* (□). Comme ces deux interactions composites sont considérées comme possibles, on peut ajouter leur signature au contexte interactionnel (7). Ce contexte augmenté E' permet de définir deux nouvelles interactions composites comme possibles. Au cycle suivant (8), l'agent énonce l'interaction *tourner gauche face à un espace vide* (△). Quatre interactions composites peuvent être mises à jour (9). On obtient d'un côté les interactions composites [*toucher un espace vide devant* □] et [*avancer* ▷] (10), liées à un espace vide devant l'agent. On peut noter que leur signature est cohérente avec le contexte interactionnel (11). D'un autre côté, on obtient les interactions composites [*tourner à gauche face à un espace vide* △][*toucher un mur gauche* ■] et [*tourner à droite face à un espace vide* ▽][*toucher un mur droite* ■] (12). Ces deux interactions sont liées à la présence d'un mur situé derrière l'agent, hors de portée de son système interactionnel. Leurs signatures sont vides et ne peuvent pas être utilisées pour compléter le contexte interactionnel. En revanche, le fait que ces interactions soient considérées comme possibles implique la présence des objets qui affordent les interactions *toucher un mur à gauche* (■) et *toucher un mur à droite* (■), accessibles en énonçant respectivement les interactions *tourner à gauche* (△) et *tourner à droite* (▽). Si l'agent énonce l'une de ces deux interactions, l'un de ces objets sera suivi et permettra de compléter le contexte interactionnel au pas suivant.

appliquer une mise à jour de la mémoire à partir d'une interaction sans l'énacter, ce qui permet d'obtenir une mémoire spatiale et un contexte interactionnel simulés. Ce contexte peut à son tour ajouter de nouvelles interactions composites à la mémoire simulée. La simulation peut ainsi continuer jusqu'à ce que la mémoire simulée ne contienne plus d'interactions composites. Comme nous le verrons plus tard, ce système de simulation sera utile pour définir des interactions menant à un contexte interactionnel pertinent pour un apprentissage, ou l'énaction d'une interaction dotée d'une valeur de satisfaction élevée.

Ce mécanisme permet également d'améliorer l'apprentissage des signatures d'interactions composites. En effet, si l'agent énonce la première interaction composite, et que l'interaction obtenue après mise à jour est une interaction considérée comme possible avec une certitude élevée dans le contexte courant, et dont la signature est considérée comme fiable, alors nous pouvons considérer que l'interaction initiale est un succès (ou un échec si l'interaction obtenue est considérée comme impossible). Il est donc possible de renforcer la signature de l'interaction initiale sans avoir à l'énacter entièrement, et ce, même si le path n'était pas énonçable : la présence de l'objet qui afforde l'interaction mise à jour est en effet indépendante de son accessibilité. Il est également possible de détecter des erreurs dans les signatures des interactions, notamment si la mémoire contient des signatures incompatibles entre elles, ou incompatibles avec le contexte interactionnel.

Ce mécanisme permet donc de localiser et suivre des éléments en mémoire, même si ceux-ci échappent au système interactionnel de l'agent, et de compléter le contexte interactionnel, et ce, sans utiliser une structure topographique pour représenter l'espace. Les informations fournies par ce mécanisme permettent de prédire le résultat d'interactions et d'effectuer des simulations pour prédire le contexte après leur énonciation. L'espace qui entoure l'agent est caractérisé par l'ensemble des interactions énonçables ou potentiellement énonçables.

Mécanisme d'exploitation de la mémoire spatiale

Le mécanisme d'exploitation permet à un agent de générer des comportements en exploitant le contenu de sa mémoire spatiale. Ce mécanisme a pour vocation de définir les interactions que l'agent peut effectuer dans le contexte courant. La décision de l'interaction *intention* i_{t+1} du cycle suivant dépend d'un mécanisme de décision qui exploite les informations issues du mécanisme d'exploitation de la mémoire. Le mécanisme d'exploitation agit ainsi comme un filtre qui élimine les interactions qui ne peuvent pas être énoncées avec succès. Afin de tester ce mécanisme, nous utiliserons un système de décision rudimentaire ne tenant compte que des interactions sélectionnées par le mécanisme d'exploitation et de leur valeur de satisfaction propre.

Le mécanisme d'exploitation est composé de deux systèmes de sélection : le premier implémente une forme de curiosité qui pousse l'agent à apprendre et renforcer ses signatures d'interaction, l'autre définit la liste des interactions composites énonçables. Il faut noter que le mécanisme d'apprentissage dépend de l'implémentation des signatures d'interaction. Enfin, le mécanisme de décision, pouvant tenir compte d'autres mécanismes, sélectionne l'interaction la plus profitable

de cette liste.

Mécanisme d'apprentissage

Le premier système de sélection implique de définir la fiabilité d'une interaction composite, c'est-à-dire le degré de confiance que l'on peut accorder à sa signature. Nous noterons $\phi_{i,t}$ une grandeur caractérisant la fiabilité de l'interaction composite i au cycle de décision t . La fiabilité peut reposer sur la valeur de certitude de succès ou d'échec d'une interaction, ou sur le nombre de prédictions correctes de succès ou d'échec.

Une première solution repose sur le nombre de prédictions correctes consécutives de succès et d'échecs. La fiabilité est définie par :

$$\phi_{i,t} = \min(nb_{success}^t, nb_{faillure}^t) \quad (7.2)$$

avec $nb_{success}^t$ le nombre de prédictions correctes consécutives de succès et $nb_{faillure}^t$ le nombre de prédictions correctes consécutives d'échec au cycle de décision t . Ces nombres sont incrémentés à chaque prédiction (de succès ou d'échec) qui s'avère correcte, et remis à zéro en cas d'erreur.

Cette définition de la fiabilité est recommandée lorsque le système interactionnel et le mécanisme d'apprentissage des signatures d'interaction nécessitent une grande fiabilité des signatures d'interaction. En effet, une erreur de prédiction met la fiabilité d'une interaction à zéro. Nous avons utilisé cette définition dans la première expérimentation.

La seconde définition repose sur les certitudes de succès et d'échec. En effet, une interaction est soit possible, soit impossible. Les certitudes doivent donc tendre vers $c_1(i, E_t) = 1$ et $c_2(i, E_t) = 0$ pour la prédiction d'un succès et vers $c_1(i, E_t) = 0$ et $c_2(i, E_t) = 1$ pour la prédiction d'un échec. La fiabilité est donc définie par :

$$\phi_{i,t} = |c_1(i, E_t) - c_2(i, E_t)| \quad (7.3)$$

Cette méthode permet de repérer les interactions qui ont été peu testées dans le contexte interactionnel actuel.

Le mécanisme d'apprentissage proposé sélectionne l'interaction considérée comme la moins fiable dans le but de tester le résultat de son énoncé dans le contexte actuel si cela s'avère nécessaire. Une interaction est sélectionnée selon les conditions suivantes : dans un premier temps, on cherche l'interaction composite la moins fiable dont le path est considéré comme fiable et énonçable dans le contexte interactionnel courant. Si la fiabilité de cette interaction est considérée comme insuffisante (i.e. inférieure à un seuil), elle sera sélectionnée. Il est cependant possible d'utiliser des conditions plus permissives concernant la validité du path : nous proposons en effet de considérer une interaction composite candidate comme *testable* si son path n'est pas prédit comme un échec. Dans le cas où le path d'une interaction échoue trop fréquemment, cette condition peut être supprimée (de façon permanente ou non). Cette condition s'est avérée nécessaire pour permettre l'apprentissage des interactions composites dont le résultat du path ne

peut être prédit, car constitué d'une interaction composite dont la signature est vide. On peut noter que ce mécanisme ne retourne qu'une seule interaction composite au système de décision. Il s'agit d'un moyen de forcer la sélection de cette interaction pour permettre son apprentissage.

Mécanisme d'exploitation

Le mécanisme d'exploitation de la mémoire spatiale guide le comportement de l'agent. Il est utilisé si aucune interaction n'est sélectionnée par le mécanisme d'apprentissage. Ce mécanisme sélectionne simplement les interactions composites considérées comme testables et non impossibles, c'est-à-dire les interactions pour lesquelles le path est énable et dont le résultat n'est pas prédit comme un échec :

$$\{\varphi_k\} \subseteq \Phi_t / (\varphi_k \text{ testable} \wedge c_2(\varphi_k, E_t) < \tau) \quad (7.4)$$

La liste des interactions ainsi constituée est ensuite utilisée par un mécanisme de décision qui sélectionne l'interaction intention du cycle de décision suivant, i_t . Le mécanisme de décision peut se baser sur cette liste, ou simplement l'utiliser pour filtrer les décisions d'autres mécanismes de sélection.

Système de décision

Pour nos expériences, nous avons défini un système de décision rudimentaire qui sélectionne l'interaction composite disposant de la plus grande valeur de satisfaction parmi les interactions fournies par la mémoire spatiale. L'interaction intention sera sélectionnée par :

$$i_{t+1} = i \in \{i_k\} / s_i = \max_{i_k}(s_{i_k}) \quad (7.5)$$

L'agent réagit ainsi uniquement aux objets présents dans son contexte courant. La sélection de l'interaction intention sera différente selon si l'interaction doit être énable dans le but d'apprendre sa signature d'interaction (par le biais du mécanisme d'apprentissage) ou dans le but de satisfaire la motivation interactionnelle de l'agent (par le biais du mécanisme d'exploitation).

Dans le cas d'un apprentissage, deux cas sont possibles : si le contexte est favorable à l'apprentissage de l'interaction sélectionnée i_{t+1} , c'est-à-dire qu'il permet de tester et renforcer la signature de cette interaction, alors l'agent énable directement cette interaction. Dans le cas contraire, le mécanisme de décision sélectionne une *interaction épistémique* $i_{epistémique}$ permettant d'obtenir un contexte interactionnel E_{t+1} favorable à l'apprentissage de l'interaction sélectionnée. On met ici à contribution les capacités de simulation de la mémoire spatiale pour déterminer si le contexte après avoir énable l'interaction épistémique (en supposant qu'elle réussisse) offre des conditions favorable à l'apprentissage. L'agent énable d'abord cette interaction épistémique $i_{epistémique}$. Si le contexte interactionnel obtenu est favorable à l'apprentissage, l'agent énable l'interaction i_{t+1} initialement sélectionnée. Dans le cas contraire, la sélection est annulée et le

mécanisme de sélection est à nouveau sollicité pour sélectionner une nouvelle liste d'interactions à partir du nouveau contexte interactionnel.

Dans le cas d'une exploitation, on observe deux cas similaires : si l'interaction i_{t+1} sélectionnée est considérée comme énable, l'agent énable cette interaction. Dans le cas contraire, le mécanisme de décision sélectionne une interaction épistémique permettant d'obtenir un contexte interactionnel pouvant définir si l'interaction choisie est énable. L'agent énable l'interaction épistémique, puis, si l'interaction i_{t+1} peut être définie comme énable, l'interaction sélectionnée initialement est énable. Si par contre l'interaction est définie comme impossible, le mécanisme de sélection est à nouveau sollicité.

Notons que nous ne définissons pas de périodes d'apprentissage et d'exploitation. Les mécanismes sont utilisés en fonction de la fiabilité des interactions composites. Cependant, plus les interactions composites sont fiables, moins le mécanisme d'apprentissage sera utilisé.

Nos implémentations ont montré que le mécanisme d'apprentissage n'est pas forcément nécessaire si la longueur maximale des interactions composites est très limitée. En revanche, il s'avère indispensable dès que l'on veut augmenter la portée de la mémoire spatiale. En effet, les interactions composites disposant d'un path long nécessitent de connaître les signatures des interactions composites constituant leur path. Les erreurs peuvent ainsi s'accumuler jusqu'à rendre ces interactions composites inutilisables, car leurs signatures ne reflètent plus les objets qui les affordent.

Perception active

Nous avons défini un mécanisme permettant à l'agent de détecter les objets qui l'entourent et de suivre leurs positions dans l'espace. Cependant, le contexte et les informations fournies par la mémoire spatiale ne permettent pas toujours de définir le résultat d'une interaction sélectionnée. Nous proposons un mécanisme implémentant une forme de perception active permettant à l'agent de sélectionner des interactions dans le but d'obtenir activement des informations sur son environnement, indépendamment de leurs valeurs de satisfaction. Ce mécanisme se base sur la propriété selon laquelle une interaction énable informe l'agent sur certains aspects de l'environnement. Un agent peut donc tenter d'énabler une interaction pour savoir si l'objet post-énabler qui lui est associé est présent, et donc de détecter la présence de l'objet qui afforde l'interaction sélectionnée.

Ce mécanisme détermine les interactions permettant de générer un contexte interactionnel contenant les informations requises par le système de sélection. Nous appelons *énabler épistémique* un cycle de décision utilisé dans le but d'obtenir des informations sur l'environnement, et *interaction épistémique* l'interaction utilisée dans ce but, en référence aux actions épistémiques de Gatti [41]. Nous exploitons ici la possibilité de simuler les effets des interactions sur la mémoire spatiale pour définir le contexte interactionnel après énable, supposée réussie, de l'interaction épistémique.

Une interaction épistémique peut être nécessaire dans deux cas : si le contexte ne permet pas de définir si une interaction est énable, ou, dans le cas d'un apprentissage, si le contexte

ne permet pas de renforcer la signature d'une interaction. Dans le premier cas, une interaction épistémique doit permettre de générer, en cas de succès, un contexte interactionnel dans lequel le résultat de l'interaction préalablement sélectionnée puisse être défini avec une certitude élevée. Le système de décision a recours à ce mécanisme si le résultat de l'énaction de l'interaction sélectionnée n'est pas défini avec une certitude suffisante, c'est-à-dire si :

$$|c_{success}(i_{t+1}, E_t)| < \tau \quad (7.6)$$

avec τ seuil de certitude et $c_{success}$ la fonction qui donne la certitude que l'interaction i puisse être énoncée avec succès dans le contexte interactionnel E .

Le but d'une interaction épistémique est de générer un contexte contenant les informations recherchées. Il faut donc réaliser une prédiction du contexte que l'on obtiendrait après énonction d'une interaction épistémique. La prédiction d'un contexte s'effectue en utilisant les signatures des interactions composites stockées dans la mémoire Spatio-Séquentielle.

Soient \mathcal{T}_t (True) la liste des interactions composites considérées comme possibles, et \mathcal{F}_t (False) la liste des interactions composites considérées comme impossibles, au cycle de décision t . On note $\mathcal{T}_{i,t+1}$ et $\mathcal{F}_{i,t+1}$ les listes d'interactions mises à jour par l'interaction i . Ces listes sont définies par :

$$\begin{aligned} \mathcal{T}_{i,t+1} &= \{m(i, \varphi_k)\} / \varphi_k \in \mathcal{T}_t \\ \mathcal{F}_{i,t+1} &= \{m(i, \varphi_k)\} / \varphi_k \in \mathcal{F}_t \end{aligned} \quad (7.7)$$

avec m la fonction de mise à jour des interactions composites. On note $E_{i_{ep},t+1}$ le contexte simulé obtenu après énonction d'une interaction épistémique i_{ep} , à partir des signatures des interactions de $\mathcal{T}_{i_{ep}}$ et $\mathcal{F}_{i_{ep}}$. Le mécanisme de sélection cherche une interaction épistémique pouvant générer un contexte tel que :

$$|c_1(i_{t+1}, E_{i_{ep},t})| \geq \tau \quad (7.8)$$

Une interaction épistémique sera sélectionnée pour l'information que son énonction peut apporter, mais également pour le contenu de la mémoire après énonction. Notons qu'une interaction épistémique n'a pas besoin d'être considérée comme énonctionnable : en effet, celle-ci était énonctionnable, alors l'information manquante pourrait être définie avec une certaine certitude grâce à sa signature. Cependant, elle ne doit pas être considérée comme impossible. L'interaction épistémique sera sélectionnée parmi les interactions candidates selon plusieurs critères :

- elle doit être *testable*, afin d'obtenir l'information dont on a besoin, caractérisée par sa séquence finale. Dans nos implémentations, les interactions épistémiques sont des interactions primitives, donc testables pour tout cycle de décision t .

- la valeur de satisfaction de l'interaction épistémique doit être maximale, parmi les interactions candidates. En effet, une interaction épistémique doit avoir un impact minimum sur la

satisfaction de l'agent⁸.

- l'interaction intention doit toujours être *testable* (i.e. le path doit être énable) après éaction de l'interaction épistémique. En effet, une interaction épistémique est inutile si son éaction rend impossible l'éaction de l'interaction intention.

Dans le cas d'un apprentissage, on cherchera à maximiser le renforcement d'une signature d'interaction. On favorise donc les contextes interactionnels dans lesquels l'interaction à tester n'a été que peu ou pas testée. L'estimation du nombre de tests d'une interaction dans un contexte interactionnel dépend bien entendu de la façon dont sont implémentés les signatures des interactions.

Hypothèse vestibulaire

La plupart des êtres vivants disposent d'organes sensoriels leur permettant d'estimer leurs déplacements dans l'espace (par exemple le système vestibulaire chez les vertébrés). En partant de ce constat, nous avons testé une variante de notre mémoire spatiale prenant en compte des informations provenant d'un mécanisme donnant des informations sur les déplacements de l'agent. Ce mécanisme fournit, à chaque interaction éactée, le *mouvement* (sans signification ni sémantique définie *a priori*) effectué. On note μ un mouvement et \mathcal{M} l'ensemble des mouvements que peut observer le système vestibulaire de l'agent. L'ensemble \mathcal{M} doit respecter l'hypothèse suivante :

Hypothèse 7. *Unicité du mouvement : l'éaction de chaque interaction primitive i de I produit un et un seul mouvement μ de \mathcal{M} .*

Chaque interaction primitive est ainsi associée à un mouvement. L'association s'effectue en observant le mouvement généré par l'éaction de l'interaction éactée. L'intérêt du système vestibulaire repose sur l'hypothèse suivante :

Hypothèse 8. *plusieurs interactions primitives peuvent produire le même mouvement.*

D'où la propriété suivante :

Propriété 17. $\text{card}(\mathcal{M}) \leq \text{card}(I)$

Nous avons modifié le mécanisme de la mémoire spatiale pour tenir compte de ces hypothèses. En premier lieu, les interactions composites sont modifiées : puisque le path correspond au mouvement nécessaire pour atteindre un objet, nous pouvons considérer un path comme une séquence de mouvements et non plus d'interactions. Une interaction composite φ peut donc s'écrire sous la forme :

8. On peut mettre cette propriété en relation avec les saccades oculaires générées par le colliculus des mammifères, qui n'impliquent des mouvements de la tête et du corps, plus contraignant que des mouvements des yeux, que si le mouvement des yeux n'est pas suffisant pour observer une position de l'espace.

$$\begin{aligned}
 \varphi &= [\varphi_p, \varphi_f] \\
 \varphi_p &= [\mu_1, \dots, \mu_{k-1}], \varphi_f = [i_k, \dots, i_n] \\
 n &\in \mathbb{N}, k \in [1; n]
 \end{aligned} \tag{7.9}$$

Avec φ_p le path de l'interaction composite φ , constitué d'une séquence de *mouvements* $\mu \in \mathcal{M}$ et φ_f la séquence finale de φ , constituée d'une séquence d'interactions primitives $i \in I$.

L'utilisation des mouvements pour caractériser le path a pour effet de réduire le nombre d'interactions composites possibles, puisque plusieurs séquences d'interactions peuvent donner une même séquence de mouvements. Par contre, le path n'étant plus défini par une séquence d'interactions, on ne peut pas définir *a priori* la valeur de satisfaction d'une interaction composite. Il n'est pas non plus possible de déterminer si le path est énable, puisque plusieurs séquences d'interactions peuvent correspondre à la séquence de mouvements. L'énable du path d'une telle interaction composite nécessite de définir une séquence d'interactions permettant de parcourir le path de cette interaction. Nous avons donc modifié le mécanisme de sélection pour reconstituer le path d'une interaction composite en fonction du contenu de la mémoire spatiale.

Le principe est le suivant :

soit une interaction composite φ dont on veut reconstituer le path φ_p de longueur k_φ .

Soit une interaction composite j de longueur $n_j \leq k_\varphi$, avec un path de longueur k_j . L'interaction j est dite *inscrite* dans φ si :

$$\forall a \in [1; k_j], \mu_{j,a} = \mu_{\varphi,a} \wedge \forall b \in [k_j + 1; n_j], m_{\varphi,b} \text{ movement of } i_{j,b} \tag{7.10}$$

Ainsi, si j est considérée comme possible, alors il est possible d'effectuer une partie du path φ_p en énant les interactions qui composent la séquence finale de j . Reconstruire le path d'une interaction composite revient à définir un ensemble d'interactions composites possibles et inscrites dans ce path tel que chaque mouvement du path puisse être remplacé par une interaction primitive.

Notons que si plusieurs interactions peuvent remplacer un mouvement, on choisira celle disposant de la plus grande valeur de satisfaction. La séquence permettant d'énter le path sera donc toujours la séquence offrant la valeur de satisfaction la plus élevée parmi les séquences possibles. Le mécanisme d'exploitation de la mémoire spatio-séquentielle fournit ainsi une liste d'interactions composites "reconstituées" considérées comme énable. La valeur de satisfaction d'une interaction composite est définie à partir de l'interaction reconstituée, et calculée comme la somme des valeurs de satisfactions des interactions primitives qui la composent.

Une variante de la première expérimentation, implémentant le système vestibulaire nous a permis d'observer une amélioration significative de l'apprentissage des signatures d'interaction : l'apprentissage est plus rapide, et, comme les interactions composites sont moins nombreuses et plus souvent testées, leurs signatures sont plus cohérentes avec les objets qui affondent ces interactions. Il faut cependant noter que l'hypothèse vestibulaire n'apporte un avantage que dans le cas où un grand nombre d'interactions primitives génèrent un même mouvement.

7.2.2 Espace extra-personnel

Hypothèses

Le principe de la mémoire spatiale à longue portée repose sur l'hypothèse de l'espace H1, impliquant que la structure de l'espace est invariante. Cela signifie qu'un objet affordant une interaction i peut être déplacé par rapport à l'agent sans que ses propriétés (taille, forme...) ne soient modifiées. Une certaine configuration d'éléments de l'environnement peut donc, si sa position relative par rapport à l'agent est modifiée, correspondre à l'objet qui afforde une interaction.

Le mécanisme de construction d'une mémoire spatiale, vu au chapitre précédent, permet la construction d'une structure informationnelle qui permet de localiser des instances d'objet et de définir leur position sous la forme de la transformation spatiale permettant de retrouver ces objets. Contrairement à la mémoire spatiale péri-personnelle, on ne cherche pas à définir une signature d'interaction pour les instances d'objets pour chaque transformation géométrique, mais à définir la position des instances d'objet dans le but de s'en rapprocher ou de s'en éloigner.

Le mécanisme d'exploitation d'une mémoire spatiale à longue portée doit permettre à l'agent de satisfaire sa motivation propre à court et moyen terme. C'est-à-dire qu'on ne cherche pas seulement à énoncer les interactions possibles les plus satisfaisantes, comme c'était le cas avec la mémoire péri-personnelle, mais on doit également tenir compte des instances d'objets lointains qui pourraient permettre l'énonciation future d'interactions. Le principe du mécanisme que nous proposons consiste à attribuer une valeur d'utilité aux interactions en fonction du rapprochement ou de l'éloignement de l'ensemble des instances d'objet distantes qu'elles permettent, et de la valeur de satisfaction des interactions affordées par ces instances d'objet.

Afin d'étudier le mécanisme d'exploitation et son fonctionnement, nous proposons dans un premier temps de formaliser un mécanisme exploitant une version simplifiée et codée en dur de la mémoire spatiale, comparable à celle utilisée par un agent implémentant un modèle SRI. Nous formaliserons ensuite une version du mécanisme exploitant la structure construite par ce mécanisme.

Mémoire spatiale simplifiée

Nous proposons un modèle simplifié de la structure construite par le mécanisme permettant de construire la mémoire spatiale, afin de se focaliser sur l'étude du mécanisme exploitant cette structure. La mémoire spatiale formalisée dans cette section se base sur les hypothèses du modèle Spatial RI (SRI).

Formalisation de la mémoire spatiale

Cette structure simplifiée est constituée d'un espace fini de même nombre de dimensions que l'environnement de l'agent. La mémoire spatiale est constituée d'un ensemble de positions p , définies comme des régions indivisibles de l'espace dans laquelle une interaction énoncée peut être localisée, de façon directe (par son énonciation) ou indirecte (par son suivi). Une partie des

positions est définie par l'ensemble des positions dans lesquelles une interaction peut être énoncée, que l'on note P . L'espace non observable peut être discrétisé en positions de façon arbitraire. Il est cependant plus judicieux de conserver une densité de position homogène avec celle de l'espace observable, afin que le déplacement des interactions énoncées reste homogène. L'ensemble des positions de l'espace global est noté P' .

Dans le modèle SRI (décrit au chapitre 4), une interaction énoncée est associée à la position p où elle a été énoncée, pour former un *acte*. Les actes énoncés sont placés sur la mémoire spatiale lorsque la position est définie dans l'espace. À chaque acte placé en mémoire, on associe une valeur o , incrémentée à chaque cycle de décision pour caractériser l'âge de cet acte. Cet âge permet de définir la fiabilité de la position de l'acte, et de le supprimer lorsqu'il est trop ancien. En effet, nous ne cherchons pas à cartographier l'environnement : la mémoire spatiale constitue une mémoire à court terme permettant de définir un contexte spatial exploitable pour l'agent.

Afin de mettre à jour la position des actes stockés en mémoire, on définit, pour chaque interaction primitive et primaire i , la transformation géométrique τ associée au déplacement de l'environnement par rapport à l'agent provoqué par l'énoncé de i , que l'on note τ_i . Dans la mémoire spatiale, une transformation géométrique se traduit par un déplacement d'une position p vers une autre position p' . Une transformation géométrique τ peut ainsi se définir par la fonction suivante :

$$\tau : P' \rightarrow P' \cup \{null\}$$

Notons que comme l'ensemble des positions P' est discret, la transformation appliquée à la mémoire spatiale est une approximation de la transformation géométrique τ . Notons également qu'un acte, déplacé par une transformation géométrique τ , peut se trouver en dehors de la mémoire spatiale. La position obtenue n'est alors pas définie par la mémoire spatiale (position *null*). Un élément qui sort de la mémoire spatiale est supprimé, car considéré comme hors de portée de la mémoire.

La mémoire spatiale est mise à jour de la façon suivante : à chaque cycle de décision, les actes présents en mémoire sont déplacés selon la transformation géométrique τ_{e_t} liée à l'interaction primaire énoncée e_t . Les actes énoncés au dernier pas sont ensuite placés en mémoire s'ils sont fournis avec une position, sous la forme d'un couple (a, o) où a est un acte énoncé, et o une variable caractérisant l'âge de cet élément. Les actes sont effacés au bout d'un certain nombre de pas (i.e. o supérieur à un seuil). Les expérimentations effectuées pour tester le mécanisme de construction de la mémoire spatiale (présentées au chapitre 11) ont montré que la mémoire spatiale permet de suivre les objets pendant un nombre limité de cycles de décisions (typiquement une dizaine dans nos implémentations). Passé un certain nombre de cycle, la position devient trop incertaine pour être exploitable. On utilise l'âge o d'un acte stocké en mémoire pour définir sa fiabilité : en effet, plus un acte est ancien, plus il est possible que des erreurs dans les déplacements se soient accumulées.

Reconnaissance des instances d'objet

Le mécanisme d'exploitation de la mémoire spatiale doit dans un premier temps reconnaître et localiser les objets susceptibles d'afforder les actes. Le mécanisme de reconnaissance des instances d'objet est similaire à celui décrit au chapitre 6, mais s'applique à l'espace global, c'est-à-dire l'espace intégré par la mémoire spatiale. Nous proposons de définir la position d'une instance d'objet dans l'espace en se basant sur la transformation géométrique τ qui la sépare de l'objet défini par la signature. La reconnaissance et la localisation d'une instance d'objet peut ainsi s'effectuer en appliquant à la mémoire spatiale une transformation géométrique τ . On appelle M_τ l'image de la mémoire spatiale M par la transformation τ . Celle-ci permet de définir le contexte que l'on obtiendrait si l'agent avait effectué cette transformation dans l'environnement, que l'on appelle *image du contexte par τ* , noté $E_{\tau,t}$. Le contexte image est défini par l'ensemble :

$$E_{\tau,t} = (i,p) / (i,p) \in M_{\tau,t} \wedge p \in P \quad (7.11)$$

Avec P l'ensemble des positions de l'espace *observable* dans lequel une interaction peut être énoncée.

Pour toute transformation τ , une instance de l'objet qui afforde une interaction i est dite présente en τ avec une certitude de $c_{success}(i, E_{\tau,t})$, correspondant à la certitude de présence de l'objet dans l'image $M_{\tau,t}$ de la mémoire spatiale. Notons qu'en définissant la position des instances d'objets en termes de transformations géométriques par rapport à la signature de l'objet, il n'est pas nécessaire de connaître la position de l'agent dans la mémoire spatiale. En effet, la transformation est définie entre les positions des actes de la signature et ceux de la mémoire spatiale : aucune position de référence n'est nécessaire. Ce principe est illustré par la Figure 7.4.

Il est possible de définir la transformation de la mémoire spatiale comme une séquence de transformation géométriques liées à des interactions primitives. La position d'une instance d'objet est alors définie par une (ou plusieurs) séquences d'interaction. La distance d'une instance d'objet est définie par la longueur de la plus courte séquence caractérisant cette position. Notons que ceci ne définit pas un mécanisme de *path planning* : on définit bien la séquence d'interactions permettant d'atteindre l'objet, mais ce chemin est construit indépendamment de la possibilité de l'énoncer. Cette définition de la position présente l'avantage de définir les distances par le nombre d'interactions nécessaires pour atteindre une instance d'objet : un objet peut en effet être spatialement proche de l'agent, mais peut nécessiter une longue séquence d'interactions pour être atteint. Cette notion de distance "interactionnelle" reflète ainsi mieux les distances du point de vue de l'agent.

Mécanisme d'exploitation de la mémoire spatiale

Un mécanisme de décision est proposé pour permettre à un agent d'agir en fonction des instances d'objets qui l'entourent, dans le but de satisfaire sa motivation interactionnelle à court

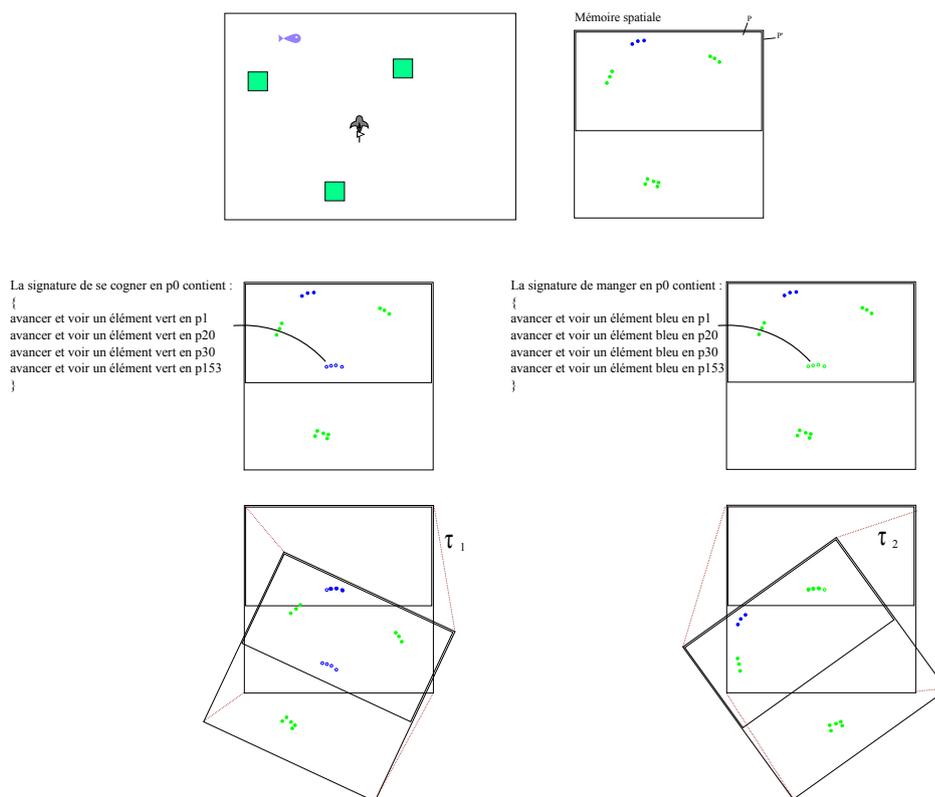


FIGURE 7.4 – Principe de la détection des instances d’objets. En haut, l’agent dans son environnement et le contenu de la mémoire spatiale après éaction de l’acte *avancer en p_0* (avec p_0 une position non définie dans l’espace). Au milieu, les actes désignés par les signatures d’interaction des actes *se cogner en p_0* et *manger en p_0* sont superposés à la mémoire spatiale (les positions sont définies arbitrairement pour illustrer). Les actes désignés par les signatures d’interaction sont représentés par des cercles vides, ceux intégrés par la mémoire spatiale par des disques pleins. Ces deux signatures ne correspondent pas au contexte actuel : les deux actes associés ne sont pas éactables. En bas, on déplace la mémoire spatiale par rapport aux signatures en appliquant des transformations géométriques. La transformation τ_1 permet une forte certitude de l’acte *manger en p_0* , et la transformation τ_2 permet une forte certitude de l’acte *se cogner en p_0* . On considère qu’une instance de l’objet qui afforde l’acte *manger en p_0* est présente en τ_1 et une instance de l’objet qui afforde l’acte *se cogner en p_0* est présente en τ_2 . Ceci signifie que si l’agent se déplace de τ_1 , il pourra éactier l’acte *manger en p_0* .

(hédonisme objectif) et moyen terme (hédonisme tempéré). L’idée est de permettre à l’agent de se diriger vers les instances d’objets qui affordent des actes dotés d’une forte valeur de satisfaction, afin de lui permettre d’interagir avec eux, et s’éloigner des instances d’objets qui affordent des actes à faible valeur de satisfaction, même si cela implique d’aller à l’encontre sa motivation interactionnelle. Notons que l’influence d’une instance d’objet doit être inversement proportionnel à sa distance par rapport à l’agent. En effet, l’agent aura plus de chance d’interagir avec un objet

proche (de façon volontaire ou non) dans un avenir proche.

Le système de décision détermine les actes dont l'énaction permet de se rapprocher d'une instance d'objet attractive ou de s'éloigner d'une instance d'objet répulsive. Cependant, ce système ne définit pas un but unique à atteindre. Il faut donc tenir compte du déplacement de toutes les instances d'objets, par rapport à l'agent, que générerait l'énaction d'un acte candidat. Nous mettons à profit les possibilités prédictives de la mémoire spatiale pour comparer la distance des instances d'objets avant et après énonction d'un acte candidat. Le mécanisme commence par calculer une valeur, que nous appelons *proximité globale* d'un objet, pour chaque type d'objet (ou, puisque les objets sont liés aux actes, pour chaque acte).

Définition 32. La *proximité globale* ν_i d'un objet affordant l'interaction ou de l'acte i est une grandeur qui caractérise la "ressemblance" entre l'environnement défini par la mémoire spatiale à un cycle de décision t et la signature d'interaction de l'interaction (ou de l'acte) i .

La proximité globale d'une interaction ou d'un acte i est calculée en tenant compte de la distance et de la certitude de chaque instance de l'objet qui afforde i :

$$\nu_{i,t}(M) = \int_{\tau \in T} c_{success}(i, E_{\tau,t}) \times f(d(\tau)) \quad (7.12)$$

Nous notons $d(\tau)$ la distance définie par la position τ et $f : \mathbb{R}^+ \rightarrow]0; 1]$ une fonction qui caractérise l'influence des instances d'objets en fonction de leurs distances. Nous utilisons, dans nos implémentations, la fonction $f : x \mapsto e^{-\gamma \times x}$, avec γ un coefficient qui caractérise la décroissance de l'influence des objets en fonction de leur éloignement. Si ce coefficient est trop grand, les instances d'objets lointaines risquent de ne plus avoir assez d'influence pour orienter les comportements de l'agent. S'il est trop faible, les instances d'objets lointaines auront une trop forte influence sur le comportement de l'agent, qui ne sera plus adapté à son contexte immédiat.

La mémoire spatiale calcule ensuite, pour chaque interaction ou acte candidat i_c , l'image prédite de la mémoire après énonction de i_c , qui servira à calculer la proximité globale après énonction de i_c . La variation de proximité globale $\Delta\nu_{i_c}$ est pondérée par la valeur de satisfaction de l'interaction ou de l'acte affordé par l'instance d'objet considérée : en effet, le rapprochement d'une instance d'objet sera positif si l'objet afforde une interaction ou un acte à forte valeur de satisfaction, et négatif dans le cas contraire :

$$\Delta\nu_{i_c} = \sum_{a \in A} (\nu_a(M_{\tau_{i_c}}) - \nu_a(M)) \times s_a \quad (7.13)$$

Avec s_a la valeur de satisfaction associée à l'énaction de l'acte a .

La Figure 7.5 illustre le principe de proximité globale et de variation de proximité. Le mécanisme ajoute ensuite une valeur d'utilité aux actes candidats, pour définir une valeur de *satisfaction globale* s'_i :

$$s'_{i_c} = s_{i_c} + \beta \times \Delta\nu_{i_c} \quad (7.14)$$

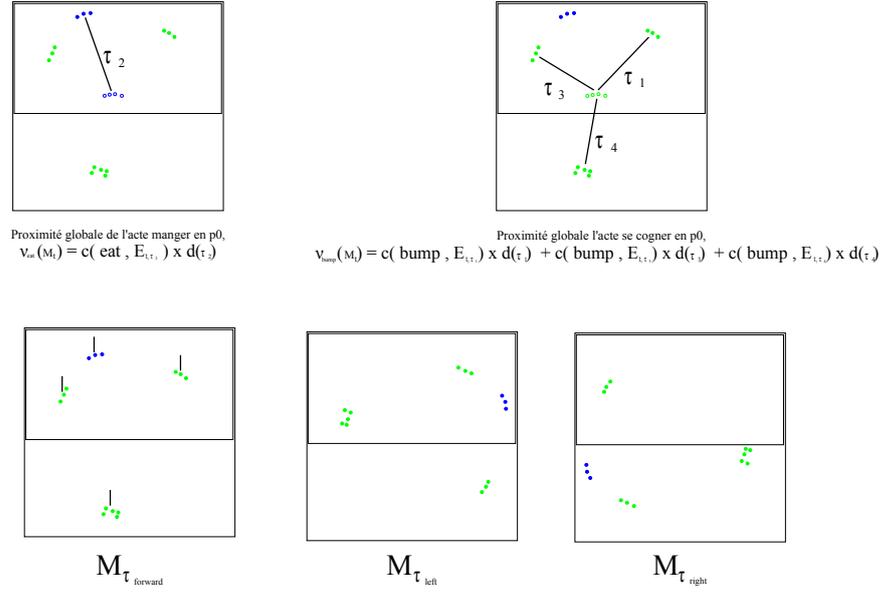


FIGURE 7.5 – Proximité globale et variation de proximité. On reprend l'exemple de contexte de la figure 7.4. En haut, on définit les transformations caractérisant la position des instances d'objets affordant les actes *manger en p_0* et *se cogner en p_0* . On peut définir la proximité globale comme la somme des certitudes de présence des instances d'objets pondérées par une valeur liée à leur distance. On supposera dans cet exemple que toutes les transformations autres que celles affichées donnent une certitude de présence de 0. En bas, trois actes sont candidats : *avancer en p_0* , *tourner à gauche en p_0* et *tourner à droite en p_0* . Les transformations sont utilisées pour former les contextes prédits $M_{\tau_{forward}}$, $M_{\tau_{left}}$ et $M_{\tau_{right}}$. On calcule la proximité globale des actes *manger p_0* et *se cogner p_0* pour chacune de ces prédictions, afin de définir la variation de proximité générée par l'énaction de chaque acte candidat.

avec $\beta \in \mathbb{R}^+$ le coefficient d'influence de la mémoire spatiale.

Le coefficient d'influence de la mémoire spatiale caractérise l'influence des objets distants par rapport aux valeurs de satisfaction des interactions. Si ce coefficient est grand, l'agent aura tendance à se diriger vers les instances d'objets attractives en ne tenant que peu compte de la valeur de satisfaction des interactions permettant de l'atteindre. Si le coefficient est petit, l'agent cherchera à énoncer les interactions dotées d'une forte valeur de satisfaction, et ne se dirigera vers les instances d'objets que si elles sont suffisamment proches et atteignables par des interactions dotées d'une forte valeur de satisfaction.

Le mécanisme de décision peut alors sélectionner l'acte pour lequel la valeur de satisfaction globale est la plus élevée. L'acte sélectionnée sera ainsi celui offrant le meilleur compromis entre la valeur de satisfaction offerte par son énonciation, et le rapprochement (ou l'éloignement) des instances d'objet, permettant l'énonciation future d'un acte doté d'une forte valeur de satisfaction, ou permettant d'éviter d'énoncer un acte doté d'une faible valeur de satisfaction.

Mécanisme d'exploitation d'une mémoire spatiale agnostique

Nous avons adapté le mécanisme d'exploitation pour le tester avec la mémoire spatiale construite par le mécanisme de construction de la mémoire spatiale. Ce mécanisme se base sur les mêmes principes que celui décrit dans la section précédente : le but de ce mécanisme consiste à définir les interactions qui permettent de se rapprocher des instances d'objets attractifs et de s'éloigner des instances d'objets répulsifs.

Nous exploitons le découpage fonctionnel des lieux et lieux composites décrit au chapitre 6 : les instances d'objets sont localisées à l'aide d'un ensemble de lieux composites, donnant un ensemble de chemins permettant d'observer (directement ou indirectement) l'instance dans une région de l'espace pouvant être rapprochée en utilisant une certaine interaction qui définit ce lieu. Ainsi, chaque lieu primitif indique, par définition, une interaction permettant de se rapprocher d'une instance d'objet. La première interaction du path d'un lieu composite (dans le cas d'un path de longueur non nulle) définit une interaction permettant de se rapprocher d'une instance d'objet, en supposant que l'agent utilise le path de ce lieu.

Cependant, le path d'un lieu composite ne caractérise pas toujours le chemin le plus court pour atteindre une instance d'objet. Dans certains cas, le premier élément du path peut être une interaction permettant de s'éloigner de l'instance. Nous devons donc, dans un premier temps, estimer la distance des instances d'objets, afin de ne prendre en compte que les lieux composites définissant le chemin le plus court pour les atteindre.

Chaque lieu l définit une distance, que l'on note d_l : la distance d'un lieu primitif correspond à la distance des positions qui le caractérisent. La distance d'une interaction composite est définie comme la somme de la distance donnée par son lieu final et de la longueur de son path. En effet, le path caractérise les interactions que l'agent doit effectuer pour que l'instance soit observée dans le lieu final, ce qui augmente le nombre total d'interactions nécessaires pour pouvoir détecter une instance à une distance caractérisée par le lieu final.

Le calcul de la distance d'une instance d'objet est déterminé par les lieux composites caractérisant sa position qui définissent la distance la plus courte. Il faut cependant tenir compte de la fiabilité de ces lieux composites. À force de mise à jour de l'estimation de la position, un lieu composite est en effet de moins en moins susceptible de définir effectivement la position de l'instance. Il convient donc de retirer les lieux composites les moins fiables de l'estimation de la distance d'une instance.

Nous proposons une méthode permettant d'estimer la distance d'une instance d'objet ω en se basant sur les distances définies par les lieux composites et leurs certitudes respectives. Cette méthode se base sur la recherche du premier maximum local de certitude en fonction des distances définies par les lieux. On commence par déterminer la plus grande certitude pour chaque distance d , que l'on note $c_{max,\omega,d}$:

$$c_{max,\omega,d} = \max_{l \in \Omega_\omega/d_l=d} (c_{l,\omega}) \quad (7.15)$$

avec $c_{l,\omega}$ la certitude que l'instance d'objet ω soit dans la région définie par le lieu l . La distance

minimale de l'instance d'objet ω est définie par :

$$d_{min,\omega} = \min(d) / (c_{max,\omega,d} > c_{max,\omega,d-1} \wedge c_{max,\omega,d} \geq c_{max,\omega,d+1}) \quad (7.16)$$

La distance estimée de l'instance ω est ensuite définie comme la valeur moyenne des distances d_l des lieux l "proches" de la distance minimale, c'est-à-dire à une distance ϵ de la valeur minimale, pondérée par la certitude $c_{l,\omega}$ de ces lieux :

$$d_\omega = \frac{\sum_{l \in \Omega_\omega / d_l \in [d_{min,\omega} - \epsilon; d_{min,\omega} + \epsilon]} c_{l,\omega} \times d_l \times \gamma^{|d_{min} - d_l|}}{\sum_{l \in \Omega_\omega / d_l \in [d_{min,\omega} - \epsilon; d_{min,\omega} + \epsilon]} \gamma^{|d_{min} - d_l|}} \quad (7.17)$$

avec $\gamma \in [0; 1]$ un coefficient caractérisant l'influence des lieux autour de la distance d_{min} . Notons que cette distance estimée peut ne pas refléter la distance réelle, notamment si les lieux composites retenus sont constitués d'un path de grande longueur. Cependant, nous obtenons une estimation de la distance en termes de nombre d'interactions minimum connu permettant d'atteindre une instance d'objet.

Nous pouvons à présent définir les interactions candidates permettant de se rapprocher des instances d'interaction en se basant sur les lieux composites pour lesquels $d_l \in [d_\omega - 1; d_\omega + 1]$:

-si un lieu composite a un path de longueur nulle, alors l'instance d'objet peut être rapprochée par l'interaction qui définit le lieu final.

-si le lieu composite a un path de longueur non nulle, alors la première interaction du path permet de rapprocher l'instance d'objet (d'un point de vue interactionnel) d'une région de l'espace observable.

Nous obtenons ainsi les deux informations nécessaires à notre mécanisme d'exploitation : les interactions permettant le rapprochement d'une instance d'objet et une distance pour caractériser l'influence de cette instance. Le mécanisme de sélection est similaire au mécanisme présenté précédemment, à ceci près que la variation de distance est toujours égale à 1 (car l'agent va se rapprocher d'une interaction de l'instance d'objet). On ne peut, en revanche, que définir les interactions qui permettent de se rapprocher, et non celles qui permettent de s'éloigner des instances d'objet.

On redéfinit les équations du mécanisme de sélection : on appelle R_ω l'ensemble des interactions primitives qui permettent de se rapprocher d'une instance d'objet ω , affordant l'interaction i_ω et située à une distance d_ω . La variation de proximité globale provoqué par l'énaction d'une interaction candidate i_c est définie par :

$$\Delta \nu_{i_c} = \sum_{\omega \in \mathcal{M}} \begin{cases} f(d_\omega) \times s_{i_\omega} & \text{if } i_c \text{ in } R_\omega \\ 0 & \text{else} \end{cases} \quad (7.18)$$

Avec f fonction qui caractérise l'influence des instances d'objets en fonction de leur distance. Le mécanisme de sélection choisit, comme précédemment, l'interaction i_c pour laquelle la valeur de satisfaction globale s'_{i_c} (équation 7.14) est la plus élevée.

7.3 Conclusion

Le mécanisme de la mémoire péri-personnelle ne nécessite aucune hypothèse sur la topographie de l'espace, et de ce fait, n'a pas besoin d'un mécanisme pour construire une structure caractérisant l'espace. Ce mécanisme ne nécessite que peu de calculs pour définir les interactions les plus profitables à l'agent. Cette propriété rend ce mécanisme particulièrement adapté à un système nécessitant une forte réactivité. La possibilité de compléter le contexte interactionnel, et d'ajouter des entrées à ce contexte pour tenir compte des objets situés en dehors du système interactionnel de l'agent permet de fournir des informations supplémentaires à d'autres mécanismes décisionnels. Cependant, ce mécanisme est limité à l'espace proche de l'agent. La portée peut être augmentée en augmentant la longueur des interactions composites. Malheureusement, le nombre d'interactions composites croît de façon exponentielle avec la longueur. Le mécanisme de la mémoire péri-personnelle est donc limité à l'espace proche seul. Cette limitation implique la présence d'un second mécanisme pour caractériser l'espace sur une distance plus importante.

Le mécanisme de sélection de la mémoire extra-personnelle nécessite deux informations pour générer des comportements permettant à l'agent de se rapprocher ou s'éloigner des objets :

- la distance des instances d'objets reconnues, afin d'en caractériser l'influence
- la variation des distances produites par l'énaction des interactions, pour définir le rapprochement généré par les interactions candidates.

Ces informations sont fournies par la mémoire spatiale que nous avons décrit au chapitre précédent (Chapitre 6). Le mécanisme d'exploitation de la mémoire spatiale prend en compte l'ensemble des éléments présents dans l'environnement et attribue une valeur d'utilité aux interactions qui dépend des conséquences de leur énaction sur l'environnement. Ce mécanisme permet à un agent de sélectionner ses interactions dans le but de satisfaire sa motivation propre à moyen terme, en permettant de se rapprocher des instances d'objets offrant des interactions à forte valeur de satisfaction, même si cela implique d'énacter des interactions peu satisfaisantes.

Chapitre 8

Choix d'implémentation pour la validation

8.1 Indépendance entre l'agent et l'environnement

Conformément aux spécifications définies par le modèle du Radical Interactionism (RI), décrit au chapitre 4, la communication entre l'agent et l'environnement ne s'effectue que par l'échange d'interactions primitives. Les interactions sont implémentées sous la forme de structures comprenant une action, une perception et une valeur de satisfaction. Le système de décision n'a jamais accès aux actions et perceptions d'une interaction. Les interactions intentions sont transmises par le système de décision à une interface qui applique le schéma sensorimoteur et contrôle son exécution. Les interactions énoncées sont reconstituées par cette interface, qui détermine l'interaction énoncée à partir de l'action de l'interaction intention et de la perception observée à la fin de l'interaction agent-environnement. L'interaction énoncée est ensuite envoyée au système de décision de l'agent.

8.2 Mécanismes de construction de structures pour la prise de connaissance de l'environnement

Les signatures d'interaction et les fonctions de certitude de succès et d'échec constituent un système prédictif permettant d'évaluer la faisabilité d'une interaction. Cette prédiction est effectuée en se basant sur un ensemble d'interactions caractérisant l'environnement intégré par l'agent.

Nous avons défini la signature d'interaction comme une structure permettant de caractériser un contexte d'éléments de l'environnement par la façon dont l'agent peut l'expérimenter par le biais de ses interactions. Elle permet ainsi de générer un lien entre les interactions qui permettent de considérer un objet et l'interaction abordée par cet objet. L'agent peut ainsi constituer des modèles exploitables d'objets à partir d'interactions, sans qu'aucune autre information que les

interactions qu'il énonce ne soit nécessaire.

L'implémentation des signatures d'interaction à base de neurones artificiels permet d'extraire du vecteur d'entrée les entrées pertinentes pour prédire la valeur de sortie. La signature permet ainsi de définir une association forte entre une interaction et les interactions qui permettent de prédire le résultat de son énoncé éventuelle. Les signatures d'interaction sont constituées d'un vecteur générant une association orientée entre une interaction et l'ensemble d'interactions primitives qui permettent de prédire le résultat de son énoncé.

Cette propriété donne au système une capacité à prédire le résultat des interactions du cycle de décision suivant. On peut s'attendre à ce que le système puisse regrouper des interactions liées à un même objet de l'espace, si ces interactions permettent de prédire le résultat d'une même interaction.

La prédiction sur la faisabilité d'une interaction s'effectue par l'utilisation des fonctions de certitudes de succès et d'échec. Ces fonctions se basent sur la présence ou l'absence de la signature d'une interaction dans le contexte interactionnel courant. Le système met à jour les signatures des interactions énoncées à partir des contextes interactionnels passés. L'apprentissage s'effectue uniquement par l'expérience que l'agent a avec son environnement par le biais de ses interactions.

8.3 Mécanisme de construction de la Mémoire Spatiale

Dans un premier temps, ce mécanisme doit permettre la construction d'une structure intégrant l'espace qu'il peut observer, c'est-à-dire les positions de l'espace où une interaction peut être énoncée. Notons que dans le modèle SRI, la topologie de l'espace est définie *a priori* (hypothèse du tectum optique). La construction de cette topologie repose sur l'utilisation des signatures d'interactions apprises par le mécanisme de construction de structures pour la prise de connaissance de l'environnement. L'émergence de la structure intégrant l'espace observable repose en effet sur les associations définies par les signatures d'interaction.

La seconde partie de ce mécanisme doit permettre la construction d'une structure intégrant l'espace global et permettre de "suivre" les objets détectés. L'algorithme intégrant la topologie de l'espace global utilise des structures similaires aux signatures d'interaction. Celles-ci génèrent une association forte entre une région de l'espace et les régions qui permettent de prédire la présence d'un objet dans cette région. Ces associations permettent de lier des positions de l'espace par un mouvement qui permet de passer de l'une à l'autre par le biais de structures appelées *lieux composites*. L'aspect séquentiel des lieux composites permet de définir des régions de l'espace non observable et d'établir des liens entre ces régions, caractérisés par les interactions qui permettent de passer d'une région à une autre. L'agent peut ainsi caractériser la topologie de l'espace sans que celle-ci n'ait besoin d'être définie *a priori*.

Les positions de l'espaces peuvent être définies de deux façons : dans l'approche RI, les positions sont définies par des séquences d'interactions primitives. Chaque position est définie par la séquence d'interactions primitives qui permet d'atteindre cette position. Si le système se base sur le modèle SRI, on peut utiliser des positions définies par leurs coordonnées en référentiel

égocentré. Nous définissons un *lieu* comme un ensemble de positions de l'espace pour lesquelles l'interaction permettant de s'en rapprocher et la distance, définie par le nombre minimal d'interactions primitives à énoncer pour l'atteindre, ou par sa distance géométrique par rapport à l'agent dans le cas du modèle SRI, sont similaires.

Une *instance d'objet* est définie par l'interaction qu'elle peut afforder, et une position dans l'espace. Cette position est définie par l'ensemble des lieux et lieux composites qui contiennent cette instance d'objet. Définir ainsi une instance d'objet permet de définir comment s'en rapprocher, ainsi que son influence (liée à sa distance).

8.4 Mécanismes d'exploitation de la Mémoire Spatiale

8.4.1 Espace péri-personnel

L'espace proche est caractérisé par des séquences d'interactions appelées *interactions composites*. Une interaction composite est constituée de deux séquences d'interactions. La seconde séquence permet de caractériser la présence de l'objet qui afforde l'interaction composite (puisque la possibilité d'énoncer une séquence d'interactions est liée à la présence de l'objet qui afforde cette séquence dans l'environnement). La première séquence caractérise la suite d'interactions que l'agent doit effectuer pour atteindre cet objet, indépendamment de la possibilité de l'énoncer (puisque la présence d'un objet est indépendante de son accessibilité), et définit ainsi une position dans l'espace conforme à l'espace représentatif de Poincaré [107].

Comme chaque interaction composite permet de caractériser la présence (ou l'absence) d'un objet à une certaine position de l'espace, il devient possible de caractériser la configuration de l'environnement proche de l'agent à l'aide de l'ensemble des interactions composites considérées comme possible. Un tel modèle de l'environnement ne nécessite aucune autre information que les interactions et leurs signatures.

Le système de suivi des objets repose sur l'aspect séquentiel des interactions composites. En effet, une interaction composite caractérise la position d'un objet par la séquence d'interactions qui permet de l'atteindre. Ainsi, si l'agent énonce la première interaction de cette séquence (dans le but d'atteindre l'objet ou non), alors l'objet est toujours présent, et à une position caractérisée par les interactions suivantes de la séquence initiale. L'objet et sa position peuvent donc être caractérisés par une nouvelle interaction composite. Le système de suivi des objets repose ainsi sur les propriétés des interactions composites et ne nécessite pas d'apprentissage.

8.4.2 Espace extra-personnel

L'agent prend en compte l'ensemble des instances d'objet détectées dans son environnement. Comme nous l'avons vu précédemment, chaque instance est caractérisée par des interactions permettant de s'en approcher ou de s'en éloigner, ainsi qu'une "distance" qui permet de définir son influence dans les décisions de l'agent.

L'agent tient compte dans un premier temps des interactions qu'il peut effectuer, c'est-à-dire considérées comme énonçables d'après les informations fournies par le mécanisme de construction de structures pour la prise de connaissance de l'environnement. La sélection de l'interaction intention du cycle de décision suivant se base sur la valeur de satisfaction des interactions possibles. Cependant, l'agent doit également tenir compte des affordances présentes dans son environnement afin de maximiser sa valeur de satisfaction à moyen terme. On ajoute donc, aux valeurs de satisfaction, une valeur d'utilité définie par le rapprochement des instances d'objet permis par l'énonciation des interactions.

Troisième partie

Expérimentations

Cette partie décrit les différentes expérimentations menées pour mettre au point les contributions et mettre en évidence l'observation des propriétés d'émergence de comportements d'apprentissage lors de la mise en œuvre des contributions sur des agents artificiels ou robotiques.

Deux situations sont particulièrement expérimentées : 1) la situation d'une mémoire péri-personnelle, mécanisme simple prenant en compte l'environnement avec lequel l'agent peut interagir directement (environnement spatial immédiat); 2) la situation d'une mémoire spatiale extra-personnelle, mécanisme plus élaboré faisant l'hypothèse d'une capacité à enacter des interactions avec des éléments de l'environnement non immédiatement accessibles.

L'étude de la mémoire spatiale extra-personnelle a été menée en deux temps : 1) en utilisant d'abord un mécanisme codé en dur implémentant une mémoire spatiale pour faciliter l'étude de l'intégration de cet effet mémoire dans l'apprentissage développemental; 2) en remplaçant ensuite la mémoire *codée en dur* par un dispositif agnostique, dont la structure se met en place en fonction de l'expérience et de l'apprentissage développemental de l'agent.

Chapitre 9

Expérimentation d'un dispositif de mémoire péri-personnelle

9.1 Description de l'environnement

9.1.1 Configuration de l'expérimentation

Il s'agit d'expérimenter un mécanisme de mémoire spatiale péri-personnelle pour un agent autonome. Ce mécanisme est en charge de la boucle sensorimotrice de l'agent. La mémoire spatiale est limitée à l'environnement immédiat de l'agent, et nous choisissons la configuration environnementale dite du *small loop problem* [46], ainsi que quelques variations, pour mener l'expérimentation. Cette configuration, définie initialement pour offrir des régularités séquentielles, nous permet de nous limiter à des agents disposant d'un système sensoriel à courte portée, et d'un ensemble d'interactions restreint, propice à l'étude d'un système disposant d'une structure intégrant l'espace immédiat de l'agent.

Nous avons développé un environnement basé sur l'environnement *Vacuum* proposé par Russell et Norvig [112]. Développé en Java, cet environnement est discret et ne contient que deux types d'éléments : les murs et les espaces vides. La configuration de l'environnement est construite de façon à ce qu'en toute position de l'environnement accessible à l'agent, l'un des trois contextes spatiaux suivants apparaisse :

- des lignes droites (espace vide devant l'agent, murs à droite et à gauche),
- des virages à gauche (espace vide à gauche de l'agent, murs devant et à droite),
- des virages à droite (espace vide à droite de l'agent, murs devant et à gauche).

Nous modifierons dynamiquement la configuration de l'environnement, en particulier avec un contexte voie sans issue (mur devant, à droite et à gauche), quand il apparaîtra que l'agent a stabilisé son comportement dans son environnement initial. Il s'agit en effet d'expérimenter les capacités d'apprentissage dynamique observables par une adaptation à cette évolution de l'environnement. L'illustration 9.1 décrit les différentes configurations utilisées dans l'expérimentation.

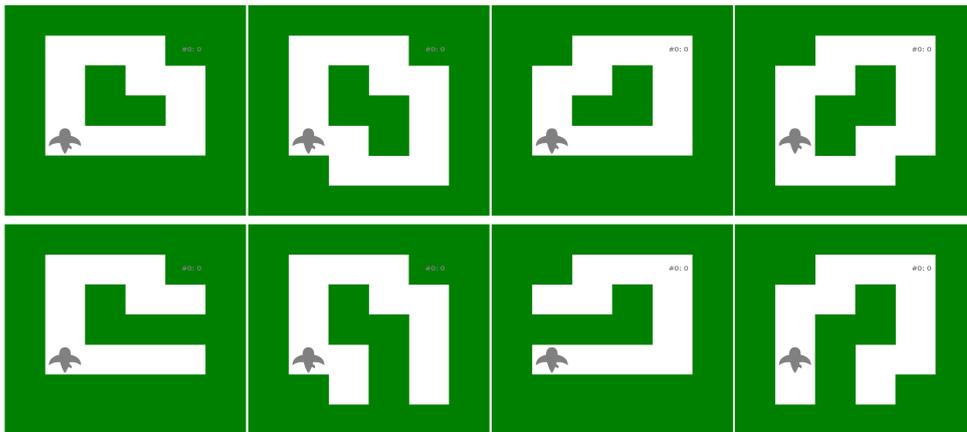


FIGURE 9.1 – Configurations de l'expérimentation sur la mémoire peri-personnelle : Rangée du haut : configurations de test de la mémoire péri-personnelle. De gauche à droite : *small loop*, *double loop*, *reverse small loop*, *reverse double loop*. Rangée du bas, les mêmes configurations après ajout d'un élément mur pour former un nouveau contexte environnemental.

Le couplage agent/environnement offre un ensemble de 12 interactions permettant à l'agent d'expérimenter et agir sur son environnement. Ces interactions sont listées dans la Table 9.1, avec leur symbole et leur valeur de satisfaction. Ces valeurs sont similaires à celles utilisées dans les premières implémentations du projet IDEAL. L'agent "aime" *avancer d'un pas* mais "déteste" *se cogner dans un mur*. Les interactions *toucher* et *tourner* ont une valeur de satisfaction négative mais faible en valeur absolue.

TABLE 9.1 – Liste des douze interactions primitives utilisée dans le système de test. Le symbole de chaque interaction est celui qui sera utilisé dans les figures de ce chapitre. La valeur de satisfaction associée à chaque interaction est indiquée entre parenthèse.

□	toucher devant un espace vide (-1)
■	toucher devant un mur (-2)
▤	toucher à droite un espace vide (-1)
▥	toucher à droite un mur (-2)
▧	toucher à gauche un espace vide (-1)
▨	toucher à gauche un mur (-2)
▷	avancer d'un pas (5)
▶	se cogner contre un mur (-10)
∪	tourner à droite de 90° face à un espace vide (-3)
∩	tourner à droite de 90° face un mur (-4)
∪	tourner à gauche de 90° face à un espace vide (-3)
∩	tourner à gauche de 90° face à droite un mur (-4)

Nous limiterons cette étude aux interactions composites dont la séquence finale est constituée d'une unique interaction primitive. En effet, les interactions et l'environnement étant très simples, l'objet affordant une séquence finale de longueur supérieure à 1 ne serait constitué que de la somme des objets affordant les interactions qui constituent cette séquence. Ainsi, la signature d'une interaction composite $[i_1, \dots, i_{k-1}] [i_k, \dots, i_n]$ serait simplement la somme des signatures des interactions composites $[i_1, \dots, i_{k-1}] [i_k]$ à $[i_1, \dots, i_{n-1}] [i_n]$. Nous emploierons le terme *interaction finale* pour désigner une séquence finale d'une interaction composite dont la longueur est de 1.

Définition 33. *Interaction finale : séquence finale d'une interaction composite constituée d'une unique interaction primitive.*

9.1.2 Simplifications de la prise en compte du contexte interactionnel

Comme expliqué au chapitre 4, le contexte interactionnel d'un agent à un cycle de décision t est constitué de la ou les interactions énoncées à la fin du cycle de décision. L'agent apprend à se servir de ce contexte pour prédire les interactions qui peuvent être énoncées avec succès au cycle de décision $t+1$. L'expérimentation porte sur la capacité de mémoire spatiale à rendre compte de l'environnement immédiat et des objets présents dans l'environnement. Pour cela, nous avons introduit un mécanisme permettant d'intégrer des "modèles" des objets de l'environnement. En effet, la notion d'objet n'a pas de sens pour l'agent, ce sont les modèles interactionnels tels qu'ils peuvent être interagis pendant l'expérimentation qui font dire à l'observateur que l'agent prend en compte les objets ayant un sens dans l'environnement pour cet observateur (mur, espace vide, ...). Nous limitons le contexte interactionnel aux interactions "toucher", afin de réduire la taille des signatures d'interaction et, de fait, la durée d'apprentissage. En effet, toutes les autres interactions (excepté *avancer d'un pas*) renseignent sur la présence d'un élément situé devant l'agent, ce qui est redondant avec les interactions *toucher devant*. Il s'agit également d'une simplification du mécanisme des interactions épistémiques : puisque les seules interactions permettant de compléter le contexte ne produisent pas de mouvement, il n'est pas nécessaire de simuler le contexte après énonciation de l'interaction épistémique, puisque ce nouveau contexte sera identique au contexte courant augmenté, bien sûr, du résultat de l'interaction épistémique.

Je propose également de considérer les échecs des interactions primitives dans le contexte interactionnel : chaque interaction dispose d'une et une seule interaction opposée. Ainsi, le succès d'une interaction implique l'échec de son opposée, et d'une certaine façon l'absence de l'objet qui caractérise le succès de cette interaction opposée. De plus, les objets qui affordent les interactions utilisées pour définir le contexte sont mutuellement incompatibles, puisqu'il n'y a que deux possibilités : la présence de l'un implique l'absence de l'autre. Cette simplification n'aurait pas été possible si l'on avait tenu compte des interactions *avancer* et *se cogner* dans le contexte interactionnel, puisque les objets post-interaction de ces interactions sont localisées à des positions différentes : l'objet post énonciation de *avancer* est un espace vide sous l'agent, tandis que celui de *se cogner* est un mur devant l'agent. On ajoute ainsi un troisième état, l'échec, pour définir les éléments du contexte interactionnel.

Cette simplification permet à l'agent de tester deux entrées du contexte interactionnel simultanément. Notons que les poids associés aux entrées de deux interactions opposées seront toujours de valeurs opposées, puisqu'à chaque apprentissage, les valeurs de ces entrées seront de valeurs opposées. Cette simplification permet également de simplifier l'affichage des signatures, en ne montrant, par exemple, que les poids associés aux interactions *toucher un espace vide* ; les poids associés aux interactions *toucher un mur* étant de valeur opposée.

L'absence d'erreur dans l'énaction des interactions et la discrétisation de l'environnement nous permet de définir une troisième simplification : le système interactionnel, tel que nous l'avons défini précédemment fait que les objets qui affordent les interactions ont tous la même taille. Tous les objets sont donc considérés indépendamment de leur taille et de leur orientation. Ainsi, si une entrée du contexte interactionnel ne permet pas de définir la présence ou l'absence de l'objet qui afforde une interaction donnée, il devient possible de supprimer cette entrée de la signature de cette interaction. Cette suppression est effectuée si un même contexte interactionnel peut conduire à deux résultats différents, ou si deux contextes opposés conduisent à un même résultat.

Cette simplification permet de réduire le temps d'apprentissage en éliminant les interactions non pertinentes des signatures. Notons que cette simplification ne peut être employée que si le contexte sur lequel se base le mécanisme d'apprentissage ne contient jamais d'erreurs. Une interaction dont toutes les entrées du contexte interactionnel sont supprimées de sa signature sera définie comme dé-corrélée.

Définition 34. *une interaction est dite dé-corrélée si toutes les entrées de sa signature sont supprimées.*

Une interaction dé-corrélée est affordée par un élément de l'environnement qui échappe au système interactionnel de l'agent. Comme nous le verrons plus tard, ces interactions peuvent être détectés de façon indirectes et renseigner l'agent sur des éléments qui échappent à son système interactionnel.

9.1.3 Détails de l'implémentation

Le mécanisme simplifié de la mémoire spatiale n'est pas tolérant aux erreurs. La mémoire spatiale est considérée comme un ensemble de quatre contextes interactionnels consécutifs et trois couples de listes (\mathcal{T}, \mathcal{F}) associés à ces contextes, constituant chacun un niveau de fiabilité différent (Figure 9.2). Ceci permet de séparer le contexte interactionnel, exempt d'erreur, des contextes complétés par la mémoire spatiale, pouvant contenir des erreurs liées aux signatures en cours d'apprentissage. Chaque niveau permet la détection d'interactions composites supplémentaires reconnues comme possibles d'après le contexte du niveau précédent.

Le premier contexte E ne prend en compte que l'interaction énéacté e_t , et l'interaction qui lui est opposée, considérée comme un échec. Ce contexte ne peut donc pas contenir d'erreur et sert à l'apprentissage et au renforcement des signatures d'interaction. Le premier couple de listes d'interactions conserve en mémoire les interactions reconnues d'après ce premier contexte et

considérées comme fiables. Ces interactions permettent de générer un second contexte E' , ainsi doté d'une certaine fiabilité. Les couples de listes suivants permettent la reconnaissance et le stockage de toutes les interactions reconnues comme possibles (pour les listes \mathcal{T}) ou impossibles (pour les listes \mathcal{F}). Elles sont principalement utilisées pour définir un contexte estimé permettant de définir la faisabilité des paths des interactions composites, ainsi que pour la détection des erreurs, en permettant la détection d'incohérences dans les signatures des interactions composites qu'elles contiennent. Elles permettent ainsi de détecter rapidement les erreurs et incohérences des signatures d'interaction en cours d'apprentissage.

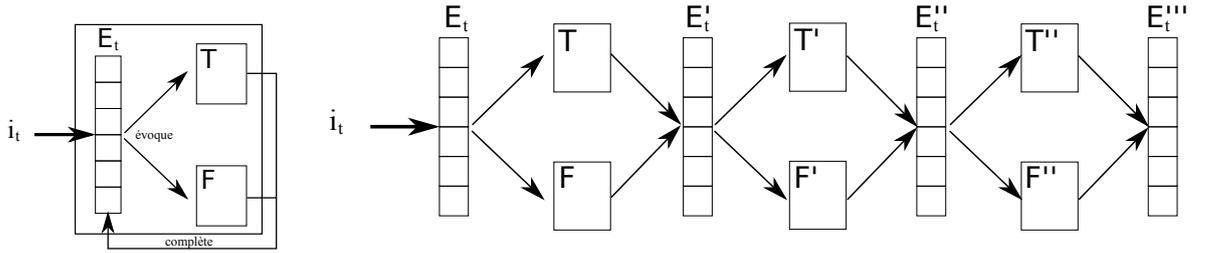


FIGURE 9.2 – on "déroule" le mécanisme de mémoire spatiale. Au lieu de compléter le contexte interactionnel E_t à l'aide des listes \mathcal{T} et \mathcal{F} , en accumulant des erreurs dans E_t , chaque couple de liste est utilisé pour compléter un nouveau contexte interactionnel, plus complet mais moins fiable que le précédent. Dans notre système de tests, trois contextes augmentés et couples de listes consécutifs permettent de compléter l'expérience que l'agent a de son environnement, en définissant différents niveaux de fiabilité des informations issues du système spatio-séquentiel.

La fiabilité des signatures d'interaction est également définie pour tenir compte de l'intolérance aux erreurs des signatures. Dans cette implémentation, la fiabilité est définie par le nombre de prédictions correctes consécutives de succès et d'échec. Nous définissons une interaction comme *fiable* si sa signature a permis de prédire correctement et consécutivement un certain nombre de succès et d'échecs. Des essais préliminaires ont montré qu'un seuil de dix succès et de dix échecs correctement prédits de façon consécutive est un bon compromis entre la fiabilité des interactions obtenues et le temps d'apprentissage.

Il peut arriver qu'une interaction composite ne puisse pas échouer. Ceci est possible si le path d'une interaction composite génère un objet post-énaction qui contient l'objet pré-énaction qui afforde l'interaction finale. Par exemple, dans l'ensemble d'interactions définie dans le tableau 9.1, si l'agent énonce l'interaction *tourner à droite face à un espace vide*, il pourra énoncer l'interaction *avancer*. L'interaction composite $[\curvearrowright] [\triangleright]$ ne peut donc pas échouer.

La fiabilité des interactions composites qui ne peuvent pas échouer est plus difficile à définir, puisqu'on ne peut pas garantir avec certitude qu'une interaction n'échouera jamais. Je propose de baser la fiabilité de ces interactions sur le nombre d'énactions et le nombre de prédictions correctes. Une interaction qui ne peut pas échouer considérée comme fiable sera dite *toujours vraie* :

Définition 35. *une interaction composite toujours vraie est une interaction composite qui ne*

peut pas échouer et dont la signature est considérée comme fiable par le mécanisme de décision.

Dans notre implémentation, une interaction composite sera considérée comme *toujours vraie* si sa signature a permis de prédire correctement au moins dix succès, si elle a été énoncée avec succès 25 fois et si elle n'a jamais échoué. Les interactions toujours vraies sont utilisées de la même façon que les interactions fiables par le mécanisme de décision.

9.1.4 Implémentation des signatures d'interaction

Je propose une implémentation du mécanisme basée sur un réseau de neurone formel monocouche (aussi appelé *perceptron*), bien qu'il soit raisonnable de penser que d'autres mécanismes puisse être envisagés. Cependant, cette approche m'a semblé être la plus pertinente pour implémenter les propriétés définies par le mécanisme de construction de structures pour la prise de connaissance de l'environnement (décrit au chapitre 5). Les réseaux de neurones formels sont des modèles très simples, mais qui permettent de catégoriser tout vecteur d'entrée qu'on leur présente, à partir d'un échantillon utilisé pour entraîner le réseau. Ces modèles conviennent parfaitement pour implémenter notre mécanisme puisque l'on cherche à catégoriser les contextes interactionnel permettant soit le succès soit l'échec d'une interaction. L'apprentissage peut s'effectuer à chaque énoncé d'une interaction, en comparant le résultat de l'énoncé avec la prédiction. En outre, ils permettent de visualiser les structures des objets construites par l'agent en analysant les poids synaptiques, caractérisant les interactions considérées comme permettant de détecter l'objet qui afforde une interaction. Dans notre implémentation, chaque interaction est associée à un neurone formel dont les poids synaptiques constituent la signature.

À chaque interaction composite $\varphi \in \Phi$ est associée un neurone prenant en entrée le contexte courant E_t , mis sous la forme d'un vecteur (voir section ci-après), et un ensemble de poids $W_\varphi = w_{\varphi,1}, \dots, w_{\varphi,n}, b_\varphi$ avec $n = \text{card}(I)$, I ensemble des interactions primitives, et où $w_{\varphi,k}$ est un poids associé à une entrée du contexte interactionnel et b_φ est le biais du neurone.

Le contexte interactionnel

Nous devons dans un premier temps définir un contexte servant d'entrée au réseau de neurone. Le contexte a la forme d'un vecteur donnant, à chaque cycle de décision t , le résultat (énoncé ou non) de chaque interaction, ce qui permet de caractériser le contexte E_t sous une forme exploitable par un réseau de neurones.

Nous définissons dans un premier temps une fonction $\epsilon_i : \mathbb{N} \rightarrow \{0;1\}$ qui caractérise le résultat d'une interaction primitive i au cycle t . Cette fonction est définie par :

$$\epsilon_i(t) = \begin{cases} 1 & \text{si } i \in E_t \text{ (énoncée)} \\ 0 & \text{sinon (non énoncée)} \end{cases} \quad (9.1)$$

Notons que dans notre système simplifié, le contexte prend en compte l'échec des interactions. On ajoute un troisième état pour caractériser l'échec : $\epsilon_i(t) = -1$ (*failure*). Une interaction

échoue si une interaction opposée est énoncée. Une interaction est définie comme *non énoncée* si elle n'a pas été énoncée et si elle n'a pas échoué. Le contexte interactionnel E_t est mis sous la forme d'un vecteur $[\epsilon_{i_1,t}, \dots, \epsilon_{i_n,t}] \in [-1; 1]^n$, donnant le résultat de chaque interaction primitive $i \in I$. Ce contexte caractérise la façon dont l'agent a expérimenté son environnement au cours du dernier cycle de décision.

Un autre intérêt de définir le contexte interactionnel sous la forme d'un vecteur est, nous le verrons dans la section suivante, de permettre à l'agent de compléter son expérience de l'environnement, à l'aide des informations fournies par la mémoire spatiale. Le vecteur pourra en effet être complété en donnant la certitude qu'une interaction aurait pu être énoncée comme un succès ou un échec, indiquant la présence de l'objet post-énoncement associé, avec une certitude absolue de succès si $\epsilon_{i,t} = 1$ et une certitude absolue d'échec si $\epsilon_{i,t} = -1$.

Rappel sur le neurone formel

Le neurone formel est une représentation mathématique simplifiée des neurones biologiques. Il est en général constitué d'un ensemble d'entrées, correspondant aux *dendrites* d'un neurone biologique, auxquelles sont associées des valeurs appelées *poids synaptiques*, et d'une sortie, correspondant à l'*axone* d'un neurone biologique. Le neurone applique à la valeur de sortie une *fonction d'activation*, permettant d'obtenir la valeur de sortie du neurone.

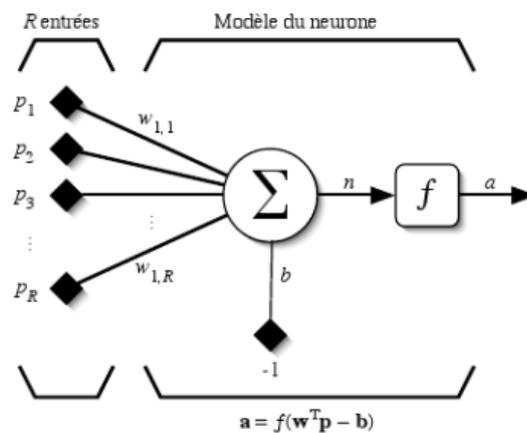


FIGURE 9.3 – Un neurone formel

L'implémentation des signatures utilise une variante du neurone formel de McCulloch et Pitts [84]. Ce neurone calcule la somme des entrées pondérées par leurs poids synaptiques. J'utilise une fonction sigmoïde centrée sur 0 comme fonction d'activation. Cette fonction permet en effet d'obtenir les trois états de sortie dont nous avons besoin : succès (1), échec (-1) et neutre (0). Un biais b est ajouté à l'ensemble des poids. J'ai choisi ce type de neurone pour sa simplicité et pour sa facilité de mise en œuvre, mais également pour sa robustesse dans la reconnaissance de *patterns* dans les vecteurs d'entrées. La fonction caractérisant ces neurones est défini par :

$$y = g \left(\sum_{k \in [1;n]} (x_k \cdot w_k) + b \right) \quad (9.2)$$

Les poids synaptiques d'un neurone formel sont ajustés au cours d'un d'apprentissage. Nous utilisons un apprentissage basé sur la *loi Delta*. Notons que le biais est ici considéré comme un poids w_{n+1} associé à une entrée dont la valeur est toujours à 1 :

$$\begin{aligned} \Delta &= x_{k,t+1} - y_t \\ w_k^{t+1} &= w_k^t + \alpha \times x_{k,t} \times \Delta \\ \forall k \in [1; n+1], \alpha &\text{ learning rate, } \alpha \in [0; 1] \end{aligned} \quad (9.3)$$

Où $x_{k,t+1}$ est la sortie attendue et y_t la sortie observée. L'intérêt de ce modèle est qu'il permet de faire émerger les entrées pertinentes pour la prédiction de la valeur de sortie.

Implémentation des signatures

La signature d'une interaction i (ou d'une interaction composite φ) est constituée par le vecteur des poids W_i de son neurone. La signature d'une interaction est ainsi caractérisée par un vecteur $S_i = [w_{i,1}, \dots, w_{i,n}, b_i] \in \mathbb{R}^{n+1}$. Le vecteur de poids permet de définir, pour chaque interaction primitive i_k de I , la certitude que i_k permettent de détecter l'objet qui afforde une interaction i : chaque poids donne la corrélation (positive ou négative) entre la détection de l'objet post-énaction d'une interaction i_k et le résultat de l'énaction de i : les interactions liées à des connexions dotées d'un poids positif permettent de détecter l'objet, tandis que les interactions liées à des connexions dotées d'un poids négatif permettent de détecter l'absence de l'objet.

certitude Nous supposons que les certitudes de succès et d'échec sont mutuellement exclusives. Il est donc possible de définir une valeur de certitude unique caractérisant la certitude de succès et d'échec, que l'on note $c : \Phi \times \mathcal{P}(I) \rightarrow [-1; 1]$, avec une certitude absolue de succès si $c(\varphi, E_t) = 1$ et une certitude absolue d'échec si $c(\varphi, E_t) = -1$. L'implémentation du mécanisme avec un neurone formel permet de définir la fonction de certitude comme :

$$\begin{aligned} c(\varphi, E_t) &= g \left(\sum_{k \in [1;n]} (\epsilon_k(t) \cdot w_{\varphi,k}) + b_\varphi \right) \\ g(x) &= \left(\frac{2}{1 + e^{-x}} \right) - 1 \end{aligned} \quad (9.4)$$

Avec φ une interaction composite, $\epsilon_k(t)$ le résultat de la $k^{\text{ième}}$ interaction primitive de l'ensemble I , $w_{\varphi,k}$ le poids de la signature de φ associé à la $k^{\text{ième}}$ interaction primitive, et b_φ le biais de la signature de φ .

Une fonction d'activation sigmoïde permet de restreindre la valeur de certitude dans l'intervalle $[-1, 1]$. La sortie de chaque neurone donne alors la certitude de succès ou d'échec de l'interaction qui lui est associé (Figure 9.4).

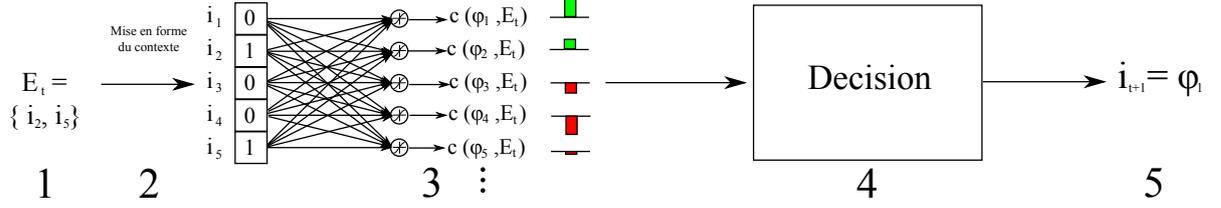


FIGURE 9.4 – Implémentation des signatures d’interactions. 1) l’agent expérimente le contexte E_t . Celui-ci est mis sous la forme d’un vecteur afin de pouvoir être utilisé par le réseau neuronal implémentant les signatures (2). Les signatures déterminent la certitude de chaque interaction $\varphi \in \Phi$ (3). Ces certitudes sont utilisées par le système de décision (4) pour sélectionner l’interaction intention i_{t+1} du cycle suivant (5).

Apprentissage des signatures Le mécanisme d’apprentissage des signatures permet, à chaque cycle de décision t , d’apprendre et/ou renforcer les signatures des interactions à partir du contexte interactionnel observé au pas $t-n$ (c’est-à-dire avant l’énaction) et du résultat de l’énaction des interactions. L’apprentissage s’effectue chaque fois qu’une interaction se termine comme un succès ou un échec, en utilisant la *loi delta* (équation 9.5). Il s’agit donc d’un apprentissage supervisé, puisque le mécanisme fournit à la fois l’entrée et la sortie du réseau. L’apprentissage s’effectue en continu tout au long de la vie de l’agent. Chaque prédiction est ainsi basée sur l’ensemble des expériences passées de l’agent.

$$\begin{aligned} \Delta &= res_\varphi(t) - c(\varphi, E_{t-n}) \\ w_{\varphi,k}^{t+1} &= w_{\varphi,k}^t + \alpha \times \epsilon_k(t-n) \times \Delta \\ \forall k \in [1; n+1], \alpha &\text{ learning rate, } \alpha \in [0; 1] \end{aligned} \quad (9.5)$$

Avec $res_\varphi(t)$ le résultat de φ dans le cas où elle a été énoncée avec succès ($res_\varphi(t) = 1$) ou si elle a échoué ($res_\varphi(t) = -1$), $c(\varphi, E_{t-n})$ la prédiction de φ avant que celle-ci soit énoncée au cycle de décision $t-n$ (avec n le nombre de cycle de décision nécessaire pour énoncer φ), $w_{\varphi,k}$ un poids de la signature de φ , avant et après mise à jour, et $\epsilon_k(t-n)$ l’entrée du contexte interactionnel associé à ce poids, au cycle de décision $t-n$.

Notons que le biais b_φ est comparable à un poids $w_{\varphi,n+1}$ associé à une entrée ϵ_{n+1} pour laquelle la valeur est 1 à chaque cycle t . Le système d’apprentissage est illustré par la Figure 9.5.

Réversibilité des signatures Notre implémentation des signatures permet une réversibilité aisée : lorsqu’une interaction φ est considérée comme énonçable (ou non énonçable) d’après la mémoire spatiale, mais dont le résultat ne peut être prédit d’après le contexte interactionnel courant, on peut compléter ce dernier en ajoutant (respectivement en soustrayant) le vecteur de poids caractérisant la signature de φ . La valeur des poids de la signature donne en effet la certitude que chaque interaction primitive ait pu permettre de détecter l’objet pré-énaction de φ , et donc la certitude qu’elle aurait pu être énoncée au cycle de décision précédent. Ainsi, le

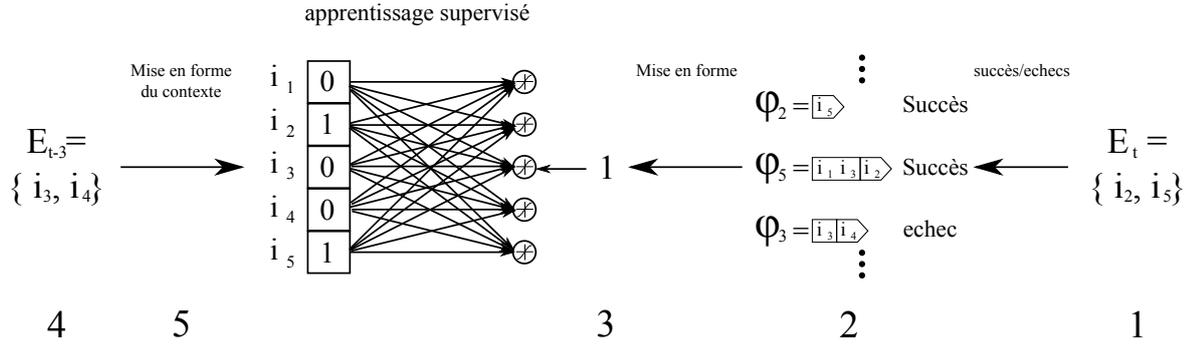


FIGURE 9.5 – Apprentissage des signatures. À la fin du cycle de décision t , l'agent expérimente le contexte E_t (1). Le succès ou l'échec de chaque interaction φ est caractérisé (2), et utilisé pour fournir les valeurs de sortie de référence du réseau neuronal implémentant les signatures (3). Les contextes E_{t-n} des cycles précédents (4) sont mis sous la forme de vecteurs (5). La signature de chaque interaction φ complétée comme un succès ou un échec est renforcée à l'aide du contexte courant E_t et du contexte précédent E_{t-n} (avec n nombre de cycle de décision pour énoncer φ).

mécanisme peut générer un contexte complété $E't = [\epsilon'_1, \dots, \epsilon'_n]$, défini par :

$$E'_t = E_t + \sum_{\varphi \in \Phi} S'_\varphi \times c_{\varphi,t} \quad (9.6)$$

avec S'_φ la signature de φ à laquelle on a retiré le biais et $c_{\varphi,t}$ la certitude que l'interaction composite φ puisse être énoncée, fourni par la mémoire spatiale, indépendamment du contexte interactionnel E_t . Notons que pour tout $k \in [1; n]$, ϵ'_k est borné par $[-1; 1]$.

Une signature S'_i est ajoutée si elle est *compatible* avec le contexte interactionnel courant E_t . Une signature est dite *compatible* si elle satisfait la condition suivante :

$$\forall k \in [1; n], c_{\varphi,t} \times w_{\varphi,k} \times \epsilon_{k,t} \geq 0 \quad (9.7)$$

Une interaction définie comme non compatible permet la détection d'erreurs dans sa signature, ou dans le mécanisme à l'origine de la prédiction $c_{\varphi,t}$.

9.1.5 Implémentation du mécanisme de sélection

Le mécanisme de sélection est constitué de deux parties, l'une permettant l'apprentissage des signatures d'interaction, et l'autre permettant l'exploitation de la mémoire spatiale. Les tests ont été menés tout d'abord sans le mécanisme d'apprentissage, puis avec ce mécanisme.

Mécanisme d'apprentissage

La sélection du mécanisme d'apprentissage s'effectue en deux étapes : une première étape détermine, lorsque c'est possible, une interaction pouvant être testée dans le contexte interac-

tionnel courant E_t . Dans le cas contraire, le mécanisme détermine une interaction pour laquelle une interaction épistémique pourrait, en cas de succès, générer un contexte favorable à son apprentissage. Si le mécanisme d'apprentissage ne sélectionne aucune interaction, le mécanisme d'exploitation est alors utilisé.

Pour pouvoir être sélectionnée, une interaction doit être non encore étiquetée comme fiable, toujours vraie ou décorrélée. Il n'est en effet pas nécessaire de tester une interaction si celle-ci est déjà considérée comme fiable. Les interactions composites constituant les sous séquences qui la composent doivent être fiables. En effet, l'apprentissage d'une interaction composite repose sur la fiabilité des interactions composites qui composent l'interaction sélectionnée. Ceci permet à l'agent d'apprendre des interactions composites de longueur croissante. Le path doit également être énable d'après la mémoire spatiale. Cependant, cette condition peut être trop restrictive, notamment au début de l'apprentissage. Pour éviter ce problème, une interaction peut être sélectionnée si son path n'est pas considéré comme impossible. Cependant, si le path échoue trop souvent, la condition d'origine sera restaurée pour cette interaction : dans notre implémentation, si le path d'une interaction échoue dix fois, cette interaction ne peut plus être sélectionnée que si son path est considéré comme énable.

L'apprentissage est optimisé en permettant d'alterner les contextes permettant des prédictions de succès et d'échec des interactions. Ainsi, on évite qu'une interaction soit testée plusieurs fois de façon consécutive dans un même contexte, ce qui est notamment possible dans le cas d'interactions ne produisant pas de mouvement.

Afin d'alterner les tests de prédictions de succès et d'échec, on ajoute la condition suivante : l'interaction est candidate si elle est prédite comme un succès ($c(S_\varphi, E_t) > 0$) et si le nombre de prédictions de succès correctes consécutives est inférieur au nombre de prédictions d'échec correctes consécutives, ou si elle est prédite comme un échec ($c(S_\varphi, E_t) < 0$) et si le nombre de prédictions d'échec correctes consécutives est inférieur au nombre de prédictions de succès correctes consécutives. Le cas où le résultat d'une interaction ne peut être prédit ($c(S_\varphi, E_t) = 0$) est pris en compte dans le cas d'une sélection nécessitant une interaction épistémique. Notons également que dans le cas d'une interaction qui n'a jamais échoué et dont la signature a permis de prédire un succès à plusieurs reprises (une interaction potentiellement toujours vraie), l'interaction peut être testée si elle est prédite comme un succès.

Dans chaque étape du mécanisme de sélection, le mécanisme sélectionne l'interaction, parmi les interactions candidates, celle considérée comme la moins fiable, c'est-à-dire l'interaction pour laquelle le minimum, entre le nombre de prédictions de succès correctes consécutives et le nombre de prédictions d'échec consécutives, est le plus faible. En cas d'égalité entre plusieurs interactions, le mécanisme sélectionne la première d'entre elles dans l'ordre donné par la liste des interactions composites.

Mécanisme d'exploitation

Le mécanisme d'exploitation a un fonctionnement proche de celui décrit dans le chapitre 7. Le mécanisme doit cependant être adapté selon qu'il est utilisé ou non avec le mécanisme d'apprentissage. Le but du mécanisme est de fournir la liste des interactions composites considérées comme énonçables dans le contexte interactionnel actuel. Il convient donc de sélectionner les interactions composites énonçables, c'est-à-dire :

- considérées comme possibles d'après leur signature,
- dont le path est énonçable d'après la mémoire spatiale.

Le mécanisme détermine les interactions dont le path est énonçable d'après la mémoire spatiale et dont la certitude de succès est considérée comme suffisante. Dans le cas d'une utilisation avec le mécanisme d'apprentissage, nous pouvons considérer qu'une interaction peut être considérée comme énonçable si la certitude de succès est supérieure ou égale à 0. Notons qu'une certitude de 0 indique seulement que l'interaction n'est pas considérée comme impossible. Dans le cas où le mécanisme d'apprentissage n'est pas utilisé, il faut permettre la sélection d'une interaction considérée comme impossible. Sa signature pourrait en effet contenir des erreurs, ce qui, dans certains contextes, pourrait donner une certitude de succès erronée. Afin de contourner ce problème, on définit un seuil de certitude négatif, ce qui permet la sélection d'interactions prédites comme un échec, mais dont le faible nombre de tests, caractérisé par des poids de faible valeur, permet de douter de la validité de cette prédiction. Ce seuil est fixé à -0.2. En effet, avec les paramètres utilisés, une certitude de 0.2 (en valeur absolue) indique que la signature n'a été testée qu'une seule fois dans le contexte interactionnel actuel, ce qui est insuffisant pour définir la pertinence de la signature.

Une autre différence entre les deux versions du mécanisme est que l'on utilisera des contextes différents pour définir la certitude et la validité des paths : l'utilisation d'un mécanisme d'apprentissage permet un certain niveau de fiabilité des signatures d'interaction, ce qui permet d'utiliser les contextes les plus complétés. Le mécanisme d'exploitation utilisé avec le mécanisme d'apprentissage utilisera ainsi le troisième contexte complété (E''') pour générer les prédictions et les mémoires $\mathcal{T}3$ et $\mathcal{F}3$ pour définir la faisabilité du path, tandis que la version n'utilisant pas le mécanisme d'apprentissage utilisera le second contexte (E'') et les mémoires $\mathcal{T}2$ et $\mathcal{F}2$, jugées plus fiables. En effet, sans mécanisme d'apprentissage, l'accumulation d'erreurs dans le troisième contexte interactionnel E''' ne permet pas à l'agent de générer des comportements adaptés à son environnement.

Mécanisme de décision

Le mécanisme de décision sélectionne ensuite l'interaction composite la plus satisfaisante parmi celles définies par les mécanismes de sélection. Notons que le mécanisme d'apprentissage définit une liste contenant une seule interaction, permettant de forcer le mécanisme de décision à tester cette interaction. Le mécanisme de décision décrit ici sélectionne simplement l'interaction

composite pour laquelle la valeur de satisfaction est maximale. Notons que, pour éviter que l'agent ne sélectionne en boucle une même interaction ne produisant pas de mouvement, et donc un changement de contexte environnemental, il faut détecter et éliminer ces interactions de la sélection. Une interaction composite ne produit pas de mouvement si son interaction finale est désignée par sa propre signature. En effet, cette condition implique que l'objet pré-interaction et post-interaction de cette interaction soient identiques. L'interaction, une fois énoncée, pourra être à nouveau énoncée indéfiniment. Il faut cependant noter que si l'interaction a une valeur de satisfaction élevée, il peut être intéressant pour l'agent d'énoncer en boucle cette interaction. On pourra alors éviter le bouclage en ajoutant une forme d' "ennui" qui réduit la satisfaction d'une interaction primitive quand elle est trop souvent énoncée. Cependant, les interactions de notre système qui ne produisent pas de mouvement, c'est-à-dire les interactions *toucher* et *se cogner*, ont toutes une valeur de satisfaction négative, ce qui rend inutile un tel mécanisme. La condition de la sélection est donc définie par :

$$i_{t+1} \in \Phi / s_{i_{t+1}} = \max_{\varphi \in \Phi} (s_{\varphi}) \wedge (i_{t+1,f} \notin S_{i_{t+1}} \vee s_{i_{t+1}} > 0) \quad (9.8)$$

Avec i_{t+1} l'interaction sélectionnée, $i_{t+1,f}$ l'interaction finale de i_{t+1} , Φ l'ensemble des interactions composites, s_{φ} la valeur de satisfaction d'une interaction composite φ et S_{φ} la signature d'interaction de φ .

Mécanisme de sélection des interactions épistémiques

Dans le cas où la certitude de succès est strictement positive, l'agent énoncera cette interaction. Dans le cas contraire, c'est-à-dire si le résultat ne peut pas être prédit, une interaction épistémique sera sélectionnée et énoncée. La sélection d'une interaction épistémique dépend de l'information recherchée. Dans le cas d'une interaction sélectionnée pour permettre un apprentissage, l'interaction épistémique doit permettre de définir un contexte propice à l'apprentissage de la signature de cette interaction. Dans le cas d'une interaction sélectionnée par le mécanisme d'exploitation, on sélectionnera l'interaction épistémique permettant de définir un contexte dans lequel la certitude du résultat de l'énonciation de cette interaction sera maximale (en valeur absolue).

Dans le premier cas, l'interaction épistémique choisie sera celle proposant une information sur l'entrée non *déconnectée*, c'est-à-dire que l'interaction correspondante n'a pas été supprimée de la signature d'interaction, dont le poids est le plus faible (en valeur absolue) de la signature de l'interaction que l'agent veut tester. L'agent peut ainsi tester cette entrée afin de savoir si l'objet qui afforde l'interaction sélectionnée peut être détecté par le biais de cette interaction épistémique. Cette méthode permet de tester et renforcer un à un les poids des signatures. L'interaction épistémique $i_{ep,\varphi}$ permettant l'apprentissage d'une interaction φ est défini par :

$$i_{ep,\varphi} = i_k \in I / |w_{\varphi,k}| = \min_{l \in [1;n]} (|w_{\varphi,l}|) \quad (9.9)$$

Avec I l'ensemble des interactions primitives, $w_{\varphi,k}$ le poids de la signature de φ associé à la $k^{\text{ième}}$ interaction de l'ensemble I , et $n = \text{Card}(I)$

Dans le second cas, l'interaction épistémique choisie sera celle proposant une information sur l'entrée dont le poids est le plus fort (en valeur absolue). En effet, la valeur absolue du poids d'une signature d'interaction caractérise la fiabilité du lien entre l'interaction primitive associée au poids et l'interaction sélectionnée, car considéré comme celui qui a été le plus testé. Soit E_i le contexte que l'on obtiendrait après éaction d'une interaction i . L'interaction épistémique $i_{ep,\varphi}$ permettant de valider une interaction φ est définie par :

$$i_{ep,\varphi} = i \in I / |c(\varphi, E_{i_{ep,\varphi}})| = \max_{i \in I} (|c(\varphi, E_i)|) \quad (9.10)$$

Avec $E_{i_{ep,\varphi}}$ le contexte interactionnel que l'on obtiendrait en supposant le succès de $i_{ep,\varphi}$, et $c(\varphi, E)$ la certitude de succès et d'échec d'une interaction φ dans un contexte interactionnel E .

Une fois l'interaction épistémique éactée, avec succès ou non, le mécanisme de décision détermine si l'interaction préalablement sélectionnée respecte toujours les conditions requises, que ce soit pour un apprentissage ou une exploitation. Si c'est le cas, cette interaction est éactée. Dans le cas contraire, le mécanisme annule la dernière décision et sélectionne une nouvelle interaction.

Mécanisme de détection des erreurs

Sans avoir accès aux objets physiques qui affordent les interactions, il est possible de détecter la présence d'erreurs dans leurs signatures. La détection d'erreur se base sur la détection d'incohérences entre les contextes et les signatures des interactions stockées par la mémoire spatio-séquentielle. Deux mécanismes permettent de détecter de telles incohérences.

Le premier mécanisme consiste en une détection des incohérences entre les signatures des interactions stockées par la mémoire spatio-séquentielle et le contexte interactionnel :

si $\exists \varphi \in \mathcal{T} / c(\varphi, E_t) < 0 \vee \exists \varphi \in \mathcal{F} / c(\varphi, E_t) > 0$, alors l'interaction φ n'est pas cohérente, puisque l'objet qu'elle désigne n'est pas compatible avec le contexte réellement expérimenté par l'agent, qui est exempt d'erreur. Il n'est cependant pas possible de savoir si l'erreur provient de l'interaction φ , ou de l'interaction composite φ' initialement intégrée par la mémoire spatiale, et dont les mises à jour consécutives ont mené à φ . La fiabilité de ces deux interactions est alors remise à zéro, c'est-à-dire qu'elles sont considérées comme des interactions jamais testées, ce qui permet de recommencer leur apprentissage sans toutefois remettre à zéro les poids déjà appris. Notons que ce mécanisme est le plus efficace des deux proposés, car chaque incohérence détectée conduit à la mise à zéro de la fiabilité d'au plus deux interactions composites.

Le second mécanisme détecte les incohérences lors de la construction des contextes augmentés E' à E''' à partir des interactions stockées en mémoire. Une incohérence est détectée si deux interactions gérées dans la mémoire proposent des états incompatibles pour une même entrée d'un contexte interactionnel :

$$\begin{aligned}
& \exists \varphi_1, \varphi_2 \in \mathcal{T}, \exists k \in [1; n] / w_{\varphi_1, k} \times w_{\varphi_2, k} < 0 \\
& \vee \exists \varphi_1, \varphi_2 \in \mathcal{F}, \exists k \in [1; n] / w_{\varphi_1, k} \times w_{\varphi_2, k} < 0 \\
& \vee \exists \varphi_1 \in \mathcal{T}, \varphi_2 \in \mathcal{F}, \exists k \in [1; n] / w_{\varphi_1, k} \times w_{\varphi_2, k} > 0
\end{aligned} \tag{9.11}$$

Avec \mathcal{T} et \mathcal{F} les listes des interactions considérées comme possibles et impossibles, et $w_{\varphi, k}$ le poids de la signature de φ associé à la $k^{\text{ième}}$ interaction de l'ensemble I .

Dans ce cas, φ_1 et φ_2 ne sont pas cohérentes. Cependant, il est impossible de déterminer laquelle de ces deux interactions contient des erreurs. Il est également impossible de dire si l'erreur provient des interactions relevées ou des interactions composites initialement détectées. Ainsi, toutes les interactions concernées voient leur fiabilité remise à zéro. Ce mécanisme est donc peu efficace puisque chaque incohérence détectée peut produire la mise à zéro de la fiabilité d'un grand nombre d'interactions. Afin de limiter le temps nécessaire pour apprendre à nouveau les interactions détectées, je propose de ne pas remettre à zéro la fiabilité des interactions fiables, mais seulement de réduire leur fiabilité, afin de réduire le nombre de tests nécessaires pour que ces interactions soient à nouveau considérées comme fiables. Les interactions fiables étant en effet les interactions les plus testées, il est moins probable que leurs signatures contiennent des erreurs.

Je propose un troisième mécanisme dédié aux interactions qui ne peuvent pas échouer. Ces interactions sont caractérisées par le fait qu'un de leurs sous-chemins est défini par une interaction composite dont l'objet pré-énaction inclut celui de l'interaction considérée. De façon à simplifier le mécanisme de sélection, je considère uniquement les interactions toujours vraies pour lesquelles tous les sous-chemins sont abordés par un objet incluant celui qui aborde l'interaction considérée. Ainsi, nous pouvons considérer que lorsqu'une interaction toujours vraie est considérée comme possible, alors elle est également énable. Ainsi, lorsque l'énaction du path d'une interaction toujours vraie, définie comme possible, échoue, on considère que le contenu du contexte précédant l'énaction ne fait pas partie de l'objet qui aborde cette interaction. On peut ainsi procéder à la déconnexion des entrées du contexte concernées.

9.2 Expérimentation

Les essais ont été menés avec des interactions composites limitées à une longueur de deux, puis de trois interactions primitives. Le mécanisme de décision a d'abord été testé sans mécanisme d'apprentissage, de façon à observer comment des signatures et des comportements peuvent émerger simultanément. Le mécanisme d'apprentissage a ensuite été ajouté pour obtenir des signatures fiables qui ne soient pas seulement basées sur le comportement émergent de l'agent, mais qui caractérisent les objets qui abordent les interactions associées à ces signatures.

9.2.1 Affichage des résultats

Les résultats issus des expérimentations de ce chapitre sont nombreux et de nature variée. Différents affichages permettent de résumer ces résultats pour en extraire des informations per-

tinentes. Les propriétés des signatures d'interactions sont présentées pour observer d'une part la position des objets qu'elles caractérisent, et d'autre part le type d'élément de l'environnement qu'elles caractérisent. La *position* et le *type* des objets sont bien sûr définis du point de vue d'un observateur extérieur, puisque l'agent n'a pas accès à ces informations.

Affichage des signatures d'interaction

Le contexte interactionnel, tel que nous l'avons défini précédemment, comporte six entrées, correspondant aux six interactions primitives *toucher*. Les signatures des interactions sont donc constituées d'un vecteur de six poids. Cependant, les interactions *toucher un espace vide* et *toucher un mur* sont *opposées*, c'est-à-dire que l'échec de l'une conduit à l'énaction de l'autre, et le contexte interactionnel que nous proposons dans cette expérimentation prend en compte les échecs des interactions. De ce fait, la valeur des poids associés aux interactions *toucher un espace vide* seront opposées aux valeurs des poids associés aux interactions *toucher un mur*. Nous proposons ainsi de n'afficher que les trois valeurs des poids associées aux interactions *toucher un espace vide*. Afin de faciliter la lecture des signatures, chaque poids est affiché sous la forme d'un carré dont la couleur représente la valeur : vert pur pour 1 et rouge pur pour -1. Ces carrés sont disposés de façon à respecter la position, par rapport à l'agent, de l'objet qui afforde l'interaction associée au poids. La Figure 9.6 illustre l'affichage des signatures.

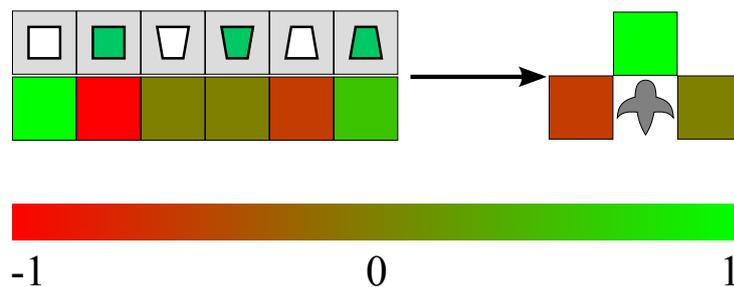


FIGURE 9.6 – Affichage simplifié des signatures d'interaction. En haut à gauche, une signature d'interaction, constituée d'un vecteur de six poids. En haut à droite, l'affichage simplifié : seules les trois valeurs associées aux interactions *toucher un espace vide* sont affichées. Les positions des poids respectent les positions, par rapport à l'agent, qui affordent les trois interactions primitives. L'agent est supposé au milieu des trois carrés, orienté vers le haut. En bas, le gradient des couleurs associés aux poids.

Les positions des objets affordant les interactions peuvent être déterminées par la valeur absolue des poids des signatures d'interaction. En effet, on considère que chaque interaction primitive permet de considérer, par son succès, un objet (ou son absence en cas d'échec de l'interaction), situé à une certaine position de l'espace. Les signatures permettent ainsi d'estimer la position de l'objet qui afforde leur interaction à partir des positions désignées par les interactions composant ces signatures. Le premier dispositif d'affichage permet de rendre compte des positions des objets qui affordent les interactions en fonction de leurs signatures. Dans cet affichage, on

place les interactions primitives aux positions occupées par leurs objets post-énaction par rapport à l'agent. Les interactions composites sont représentées sous la forme de points dont la couleur dépend de la position réelle (inconnue de l'agent) des objets qui les affordent, et dont la position est définie par les valeurs absolues des poids de leurs signatures. Ceci permet de regrouper les interactions liées à une même interaction primitive (par un poids positif ou négatif), et donc liées à une même position de l'espace. Un lien est tracé entre une interaction composite et l'interaction composite de longueur 1 qui constitue son interaction finale. Ces deux interactions caractérisant par définition un même objet, ces liens nous permettront de visualiser les "mouvements" des objets, par rapport à l'agent, générés par l'énaction du path de ces interactions composites. Le système d'affichage est illustré par la Figure 9.7.

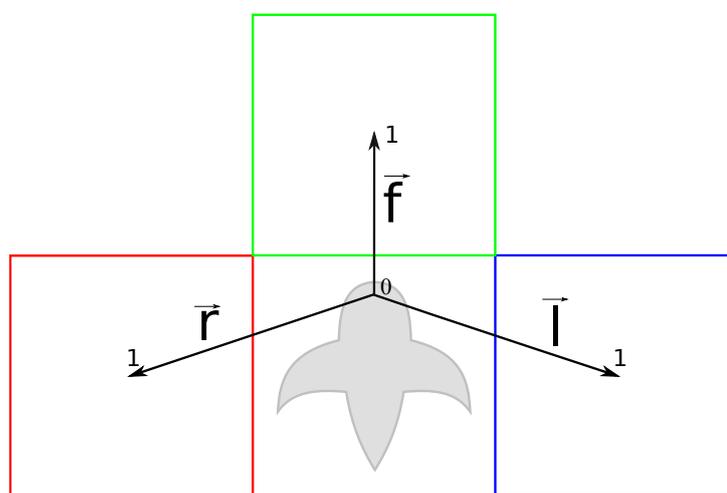


FIGURE 9.7 – Affichage de la position des interactions composites définie par leurs signatures. Les carrés rouge, vert et bleu représentent la position des objets qui affordent respectivement les interactions primitives *toucher à gauche un espace vide*, *toucher devant un espace vide*, *toucher à droite un espace vide*, par rapport à l'agent (au milieu des trois carrés). La position d'une interaction composite φ est défini par $|w_{\varphi,0}| \cdot \vec{f} + |w_{\varphi,2}| \cdot \vec{r} + |w_{\varphi,4}| \cdot \vec{l}$, avec $w_{\varphi,0}$ le poids associé à l'entrée *toucher un espace vide devant*, $w_{\varphi,2}$ le poids associé à l'entrée *toucher un espace vide à droite*, et $w_{\varphi,4}$ le poids associé à l'entrée *toucher un espace vide à gauche*.

Un second affichage permet d'observer le regroupement des interactions composites affordées par un même type d'objet, indépendamment de sa position. Dans cet affichage, les interactions sont représentées par des points dont la couleur dépend de l'objet réel qui les affordent. Ces points évoluent librement dans la surface d'affichage. On applique dans un premier temps une force d'attraction entre les interactions composites qui partagent une même interaction finale, puisque caractérisant un même objet par définition. On observe ensuite les similarités entre les signatures : des signatures identiques caractérisent un même objet à une même position de l'espace. On applique alors une force d'attraction entre les interactions dont les signatures sont similaires, et une force répulsive entre les interactions dont les signatures sont incompatibles.

Nous pouvons ainsi observer l'émergence de *bundles* d'objets implicitement définis par les signatures. Ces *bundles* caractérisent les interactions évoquées simultanément par un même contexte interactionnel. La Figure 9.8 résume le principe de ce système d'affichage.

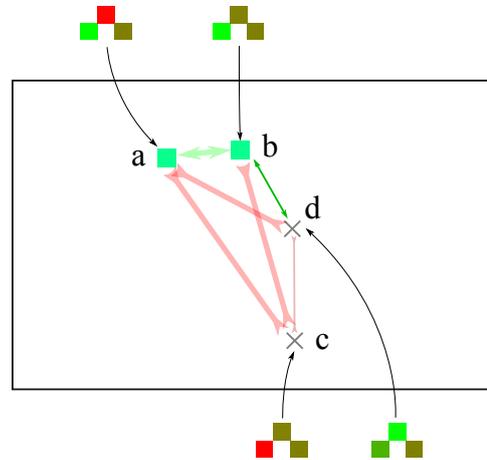


FIGURE 9.8 – Affichage des *bundles* d'objets. Une force d'attraction est appliquée entre les interactions partageant une même interaction finale. Une force d'attraction ou de répulsion est ensuite appliquée entre chaque paire d'interactions en fonction de la similarité entre leurs signatures. Dans cet exemple, les interactions *a* et *b* sont fortement attirées et se rapprochent l'une de l'autre, car leurs signatures sont fortement liées positivement à une même interaction. Les interactions *a* et *c*, et *b* et *c* se repoussent, car leurs signatures caractérisent des contextes mutuellement incompatibles. Les *bundles* sont formés par des groupes d'interactions proches.

Affichage du comportement de l'agent

Les comportements sont représentés sous la forme d'une *timeline* donnant la séquence d'interactions primitives utilisées dans chaque configuration environnementale. À ces *timelines* sont superposées les interactions composites sélectionnées par le mécanisme à chaque cycle de décision, à l'origine des interactions primitives énoncées.

Nous observerons également l'évolution du contenu de la mémoire spatiale dans les différentes configurations environnementales, dans le but de mettre en évidence la façon dont la mémoire permet à l'agent de construire son contexte interactionnel, et comment elle permet de suivre les objets préalablement détectés. Notons que seules les interactions pertinentes de la mémoire, c'est-à-dire celles permettant la construction des contextes augmentés et le suivi des objets seront affichées. Le listing complet de la mémoire spatiale est affiché en annexe A1. La Figure 9.9 montre l'affichage permettant de suivre le contenu de la mémoire spatiale.

9.2.2 Étude du comportement sans mécanisme d'apprentissage

Nous observons dans un premier temps le comportement de l'agent, c'est-à-dire les séquences d'interaction utilisées dans chaque contexte spatial, mais également les interactions sélectionnées

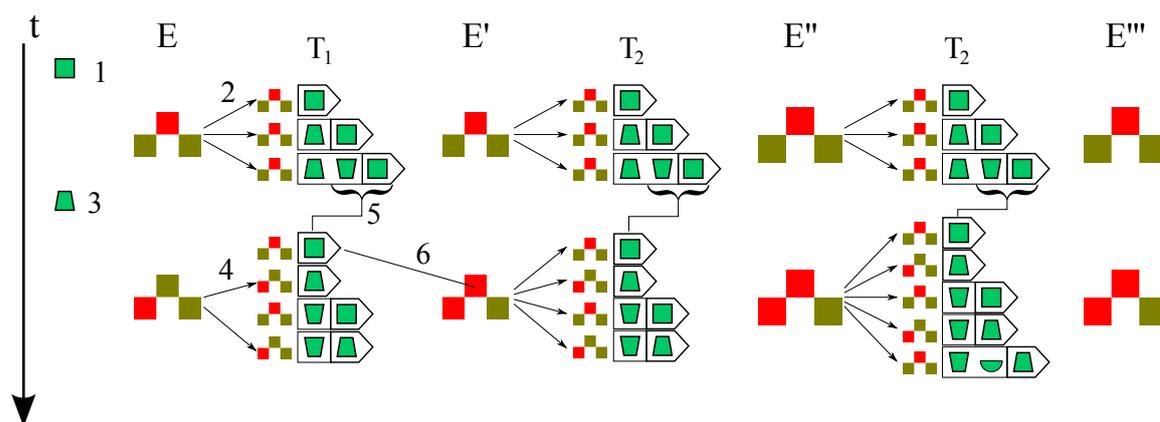


FIGURE 9.9 – Suivi du contenu de la mémoire spatiale. Chaque ligne représente un cycle de décision. Gauche à droite : les interactions énoncées, le contexte interactionnel E , la mémoire $\mathcal{T}1$, les contextes augmentés E' à E''' et le contenu des mémoires $\mathcal{T}2$ et $\mathcal{T}3$. Les interactions pertinentes des mémoires sont affichées avec leurs signatures. Dans cet exemple, l'agent énonce au premier cycle de décision, une interaction (1). Le contexte interactionnel obtenu évoque la possibilité d'énoncer certaines interactions composites (2). Au cycle suivant, l'agent énonce une autre interaction (3). Le contexte permet d'évoquer un nouvel ensemble d'interaction (4). Les mémoires spatiales sont mises à jour (5) pour compléter les mémoires. L'une de ces interactions permet, par sa signature, de compléter le contexte augmenté E' (6), ainsi que les contextes suivants (E'' et E''').

par l'agent et le contenu de la mémoire spatio-séquentielle, à l'origine de ces décisions. Nous comparerons ensuite les signatures d'interaction pour observer l'émergence des positions dans l'espace et des types d'objet.

On laisse dans un premier temps l'agent évoluer dans son environnement, jusqu'à obtenir une stabilisation du comportement, c'est-à-dire que les séquences utilisées pour chaque contexte spatial ainsi que les signatures d'interaction n'évoluent plus.

Dans le cas d'une utilisation d'interactions composites limitées à une longueur de deux interactions primitives, le comportement se stabilise très vite et est d'une grande efficacité : dans toutes les configurations environnementales, il faut moins de 500 cycles de décision pour obtenir un comportement stable. Les séquences utilisées dans chaque contexte spatial sont données en Figure 9.10. Ces séquences illustrent comment l'agent arrive à caractériser sa situation : il effectue une suite d'interactions épistémiques, dans l'ordre : toucher un espace vide devant, toucher un espace vide à droite. Ces deux interactions sont en effet suffisantes pour définir le contexte spatial dans lequel se trouve l'agent, parmi les trois contextes initiaux. L'agent énonce les interactions épistémiques jusqu'à ce que le contexte interactionnel qui se construit soit favorable à l'énonciation d'une interaction composite disposant d'une forte valeur de satisfaction. L'agent utilise principalement trois interactions composites, énonçables dans chacun des trois contextes spatiaux :

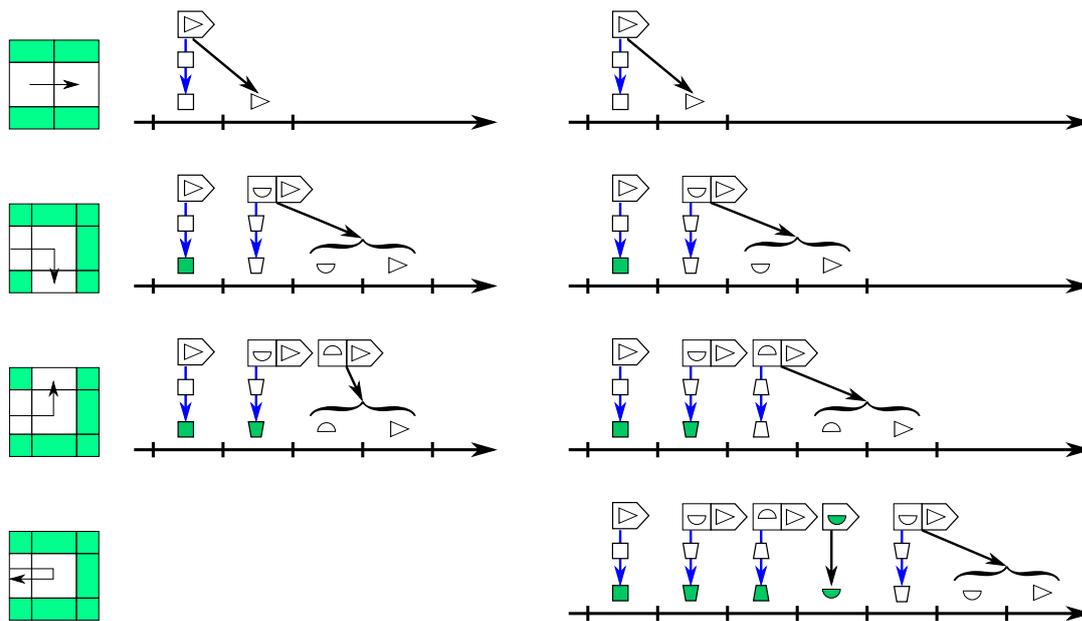


FIGURE 9.10 – Séquences d’interactions utilisées par l’agent dans chaque contexte spatial. De haut en bas : *ligne droite*, *virage à droite*, *virage à gauche* et *voie sans issue*. À gauche sont affichées les séquences utilisées avant ajout du contexte spatial *voie sans issue*, et à droite sont affichées les séquences utilisées après ajout de ce contexte. Sur chaque *timeline*, la ligne supérieure montre les interactions composites sélectionnées par l’agent. La ligne du bas montre les interactions primitives énoncées à chaque pas. Une flèche noire indique que l’agent énonce une interaction composite. Une flèche bleue indique l’énonciation d’une interaction épistémique (l’interaction est affichée sur la flèche). Nous pouvons voir qu’il n’y a pas de différence dans les séquences utilisées dans les contextes spatiaux *ligne droite* et *tourner à droite*. La séquence utilisée dans le contexte *virage à gauche* diffère. En effet, avant l’ajout du bloc mur, l’agent associe l’interaction [tourner à gauche face à un espace vide][*avancer*] à la présence d’un mur à sa droite. Il énonce ainsi cette interaction après avoir énoncé l’interaction toucher un mur à droite. Après ajout du contexte *voie sans issue*, l’agent corrige ce comportement. Il énonce ainsi une interaction épistémique pour considérer le contenu de l’espace à sa gauche.

- [*avancer*] dans le contexte spatial *ligne droite*
- [*tourner à droite face à un espace vide*][*avancer*] dans le contexte spatial *virage à gauche*
- [*tourner à gauche face à un espace vide*][*avancer*] dans le contexte spatial *virage à droite*.

Cette spécialisation des interactions pour chaque contexte spatial montre que l’agent a attribué une sémantique à chaque contexte, chacun étant observable d’après une certaine séquence d’interactions épistémiques, et affiche une certaine interaction composite dotée d’une valeur de satisfaction positive. Le comportement de l’agent peut se résumer à l’organigramme affiché à la Figure 9.11.

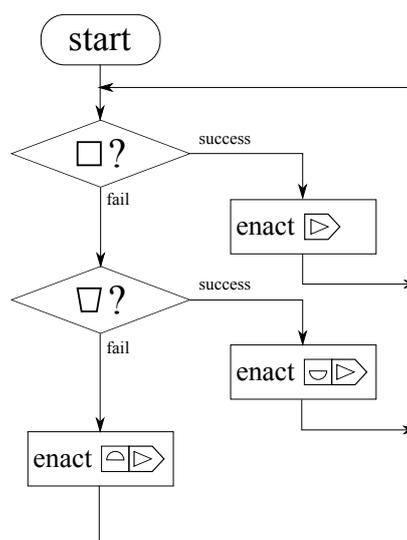


FIGURE 9.11 – Organigramme résumant le comportement de l’agent. Deux interactions épistémiques suffisent à caractériser le contexte dans lequel se trouve l’agent, ce qui lui permet de sélectionner une interaction composite dotée d’une valeur de satisfaction positive.

Lorsque l’on observe la spécialisation des interactions, on remarque que l’agent ne tient pas compte du côté gauche (Figure 9.12 graphe de gauche). Une faible fraction des interactions se sont spécialisées, et beaucoup contiennent des erreurs. En observant les signatures d’interaction, on se rend compte que de nombreuses erreurs sont présentes. Par exemple, l’agent attribue l’interaction *tourner à gauche face à un espace vide* à la présence d’un mur situé à sa droite. En effet, il n’utilise cette interaction que dans le cas d’un contexte spatial *virage à gauche*, qui comprend un espace vide qui afforde cette interaction, et un mur à droite. Ce second élément étant présent à chaque fois que l’interaction est énoncée, il est considéré comme faisant partie de l’objet qui afforde cette interaction. Après stabilisation du comportement, cette interaction n’est utilisée que dans le contexte spatial *virage à gauche*. Ainsi, cette signature, même fautive, reste vraie dans le cadre de son utilisation. Ce genre d’erreur est à l’origine de l’efficacité des séquences utilisées : dans le contexte spatial *tourner à gauche*, l’agent, après avoir énoncé l’interaction *toucher un mur à droite*, énonce l’interaction composite [*tourner à gauche face à un espace vide*][*avancer*) sans énoncer une interaction épistémique supplémentaire. La séquence permettant d’agir dans ce contexte spatial utilise ainsi une interaction épistémique de moins que ce qui aurait été nécessaire si la signature avait été correcte.

On notera également que l’agent a identifié, dans chaque configuration environnementale, trois types d’objets (Figure 9.12, graphe de droite). Les interactions primitives sont réparties entre les *bundles* de la façon suivante :

-bundle a : { *toucher un espace vide devant, avancer d’un pas* }. Ce *bundle* caractérise la présence d’un espace vide devant l’agent.

-bundle b : { *tourner à gauche face à un espace vide, toucher un mur à droite, tourner à*

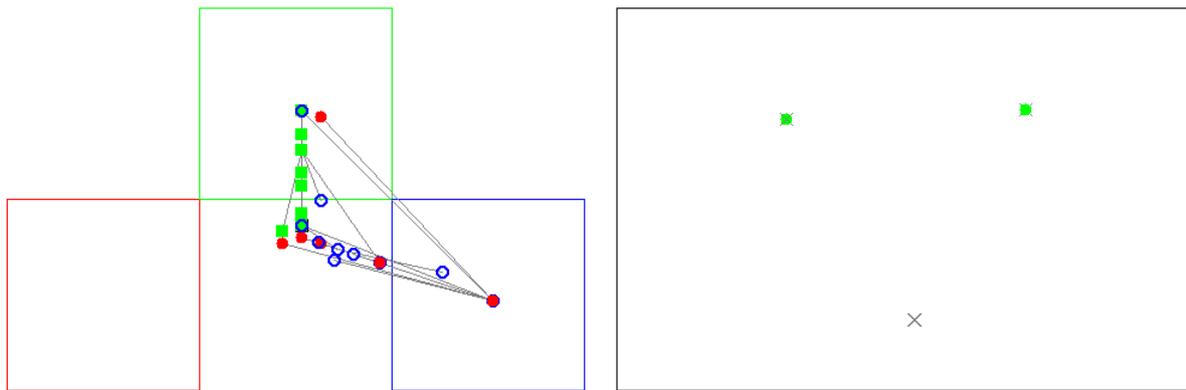


FIGURE 9.12 – Spécialisation des interactions composites en fonction des positions (à gauche), et du type d'objet (à droite). Cette répartition est obtenue dans les quatre configurations de l'environnement. À gauche : les trois carrés correspondent aux trois positions que l'agent peut observer par le biais de ses interactions. Les interactions composites sont représentées par des points dont la position dépend des poids de leur signature, pour indiquer à quelle région de l'espace elles font référence. La couleur de ces points montrent l'affectation réelle des interactions composites (inconnues de l'agent) : rouge : coté gauche, vert : devant, bleu : coté droit, noir : l'interaction est associée à une position qui échappe au système interactionnel de l'agent. Notons que ces interactions, une fois dé-corrélées, ne sont plus représentées. Nous pouvons observer que peu d'interactions composites se sont spécialisées pour une position donnée, et aucune ne caractérise un objet situé à gauche de l'agent. À droite : les interactions composites sont regroupées en fonction des similitudes des signatures afin de faire émerger des *bundles* d'objets. Les interactions sont représentées par des croix grises si elles sont liées à un espace vide, et par des points verts si elles sont liées à un mur. Les interactions composites ont formé trois *bundles*, qui ne correspondent pas aux objets qui affordent réellement ces interactions.

droite face à un mur }. Ce *bundle* caractérise la présence d'un mur à droite de l'agent ou un espace vide à gauche de celui-ci. Ce *bundle* illustre le fait que pour l'agent, un espace vide situé sur l'un de ses cotés est équivalent à la présence d'un mur de l'autre coté.

-bundle c : { *toucher un espace vide à droite, toucher un espace vide à gauche, tourner à droite face à un mur, toucher un mur à gauche, toucher un mur devant, se cogner, tourner à gauche face à un mur* }

Ces *bundles* montrent que les objets sont considérés selon la façon dont l'agent les expérimente et interagit avec eux : ainsi, puisque la présence d'un mur sur l'un des cotés afforde *tourner* de l'autre coté face à un espace vide, alors des interactions, pourtant liées à des éléments différents, sont considérées comme affordées par un même objet.

Nous étudions maintenant le contenu de la mémoire spatiale pendant l'énaction de la séquence utilisée dans le contexte spatial *virage à gauche*, dans la configuration *small loop*. Le contenu de la mémoire est affiché dans la Figure 9.13. Cette séquence est en effet la plus longue et donc la

plus propice à l'observation d'une persistance des objets. Nous pouvons ainsi observer l'agent construire un contexte interactionnel augmenté : au pas 1, le bloc devant l'agent est intégré sous la forme d'une interaction composite, qui mise à jour au pas 2, indique à l'agent la présence d'un objet qui empêche l'énaction de l'interaction avancer. Les contextes augmentés caractérisent le succès de deux interactions primitives : toucher un mur devant et toucher un mur à droite. Cette parallélisation permet à l'agent de tenir compte de plusieurs éléments simultanément.

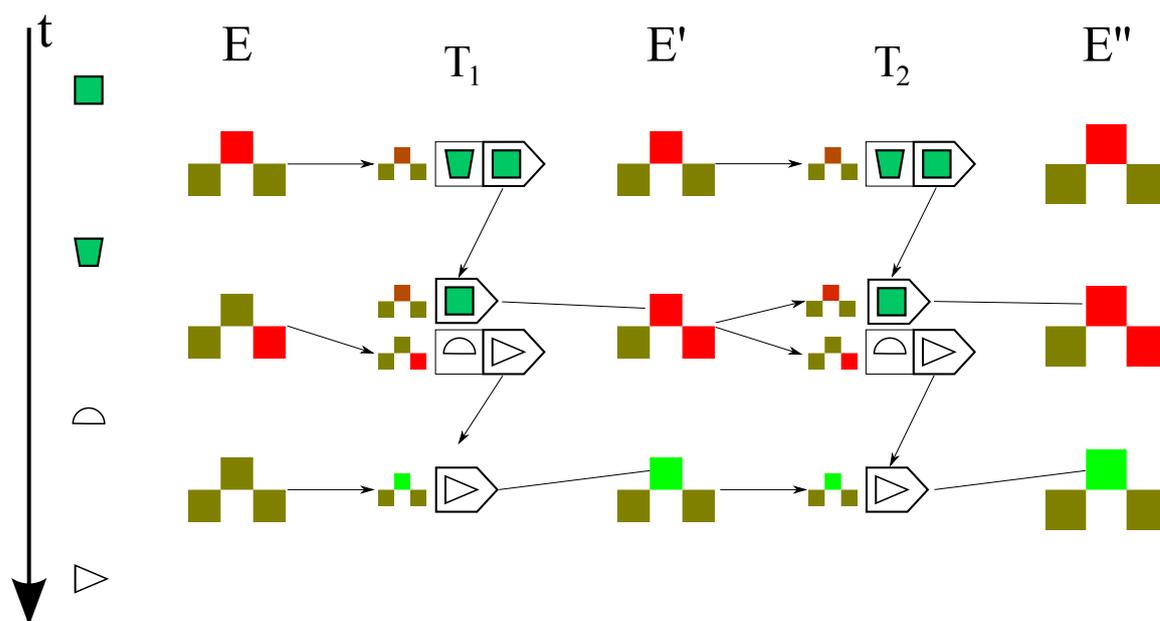


FIGURE 9.13 – Suivi des objets par la mémoire spatio-séquentielle dans le contexte spatial *virage à gauche*. Nous pouvons constater que l'interaction composite [*tourner à gauche face à un espace vide*][*avancer*] est associée à la présence d'un mur à droite de l'agent, ce qui conduit l'agent à énoncer cette interaction à partir du cycle de décision 3.

Le contexte obtenu permet à l'agent de savoir que les interactions [*avancer*] et [*tourner à droite face à un espace vide*][*avancer*] ne sont pas possibles. En revanche, comme nous l'avons vu précédemment, l'agent a attribué l'interaction [*tourner à gauche face à un espace vide*][*avancer*] à la présence d'un mur à droite. Le contexte obtenu permet de valider la faisabilité de cette interaction, qui est énoncée au pas suivant. On peut remarquer que l'objet qui afforde l'interaction *avancer* est intégré dans les contextes augmentés au pas 3. On assiste ici à une association entre deux interactions considérées comme affordées par deux éléments distincts. Cette observation montre que les objets peuvent être définis par leur contexte : ici, l'interaction *tourner à gauche face à un espace vide* est associée à un mur à droite, ce qui est vrai tant que l'agent n'utilise cette interaction que dans les contextes spatiaux *virage à gauche* et *virage à droite*. Les interactions composites permettent ainsi de lier différentes façons d'expérimenter un objet.

On pourrait penser que ce lien entre des contextes définis par des éléments distincts puisse poser problème pour définir les mouvements de l'agent. En effet, dans cet exemple, une rotation à gauche de l'agent permet à un élément situé initialement à droite de se retrouver devant.

Cependant, il faut se rappeler que ces transformations ne sont pas des déplacements, mais des liens entre des configurations d'éléments, parfois incomplètes : ici, le contexte géré par la mémoire spatiale est constitué d'un élément *mur* à droite et d'un élément *espace vide* à gauche, non intégré par la mémoire spatiale. Après rotation à gauche de l'agent, le contexte est constitué d'un élément *espace vide* devant et d'un élément *mur*, non intégré par la mémoire spatiale. Le déplacement du premier contexte vers le second est donc correct, même si ces contextes ne sont que partiellement intégrés par l'agent.

Lorsqu'on ajoute le contexte spatial *voie sans issue*, par ajout d'un bloc mur, certaines signatures contenant des erreurs sont corrigées, notamment les interactions *tourner à gauche face à un mur*, et [*tourner à gauche face à un mur*] [*avancer*], qui ne sont plus valides, et voient leurs signatures modifiées pour tenir compte d'un objet situé à gauche de l'agent. Cette modification est visible dans la Figure 9.14 : nous pouvons observer qu'un groupe d'interactions s'est spécialisé dans les entrées du contexte interactionnel liées à la détection d'un objet à gauche de l'agent (notamment les entrées liées aux interactions *toucher un espace vide à gauche* et *toucher un mur à gauche*). L'agent a ainsi appris à intégrer les objets situés à sa gauche.

Il faut un peu plus de 1000 cycles de décision supplémentaires pour obtenir un comportement stable dans chaque configuration. Ce grand nombre de cycles s'explique par le fait que les dernières séquences apprises par l'agent doivent être testées dans le contexte spatial *voie sans issue*, contexte que l'agent ne rencontre qu'une fois chaque fois qu'il fait le tour de l'environnement. Les séquences utilisées par l'agent sont affichées dans la Figure 9.10. On observe notamment, dans le contexte spatial *virage à gauche*, que l'agent utilise une interaction épistémique supplémentaire permettant de différencier les contextes spatiaux *virage à gauche* et *voie sans issue*.

On peut noter que si l'on définit, pour les interactions primitives *tourner face à un mur*, une valeur de satisfaction inférieure à -4, l'agent reste bloqué dans le contexte spatial *voie sans issue*. En effet, après avoir construit le contexte interactionnel augmenté, dans lequel aucune des trois interactions composites habituellement utilisées ne peuvent être énoncées, l'agent doit choisir, parmi les interactions considérées comme possible, celle qui a la plus grande valeur de satisfaction. Parmi elles, on compte de nombreuses interactions composites, constituées d'interactions *toucher un mur* dont la valeur de satisfaction est de -2 ou -4. La stabilisation rapide du comportement de l'agent ne permet pas un apprentissage efficace des signatures de ces interactions. Certaines de ces signatures contiennent des erreurs, ne permettant pas à l'agent d'observer que leur énonciation ne produit pas de mouvement. Ces interactions sont ainsi sélectionnées, puisque d'une valeur de satisfaction supérieure à celle des interactions *tourner face à un mur*. Ce comportement ne caractérise pas un défaut du mécanisme : l'agent sélectionne, dans le contexte qu'il expérimente, et d'après ce qu'il a appris de ses possibilités d'interactions, les interactions composites qu'il considère comme les plus satisfaisantes. Cependant, il illustre le besoin pour l'agent d'apprendre à utiliser ses interactions, et à définir les conséquences de leur énonciation avant de les exploiter.

Le suivi du contenu de la mémoire spatio-séquentielle, illustré par la Figure 9.15, permet de comprendre les décisions du mécanisme de sélection. Il met également en évidence le suivi des

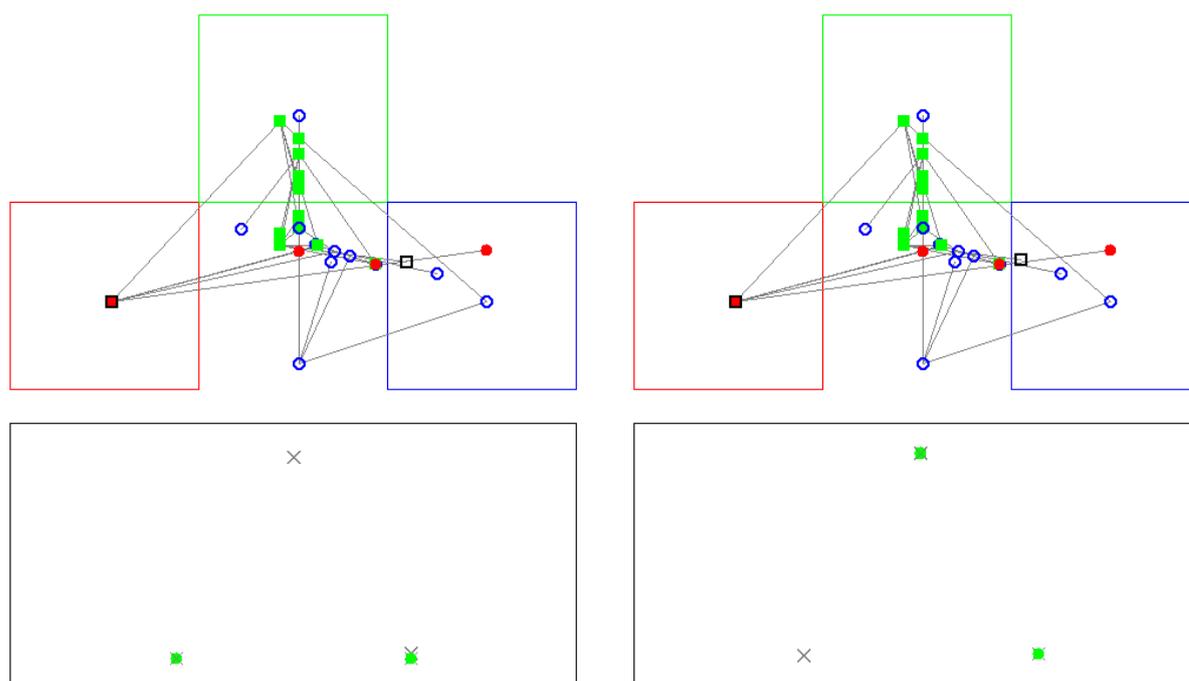


FIGURE 9.14 – Spécialisation des interactions en fonction des positions et des objets, dans les configurations *simple loop* (à gauche) et *double loop* (à droite), après avoir ajouté un bloc mur. Les différences entre les spécialisations dans chaque configuration sont minimales et n’influent pas sur les comportements de l’agent. Nous pouvons remarquer que dans les deux configurations, une interaction s’est spécialisée pour le côté gauche de l’agent. Les interactions catégorisent toujours trois types d’objets indépendamment des éléments de l’environnement.

objets lorsque l’agent effectue des rotations, notamment au cycle 2 où la mémoire conserve la présence du mur devant l’agent sous la forme de l’interaction *toucher un mur devant*, et au cycle 6, où la présence de l’espace vide est conservée, ce qui permet à l’agent d’avancer sans qu’une interaction épistémique ne soit nécessaire. Il faut toutefois noter la présence d’erreurs dans les signatures des interactions *turner*, qui conduisent à la construction de contextes interactionnels ne reflétant pas l’environnement de l’agent, comme nous pouvons l’observer au cycle 5. Ce contexte caractérise la présence d’un espace vide à droite et à gauche de l’agent, ce qui est faux pour ce dernier élément. Cependant, dans le contexte interactionnel obtenu, l’agent aura tendance à tourner à droite. Ce comportement rend non seulement cette incohérence non pertinente dans la sélection des interactions, mais ne permet pas la correction de cette incohérence puisque l’agent ne pourra jamais la mettre en évidence.

On effectue ensuite les tests avec des interactions composites de longueur 3. L’agent atteint toujours un comportement stable rapidement (moins de 500 pas). Cependant, cette rapidité l’enferme très vite dans un comportement qui n’évolue plus, ce qui limite considérablement l’apprentissage d’interactions composites de longueur 3. On constate en effet que seules 18 interactions

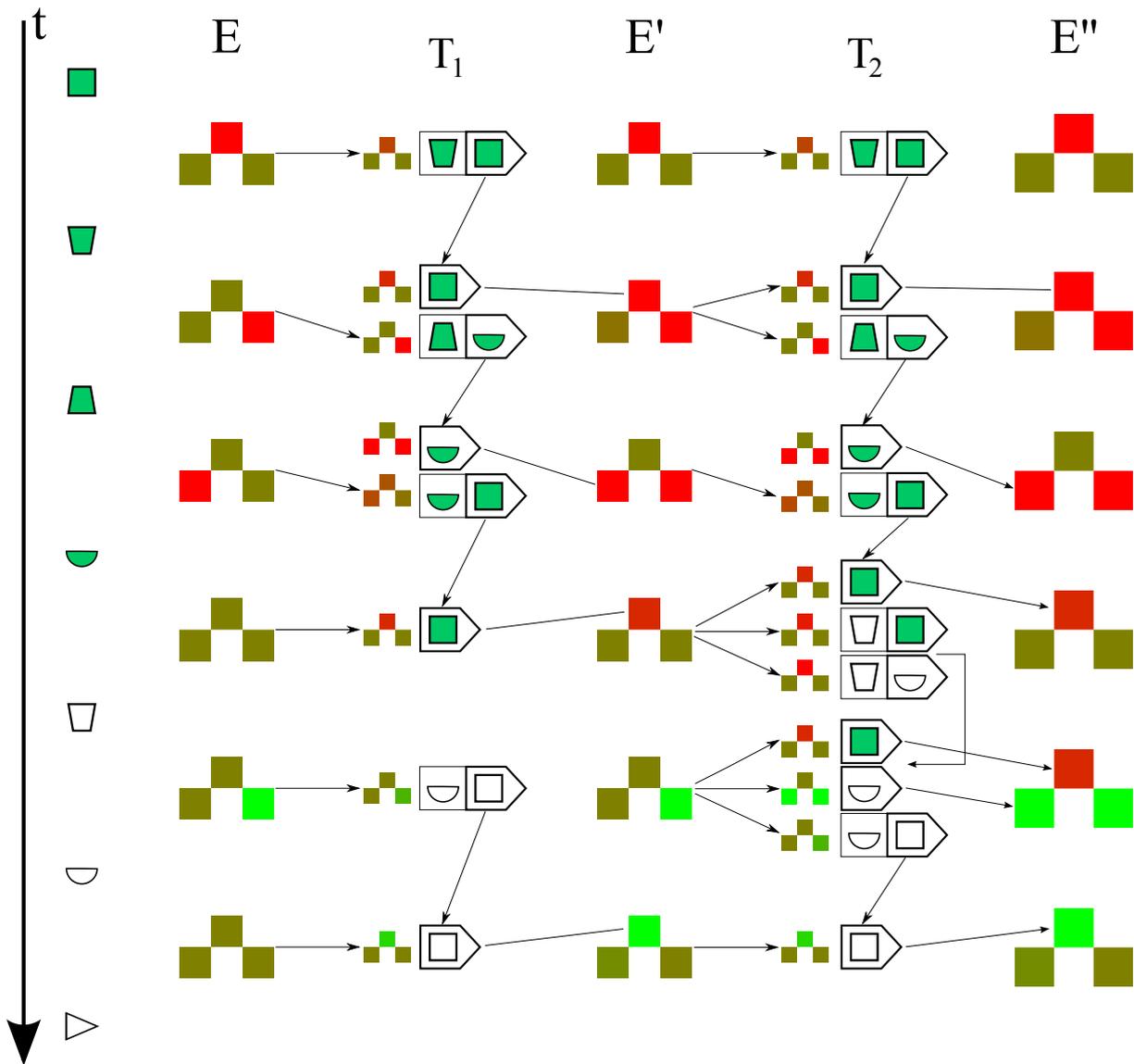


FIGURE 9.15 – Suivi des objets par la mémoire spatio-séquentielle dans le contexte spatial *voie sans issue*. Nous pouvons constater la présence d'erreurs dans les signatures des interactions *tourner*, ce qui génère des contextes parfois incorrects (cinquième cycle de décision). Ces erreurs ne perturbent cependant pas le comportement de l'agent, car ces contextes restent corrects d'après l'utilisation qu'en fait l'agent : au cycle 5, le contexte considère les interactions *toucher un espace vide à gauche* et *toucher un espace vide à droite* comme énoncées, traduisant la présence d'un espace vide à droite et à gauche de l'agent, ce qui est faux pour le côté gauche. Or, dans ce contexte spatial, l'agent aura tendance à tourner à droite. L'objet à sa gauche n'est donc pas pertinent dans la sélection de l'interaction du pas suivant. De ce fait, la mémoire spatiale ne permettra pas un suivi correct des objets.

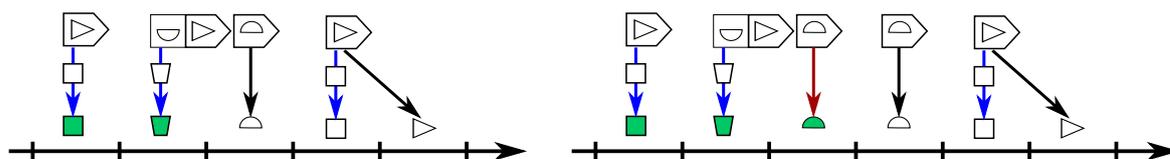


FIGURE 9.16 – Séquences d’interactions utilisées par l’agent lorsqu’il peut apprendre des interactions composites de longueur 3, dans les contextes spatiaux *virage à gauche* (à gauche) et *voie sans issue* (à droite). Les séquences utilisées dans les deux autres contextes sont identiques que celles précédemment vues précédemment. On remarque que l’agent commet systématiquement une erreur (flèche rouge) dans le contexte *voie sans issue*.

composites sur les 73 découvertes par l’agent après ajout du bloc mur sont de longueur 3, et aucune n’est associée à une signature reflétant l’objet qui les afforde, ce qui les rend inutilisables pour le mécanisme de suivi de la mémoire spatio-séquentielle.

On constate qu’avant d’ajouter le contexte spatial *voie sans issue*, le comportement de l’agent est identique à celui observé avec des interactions composites de longueur 2. Cependant, après avoir ajouté ce contexte, le comportement diffère dans les contextes *tourner à gauche* et à cause d’une erreur dans les signatures utilisées pour construire les contextes. Dans ces contextes, l’agent, après avoir éliminé la possibilité d’enacter les interactions composites $[avancer] [tourner à droite face à un espace vide] [avancer]$, utilise l’interaction *tourner à droite face à un espace vide*. L’agent commet ainsi systématiquement une erreur dans le contexte spatial *voie sans issue*, que son comportement ne permet pas de corriger (Figure 9.16).

On observe également qu’après avoir ajouté le contexte spatial *voie sans issue*, aucune interaction composite ne se dégage pour les objets situés à gauche de l’agent (Figure 9.17). Ce défaut de spécialisation est à l’origine de l’erreur observée dans le comportement de l’agent dans contexte spatial *voie sans issue* : la signature de l’interaction *tourner à gauche face à un espace vide* caractérise la présence d’un objet à droite de l’agent. Cependant, l’agent ne peut pas déterminer l’origine de cette erreur et la corriger.

Ces observations nous montrent que l’agent a besoin de tester et vérifier ses signatures d’interactions pour pouvoir les rendre utilisables par le mécanisme de la mémoire spatio-séquentielle. En l’absence de ce mécanisme d’apprentissage, les signatures se basent sur des coïncidences, qui peuvent enfermer l’agent dans un ensemble de comportements où les erreurs ne peuvent plus être détectées. Les erreurs des signatures peuvent ne pas influencer le système de décision, mais leur accumulation rend impossible l’évolution vers des mécanismes de décision plus complexes. Nous proposons donc un mécanisme d’apprentissage pour permettre à l’agent de tester les signatures de ses interactions avant de les exploiter.

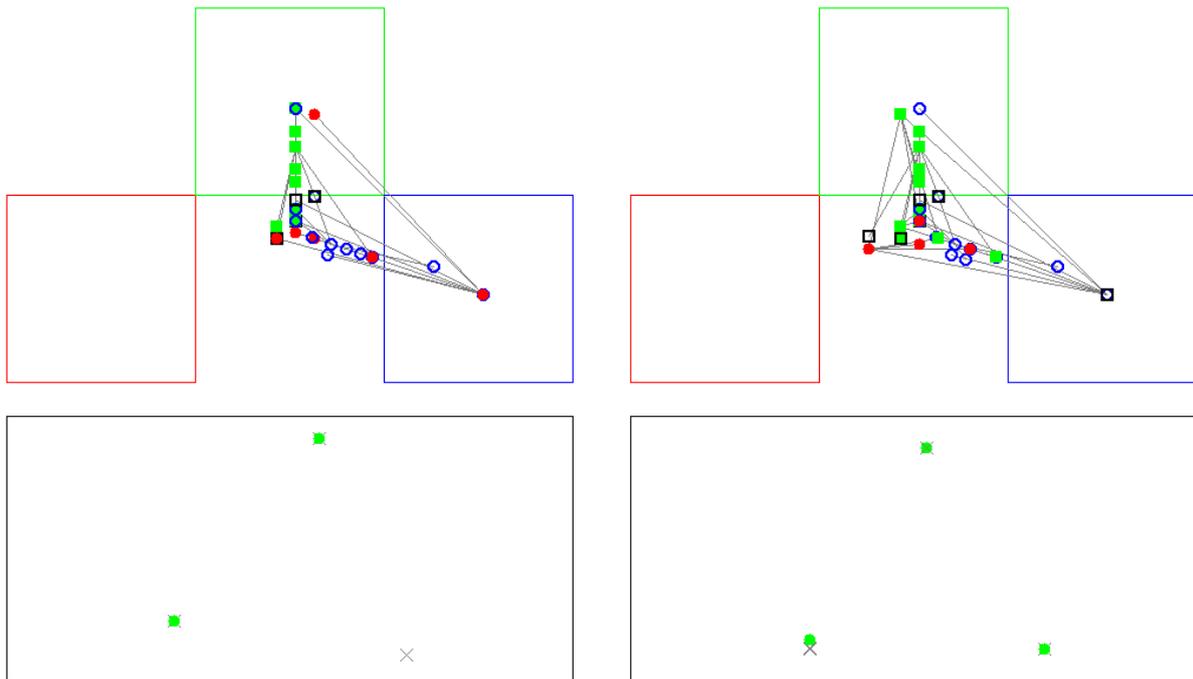


FIGURE 9.17 – Spécialisation des interactions en fonction des positions et des éléments de l'environnement dans le cas d'une utilisation d'interactions composites de longueur 3, avant (à gauche) et après (à droite) ajout du contexte spatial *voie sans issue* dans la configuration *small loop*. On constate que très peu de signatures sont correctement définies. De plus, après avoir ajouté le contexte spatial *voie sans issue*, l'agent n'apprend pas à caractériser son coté gauche, ce qui est à l'origine de l'erreur observée dans les séquences d'interaction.

9.2.3 Étude du comportement avec un mécanisme d'apprentissage

Nous testons à présent un système utilisant un mécanisme d'apprentissage destiné à tester et valider les signatures des interactions composites, de façon à dépasser les limites observées précédemment. Comme précédemment, nous testons dans un premier temps le mécanisme avec des interactions composites dont la longueur est limitée à 2, puis à 3.

Interactions composites limitées à une longueur de 2

Comme pour les essais précédents, nous laissons l'agent évoluer dans son environnement jusqu'à ce que le comportement se stabilise. On observe que la stabilisation du comportement nécessite bien plus de temps : entre 3400 et 3800 cycles de décision environ sont nécessaires, selon les configurations de l'environnement. Le comportement obtenu est moins efficace que le mécanisme dénué de système d'apprentissage avant ajout du contexte spatial *voie sans issue*, mais est plus cohérent avec le contexte de l'agent. En effet, on peut constater, en observant les signatures, qu'un grand nombre d'entre elles sont correctes (cf Table 9.2). L'agent doit donc obtenir, dans le cas d'un virage à gauche, une information sur l'objet à sa gauche avant de tourner.

La séquence contient une interaction *toucher à gauche un espace vide* en plus par rapport aux séquences obtenues précédemment.

TABLE 9.2 – nombre d’interactions composites découvertes par l’agent dans chaque configuration, le nombre d’interactions considérées comme fiables (en incluant les interactions toujours vraies), les interactions dé-corrélées (considérées comme associées à un objet que l’agent ne peut pas expérimenter), le nombre total d’interaction dont l’apprentissage est considéré comme terminé, et nombre d’interactions pour lesquelles la signature est correcte du point de vue d’un observateur extérieur.

	nb interactions	fiables	dé-corrélées	total	correctes
Simple Loop	105	71 (67.6%)	14 (13.3%)	85 (80.9%)	103 (96.2%)
Double Loop	107	49 (45.8%)	16 (15%)	65 (60.8%)	100 (93.5%)
Reverse SL	107	77 (72%)	13 (12.1%)	90 (84.1%)	101 (94.4%)
Reverse DL	105	68 (44.2%)	14 (23.1%)	82 (76.6%)	104 (97.2%)

Les séquences utilisées dans chaque contexte spatial sont affichées dans la Figure 9.18. Nous pouvons voir que dans les contextes *virage à droite* et *virage à gauche*, deux comportements sont observés en fonction des configurations. Le premier consiste à vérifier la faisabilité de l’interaction composite [*tourner à gauche face à un espace vide*][*avancer*), puis, si elle n’est pas énable, à vérifier la faisabilité de l’interaction [*tourner à droite face à un mur*][*avancer*). Le second comportement inverse l’ordre de ces séquences. Cette différence s’explique par l’ordre dans lequel l’agent a appris les interactions composites. En effet, ces deux interactions disposent de la même valeur de satisfaction. Comme le contexte interactionnel ne donne aucune indication sur la faisabilité de ces interactions, le système de décision sélectionnera celle que l’agent a découvert en premier. Cette individuation dans les comportements montrent que l’ordre dans lequel l’agent apprend les interactions composites qu’il peut utiliser a une influence non négligeable sur le comportement. Un troisième comportement est observé dans le contexte spatial *virage à gauche* après apprentissage dans la configuration *reverse small loop*. Dans cette configuration, une erreur persiste dans la signature de l’interaction composite [*tourner à gauche face à un espace vide*][*avancer*) : cette interaction est en effet associée à la présence soit d’un espace vide à gauche, soit d’un mur devant l’agent. Ainsi, après avoir éliminé la possibilité d’énacter l’interaction [*tourner à droite*][*avancer*), le contexte caractérise la présence d’un mur devant l’agent, indiquant que l’interaction [*tourner à gauche*][*avancer*) est énable. L’agent énable alors cette interaction sans faire appel à une interaction épistémique.

La présence d’erreurs malgré l’utilisation d’un mécanisme d’apprentissage montre que l’agent peut se baser sur des coïncidences pour définir les contextes propices à l’énaction d’une interaction. Dans cet exemple, une fois le comportement stabilisé, l’agent n’utilise l’interaction [*tourner à gauche face à un espace vide*][*avancer*) que dans le contexte spatial *virage à gauche*. Or, ce contexte est caractérisée par la présence d’un mur face à l’agent. L’erreur persiste car elle n’est jamais mise en évidence. La mémoire spatiale tient ainsi compte de l’aspect séquentiel du

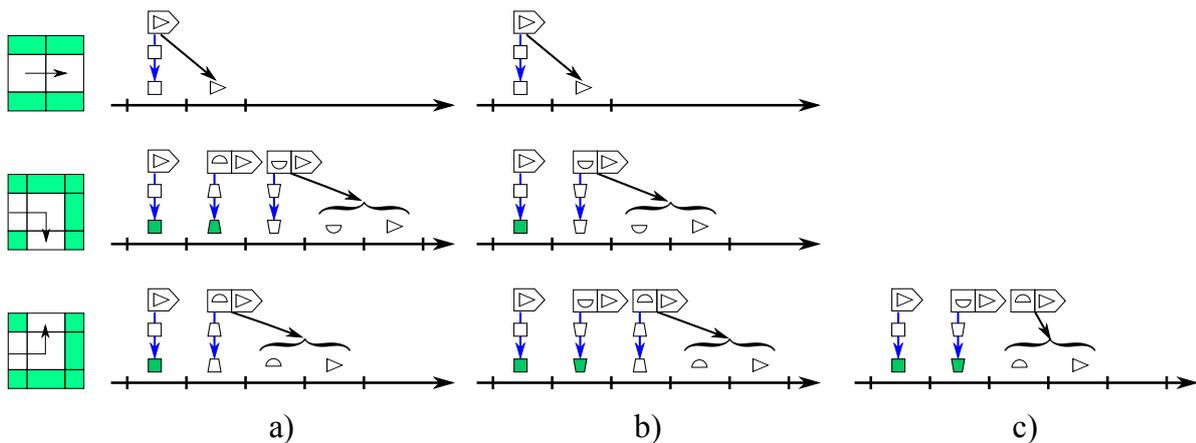


FIGURE 9.18 – Séquences d’interactions observées dans chaque contexte spatial. a) les séquences observées dans les configurations *small loop* et *double loop*. b) les séquences observées dans la configuration *reverse double loop*. Dans la configuration *reverse simple loop*, l’agent utilise les séquences b) dans les contextes *ligne droite* et *virage à gauche*. Dans le contexte *virage à droite*, il utilise la séquence c). En effet, une erreur dans la signature de l’interaction [*tourner à gauche face à un espace vide*][*avancer*) permet de considérer cette interaction comme énonçable au cycle de décision 2, permettant de se passer d’une interaction épistémique. La différence entre les séquences a) et b) est due à l’ordre dans lequel l’agent cherche à énoncer ses interactions, lié à l’ordre dans lequel il les a découvertes.

comportement de l’agent.

En observant la spécialisation des signatures d’interactions (Figure 9.19), on remarque qu’une grande majorité d’entre elles se sont spécialisées pour une position et un type d’élément précis, avec un faible taux d’erreurs : dans le premier graphe, les interactions forment trois groupes distincts correspondant aux trois positions que l’agent peut expérimenter par le biais de ses interactions. Notons que les interactions qui ne font pas partie de ces trois groupes sont pour la plupart des interactions qui ne sont pas considérées comme fiables par l’agent. Dans le second, deux groupes très compacts ont émergé. Chaque groupe ne contient que des interactions liées à un même type d’élément. L’agent a ainsi intégré trois positions observables dans son environnement, ainsi que deux objets. Les signatures permettent de générer un ensemble de *bundles* implicite des objets : si un objet est présent dans l’environnement, l’ensemble des interactions dont la signature correspond à cet objet seront reconnues et intégrées par la mémoire spatiale, donnant simultanément l’ensemble des façons de l’atteindre et d’interagir avec lui, sans qu’il ne soit nécessaire d’utiliser une modélisation des objets.

Le suivi de la mémoire spatiale dans le contexte spatial *tourner à gauche* (dans la configuration *simple loop*) nous permet d’observer la permanence de l’objet qui émerge de la mise à jour des interactions composites : chaque contexte interactionnel permet la détection d’un ensemble d’interactions, et peut être complété par les interactions composites mises à jour des mémoires

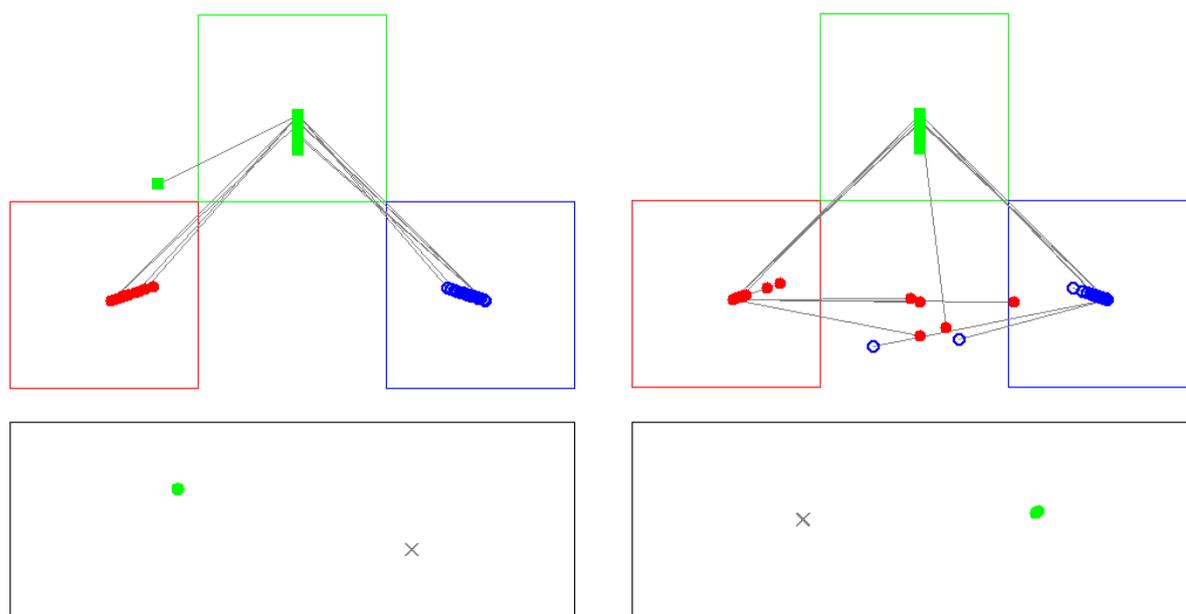


FIGURE 9.19 – Spécialisation des signatures dans les configurations *simple loop* (à gauche) et *double loop* (à droite). Les couleurs sont les mêmes que dans les graphes précédents. Les liens gris connectent les interactions composites avec les interactions primitives constituant leur interaction finale. Ces liens caractérisent le "mouvement" de l'objet caractérisé par une interaction finale quand le path est énéacté. Nous pouvons constater que la majorité des interactions se sont spécialisées dans une position et un objet particulier. Notons que dans ces deux configurations, aucune des interactions en dehors des trois positions détectables n'est considérée comme fiable.

\mathcal{T} et \mathcal{F} du contexte interactionnel précédent. Comme le path des interactions composites a une longueur maximale de 1, chaque interaction peut subir au plus une mise à jour. Chaque contexte peut ainsi recevoir au plus une interaction supposée énéactée d'après la mémoire du contexte interactionnel précédent.

Nous pouvons voir le suivi des objets dans la Figure 9.20 : au cycle de décision 4, l'agent effectue une rotation à droite. Cette interaction ne fournit aucune information au contexte interactionnel (E). Cependant, le contexte augmenté E''' caractérise la présence d'un espace vide devant l'agent, initialement à sa droite, et d'un mur à gauche de l'agent, initialement devant lui. Cette rotation dans la mémoire est implicite : en effet, les objets ont été suivis grâce à la mise à jour du path de certaines interactions composites qu'ils affordent.

On ajoute ensuite le bloc mur pour ajouter le contexte spatial *voie sans issue*. Le temps d'adaptation à ce contexte est très variable, bien que relativement court : il faut compter environ 80 à 370 cycles de décision, en fonction des configurations, pour obtenir à nouveau un comportement stable. Les séquences utilisées dans les contextes spatiaux connus sont identiques à celles observées précédemment (Figure 9.18), exception faite de la configuration *reverse simple loop*,

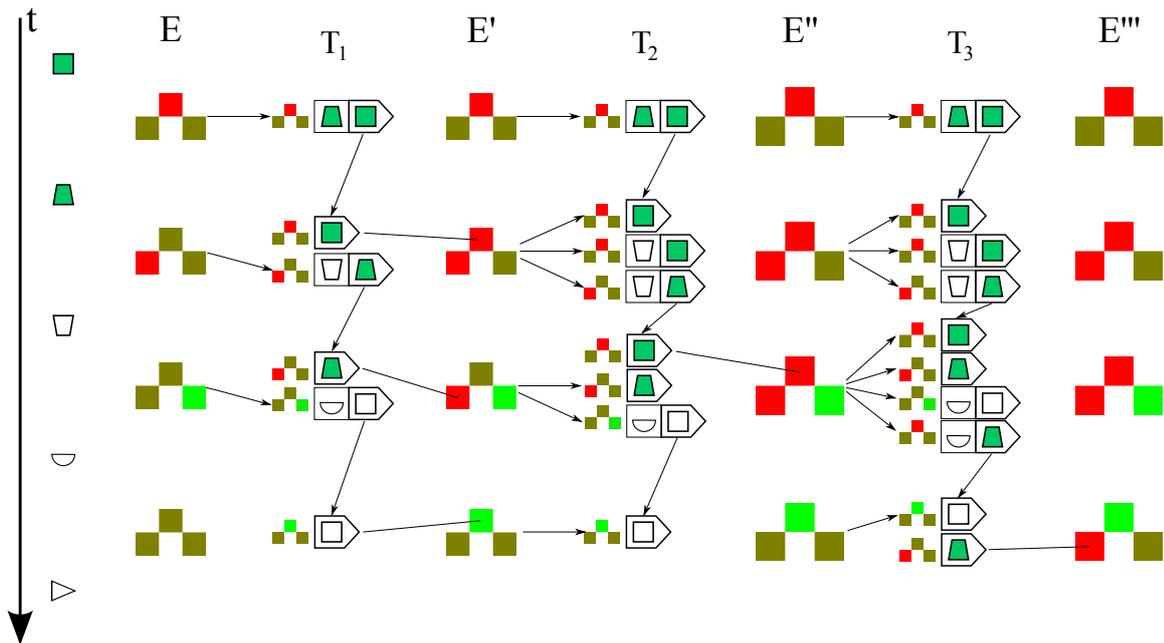


FIGURE 9.20 – Suivi de la mémoire spatio-séquentielle dans le contexte spatial *virage à droite* (configuration *simple loop*). Chaque contexte interactionnel permet l'ajout d'une information supplémentaire. L'utilisation de trois contextes interactionnels augmentés consécutifs permet à l'agent de connaître, au cycle de décision 3, le contenu de son environnement sur les trois positions qu'il peut expérimenter.

où l'erreur observée a été corrigée. L'agent utilise ainsi, dans le contexte spatial *virage à droite*, la séquence d'interaction b affichée dans la figure 9.18.

Dans le contexte spatial *voie sans issue*, on observe également des différences dans les séquences utilisées : dans les configurations *small loop* et *double loop*, l'agent utilise la séquence a) donné en Figure 9.21, et dans les configurations *reverse simple loop* et *reverse double loop*, l'agent utilise la séquence b). Comme précédemment, cette différence provient de l'ordre dans lequel l'agent a découvert ses interactions composites. On peut noter qu'une troisième séquence peut être observée si on limite la mémoire spatio-séquentielle à deux contextes augmentés (E' et E''). En effet, on observe, dans la Figure 9.22, que deux contextes interactionnels ne sont pas suffisants pour suivre les objets qui entourent l'agent tout au long de la séquence utilisée dans le contexte spatial *voie sans issue* : au cycle 4, le contexte interactionnel caractérise la présence d'un mur devant l'agent, indiquant que l'interaction [avancer] ne peut pas être énoncée, mais pas la présence du mur situé désormais à gauche de l'agent. Si celui-ci sélectionne en premier l'interaction [tourner à gauche face à un espace vide][avancer], il va utiliser une interaction épistémique pour en tester la faisabilité. La séquence c) comporte ainsi une interaction épistémique supplémentaire au cycle 5.

Les signatures d'interactions n'évoluent que très peu, voire pas du tout dans la configuration *simple loop*. Les graphes de spécialisation des interactions sont ainsi très similaires à ceux de

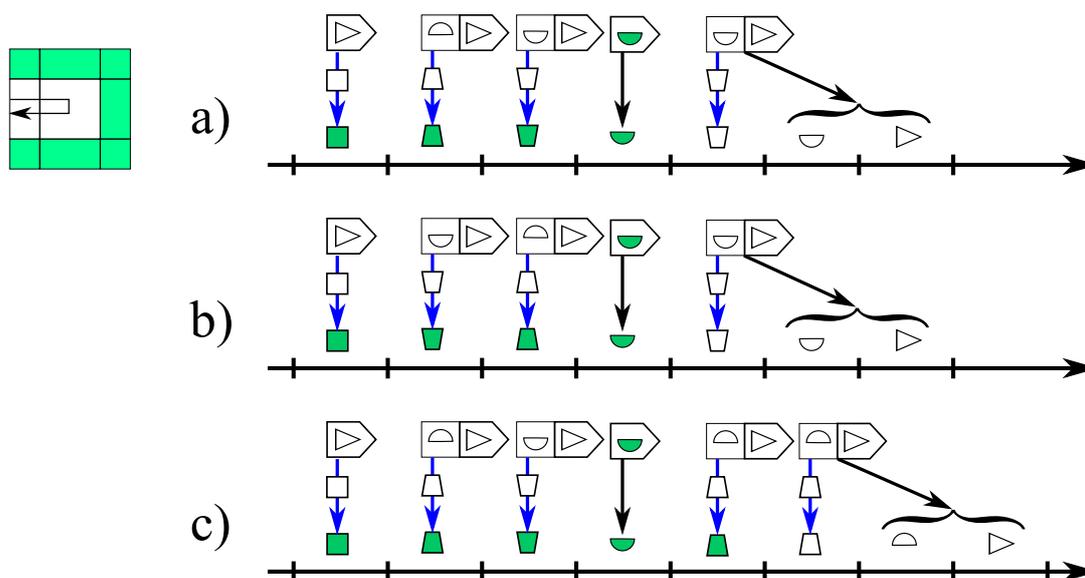


FIGURE 9.21 – Séquences utilisées dans le contexte spatial *voie sans issue*. a) séquence obtenue dans les configurations *small loop* et *double loop*. b) séquence observée dans les configurations *reverse small loop* et *reverse double loop*. La séquence utilisée dépend de l'environnement initial, qui affecte l'ordre dans la découverte des interactions composites. La séquence c) est observée si on supprime le contexte E'' : le contexte E'' ne permet pas de suivre l'ensemble des objets détectés, d'où l'utilisation d'une interaction épistémique supplémentaire pour tester la seconde énonciation de $[tourner \ à \ gauche \ face \ à \ un \ espace \ vide][avance]$.

la Figure 9.19. Ceci montre que les signatures étaient déjà majoritairement "correctes", c'est-à-dire associées aux interactions qui permettent effectivement de détecter l'objet qui afforde leurs interactions, avant l'ajout du contexte spatial *voie sans issue*. La Table 9.2 montre en effet que dans toutes les configurations, le nombre d'interactions dont la signature est correcte est supérieur à 90 %.

La Figure 9.22 montre l'évolution du contenu de la mémoire spatio-séquentielle dans le cas du contexte spatial *voie sans issue* après apprentissage dans la configuration *small loop*. Comme précédemment, chaque contexte interactionnel ajoute des informations par rapport au précédent, permettant à l'agent de détecter, au cycle de décision 3, l'ensemble des éléments qui l'entourent. Dans ce contexte interactionnel, aucune des trois interactions composites utilisées habituellement n'est énonçable. L'agent choisit alors l'interaction dont la valeur de satisfaction est la plus élevée (parmi celles produisant un mouvement, c'est-à-dire que la signature ne contient pas l'interaction elle-même). L'agent choisit l'interaction $[tourner \ à \ droite \ face \ à \ un \ mur]$. Notons que ce choix est lié à l'ordre dans lequel les interactions primitives ont été initialement définies par le concepteur de l'agent. En effet, en cas d'égalité entre plusieurs interactions, l'agent sélectionne la première dans la liste. Cette interaction permet de changer de contexte, pour arriver dans un contexte similaire à celui caractérisant le contexte spatial *virage à droite*. La fin de la séquence est identique

au comportement observé dans ce contexte. On notera, au cycle 4, la différence entre les contextes E'' et E''' : le contexte E'' indique seulement que l'interaction *toucher un mur devant* aurait pu être énoncé, caractérisant la présence d'un mur devant l'agent, mais ne donne, contrairement au contexte E''' , aucune indication sur l'élément à gauche de l'agent. Le contexte E''' , qui caractérise la présence d'un mur à gauche, indique l'impossibilité d'énoncer [*tourner à gauche face à un espace vide*][*avancer*], et évite ainsi, si cette interaction est habituellement celle choisie en premier par l'agent, le recours à une interaction épistémique.

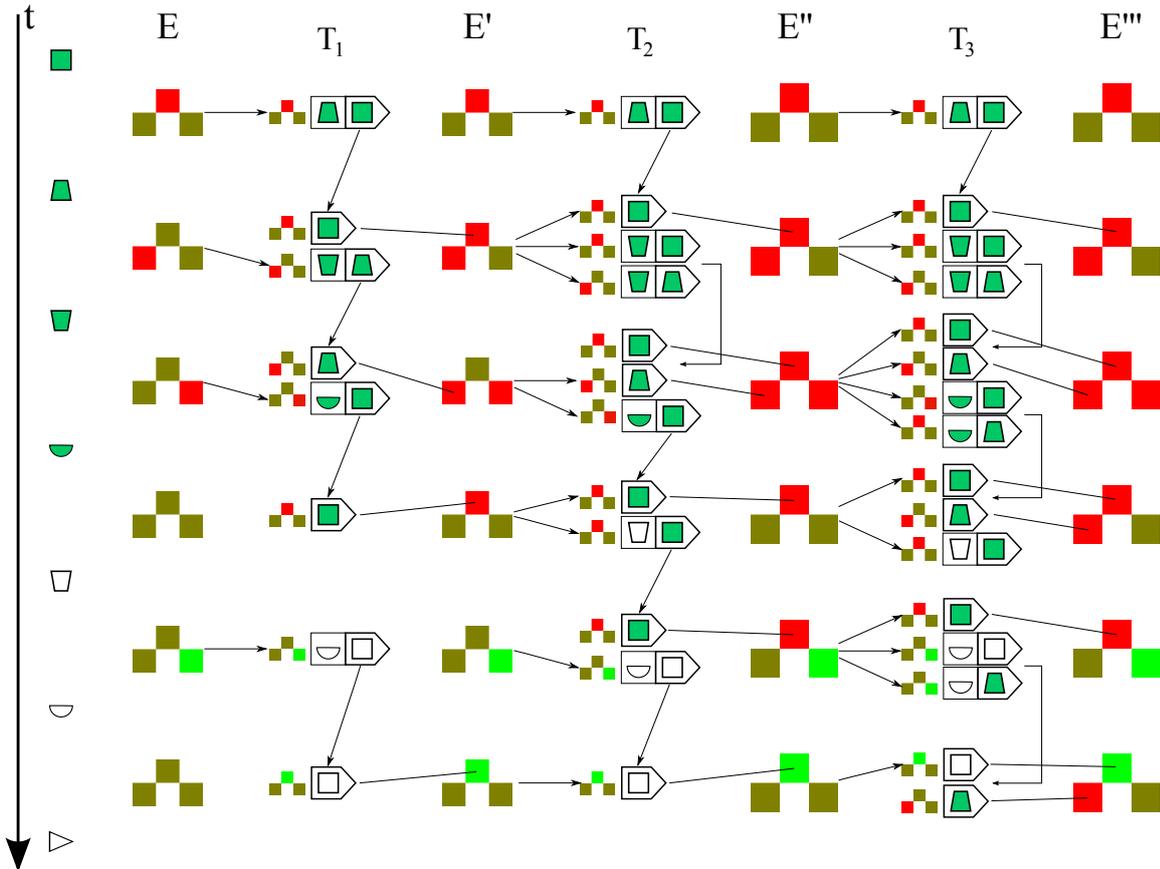


FIGURE 9.22 – Contenu de la mémoire spatio-séquentielle dans le cas du contexte spatial *voie sans issue* dans la configuration *small loop*. Nous pouvons observer l'effet de permanence des objets qui permet de compléter les contextes interactionnels successifs. Le cycle de décision 4 explique la différence entre les séquences a) et c) de la Figure 9.21 : si l'agent se base sur E'' , il sélectionnera, parmi les interactions composites lui permettant de s'éloigner du mur devant lui, la première qu'il aura apprise. Dans le cas où il utilise le contexte E''' , l'interaction commençant par *tourner à gauche face à un espace vide* est définie comme impossible. L'agent choisira ainsi l'interaction [*tourner à droite face à un espace vide*][*avancer*].

Nous avons vu que la mémoire spatiale permet de conserver et suivre des objets lorsque l'agent énonce des interactions. Le suivi permet de compléter le contexte interactionnel de l'agent et ainsi lui permet de caractériser sa situation. Cependant, ce suivi ne concerne que le déplacement

des objets (par rapport à l'agent) dans l'espace observable de l'agent. Dans la section suivante, nous étudierons comment la mémoire peut intégrer des éléments qui échappent au système interactionnel de l'agent en utilisant des interactions composites plus longues.

Interactions composites limitées à une longueur de 3

On teste ensuite le mécanisme avec des interactions composites de longueur 3. L'apprentissage est beaucoup plus long : entre 30 000 et 38 000 pas sont nécessaires pour obtenir une stabilisation du comportement. Cette durée est liée au grand nombre de séquences d'interactions découvertes par l'agent (voir Table 9.3), mais également à la difficulté de tester certaines interactions composites, qui ne peuvent être énoncées qu'à certains endroits précis de l'environnement.

TABLE 9.3 – Nombre d'interactions composites découvertes par l'agent dans chaque configuration environnementale, avec des interactions composites limitées à une longueur de 3.

	nb interactions	fiabes	dé-corrélées	total	correctes
Simple Loop	654	304 (46.5%)	163 (25%)	467 (73.5%)	409 (62.5%)
Double Loop	584	295 (50.5%)	146 (25%)	441 (75.5%)	441 (67.1%)
Reverse SL	563	230 (40.9%)	125 (22.2%)	355 (63.1%)	355 (63.2%)
Reverse DL	611	270 (44.2%)	141 (23.1%)	411 (67.3%)	411 (64%)

Nous pouvons remarquer que le nombre d'interactions fiables ou décorrélées est supérieur au nombre d'interactions correctes, ce qui implique qu'un certain nombre d'interactions considérées comme fiables comportent des erreurs dans leurs signatures.

Une fois le comportement stabilisé, on observe différents comportements, qui dépendent de la configuration utilisée. La plupart des séquences utilisées dans les différents contextes spatiaux sont identiques à celles utilisées par l'agent lorsque les interactions composites sont limitées à une longueur de 2. Comme précédemment, deux séquences sont observées (Figure 9.23), selon la configuration. Cette individuation est liée à l'ordre dans lequel l'agent découvre les interactions composites. Cependant, dans la configuration *double loop*, on observe deux comportements différents dans le contexte spatial *virage à droite*. La séquence d'interaction utilisée dépend du contexte précédente. En effet, des erreurs persistent dans la mémoire spatio-séquentielle après avoir énoncé la séquence utilisée dans le contexte spatial *virage à gauche*. De ce fait, si l'agent se retrouve dans un contexte *virage à droite* qui suit un contexte *virage à gauche*, la séquence utilisée ne sera pas la même que si elle suivait un contexte *ligne droite*.

On peut donc conclure que si des erreurs dans les signatures sont toujours présentes une fois le comportement stabilisé, l'agent basera la sélection de ses interactions sur ces erreurs, qui ne sont pas considérées comme telles, car n'étant pas préjudiciable au système de décision. De telles erreurs peuvent apparaître lorsque l'interaction n'est énoncée que dans des contextes précis, soit parce que le comportement se stabilise, soit parce que le path d'une interaction implique un contexte de départ spécifique et unique ne permettant pas d'isoler l'objet pré-énoncé de l'interaction.

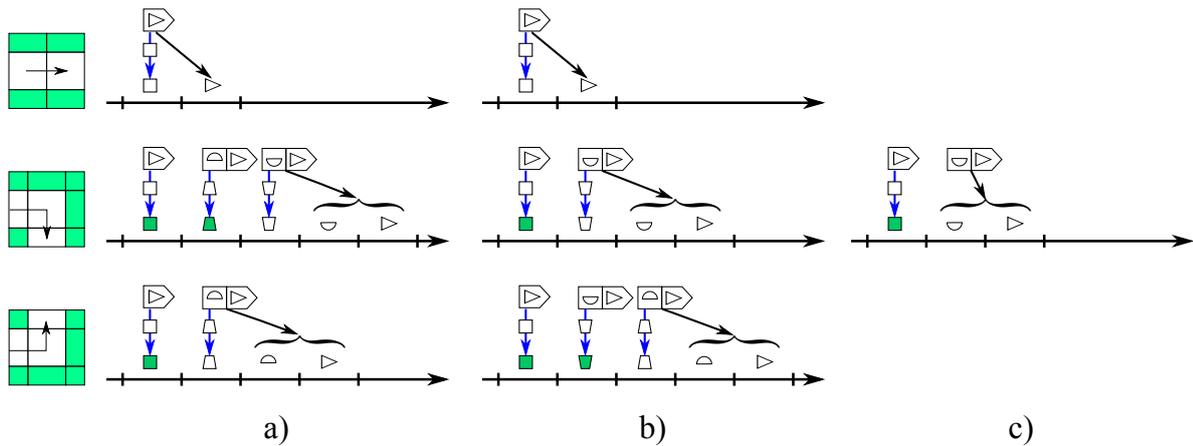


FIGURE 9.23 – Séquences d’interactions obtenues dans les différents contextes spatiaux. a) séquences observées dans la configuration *simple loop*. b) séquences observées dans les configurations *reverse simple loop* et *reverse double loop*. Cette différence vient de l’ordre d’apprentissage des interactions composites. Le comportement obtenu dans la configuration *double loop* est particulier : l’agent utilise les séquences a) dans les contextes spatiaux *ligne droite* et *virage à gauche*. Dans les contextes *virage à droite*, il utilise la séquence a) si le virage à droite est précédé par une ligne droite, et la séquence c) si le virage à droite est précédé par un virage à gauche. En effet, après avoir énéacté la séquence dans un virage à gauche, des erreurs dans le contexte caractérisent la présence d’un espace vide à droite de l’agent. Si l’agent se retrouve dans un contexte *virage à droite*, il sélectionnera et énéactera l’interaction [tourner à droite face à un espace vide][avancer), qui d’après le contexte interactionnel, est énéactable.

Les erreurs qui apparaissent dans les contextes interactionnels augmentés permettent à l’agent de tenir compte de l’aspect séquentiel de son environnement : les erreurs dans les signatures sont en effet basées sur le fait que certaines interactions composites ne sont utilisées que dans des contextes particuliers dans lesquels ces signatures sont correctes, ou au moins sans influence sur le comportement de l’agent. Notons que ces erreurs peuvent être aussi bien bénéfiques que néfastes à l’agent. En effet, les informations apportées au contexte peuvent éviter le recours à des interactions épistémiques, comme nous pouvons l’observer dans la configuration *double loop*. En revanche, des informations erronées peuvent dissuader l’agent d’effectuer une interaction composite qui lui serait profitable. Ceci a pu être observé lorsque l’agent n’utilise pas le mécanisme d’apprentissage : c’est en effet ce dernier qui pousse l’agent à tester une interaction composite, même dans un contexte défavorable, permettant la détection de telles erreurs. Le mécanisme d’apprentissage permet ainsi de limiter les erreurs préjudiciables.

En observant l’organisation et la spécialisation des signatures d’interaction, on remarque que, comme précédemment, une large majorité des signatures d’interactions sont regroupées en trois groupes, correspondant aux trois positions que l’agent peut observer, bien qu’un nombre non négligeable d’interactions soient liées à une signature caractérisant des contextes plus complexes

tenant compte de plus d'une position (Figure 9.24). Cependant, on ne relève que très peu d'erreurs dans les interactions considérées comme fiables. On notera également la présence de liens entre les positions droites et gauches, qui étaient absents précédemment. Ces liens sont formés par les interactions composites de longueur 3 permettant de suivre un objet entre ces deux positions, en effectuant par exemple deux rotations consécutives.

Nous pouvons également observer plusieurs groupements des interactions en fonction des éléments. Dans les configurations *simple loop* et *reverse simple loop*, les interactions forment deux groupes distincts composés chacun d'interactions liées à un même type d'élément. Cependant, ces groupes ne sont pas aussi compacts que précédemment, ce qui montre que, bien que les signatures soient similaires, elles ne sont pas toutes identiques. Cette différence est accentuée dans la configuration *reverse double loop* : les interactions ont également formé deux groupes distincts. Cependant, chaque groupe est divisé en deux sous-groupes, proches mais séparés : l'agent a ainsi considéré quatre types d'objets, présentant deux à deux des propriétés similaires mais non identiques. Dans la configuration *double loop*, nous retrouvons les deux groupes correspondant aux éléments espace vide et mur, mais nous observons également un troisième groupe, composé d'interactions affodés par les deux éléments. Ce groupe caractérise la formation d'objets complexes prenant en compte plusieurs éléments, à l'origine des comportements observés dans cette configuration.

Ces observations montrent que si l'agent a bien intégré les positions qu'il peut expérimenter, et les interactions qui permettent de déplacer un objet d'une position à une autre, il a également intégré des *bundles* d'objets parfois complexes et indépendants des éléments qui composent son environnement. Ces objets montrent que l'agent caractérise son environnement par la façon dont il l'expérimente, et construit des objets comme des contextes propices à l'énaction d'une interaction. Les *bundles* formés sont ainsi partiels : un objet affodant une interaction peut évoquer partiellement le contexte qui affode une autre interaction, si le contexte défini par sa signature est plus complexe.

L'analyse du contenu de la mémoire spatio-séquentielle montre que les séquences de longueur 3 permettent de suivre les objets plus longtemps. En effet, les interactions de longueur 3 permettent de suivre les objets pendant deux cycles de décision. Chaque contexte peut ainsi recevoir jusqu'à deux interactions supposées énaquées d'après la mémoire du contexte précédent. La Figure 9.25 montre le suivi du contenu de la mémoire dans le contexte spatial *virage à droite*. Nous pouvons constater, au cycle de décision 3, que le premier contexte augmenté E' caractérise l'ensemble de l'environnement observable, ce qui aurait été impossible avec des séquences limitées à une longueur de 2. On relèvera également des erreurs liées à la faible fiabilité des interactions composites de longueur 3, notamment au cycle 5 : la mise à jour d'une interaction composite évoquée au pas précédent dans la mémoire \mathcal{T}_3 conduit à la formation d'un contexte ne reflétant pas l'environnement de l'agent.

On ajoute ensuite le contexte spatial *voie sans issue*. On observe que l'agent passe un long moment dans ce nouveau contexte. Il s'ensuit une baisse non négligeable du nombre d'interactions fiables, liée à des erreurs dans certaines signatures mises en évidence par le nouveau contexte

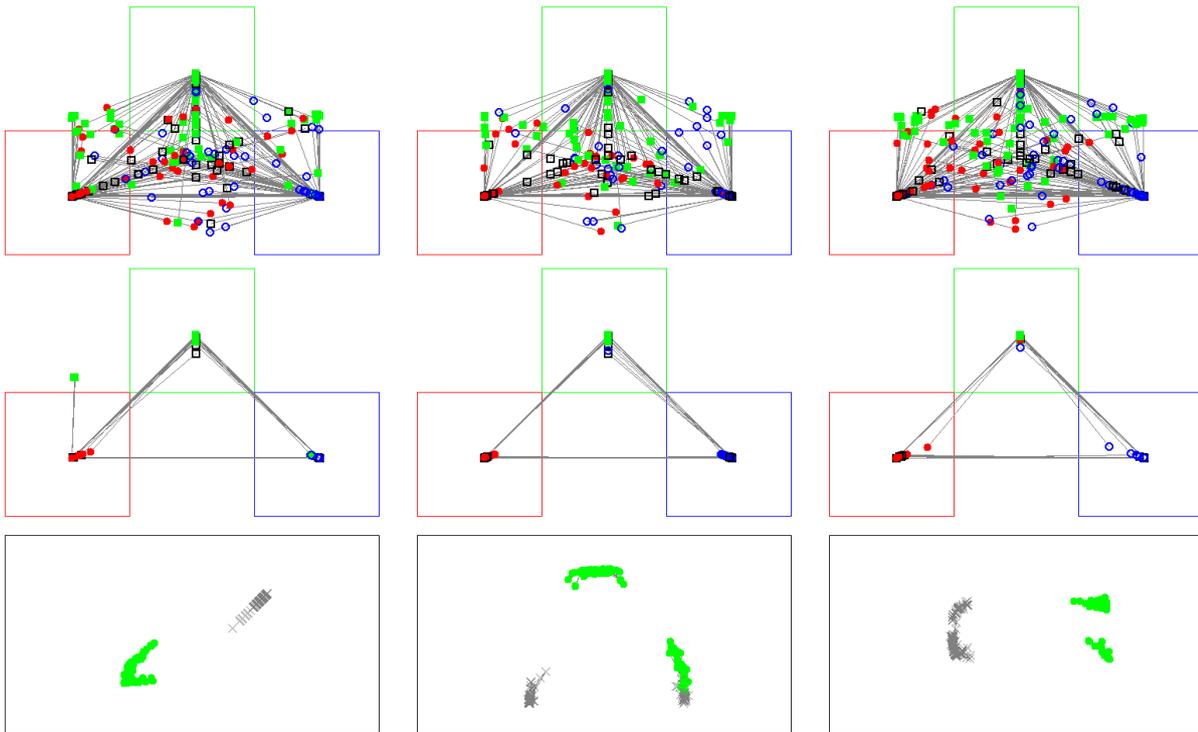


FIGURE 9.24 – Spécialisation des signatures dans les configurations *small loop* (à gauche), *double loop* (centre) et *reverse double loop* (à droite). Les graphes obtenus dans la configuration *reverse simple loop* sont très proches de ceux obtenus dans la configuration *small loop*. Les graphes du haut montrent la spécialisation en fonction de la position. Les graphes du milieu n'affichent que les interactions composites considérées comme fiable. Les graphes du bas montrent la spécialisation en fonction des éléments de l'environnement. Dans ces derniers, nous pouvons constater que des tiraillements existent dans chaque groupe, dénotant de la présence de signatures caractérisant des objets prenant en compte à la fois des espaces vides et des murs. Dans la configuration *reverse double loop*, chaque groupe forme deux sous-groupes proches mais séparés. Ces sous-groupes caractérisent le fait que les signatures définissent des contextes différents, bien que proches par leurs propriétés. Dans la configuration *double loop*, l'agent a formé deux groupes correspondant aux éléments espaces vides et murs, mais également un troisième groupe composé d'interactions composites caractérisant ces deux éléments.

spatial, et l'apprentissage d'un ensemble de nouvelles interactions composites. L'agent débute alors une nouvelle phase d'apprentissage, pendant laquelle il va tester à nouveau ses interactions, et apprendre les signatures des nouvelles interactions. Cette période d'apprentissage est très variable d'une configuration à l'autre, puisqu'il faut entre 4300 et 17500 pas avant d'atteindre un comportement stable. L'apprentissage le plus court est obtenu dans la configuration *reverse simple loop*. Cette rapidité est à mettre en relation avec le faible nombre d'interactions composites apprises par rapport aux autres configurations (cf. Tableau 9.4). On peut noter que l'ajout d'un contexte spatial a permis une hausse non négligeable du nombre d'interactions dont la signature

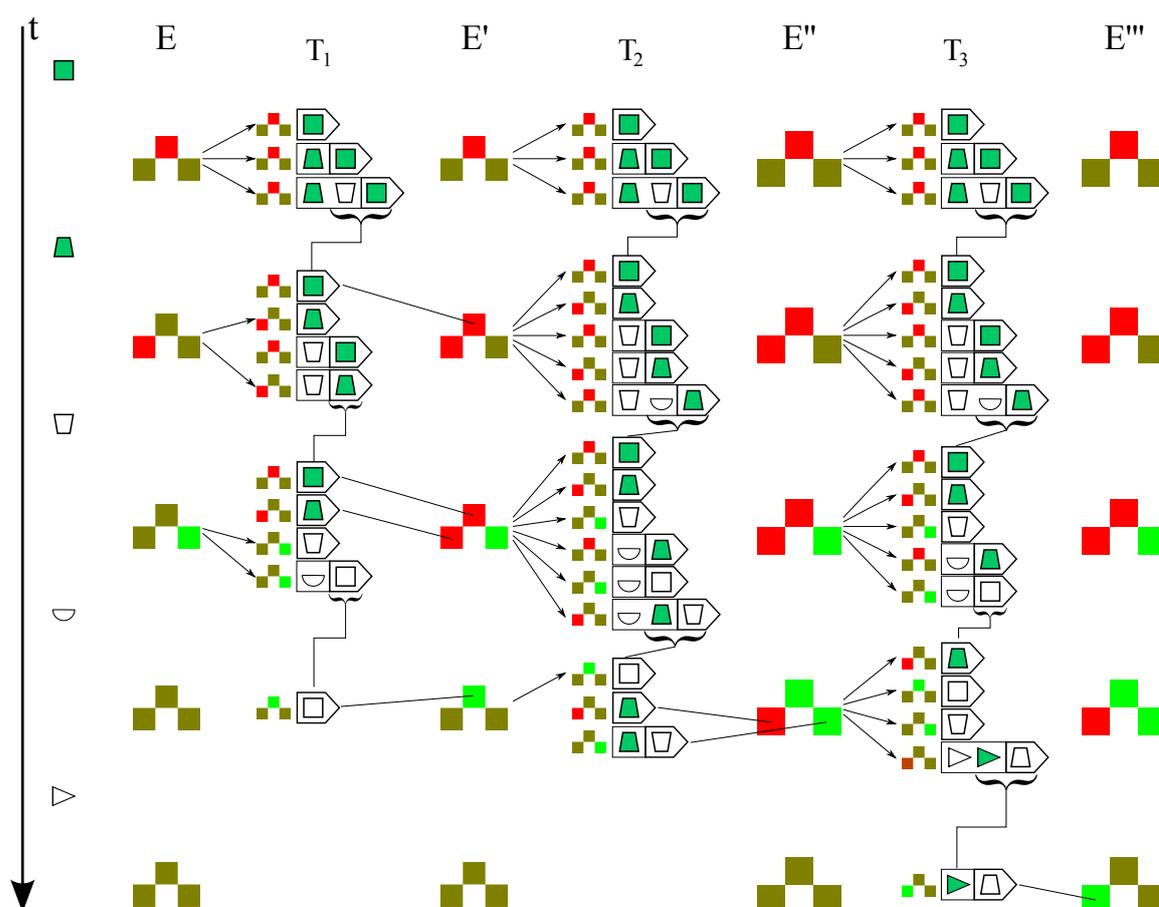


FIGURE 9.25 – Suivi de la mémoire spatio-séquentielle dans le contexte spatial *virage à droite* dans le cas d'un apprentissage dans la configuration *small loop*. On peut noter la présence d'interactions composites de longueur 3 permettant le suivi du mur initialement présent à gauche de l'agent par le premier contexte augmenté E' jusqu'au pas 3, chose impossible avec des séquences de longueur 2. On remarquera, au pas 5, la présence d'une interaction dont la signature est erronée, ce qui génère une erreur dans le contexte E''' . Ce dernier caractérise en effet la présence d'un espace vide à gauche de l'agent, là où se trouve un mur. Cette erreur est toutefois sans conséquence dans le contexte actuel. Cependant, ce type d'erreur est à l'origine du comportement observé dans le cas d'un apprentissage dans la configuration *double loop*.

est correcte. On peut donc supposer que l'apprentissage est facilité par une plus grande diversité de contextes spatiaux.

Une fois le comportement stabilisé, on n'observe que très peu de changement dans les séquences utilisées dans les contextes spatiaux déjà connus. Les interactions composites utilisées dans les trois contextes initiaux sont identiques à celles utilisées avant ajout du contexte *voie sans issue*. On notera toutefois que dans la configuration *double loop*, les erreurs apparaissant à la fin de la séquence utilisée dans le contexte *virage à gauche* ont été partiellement corrigées : en effet, en terminant cette séquence, le contexte interactionnel ne permet pas de caractériser

TABLE 9.4 – nombre d'interactions composites découvertes par l'agent dans chaque configuration, avec des interactions composites limitées à une longueur de 3, après ajout du contexte spatial *voie sans issue*.

	nb interactions	fiables	dé-corrélées	total	correctes
Simple Loop	697 (+43)	270 (41.6%)	222 (31.9%)	512 (73.5%)	496 (71.2%)
Double Loop	687 (+103)	333 (48.5%)	206 (30%)	539 (78.5%)	484 (70.5%)
Reverse SL	577 (+14)	238 (41.2%)	137 (23.7%)	375 (65%)	376 (65.2%)
Reverse DL	665 (+54)	279 (42%)	180 (27%)	459 (69%)	461 (69.3%)

la possibilité d'énacter l'interaction [*tourner à droite face à un espace vide*][*avancer*) comme précédemment, mais caractérise l'impossibilité d'énacter l'interaction [*tourner à gauche face à un espace vide*][*avancer*). L'agent utilise alors, dans un contexte *virage à droite* précédé par un virage à gauche, la séquence b), qui permet de tester la possibilité d'énacter l'interaction [*tourner à droite face à un espace vide*][*avancer*), et utilise, comme précédemment, la séquence a) si le contexte *tourner à droite* précède une ligne droite.

Dans le nouveau contexte spatial, nous obtenons quatre séquences différentes, affichées dans la Figure 9.26. Ces séquences diffèrent tout d'abord par l'interaction composite dont la possibilité d'énaction est testée en premier après élimination de la possibilité d'énacter l'interaction [*avancer*). Le choix de cette seconde interaction composite est identique à celui observé dans les autres contextes spatiaux : [*tourner à droite*][*avancer*) dans les configurations *simple loop* et *double loop*, et [*tourner à gauche*][*avancer*) dans les configurations *reverse simple loop* et *reverse double loop*. Après élimination de cette seconde interaction, une autre différence apparaît : dans les configurations *simple loop* et *reverse double loop*, une erreur apparaît dans le contexte interactionnel, après éniacton de l'interaction [*tourner à droite face à un mur*). Cette erreur caractérise la possibilité d'énacter l'interaction [*tourner à droite face à un espace vide*][*avancer*), ce qui économise une interaction épistémique au pas 5 de la séquence (séquences a et d).

Cette forte individuation montre que le comportement de l'agent n'est pas seulement influencé par l'ordre de découverte des interactions composites, mais également par les erreurs, la plupart du temps bénéfiques, que l'agent conserve après stabilisation de son comportement. Ces erreurs, qui ne seront jamais mises en évidence, font partie intégrante de la façon dont l'agent caractérise son environnement et conduisent à des variations dans la façon dont deux agents, entraînés dans des configurations différentes, interagissent avec un même contexte spatial.

Nous n'observons que peu de changement dans la répartition des interactions en fonction de la position des objets (Figure 9.27). En revanche, la répartition en fonction du type d'objet montre que dans les configurations *simple loop*, *double loop* et *reverse simple loop*, les interactions ont formé deux groupes distincts, liés aux deux types d'éléments présents dans l'environnement. Ces groupes sont plus compacts que ceux obtenus avant ajout du contexte spatial *voie sans issue*, ce qui traduit une meilleure cohérence des signatures. Dans la configuration *reverse double loop*, on observe également deux groupes distincts. Cependant, ces groupes sont moins compacts, et

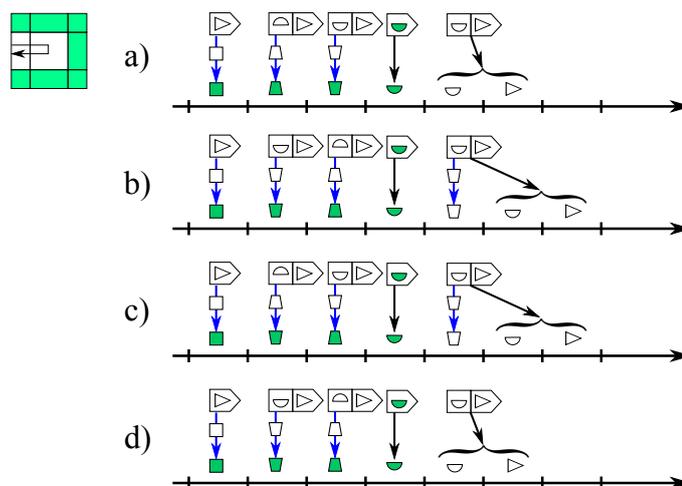


FIGURE 9.26 – Séquences d’interaction utilisées dans le contexte spatial *voie sans issue* après apprentissage. De haut en bas, les séquences utilisées après apprentissage dans les configurations *simple loop*, *double loop*, *reverse simple loop* et *reverse double loop*. Dans les deux premières configurations, l’agent teste en premier la possibilité d’énacter l’interaction $[tourner\ à\ droite\ face\ à\ un\ espace\ vide][avancer]$, tandis que dans les deux autres configurations, il teste la possibilité d’énacter $[tourner\ à\ gauche\ face\ à\ un\ espace\ vide][avancer]$ en premier. Dans les configurations *simple loop* et *reverse double loop*, une erreur dans les signatures génère, après énaction de l’interaction un contexte interactionnel favorable à l’énaction de l’interaction $[tourner\ à\ droite\ face\ à\ un\ espace\ vide][avancer]$, que l’agent énaque sans faire appel à une interaction épistémique. Cette erreur est mise en évidence dans la Figure 9.28.

l’un d’eux forme toujours deux sous-groupes proches mais séparés. Ceci montre que certaines signatures caractérisent toujours des contextes environnementaux complexes ne reflétant pas seulement les objets qui affordent les interactions.

L’analyse de la mémoire spatio-séquentielle met en évidence le phénomène de permanence de l’objet. La Figure 9.28 montre le suivi du contenu de la mémoire spatio-séquentielle pendant l’énaction de la séquence d’interaction utilisée dans le contexte spatial *voie sans issue* après apprentissage dans la configuration *small loop* (séquence a de la Figure 9.26). Nous pouvons observer que du cycle 1 au cycle 3, l’agent construit un contexte interactionnel caractérisant l’ensemble des positions qu’il peut observer. Concentrons-nous maintenant sur l’élément situé à gauche de l’agent : cet élément, un mur, est caractérisé par un ensemble d’interactions composites dont le path a une longueur de 2. Au pas suivant, l’agent énaque l’interaction *tourner à droite face à un mur*. Le mur précédemment à gauche se retrouve derrière l’agent, et est maintenant hors de portée de son système interactionnel. Cependant, une partie des interactions composites caractérisant la présence de cet élément ont été mises à jour. L’objet est ainsi caractérisé par les quatre interactions composites suivantes (notons que seules les deux interactions commençant par l’interaction primitive *tourner face à un espace vide* sont affichées dans la Figure 9.28) :

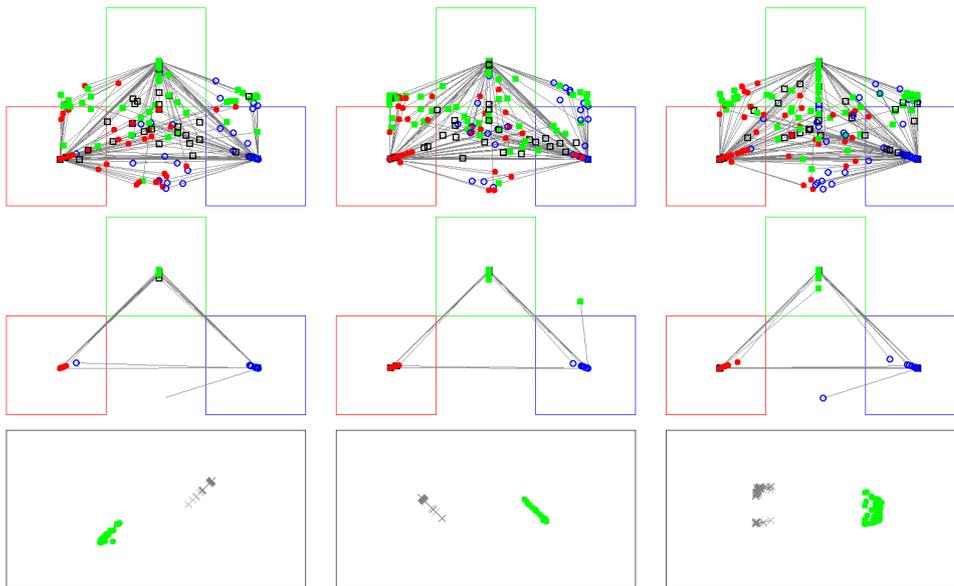


FIGURE 9.27 – Spécialisation des signatures en fonction de la position et du type d'élément, après ajout du contexte spatial *voie sans issue*, dans les configurations *small loop* (à gauche), *double loop* (centre) et *reverse double loop* (à droite). Les graphes obtenus dans la configuration *reverse simple loop* sont similaires à ceux obtenus dans le *simple loop*. On notera, par rapport aux spécialisations en fonction du type d'objet obtenues précédemment, que les interactions forment des groupes plus compacts traduisant un nombre d'erreur plus faible. Le troisième groupe observé dans la configuration *double loop* a disparu. Cependant, dans la configuration *reverse double loop*, le groupe d'interactions associées aux espaces vides est toujours scindé en deux.

- [tourner à gauche face à un espace vide][toucher un mur à gauche]
- [tourner à gauche face à un mur][toucher un mur à gauche]
- [tourner à droite face à un espace vide][toucher un mur à droite]
- [tourner à droite face à un mur][toucher un mur à droite]

Ces interactions caractérisent un objet situé derrière l'agent. Nous pouvons voir sur la Figure 9.28 que ces interactions, encadrées dans un rectangle bleu, sont *décorrélées* (signatures vides). Notons que ces interactions ne caractérisent pas seulement la présence d'un objet, mais donnent également un ensemble de séquences d'interactions permettant d'interagir avec lui. Ainsi, la mémoire spatiale permet de caractériser la présence et la position d'un objet sans qu'il ne soit nécessaire de définir explicitement sa position. Au cycle suivant, l'agent énonce l'interaction *tourner à droite face à un espace vide*. L'une des interactions caractérisant la présence du mur est mise à jour : ce dernier est maintenant caractérisé par l'interaction [toucher un mur à droite]. Le fait que cette interaction soit considérée comme énonçable implique que l'objet qui l'afforde est présent. La signature de cette interaction est donc utilisable pour compléter le contexte interactionnel de l'agent. L'objet, initialement à gauche de l'agent et perdu par le système interactionnel

de l'agent, "réapparaît" ainsi à droite de l'agent, démontrant la capacité de la mémoire spatiale à suivre un objet et à prédire le moment où il sera à nouveau observable. L'agent peut donc tenir compte de cet élément dans la sélection de l'interaction suivante, sans qu'il ne soit nécessaire d'énacter une interaction épistémique pour le percevoir. Cet élément est visible dans le contexte E'' au cycle de décision 5. Notons qu'à cause de conflits avec d'autres interactions stockées en mémoire, le mur à droite n'apparaît pas dans le contexte E''' .

Comme nous pouvons le voir, la mémoire spatiale péri-personnelle permet à l'agent de compléter son expérience de l'environnement, de façon à pouvoir caractériser sa situation, et de sélectionner l'interaction, ou la séquence d'interaction qui lui permet de satisfaire au mieux sa motivation propre. Cette mémoire permet de suivre les objets observés par l'agent, même si ceux-ci échappent au système interactionnel, comme nous avons pu l'observer dans le cas de la voie sans issue. Nous pouvons donc parler d'un phénomène de permanence de l'objet, même si celle-ci est limitée dans l'espace et dans le temps.

Nous avons également constaté que ce mécanisme est limité par le nombre d'interactions composites qui croît de façon exponentielle avec la longueur maximale des interactions composites. Afin de réduire le nombre d'interaction composite, nous proposons un mécanisme inspiré du système vestibulaire des vertébrés permettant de remplacer le path d'une interaction composite par le mouvement produit par l'énaction de ce path. Comme certaines interactions primitives produisent un même mouvement dans l'espace, le nombre de paths possibles, et donc le nombre d'interactions composites, est fortement réduit.

9.2.4 Étude d'un modèle utilisant un système vestibulaire

Nous étudions ici l'hypothèse vestibulaire. Le système interactionnel de l'agent se prête parfaitement à cette hypothèse : en effet, plusieurs interactions primitives génèrent un même mouvement, ce qui réduit significativement le nombre de *paths* d'interaction possibles. Ceci implique que les interactions composites possibles seront moins nombreuses et testées plus fréquemment.

Les douze interactions de l'ensemble I définis dans le tableau 9.1 ne définissent que quatre mouvements possibles :

- rester immobile (les 6 interactions "toucher" et *se cogner*)
- avancer (*avancer d'un pas*)
- tourner à droite (tourner à droite de 90° face à un mur ou à un espace vide)
- tourner à gauche (tourner à gauche de 90° face à un mur ou à un espace vide)

Il faut se rappeler que ces mouvements n'ont aucune signification *a priori* pour l'agent. Quand l'agent énonce une interaction primitive, celle-ci se retrouve associée au mouvement qu'elle a produit. Cependant, pour des raisons de simplification de notre implémentation, nous considérerons que l'association entre les interactions primitives et les mouvements est pré-établie. Cette simplification semble raisonnable puisque l'agent commence par tester les interactions primitives, et ne construit les interactions composites qu'une fois la séquence d'interaction observée, ce qui

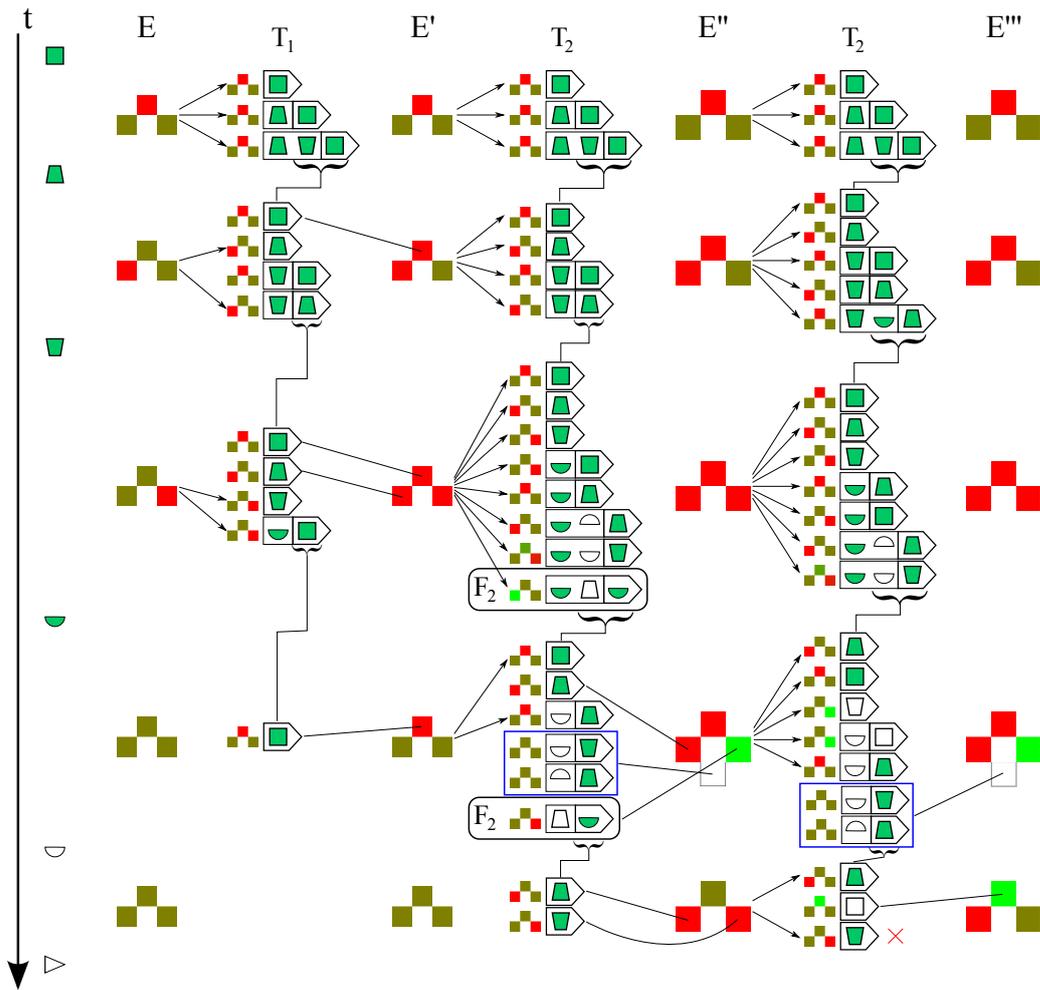


FIGURE 9.28 – Suivi de la mémoire spatio-séquentielle dans le contexte spatial *voie sans issue*, après un apprentissage dans la configuration *small loop*. Nous pouvons ici observer comment la mémoire permet de suivre un objet qui échappe au système interactionnel de l'agent : au cycle de décision 4, après une rotation à droite, on observe la présence d'interactions dont la signature est vide (cadres bleus). Ces interactions sont en effet abordées par un mur situé derrière l'agent, qui, avant la rotation, se trouvait à sa gauche. La mémoire donne les interactions composites permettant d'interagir avec cet objet. Après une seconde rotation à droite de l'agent (cycle 5), l'objet réapparaît, caractérisé par l'interaction [toucher un mur à droite]. Notons que seul le contexte E'' caractérise la présence de cet objet : des incohérences dans les mémoires \mathcal{T}_2 et \mathcal{F}_2 ont conduit à la suppression de cette information. L'erreur qui conduit l'agent à tourner deux fois à droites sans utiliser d'interactions épistémiques est liée à une erreur dans la signature d'une interaction composite observée dans la mémoire \mathcal{F}_2 (rectangles noir). Cette erreur n'est ici pas problématique puisque les contextes augmentés caractérisent correctement l'environnement réel.

implique que les mouvements associés à chaque interaction primitive aient été fournis par le système vestibulaire au moins une fois.

L'expérimentation se déroule selon les mêmes conditions que pour la version non vestibulaire de l'agent : on laisse l'agent apprendre les signatures de ses interactions composites jusqu'à obtenir un comportement stable. Nous pouvons remarquer que la durée d'apprentissage, même si elle est très variable selon la configuration utilisée, est considérablement réduite : pour la configuration *small loop*, par exemple, avec une longueur des interactions composites limitée à 2, le nombre de cycles nécessaires à l'apprentissage passe de 4000 cycles environ à 1800 cycles, et avec une longueur limitée à 3, ce nombre passe de 35 500 cycles à 5800 cycles environ. Il a été également possible de tester l'apprentissage des interactions composites de longueur 4 dans un temps raisonnable : entre 40 600 et 50 000 cycles sont nécessaires pour obtenir un comportement stable, soit un nombre de cycles de décision comparable à l'apprentissage du système non vestibulaire avec des interactions limitées à une longueur de 3.

Le pourcentage d'interactions fiables, et correctes est également plus important, comme le montre le Tableau 9.5. Ceci est principalement dû au fait que les interactions composites sont plus souvent testées, et au faible nombre d'interactions composites possibles. La possibilité d'énacter le path est en effet moins influencée par le contexte de l'agent, puisque deux interactions *opposées* peuvent générer un même mouvement. Notons cependant que le résultat est très variable en fonction de la configuration.

La spécialisation en fonction des positions et des éléments de l'environnement est également plus précise (Figure 9.29). L'agent a clairement défini les positions et les objets qu'il peut observer. On peut noter qu'avec des interactions composites de longueur limitée à 4, la spécialisation est plus importante qu'avec des interactions composites de longueur limitée à 3 sans système vestibulaire, même si le nombre d'interactions composites découvertes par l'agent est plus important. On remarque également que les interactions non spécialisées sont principalement des interactions liées à un élément qui échappent au système interactionnel de l'agent. Ces interactions composites ont un path qui pour la plupart contiennent le mouvement avancer, qui ne peut être effectué que par le biais de l'interaction *avancer d'un pas*. On retrouve ainsi le problème observé précédemment : certaines interactions composites ont un path qui nécessite un contexte particulier et peu courant de l'environnement pour être énéacté, et sont donc peu testées. Cependant, l'hypothèse vestibulaire permet de réduire significativement ce problème.

Cependant, nous n'observons pas de changements significatifs dans le comportement. L'agent utilise les mêmes séquences que précédemment. Nous pouvons noter que malgré l'utilisation d'interactions composites de longueur 4, l'agent utilise toujours les mêmes interactions composites de longueur 2. En effet, la plupart de ces interactions composites de longueur 3 et 4 sont abordées par des objets situés en dehors du système interactionnel de l'agent, et pour lesquels il n'est pas possible de définir de signature. On retrouve également certaines erreurs que nous avons pu observer précédemment, bien que les erreurs soient moins nombreuses.

Cette expérience montre que l'utilisation d'un système vestibulaire sur un agent permet d'accroître de façon significative les performances de l'apprentissage d'une mémoire spatiale, et

TABLE 9.5 – Nombre d'interactions composites découvertes par l'agent dans chaque configuration environnementale, après ajout du contexte spatial *voie sans issue*, dans le cas d'un agent utilisant un système vestibulaire, avec des séquences de longueur limitées à 2, 3 et 4 (de haut en bas).

	nb interactions	fiables	dé-corrélées	total	correctes
Simple Loop	48	30 (62.5%)	10 (20.8%)	40 (83.3%)	48 (100%)
Double Loop	48	31 (64.6%)	10 (20.8%)	41 (85.4%)	48 (100%)
Reverse SL	47	32 (68.1%)	8 (17%)	40 (85.1%)	46 (97.9%)
Reverse DL	50	32 (64%)	4 (8%)	36 (72%)	42 (84%)

	nb interactions	fiables	dé-corrélées	total	correctes
Simple Loop	194	75 (38.7%)	64 (33%)	139 (71.6%)	172 (88.7%)
Double Loop	225	98 (43.6%)	80 (35.6%)	178 (79.1%)	202 (90%)
Reverse SL	192	101 (52.6%)	60 (31.3%)	161 (83.9%)	171 (89.1%)
Reverse DL	197	89 (45.2%)	69 (35%)	158 (80.2%)	184 (93.4%)

	nb interactions	fiables	dé-corrélées	total	correctes
Simple Loop	794	258 (32.5%)	310 (39%)	568 (71.5%)	663 (83.5%)
Double Loop	791	253 (32%)	319 (40.3%)	572 (79%)	677 (90%)
Reverse SL	738	226 (30.6%)	273 (37%)	499 (67.6%)	651 (88.2%)
Reverse DL	780	260 (33.3%)	322 (41.3%)	582 (74.6%)	683 (87.6%)

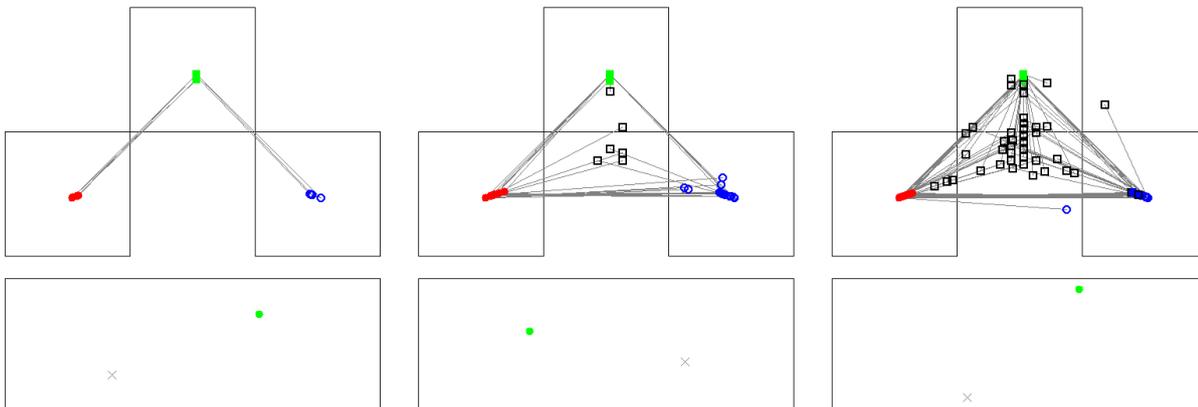


FIGURE 9.29 – Spécialisation des interactions composites en fonction de la position et des éléments dans la configuration *small loop*, avec des interactions composites de longueur limitées à 2, 3 et 4 (de gauche à droite). Les interactions composites forment des groupes compacts. On peut également remarquer que la majorité des interactions ne faisant pas partie d'un groupe sont des interactions affordées par un élément qui échappe au système interactionnel de l'agent.

ce, sans perte de performances ni de variations importantes dans le comportement de l'agent.

9.3 Conclusion

Les expérimentations présentées dans ce chapitre montrent d'une part que l'agent est capable d'apprendre à caractériser les objets qui affordent ses interactions, par le biais des signatures d'interactions, ce qui permet de former implicitement des *bundles* d'interactions. Ces *bundles* permettent de déterminer les interactions que l'agent peut énoncer, même avec un contexte interactionnel limité. D'autre part, ces expérimentations montrent que le mécanisme de la mémoire spatio-séquentielle permet à l'agent d'intégrer, localiser et suivre les objets qui l'entourent, y compris lorsqu'ils échappent à ses interactions, sans qu'aucune connaissance *a priori* sur la structure de l'espace ne soit nécessaire. Cette capacité permet à l'agent de compléter son contexte interactionnel et de caractériser son environnement proche en terme d'interactions.

Ces expérimentations ont également confirmé que le mécanisme de la mémoire spatio-séquentielle reste limité à l'espace proche. En effet, augmenter la portée de ce mécanisme implique une augmentation du nombre d'interactions composites, et donc, du temps d'apprentissage. Ceci implique qu'un autre mécanisme permettant d'intégrer l'espace lointain doit être ajouté.

Chapitre 10

Expérimentation d'un dispositif de mémoire extra-personnelle : étude des mécanismes d'exploitation de la mémoire

Ce chapitre décrit les expérimentations menées pour tester le mécanisme de décision basé sur la version codée en dur de la mémoire spatiale. Ces expérimentations ont pour but de démontrer la faisabilité des mécanismes permettant la navigation dans un environnement composé d'éléments initialement inconnus de l'agent. La mémoire spatiale est ici considérée comme déjà construite, et, selon l'hypothèse H6 (p. 83), peut fournir des renseignements sur la position des interactions énoncées précédemment par l'agent. Le mécanisme d'apprentissage des signatures est similaire, sur le principe, à celui de la mémoire spatiale péri-personnelle, mais s'applique à des interactions primitives. En effet, les interactions composites ne sont pas utilisées dans cette section. Nous avons redéfini certaines propriétés, comme la fiabilité des interactions, pour tenir compte des spécificités des environnements utilisés pour ces tests.

10.1 Description du système de test

Deux systèmes sont proposés pour tester les capacités de la mémoire spatiale. Ces systèmes offrent des environnements ouverts et continus, propices à l'utilisation et à l'analyse d'une mémoire de l'espace environnant.

Le premier système est un agent artificiel évoluant dans un environnement simulé à deux dimensions, dérivé de l'environnement utilisé pour les expériences sur la mémoire spatiale péri-personnelle (décrit au chapitre 9). L'agent n'est ici plus contraint à une grille et peut évoluer librement dans l'espace.

Le second est un robot physique évoluant dans un environnement reproduisant les caractéris-

tiques de l'environnement simulé. L'intérêt d'utiliser un robot est de pouvoir tester la robustesse de nos mécanismes dans un environnement bruité : les moteurs utilisés sont en effet peu précis et le système visuel est calibré de façon approximative.

Les deux systèmes sont contrôlés par le même mécanisme de décision, implémentant le modèle *PSRI* (Parallel Spatial Radical Interactionism). Il faut toutefois noter que le système interactionnel du robot, ainsi que la taille de la mémoire spatiale, présentent quelques différences par rapport au système simulé, différences liées à certaines contraintes imposées par les capteurs du robot.

Ces agents disposent d'un ensemble de sept interactions primaires, définies dans le Tableau 10.1, auxquelles on attache une valeur de satisfaction.

TABLE 10.1 – liste des sept interactions primaires utilisées par le système de test. Le symbole de chaque interaction est celui qui sera utilisé dans les figures de ce chapitre. La valeur de satisfaction associée à chaque interaction est indiquée entre parenthèse.

▷	avancer d'un pas (5)
▶	se cogner contre un mur (-10)
▶	avancer et manger une proie (50)
◃	tourner à gauche de 90° (-3)
◅	tourner à droite de 90° (-3)
◄	tourner à gauche de 45° (-3)
◅	tourner à droite de 45° (-3)

Notons que les interactions *avancer d'un pas*, *se cogner* et *manger* sont affordées par des éléments composant l'environnement, tandis que les interactions *tourner* sont énoncés avec succès quel que soit le contexte. Ces interactions ne sont pas localisables dans l'espace : la position des *actes* construits à partir de ces interactions n'est pas définie.

Nous ajoutons un ensemble d'interactions visuelles secondaires, constitué d'interactions permettant d'observer et de localiser des éléments de couleur se déplaçant par rapport à l'agent. Nous considérons que l'agent dispose d'un système visuel constitué d'une rétine à 1 dimension, lui donnant un champ de vision à 180° et permettant de percevoir trois couleurs : rouge, vert et bleu. La position d'une interaction visuelle peut être définie en fonction du récepteur visuel qui la perçoit et le flux optique mesuré par ce récepteur. L'interaction *se cogner*, qui ne produit pas de mouvements, et donc pas de flux optique, ne permet pas l'énonciation d'une interaction visuelle. Nous définissons ainsi un ensemble de 18 interactions visuelles, consistant à voir un élément d'une certaine couleur parmi rouge, vert, bleu se déplacer en effectuant l'une des six interactions qui produisent du mouvement. Par exemple, *avancer d'un pas et voir un élément rouge se déplacer*, *tourner à droite de 90° et voir un élément vert se déplacer...*

Le champ visuel est segmenté en un ensemble P de positions p . Afin de simplifier les transformations dans la mémoire spatiale, le champ visuel est segmenté pour former une grille régulière.

Les positions p sont ainsi uniformément réparties dans l'espace couvert par le système visuel de l'agent. Un découpage polaire a également été testé dans le cadre d'un apprentissage des signatures.

L'association d'une interaction primitive avec une position de l'espace forme un *acte* (voir chapitre 4, section 4.3). Les actes sont manipulés par nos mécanismes de la même façon que des interactions primitives : ils peuvent en effet être énoncés, et disposent de leur propre signature d'interaction. Par commodité nous appellerons *signature d'acte* la signature d'interaction d'un acte.

L'environnement est composé de quatre types d'éléments. Chaque élément est caractérisé par une couleur qui rend possible son identification par le système interactionnel de l'agent, et permet d'afforder une interaction particulière lorsqu'il se trouve en face de l'agent :

- les espaces vides (invisibles), affordant l'interaction *avancer d'un pas*,
- les murs (verts), qui affordent l'interaction *se cogner*,
- les proies (bleues), qui affordent l'interaction *manger*. Nous utilisons le terme *proie* plutôt que *cible*, car cet élément n'est pas un but que l'agent doit atteindre, mais un objet qui afforde une interaction dotée d'une valeur de satisfaction élevée, que l'agent doit définir par lui-même comme une cible. Une proie, une fois "mangée", est supprimée de l'environnement et remplacée par un espace vide. Une autre proie est ensuite ajoutée aléatoirement dans un espace vide.
- Les algues (rouges), qui affordent, comme les espaces vides, l'interaction *avancer d'un pas*.

Les trois derniers éléments (murs, proies et algues) sont opaques : l'agent ne peut pas percevoir un élément masqué par un autre. Il faut rappeler que l'agent n'a pas de connaissance *a priori* sur ces éléments, c'est-à-dire qu'il ne sait pas comment les définir ni les observer, ni quelles interactions ils affordent. La façon dont l'agent va construire les modèles des objets qui affordent ses interactions peut différer des objets que pourrait définir un observateur extérieur. Par commodité, nous les présentons du point de vue d'un observateur extérieur (humain).

10.2 Environnement simulé

L'environnement simulé est basé sur une version améliorée de l'environnement utilisé pour tester la mémoire péri-personnelle. L'environnement est construit comme une matrice dont chaque cellule peut contenir l'un des objets défini dans la section précédente. L'agent n'est cependant plus contraint à cette matrice et peut évoluer librement dans l'environnement. La Figure 10.1 donne un exemple d'environnement.

La position de l'agent dans son environnement est caractérisée par un triplet (x, y, θ) , avec $(x, y) \in \mathbb{R}^2$ les coordonnées de l'agent dans l'environnement et $\theta \in]-\Pi/2 ; \Pi/2]$ son orientation. Une translation de l'agent est stoppée lorsque la distance entre le centre de l'agent et le bord d'une cellule contenant un mur est inférieure au rayon de l'agent, provoquant également l'énoncé de l'interaction *se cogner*. Dans le cas où, après avoir énoncé une interaction permettant d'avancer, le centre de l'agent est dans une cellule de l'environnement contenant une proie, alors l'interaction

manger une proie est éactée. Cette interaction est prioritaire sur l'interaction *se cogner* : si l'agent heurte un mur et s'arrête sur une cellule contenant une proie, l'interaction *manger* sera éactée. Lorsque l'interaction *manger* est éactée, la proie située dans la même cellule que le centre de l'agent est supprimée et une autre proie est placée dans une cellule vide sélectionnée aléatoirement.

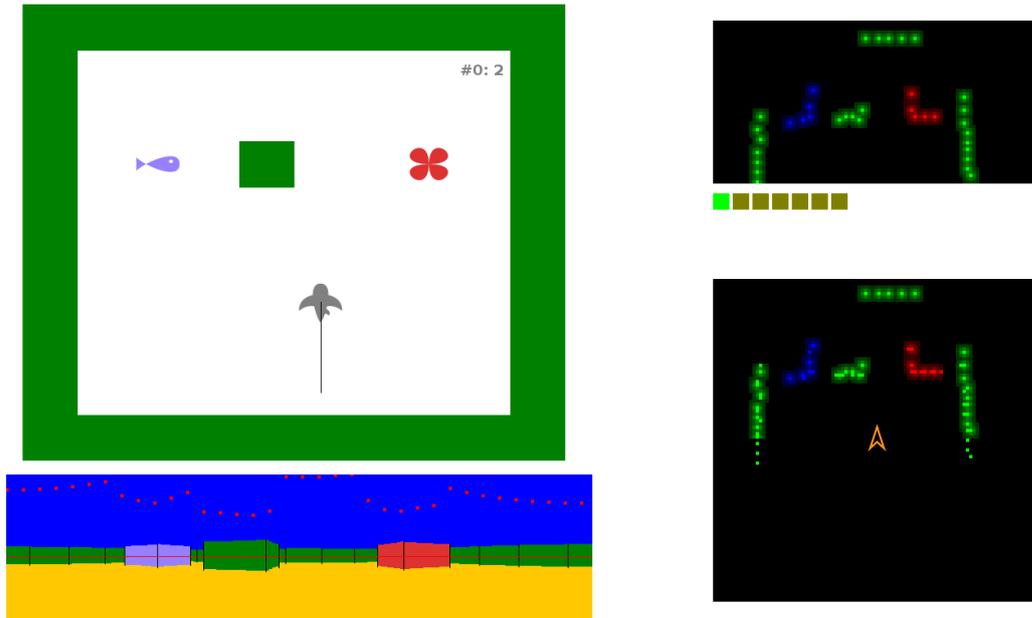


FIGURE 10.1 – Environnement simulé. En haut à gauche : l'agent dans son environnement. L'agent est représenté par un requin gris. La trace montre les déplacements de l'agent sur les 20 derniers cycles de décision. Les murs sont représentés par des carrés verts, les proies par des poissons bleus, les algues par des feuilles rouges et les espaces vides par une surface blanche. En bas à gauche : le mécanisme de rendu, donnant le point de vue de l'agent. Les points rouges montrent le flux optique simulé. Nous ne considérons que 36 récepteurs visuels, donnant une résolution de 5° . Le système visuel fournit à l'agent deux vecteurs, l'un contenant la couleur fournie par chaque récepteur visuel, et l'autre, la distance de l'élément vu par chaque récepteur. En haut à droite : le contexte interactionnel (simplifié). L'espace observable est constitué de 100×50 positions. Les sept interactions primaires sont représentées par sept carrés, donnant l'interaction éactée au cycle précédent (ici, *avancer d'un pas*). La mémoire spatiale montre l'emplacement des interactions précédemment éactées. La position de l'agent dans la mémoire (inconnue de l'agent) est désignée par un triangle orange. La mémoire spatiale est limitée à 100×100 positions pour définir l'espace global.

Le système visuel se base sur un mécanisme proche d'un système de rendu 2D, fournissant, dans le référentiel de l'agent, un modèle de l'environnement polaire donnant la distance et la couleur des éléments les plus proches, avec une résolution angulaire maximale de 1° . Le rendu est limité au champ visuel de 180° de l'agent, avec une résolution angulaire de 5° . Le système

interactionnel définit un ensemble P de $n \times m$ positions de l'espace observable où une interaction visuelle peut être énoncée. Le contexte interactionnel comporte ainsi $n \times m \times \text{card}(I_1) + \text{card}(I_2)$ entrées, où I_1 est le sous ensemble de I des interactions localisables et I_2 l'ensemble des interactions non localisables. La mémoire spatiale est une structure qui prolonge cet ensemble de positions observables pour définir un ensemble P' de $n \times n$ positions de l'espace global.

Les positions, observables ou non, sont définies de façon à permettre une discrétisation de l'espace en coordonnées cartésiennes. Ce référentiel a été choisi tout d'abord parce qu'il permet de simplifier la définition des mouvements, et ensuite parce qu'il permet un découpage homogène de l'espace. Les déplacements sont définis *a priori* en calculant la transformation effectuée par chaque interaction primitive et primaire. Dans le cas de l'interaction *manger*, comme la proie disparaît après l'énoncé de l'interaction, on définit une région, autour de l'agent, où les interactions stockées en mémoire disparaissent. Notons que le système fonctionne également sans cette modification. Cependant, l'agent continue de chercher une proie qui n'existe plus, jusqu'à ce qu'elle disparaisse de la mémoire. Cette modification est également utile dans le cas du système robotique, puisqu'elle évite de devoir déplacer les proies à chaque fois que l'une d'elle est mangée : l'agent "oublie" sa présence et ne la détecte à nouveau que lorsqu'il fait demi-tour.

10.3 Plate-forme robotique

Nos implémentations des systèmes RI sur des robots ont montré que la conception même des robots doit être repensée pour tenir compte des spécificités d'un tel système. Il est en effet apparu que chaque interaction du système interactionnel doit être implémentée par un ensemble de capteurs et d'effecteurs permettant de définir le résultat de cette interaction. Il est donc préférable de définir les actionneurs et les capteurs du robot pour implémenter les interactions de l'agent, plutôt que d'interfacer le système sensoriel avec le système interactionnel.

10.3.1 Adaptation d'un système RI sur un robot

Nous avons porté nos mécanismes d'apprentissage sur des plate-formes robotiques sur lesquelles des systèmes interactionnels pré-existants ont été adaptés. La Figure 10.2 montre les robots dans leur environnement.

Robot e-puck

Les premières versions des modèles RI et SRI ont été portées sur un robot e-puck [86]. Ce petit robot est équipé de deux moteurs et de nombreux capteurs. Parmi eux, un ensemble de capteurs à faible portée que nous avons utilisés pour implémenter les interactions *toucher* et *se cogner*. Chaque interaction *toucher* était détectée en utilisant trois capteurs de proximité. La portée de ces capteurs étant très faible, il a fallu réduire la distance parcourue par l'agent lorsqu'il effectue une interaction *avancer*, afin que les objets détectés par les interactions *toucher* soient pertinentes pour prédire le résultat de l'interaction *avancer*. Ces mêmes capteurs ont été

utilisés pour la détection de l'interaction *se cogner*, détectée lorsque la distance mesurée par les capteurs avant était inférieure à un seuil.

La principale difficulté était que les capteurs de proximités ne permettaient pas une détection efficace et fiable des éléments qui affordent les interactions de l'agent. Ces capteurs sont en effet fortement directifs, et, même en utilisant plusieurs de ces capteurs pour implémenter une interaction *toucher*, la surface couverte par ces capteurs ne couvrait pas toute la surface nécessaire pour déterminer, par exemple, si le robot pouvait avancer sans se cogner.

Robot Eddie

Les premières versions des systèmes RI dotées d'un système visuel ont été portés sur un robot Eddie. Ce robot a été équipé d'un capteur Kinect, permettant de générer une représentation en 3 dimensions de l'espace situé devant l'agent. Ce capteur est utilisé pour détecter l'interaction *se cogner*, lorsqu'un objet est suffisamment proche du robot. Ce capteur était également utilisé pour détecter les *features* visuelles : la proie est définie par un QR-code permettant sa détection et sa localisation par le capteur kinect. Cependant, le champ de vision de la caméra étant inférieur à 90°, il a fallu réduire l'angle des rotations produite par les interactions *tourner*.



FIGURE 10.2 – Implémentation de nos mécanismes dans des robots. À gauche, le robot e-puck dans le *box world* [52]. L'implémentation a été réalisé avec Laurent Hu. À droite, le robot Eddie. La proie est représentée par un QR-code. L'implémentation a été réalisée par Christian Wolf et Martin Kodys, de l'équipe IMAGINE¹⁰.

10.3.2 Conception d'un robot adapté à un système RI

Nous avons ensuite porté nos mécanismes sur un robot construit en *Lego*. La flexibilité du système *Lego* nous a permis de construire le système sensoriel de l'agent en fonction des interactions du système interactionnel, et de respecter le principe de *balance écologique* [100]. Les interactions primitives de l'agent sont ainsi définies par les deux moteurs du robot, assurant les translations

10. <http://liris.cnrs.fr/ imagine/>

et rotations générées par les interactions, et par les capteurs donnant le résultat des interactions de façon certaine. En effet, le résultat est directement défini par le système sensorimoteur, et non par une interprétation des informations fournies par les capteurs. Un système visuel permet de définir les interactions visuelles secondaires énoncées. Ce système a été construit pour respecter les contraintes du système interactionnel, c'est-à-dire un champ de vision de 180°, détection de trois couleurs et des distances.

Ce système est conçu pour se rapprocher du système simulé. J'ai utilisé un robot basé sur les *Lego Mindstorms*, et plus particulièrement, sur une brique RCX. Le modèle du robot est disponible en ligne¹¹. Le robot est contrôlé par le même mécanisme de décision que pour l'agent simulé, s'exécutant sur le même PC. La connexion entre le PC et le robot est établie à l'aide d'une connexion infrarouge (selon un protocole propre au RCX), et à l'aide d'un câble USB (pour le système visuel). La brique RCX est programmée en langage NQC (Not Quite C)¹² pour recevoir les interactions intentions à tester et retourner l'interaction énoncée au mécanisme de décision à la fin du cycle de décision.

L'environnement est constitué d'un enclos d'environ 1m x 1m, bordé par des murs. On dispose d'un ensemble d'éléments représentant les différents objets : des cubes en cartons représentent des blocs murs non traversables, et des carrés de carton colorés représentent les proies et les algues. Les cubes et les murs qui bordent l'enclos, représentant les murs, sont verts. Les carrés de carton représentant les proies sont bleus, et ceux représentant les algues sont rouges. L'utilisation de couleurs primaires permet de faciliter la détection de ces couleurs par le système visuel.

Le robot, que j'ai appelé ErnestIRL (Figure 10.3), dispose d'un ensemble d'actionneurs et de capteurs qui implémentent le système interactionnel défini dans le tableau 10.1. Le robot est équipé de deux moteurs électriques à courant continu en configuration différentielle, lui permettant d'aller tout droit, de tourner à gauche ou à droite. Les capteurs sont conçus pour permettre la détection du résultat des interactions : le robot est équipé d'un large capteur de contact frontal, lui permettant de détecter l'interaction *se cogner*. Un capteur de luminosité, placé sous le robot, permet de détecter le passage sur un carré coloré. Ce capteur est couplé à une led rouge éclairant le sol. Les carrés rouges apparaissent donc aussi lumineux que le sol blanc. Le passage sur un carré bleu génère, par contre, une forte baisse de la luminosité, permettant de détecter le passage sur une proie, et donc de détecter le succès de l'interaction *manger*. Les interactions *tourner*, qui sont toujours possibles, ne nécessitent pas de capteurs particulier.

Les interactions visuelles sont définies par une caméra panoramique, constituée d'une caméra orientée vers une surface sphérique réfléchissante. Ce système offre un champ de vision de 180°, avec une portée suffisante pour couvrir l'ensemble de l'environnement situé devant l'agent. La caméra offre une résolution de 640x480 pixels. Étant donné que l'environnement est plat, il est possible de définir la distance des éléments observés en fonction de leur position sur l'image, la caméra ne permettant pas, à cause de sa fréquence trop faible, de mesurer un flux optique avec une précision suffisante. L'image est traitée pour déterminer les interactions visuelles énoncées : dans

11. <http://liris.cnrs.fr/simon.gay/index.php?page=eirl>

12. <http://bricxcc.sourceforge.net/nqc/>

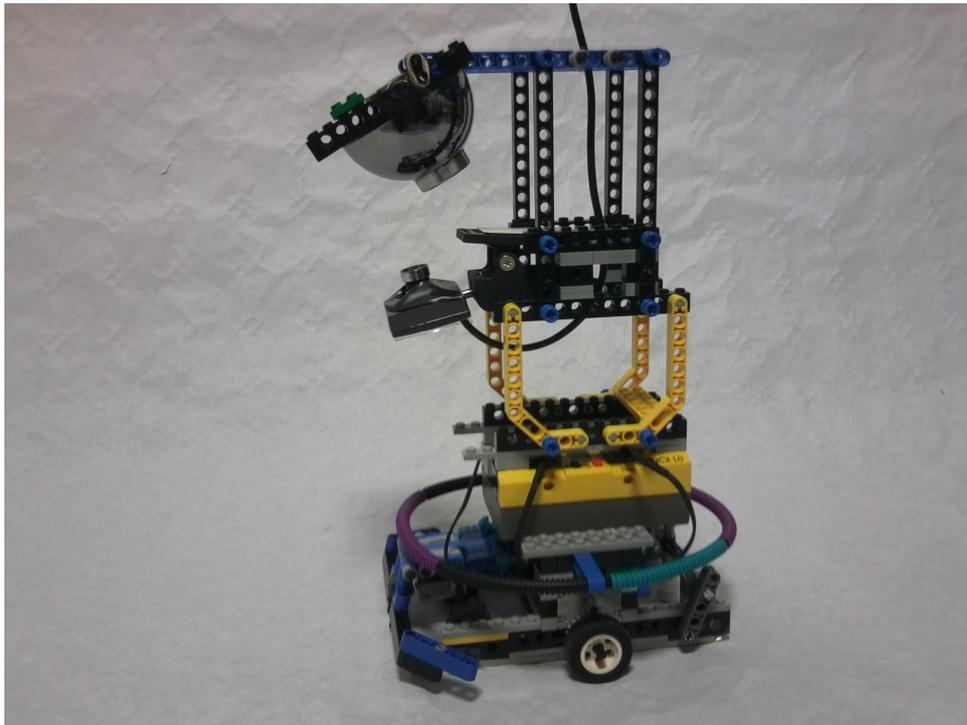


FIGURE 10.3 – Le robot ErnestIRL. Il dispose d'un capteur de contact frontal, d'un capteur de luminosité situé sous le robot, et d'un système visuel panoramique, constitué d'une webcam dirigée vers une surface sphérique réfléchissante. La propulsion est assurée par deux moteurs à courant continu.

un premier temps, les couleurs sont filtrées pour ne conserver que les trois couleurs observables. Afin de respecter l'opacité des objets, les pixels de l'image qui sont "masqués" par des pixels plus proches sont effacés. L'image obtenue est projetée pour obtenir une image plane. Cette projection a été obtenue en calibrant une surface quadrillée sur l'image obtenue avec la caméra. La Figure 10.4 montre les étapes du traitement de l'image.

L'image obtenue après projection reste cependant approximative. Nous verrons que le mécanisme de décision reste néanmoins fonctionnel malgré cette approximation. Nous pouvons également constater que la caméra est visible sur l'image finale. Nous verrons l'influence de cet élément sur les signatures des interactions. En revanche, nous éliminerons les interactions visuelles désignant la caméra de la mémoire spatiale. En effet, ces interactions restent énoncées à chaque pas. Il est donc raisonnable de penser qu'un mécanisme permettant d'apprendre le câblage de la mémoire spatiale peut apprendre qu'aucun mouvement n'est observable dans cette région. Un objet "reconnu" dans cet ensemble d'interactions ne produirait aucune variation de proximité globale, puisque restant immobile par rapport à l'agent. Je propose ainsi de ne pas tenir compte de ces interactions dans la mémoire spatiale.

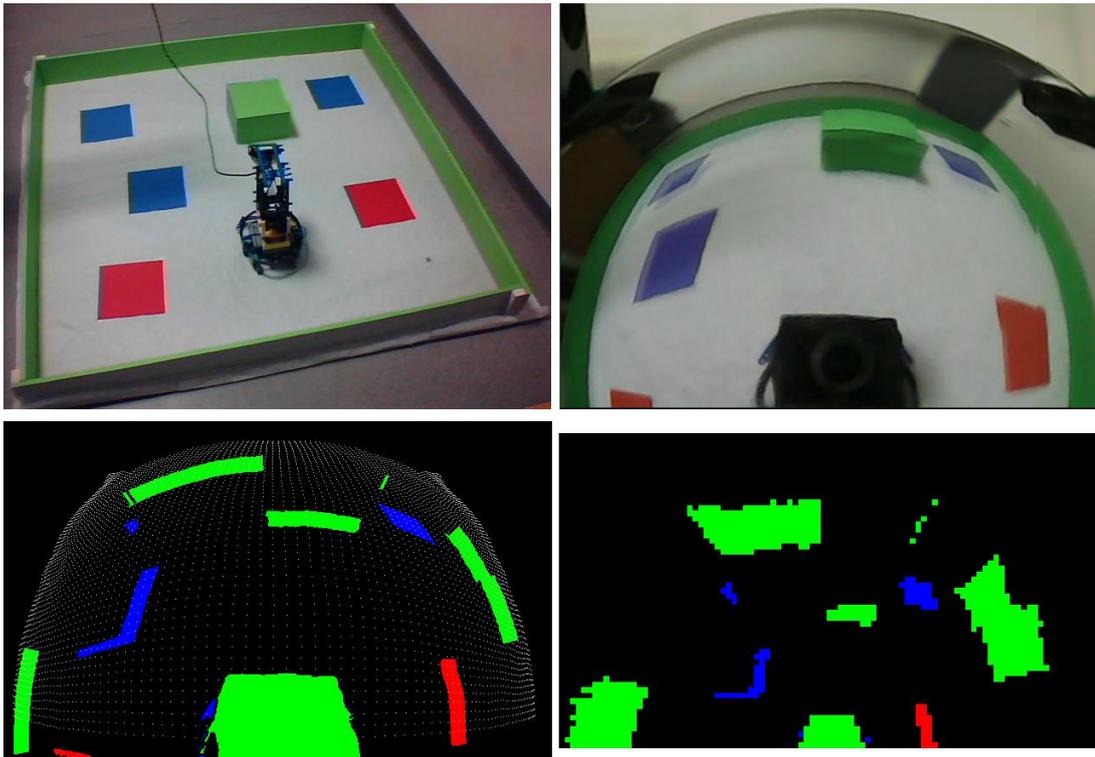


FIGURE 10.4 – Traitement de l'image. En haut à gauche : le robot dans son environnement. En haut à droite : l'image fournie par la caméra. L'image est filtrée : les couleurs sont reconnues et les éléments "masqués" sont retirés (en bas à gauche). L'image est projetée pour obtenir une image de l'environnement, qui servira à définir les interactions visuelles énoncées. Les points blancs sur l'image en bas à gauche (visibles en agrandissant l'image) correspondent aux pixels sur l'image finale. On peut remarquer que l'agent détecte sa caméra.

10.4 Formation des signatures d'actes

L'utilisation d'un système évoluant dans un environnement continu nous permet d'étudier l'émergence des signatures de façon plus précise qu'avec le système utilisé au chapitre 9. Nous rappelons que les signatures d'actes sont équivalentes aux signatures d'interactions, appliquées aux actes de l'agent. Nous observerons également les propriétés de ces signatures d'acte, qui permettront la construction d'une mémoire de l'espace au chapitre suivant. Dans cette expérimentation, nous étudions uniquement l'émergence des signatures d'acte. Nous laissons donc l'agent apprendre les signatures de ses actes, poussé par son système d'apprentissage. À cause de la durée de l'expérience (estimée à plusieurs jours avec le robot), seul l'agent simulé a été utilisé.

L'agent ne dispose pas de mémoire spatiale, car inutile pour l'apprentissage des signatures. L'espace observable est divisé en 30×15 positions, ce qui permet de définir un ensemble de 8100 *actes* visuels ($30 \times 15 \times 18$ interactions visuelles) et 7 *actes* primaires (non localisables), liés aux sept interactions primaires. Le contexte interactionnel est ainsi constitué d'un vecteur de 8107 éléments. Comme les interactions peuvent avoir plus d'une interaction opposée, on ne peut pas

tenir compte de l'échec des interactions : l'échec d'une interaction n'implique pas le succès de ses interactions opposées. Les signatures sont implémentées, comme au chapitre précédent, avec un réseau de neurone artificiel. Une signature est ainsi constituée d'un vecteur de 8108 poids (8107 entrées du contexte et un biais).

L'environnement contient initialement un grand nombre d'éléments de différents types, afin que l'agent puisse expérimenter un grand nombre d'interactions visuelles simultanément. Cependant, au fur et à mesure que les signatures des actes visuels abordés par des éléments proches deviennent fiables, il faut retirer un certain nombre de ces éléments, afin de permettre l'énaction des actes permettant de détecter un élément situé à une position lointaine sans qu'il ne soit masqué par un autre élément.

Les expérimentations ont été effectuées avec un coefficient d'apprentissage α du réseau de neurone fixé à 0.05. Ce coefficient a été choisi car il offre un bon compromis entre la vitesse d'apprentissage et la stabilité des poids synaptiques du réseau. Les coefficients d'influence γ et β de la mémoire spatiale ne sont pas spécifiés, car uniquement liés au mécanisme d'exploitation de la mémoire spatiale, qui ne sera pas utilisé ici.

10.4.1 Mécanisme d'apprentissage

Le mécanisme d'apprentissage doit respecter certaines contraintes. En effet, pour pouvoir tester une interaction visuelle secondaire, il est nécessaire de définir la faisabilité de son interaction primaire associée. Il faut ainsi donner la priorité, dans la sélection des interactions à tester, aux interactions primaires. De plus, certaines interactions visuelles sont abordées par des objets situés en dehors du système sensoriel de l'agent. On pense notamment aux actes visuels énoncés pendant une rotation : une partie du champ visuel ne peut être prédit car lié à des éléments situés derrière l'agent. Il faut donc détecter et éliminer ces interactions de la sélection.

Le mécanisme de sélection est le suivant : on définit dans un premier temps les interactions candidates :

- Une interaction primaire peut toujours être testée.
- Une interaction secondaire peut être testée si l'interaction primaire associée est prédite comme un succès avec une valeur de certitude supérieure à un certain seuil. Des essais préliminaires nous ont permis de définir ce seuil à 0.8.

L'interaction doit également avoir une certitude supérieure à un certain seuil, ce qui permet d'éliminer les interactions qui ne peuvent pas être prédites. Ce seuil, fixé à 0.2, a également été choisi d'après nos essais préliminaires. À chaque pas, l'agent sélectionne l'interaction pouvant être testée dont la certitude de succès est la plus faible (en valeur absolue), parmi les interactions candidates.

10.4.2 Affichage des résultats

Nous étudions la façon dont l'agent construit les objets qui abordent ses interactions en analysant les poids de ses signatures. Pour permettre une lecture aisée de ces ensembles de

poinds, les signatures seront affichées de la façon suivante :

- On regroupe dans un premier temps les poinds liés aux actes visuels associés à un même acte primaire. Les poinds de chaque groupe sont organisés pour respecter la topographie donnée par les positions des actes visuels. Nous rappelons que l'agent n'a pas accès à l'organisation des positions dans l'espace.

- Dans chaque groupe, chaque position est occupée par trois poinds, chacun associé à un acte définissant une couleur parmi {rouge, vert, bleu}. On définit en chaque position une couleur en référentiel RVB définie par le triplet de poinds correspondant, chaque poind caractérisant la valeur du canal de la même couleur que celle qui définit l'acte associé au poind. Les valeurs des poinds sont normalisées par rapport au poind le plus élevé (en valeur absolue). Pour chaque canal, une valeur de 0 correspond à un poind normalisé de -1, et une valeur de 1 correspond à un poind normalisé de 1. Une couleur grise indique donc que les trois poinds associés aux interactions correspondantes ont une valeur de 0.

- Les groupes sont organisés de gauche à droite pour respecter l'ordre défini par le tableau 10.1. Les poinds liés aux interactions primaires, non localisables, ainsi que le biais du neurone, sont affichés à part, sous la forme de 8 carrés. Pour ces carrés, une couleur noire indique un poind normalisé de -1, et une couleur blanche, un poind normalisé de 1.

Cette organisation nous permet d'observer la position par rapport à l'agent, ainsi que certaines propriétés des objets, telles la couleur et la taille, définies par la surface de l'espace où les poinds sont élevés.

10.4.3 Résultats expérimentaux

On laisse l'agent évoluer dans son environnement, en observant l'évolution des signatures. Au bout de 4000 à 5000 cycles de décision, les signatures des actes primaires émergent et se stabilisent. Ces signatures sont affichées dans la Figure 10.5. On observe que les signatures des actes *avancer*, *se cogner* et *manger* sont associées à des actes visuels permettant la détection d'éléments de couleur situés devant l'agent : l'acte *avancer* est lié par des poinds négatifs aux actes permettant de percevoir des objets verts et bleus, ce qui correspond à l'absence de mur et de proie, l'acte *se cogner* est associé avec des poinds positifs aux actes permettant de détecter un objet vert (les murs), et avec des poinds négatifs aux actes permettant de percevoir un objet bleu (les proies). L'acte *manger une proie* est liée aux actes permettant de détecter un objet bleu (les proies).

On remarque également que l'espace occupé par les actes désignés par les signatures de ces actes correspondent approximativement à la taille de l'agent : en effet, l'agent ne peut avancer d'un pas que si l'espace qu'il occupera après avoir avancé est constitué d'espace vide.

L'acte *se cogner* est également pris en compte dans les signatures : l'énaction de cet acte caractérise en effet la présence d'un mur devant l'agent. L'énaction de *se cogner* permet donc de détecter l'objet qui l'afforde, et qui exclut l'énaction des interactions *avancer* et *manger*. Ainsi, les actes *avancer* et *manger une proie* sont liés à l'échec de cet acte (poinds négatif), et

l'acte *se cogner*, à son succès. Le fait que l'acte *se cogner* soit lié à lui-même implique que les objets post-énaction et pré-énaction de cet acte soient identiques. Cet acte ne génère donc pas de mouvement : l'agent reste devant un mur. Les autres actes primaires sont peu pris en compte : en effet, l'énaction de ces actes renseignent seulement de la présence d'un espace vide sous l'agent, information non pertinente pour prédire le résultat des autres actes.

Les actes *tourner*, qui ne peuvent pas échouer, sont fortement liées au biais de leurs signatures, indiquant ainsi qu'elles ne sont pas liées à un objet de l'environnement, mais sont une propriété de l'agent.

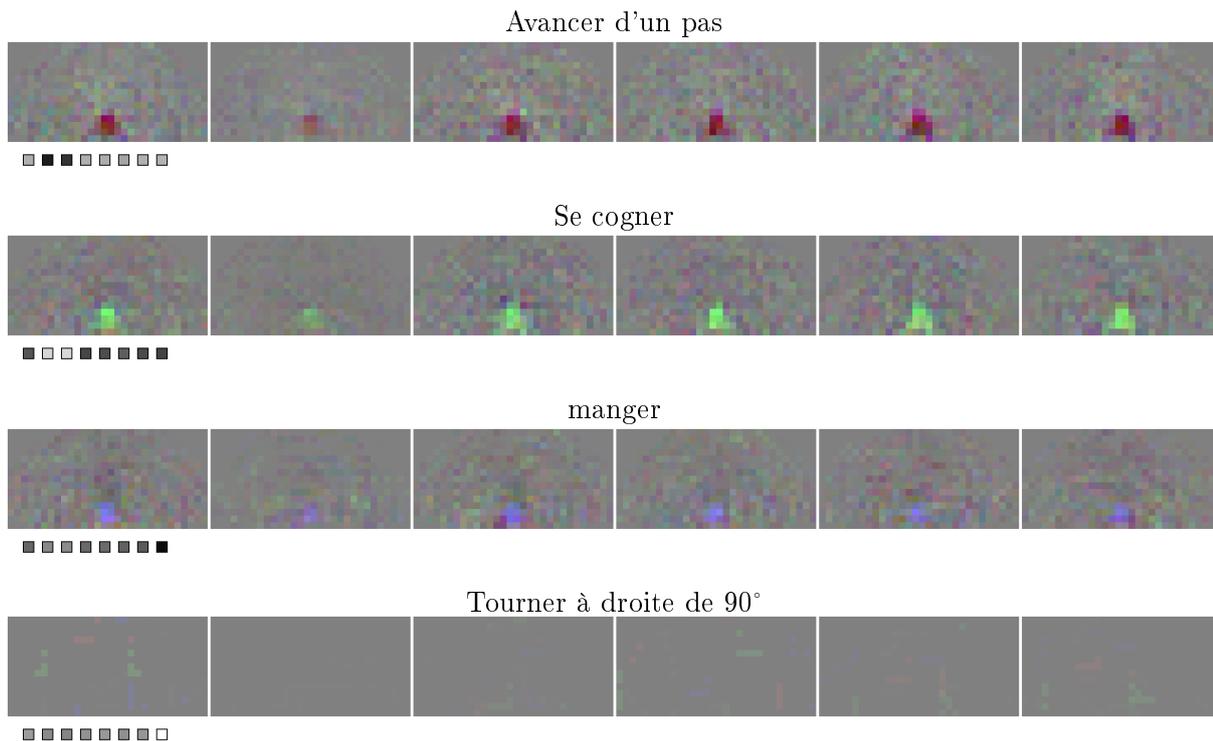


FIGURE 10.5 – Signatures des actes primaires. De haut en bas, les signatures des actes *avancer d'un pas*, *se cogner*, *manger une proie*, *tourner à gauche de 90°*. Les signatures des autres actes *tourner* ne sont pas représentées car similaires à celle de *tourner à gauche de 90°*. Les poids des actes visuels sont regroupés en fonction du type d'acte primaire associé. Dans chaque groupe, les actes visuels sont organisés pour respecter la topographie de l'espace, afin de faciliter la lecture. Les groupes sont organisés en fonction des actes primaires associés, pour respecter l'ordre défini par le tableau 10.1. Les huit carrés affichent les poids liés aux sept actes primitifs et le biais du neurone. Nous pouvons observer que les actes *avancer d'un pas*, *se cogner* et *manger* sont liés à un objet localisé devant l'agent. Ces objets sont caractérisés par le succès ou l'échec d'actes visuels permettant de détecter une certaine couleur, mais indépendamment des actes primaires auxquels ils sont associés. En effet, le même motif est présent dans chaque groupe : l'agent a ainsi appris que ces actes génèrent le même objet post-énaction.

Ces signatures montrent que l'agent a bien intégré les propriétés des objets qui l'entourent : la

couleur, la taille et la position de ces objets sont clairement distinguées. Nous pouvons également observer qu'une signature définit les actes visuels permettant de détecter la présence de l'objet qui afforde l'acte associée, en fonction des couleurs et des positions, mais indépendamment du mouvement qui leur a donné naissance : le même motif se répète sur chaque groupe de poids. On peut donc définir une équivalence entre deux actes visuels, associés à la perception d'une même couleur et localisés à une certaine position de l'espace par le fait qu'ils permettent de détecter des objets affordant un même acte.

Les signatures des actes secondaires les plus souvent énoncées commencent à émerger à partir de 8000 cycles de décision. La grande majorité des signatures se stabilisent au bout de 25 000 cycles de décision environ. La Figure 10.6 montre quelques exemples de signatures d'actes visuels.

En analysant ces signatures, on peut observer que les actes visuels sont fortement liés à un groupe restreint d'actes permettant de détecter la même couleur en une même position de l'espace. Les positions des actes désignées par les signatures sont particulières : en effet, la différence entre la position d'un acte et la position des actes désignés par sa signature correspond au mouvement produit par l'énonciation de cet acte. Ainsi, un acte a_k associé à une position p est lié, par sa signature, aux actes dont l'énonciation permet de caractériser un objet de la même couleur à une position p' telle que $\tau_{a_k}(p') = p$. Nous pouvons également constater que les signatures sont peu liées aux actes visuels basés sur l'acte primaire *manger*. En effet, l'acte *manger* est peu souvent énoncé, ce qui réduit le nombre d'apprentissages des actes visuels liés à cet acte.

Ainsi, les signatures, en plus de définir les objets qui affordent les actes, permettent également, lorsque ces actes sont associés à une position de l'espace, de caractériser le mouvement produit par chaque acte primaire. La Figure 10.7 montre la différence entre la position des actes visuels, et le barycentre des positions des actes désignés par leur signature. Le mouvement produit par l'énonciation des actes primaires *avancer d'un pas* et de tourner à droite et à gauche de 90° et 45° , en référentiel égocentré, est parfaitement reconnaissable.

L'émergence de ces liens entre les positions de l'espace par le biais des signatures rend inutile la nécessité de prédéfinir les déplacements de la mémoire spatiale dans la partie observable (c'est-à-dire les positions de P). Nous formulons l'hypothèse suivante, que nous utiliserons au chapitre suivant :

Hypothèse 9. *Les déplacements produits par l'énonciation des interactions primaires dans la partie observable de la mémoire spatiale peuvent être appris par le biais des signatures d'interaction.*

Le fait que ces déplacements puissent être appris rend possible la reconnaissance des instances d'objet sans se baser sur les préconceptions de la mémoire spatiale.

10.4.4 Détection des instances d'objet

Les liens observés précédemment permettent de "déplacer" une signature d'actes dans l'espace. Chaque acte visuel a désigné par une signature dispose lui-même de sa propre signature, qui désigne un autre acte visuel (ou un groupe restreint d'actes) dont la position correspond à

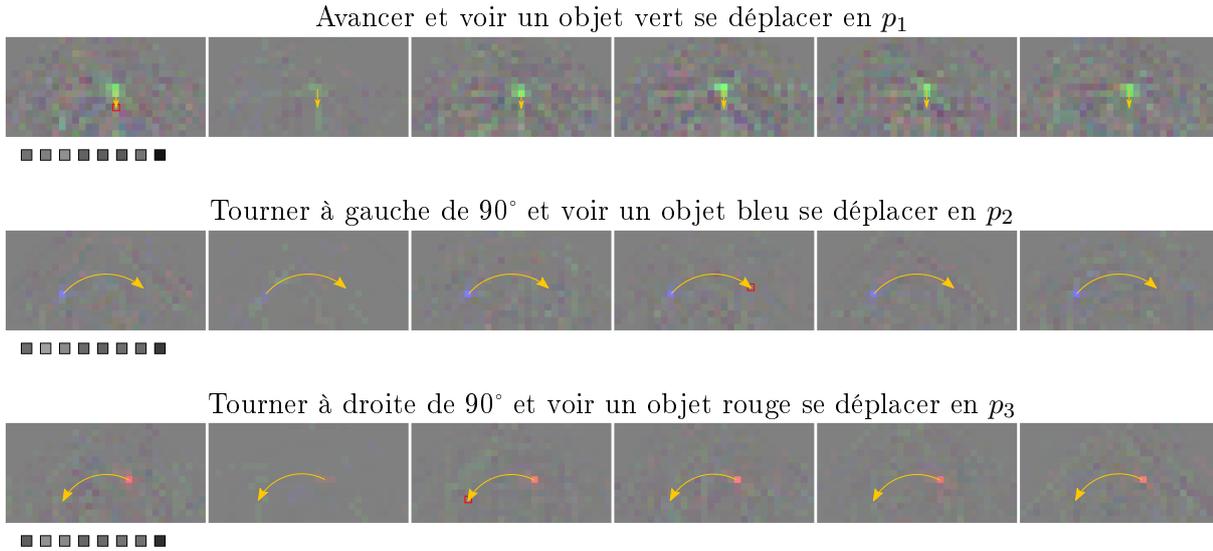


FIGURE 10.6 – Exemples de signatures d'actes visuels : *avancer et voir un élément vert*, *tourner à gauche de 90° et voir un élément bleu*, *tourner à droite de 90° et voir un élément rouge*. Les positions de ces actes sont données par les carrés rouges. On notera que les actes désignés par les signatures correspondent à un élément de même couleur. La différence entre la position des actes visuels et la position des actes désignés par leur signature est montré par une flèche orange. On reconnaît les différents déplacements provoqués par l'énaction des actes primaires : translation et rotation. Nous pouvons également remarquer que les signatures des actes visuels disposent d'un biais négatif : ces actes sont ainsi prédits comme des échecs, sauf si les actes désignés par leur signature sont présents dans le contexte interactionnel.

la position de a déplacé dans l'espace par l'interaction primaire associée à a , comme nous avons pu le constater précédemment. Il est donc possible de construire une *image* de la signature d'un acte par la transformation générée par l'énaction d'une interaction primitive i .

Définition 36. *Image d'une signature : l'image de la signature d'un acte a est la signature de a déplacée dans l'espace par une transformation τ .*

Une telle *image*, notée $S'_{a,i}$ est générée par :

$$S'_{a,i} = \sum_{b_k \in A_i} S_{b_k} \times \begin{cases} w_{a,k} & \text{if } |w_{a,k}| > 1 \\ 0 & \text{else} \end{cases} \quad (10.1)$$

avec $A_i \in A$ ensemble des actes localisables associés à l'acte primaire i , S_b la signature d'un acte b , $w_{a,k}$ le poids de la signature de l'acte a associé au $k^{ième}$ acte de l'ensemble A . Le seuil de 1 permet d'éliminer les poids considérés comme non pertinents.

Cette transformation peut être appliquée plusieurs fois consécutivement. Il devient alors possible de détecter des instances d'objet et de les localiser par la séquence de transformations

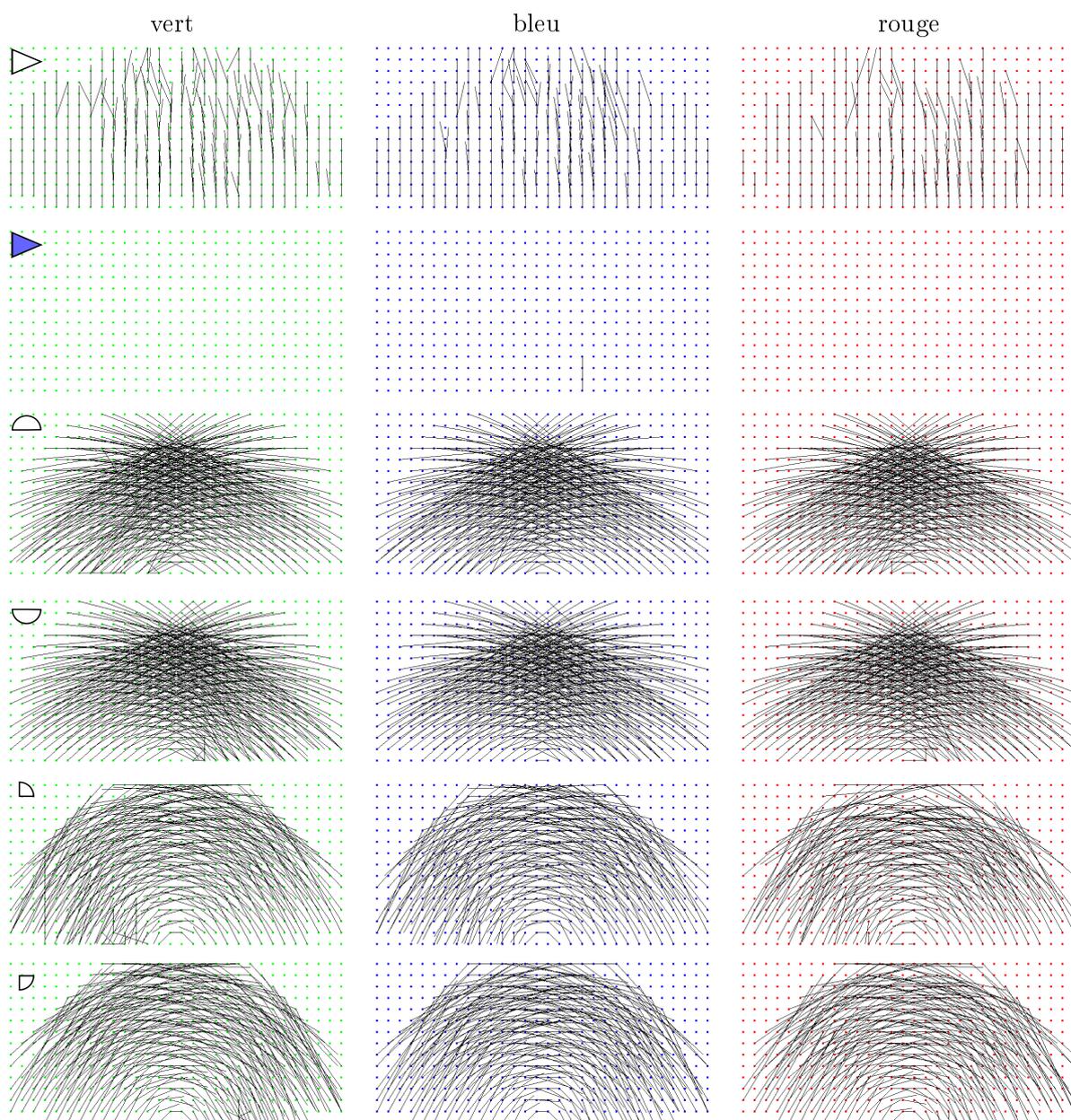
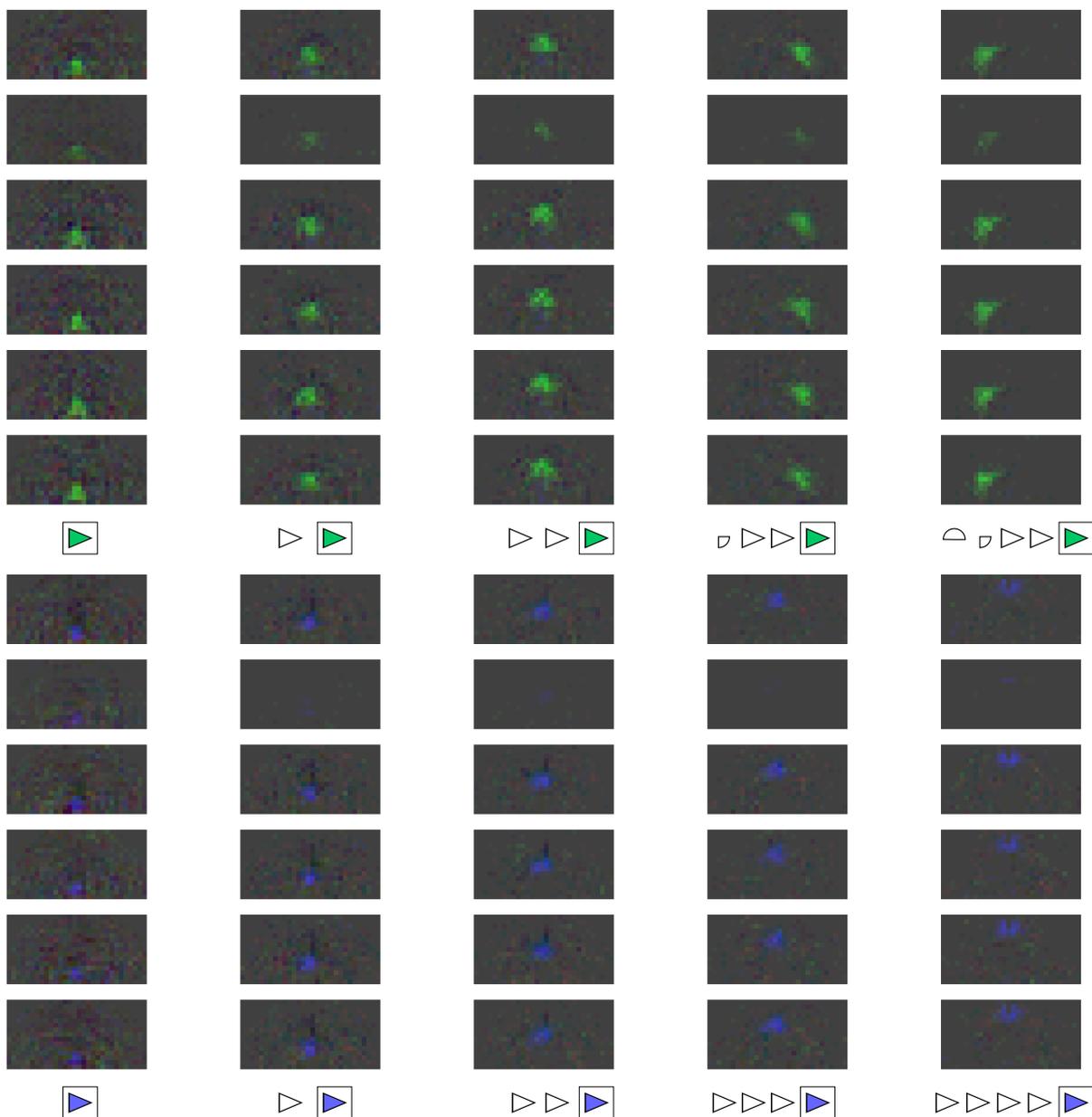


FIGURE 10.7 – Déplacements implicitement définis par les signatures des actes visuels. Un trait relie un acte visuel avec la position moyenne des actes désignés par sa signature avec un poids supérieur à 6 (si il y en a). Les graphes regroupent les actes visuels selon la couleur qui les caractérisent (vert, bleu, rouge), et l'acte primaire associée (de haut en bas, *avancer*, *manger*, *tourner à gauche de 90°*, *tourner à droite de 90°*, *tourner à gauche de 45°*, *tourner à droite de 45°*). Les transformations correspondent à une translation ou à une rotation de 90° ou de 45°. On peut noter que les signatures des actes basés sur l'acte primaire *manger* sont peu caractérisés spatialement. En effet, cet acte primaire est moins souvent éacté, ce qui réduit l'apprentissage des actes secondaires associés.

utilisée. La Figure 10.8 montre des exemples d'*image* de signature par une séquence d'interactions. Les signatures utilisées ont été obtenues après 50 000 cycles de décision. Les transformations consécutives conservent les propriétés des signatures d'origine, bien que des erreurs s'accumulent. Les erreurs sont plus nombreuses lorsque l'on utilise des actes visuels liés à des objets éloignés de l'agent : ces actes visuels sont en effet moins souvent énoncés, car les objets dont ils dépendent sont souvent masqués. Il en résulte des signatures moins fiables et pertinentes.

La possibilité de déplacer une signature d'interaction nous permet de formuler l'hypothèse de simplification suivante :

Hypothèse 10. *Une signature d'interaction peut subir une transformation géométrique dans la*



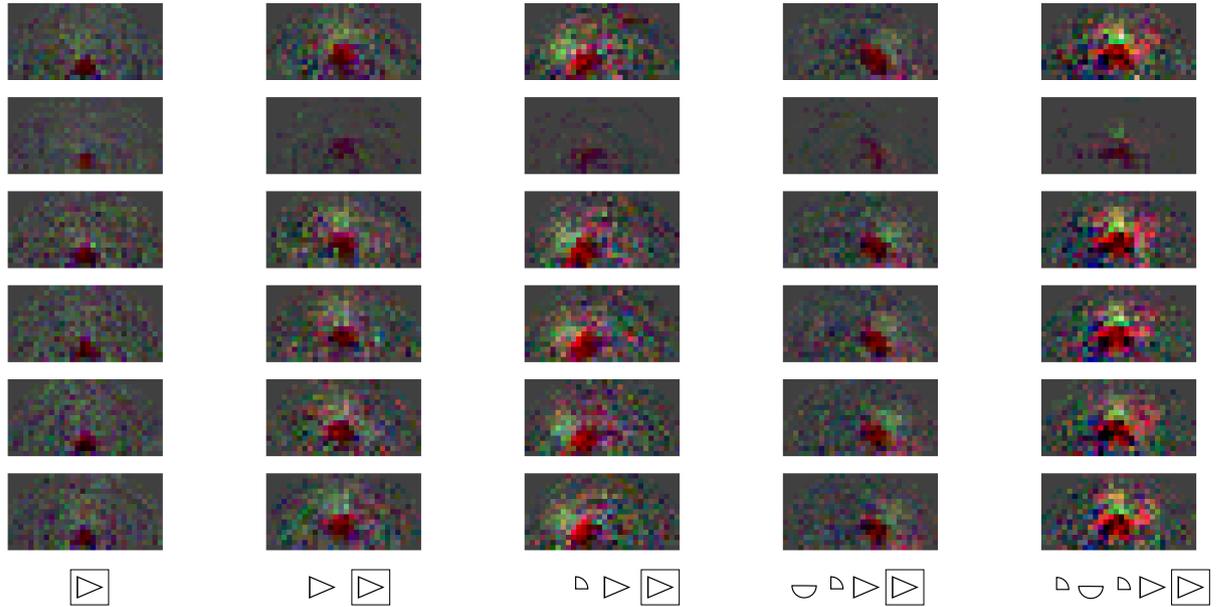


FIGURE 10.8 – Image de signatures par une séquence d'interactions. De haut en bas, signatures des interactions *se cogner*, *manger* et *avancer*. Chaque image de signature ajoute une transformation à la précédente. La séquence d'interactions associée à chaque image est affichée sous chaque image de signature. Nous pouvons voir que les propriétés des signatures d'origine (à gauche) sont conservées dans les images consécutives.

mémoire spatiale sans se baser sur des préconceptions.

Ces hypothèses seront utilisées dans les sections suivantes. Il faut rappeler que ces transformations ne font appel à aucun moment de la notion de l'espace : les signatures et images de signatures sont en effet constituées de listes. La position des poids n'est donc pas définie.

10.4.5 Effet prédictif des signatures d'interaction

En utilisant les signatures d'interaction, il est possible de prédire le contexte interactionnel que l'on obtiendrait après énonciation de chaque interaction. En effet, les signatures permettent de prédire le succès de chaque acte visuel, et donc de prédire le contexte que l'agent observerait si il énonçait l'interaction primaire associée à ces actes. Le contexte prédit E_{t+1} est composé des certitudes de succès de chaque acte dans ce contexte :

$$E_{t+1} = \{c(a_k, E_t)\}_{\forall a_k \in A} \quad (10.2)$$

Ce contexte prédit donne la prédiction de chaque acte, indépendamment de l'acte primaire associé. Pour obtenir le contexte que l'on obtiendrait en énonçant un acte primaire a , il faut limiter le contexte prédit aux actes secondaires associés à a , puisque seuls ces actes peuvent être énoncés si a est énoncé.

Comme le montre la Figure 10.9, les contextes prédits sont reconnaissables, et caractéristiques des environnements que l'on peut s'attendre à voir après énonciation d'une des interactions. Notons que certains actes qui dépendent d'éléments situés en dehors du champ visuel de l'agent peuvent être prédits avec une certitude non nulle. En effet, l'apprentissage des signatures peut prendre en compte des événements non liés si ils se produisent souvent.

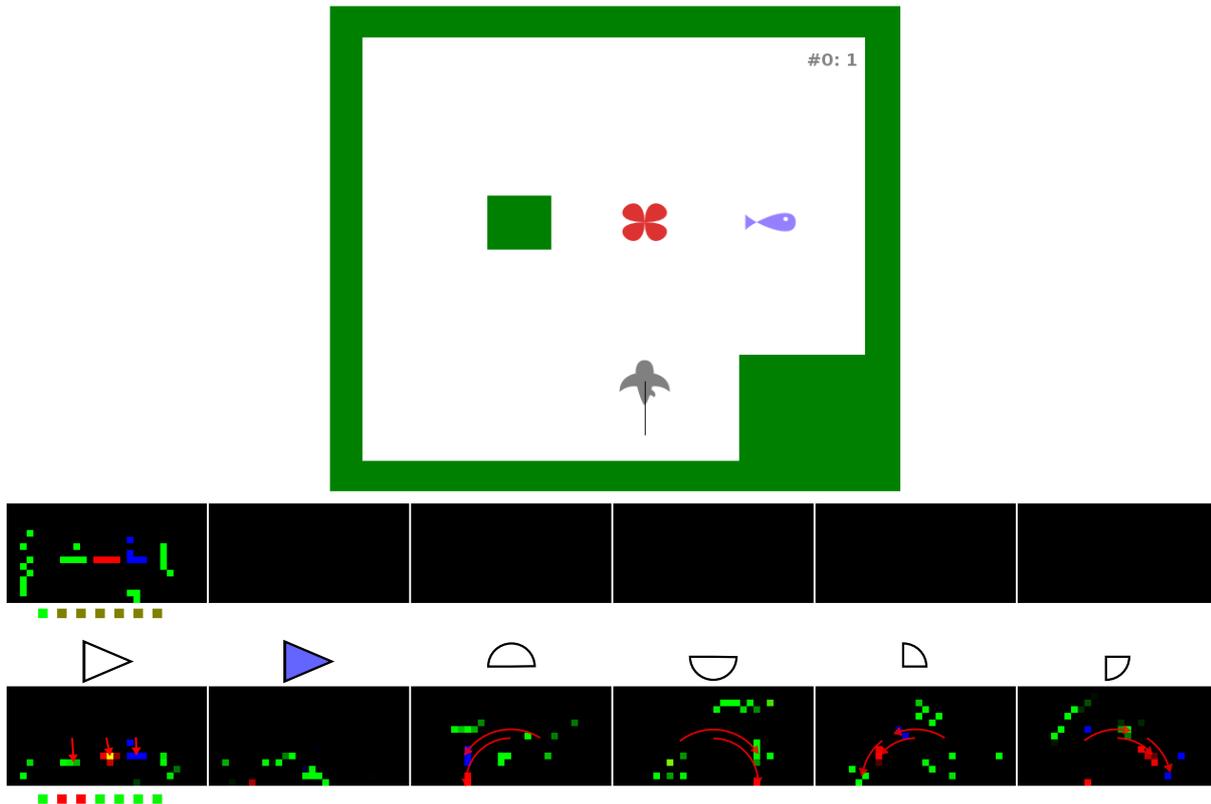


FIGURE 10.9 – Prédiction du contexte interactionnel au pas suivant. De haut en bas : l'agent dans son environnement, le contexte interactionnel courant E_t , le contexte interactionnel prédit E_{t+1} . Chaque groupe d'acte est associé à une interaction primaire différente, et constitue le contexte interactionnel prédit associé à l'énonciation de chaque interaction primaire (de gauche à droite : *avancer*, *manger*, *tourner à droite de 90°*, *tourner à gauche de 90°*, *tourner à droite de 45°*, *tourner à gauche de 45°*). Les flèches rouges comparent la position des groupes d'actes caractérisant la présence des objets du contexte courant et prédit.

10.4.6 Émergence de liens entre les actes

Nous avons vu que les signatures des actes visuels désignent des actes situés à proximité d'une même position de l'espace, indépendamment de l'acte primaire associé. En effet, ces actes permettent de détecter la présence d'un même élément à une même position de l'environnement. Les signatures permettent donc de regrouper les actes liés à un même objet. Il est intéressant de comparer la position des actes désignés par une même signature, afin d'observer l'émergence de

similarités entre les actes. Pour cela, je propose de représenter les actes sur un plan, sans tenir compte de leurs positions p . On rapproche les actes désignés par une même signature d'acte, en générant une force d'attraction entre deux actes désignés par une même signature. Cette force est proportionnelle au produit des valeurs absolues des poids de ces deux actes dans la signature qui les désignent. Afin d'identifier les groupes d'actes, nous fixerons les actes visuels associés à l'interaction *avancer* sur leurs positions réelles, afin de servir de référence. Un lien est tracé entre les actes dont les positions sont "voisines" dans l'espace, de façon à former un maillage représentant la topographie de l'espace (ce maillage n'a pas d'influence sur la position des actes sur la surface). La Figure 10.10 montre les positions des autres actes. Comme nous pouvons le voir, un grand nombre d'actes retrouvent leur position réelle (par rapport aux actes associés à *avancer*), ce qui montre qu'ils se regroupent en fonction de leur position. La Figure 10.11 montre plus en détail le maillage obtenu avec un des groupes d'actes. Les signatures permettent ainsi de regrouper les actes liés à une même position, sans qu'il ne soit nécessaire de connaître *a priori* les positions de ces actes.

Ces observations tendent à montrer que la structure de l'espace n'a pas besoin d'être connue ni représentée : en effet, les signatures permettent de regrouper les actes associés à un même objet localisé à une même position de l'espace, et relient ces actes par des liens caractérisant le déplacement produit par une interaction.

10.4.7 Utilisation des signatures pour compléter le contexte interactionnel

Les signatures d'interactions permettent de compléter le contexte interactionnel de deux façons. La première consiste à exploiter les propriétés prédictives vues précédemment, non pas sur le contexte courant E_t , mais sur le contexte précédent E_{t-1} . La prédiction permet ici de comparer le contexte réellement expérimenté par l'agent avec ce qui était prédit au cycle précédent. Si le contexte prédit E'_t contient des informations suffisamment fiables qui n'entrent pas en conflit avec les informations contenues dans le contexte réellement expérimenté E_t , alors ces informations peuvent être ajoutées au contexte E_t . Ceci peut arriver notamment quand un objet se retrouve masqué par un autre. Habituellement, lorsque l'agent se déplace, les éléments autour de lui se déplacent par rapport à lui en fonction de la position où ils ont été détectés au cycle de décision précédent. Ces déplacements conduisent à l'apprentissage de régularités dotées de fortes certitudes. Si l'un des objets se retrouve masqué, la régularité peut être utilisée pour prédire la nouvelle position de l'objet d'après ses mouvements habituels.

Nous pouvons observer cet effet mémoire en masquant un élément de l'environnement. Dans la Figures 10.12, une proie est masquée. Cependant, la prédiction d'après le contexte précédent permet à l'agent de conserver la présence de la proie et de la considérer dans son comportement.

La seconde méthode permettant de compléter le contexte interactionnel se base sur le lien entre les signatures et les objets. Comme nous l'avons vu précédemment, les signatures des actes visuels peuvent désigner plusieurs actes, caractérisant un même élément, mais associées à des interactions primaires différentes. Si un acte est considéré comme possible, alors l'objet

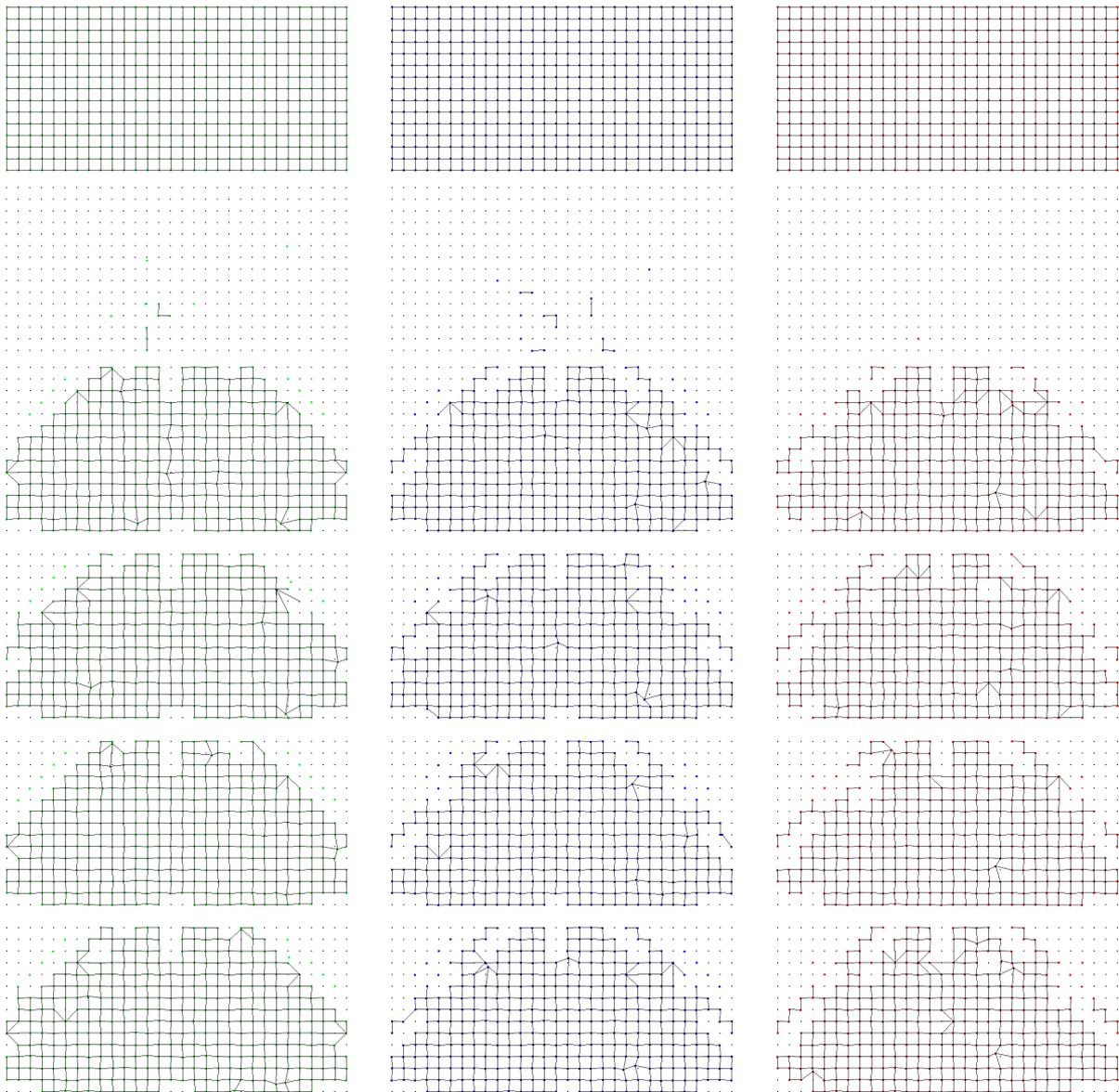


FIGURE 10.10 – Regroupement des actes désignés par une même signature. Les actes désignés par une même signature sont attirés avec une force proportionnelle au produit de leurs poids dans cette signature. Les actes associés à l'interaction *avancer* sont fixes et servent de référence. Chaque graphe affiche un groupe d'actes. De haut en bas, position des actes associées à l'interaction *avancer* (positions de référence), *manger*, *tourner à droite de 90°*, *tourner à gauche de 90°*, *tourner à droite de 45°*, *tourner à gauche de 45°*. Nous pouvons voir qu'un grand nombre d'actes sont regroupés à une position correspondant à leur position réelle (par rapport à la référence). Bien que des erreurs soient présentes, le maillage de référence émerge dans cette représentation. La portée de l'espace couvert par le maillage est limité à l'espace que l'agent peut expérimenter.

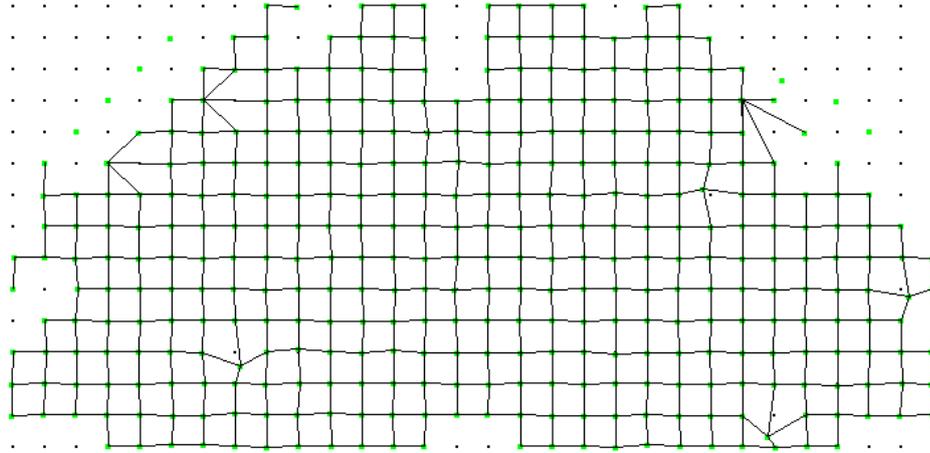


FIGURE 10.11 – Maillage obtenu par regroupement des actes visuels associés à l'acte *tourner à droite de 90°*, et permettant de détecter un objet vert. Le maillage montre que les actes se regroupent en fonction de leur position : chaque acte est attiré par un acte visuel associé à l'acte *avancer* (dont la position est fixée sur le plan), et partageant une même position p , ce qui permet de retrouver le maillage de référence.

qui l'afforde est présent. Ainsi, tous les actes permettant de percevoir cet objet peuvent être considérés comme énoncés au cycle de décision t .

Il est donc possible de définir un contexte augmenté \tilde{E}_t à partir des signatures des actes a_k reconnus comme énoncés d'après le contexte E_t :

$$\tilde{E}_t = \sum_{a_k \in A} S_{a_k} \times c(a_k, E_t) \quad (10.3)$$

Dans la Figure 10.13, nous reconstruisons un tel contexte augmenté. Comme nous pouvons le constater, le contexte est reconnaissable pour chaque ensemble d'interactions secondaires. Notons toutefois que les actes associés à l'interaction *manger* ne forment pas le contexte à cause de leur faible fiabilité.

Ces observations permettent de formuler l'hypothèse de simplification que nous utiliserons dans la section suivante :

Hypothèse 11. *Lorsqu'un acte est énoncé avec succès, nous pouvons considérer que l'ensemble des actes caractérisant un même élément et situé à une même position p peuvent également être considérés comme énoncés.*

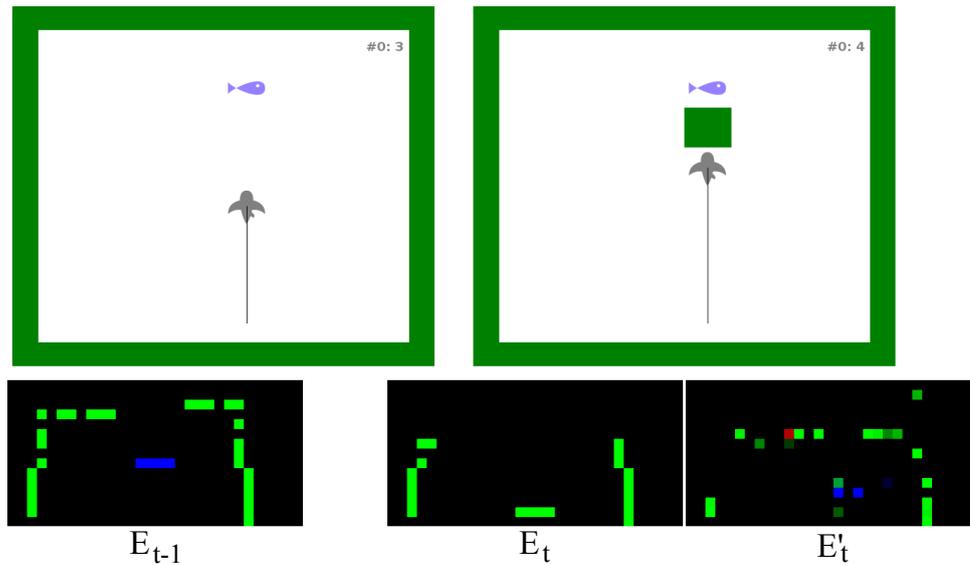


FIGURE 10.12 – Effet mémoire. En haut, l'agent dans son environnement. À droite, un bloc mur est ajouté entre l'agent et la proie, afin de masquer celle-ci. En bas, les contextes interactionnels du cycle de décision précédent E_{t-1} , le contexte courant E_t et le contexte E'_t prédit à partir du contexte E_{t-1} . Les contextes représentés ici se limitent aux actes associés à l'interaction *avancer d'un pas* (aucun des autres actes visuels n'est énéacté). Le contexte prédit E'_t contient des actes visuels liés à la présence de la proie (tache bleue), alors que celle-ci n'apparaît pas dans le contexte courant E_t .

10.5 Test d'un mécanisme d'exploitation de la Mémoire Spatiale

10.5.1 Simplifications du système de test

Cette expérimentation se concentre sur l'exploitation de la mémoire spatiale. Je propose une simplification sur le système interactionnel pour réduire le temps d'apprentissage. Cette simplification se base sur l'hypothèse 11 définie précédemment. Nous avons en effet observé qu'après apprentissage des signatures des actes visuels, les actes caractérisant un objet d'une même couleur à une même position, mais associés à des interactions primaires différentes, peuvent être considérés comme énéactés simultanément. Ainsi, je propose de regrouper les actes visuels caractérisant une même couleur localisée à une même position de l'espace. Nous réduisons ainsi significativement le nombre d'entrées du contexte interactionnel, et donc le temps d'apprentissage. Notons cependant que nous ne pourrions plus définir les signatures des actes visuels, puisque le résultat ne dépend plus de l'action de l'agent, ni utiliser ces actes dans le système de décision. Ce second point est toutefois sans conséquence sur notre système, car la valeur de satisfaction des actes visuels a été définie à 0.

La réduction du nombre d'entrées du contexte interactionnel permet d'augmenter la résolution du découpage de l'espace observable. Un nouveau découpage de 100 x 50 positions est proposé. Cette augmentation de la résolution permet une meilleure précision des éléments contenus dans

la mémoire spatiale, ainsi que des signatures d'actes. La mémoire spatiale est définie comme un ensemble de 100 x 100 positions. Dans le cas du robot, le système interactionnel définit un ensemble de 97 x 57 positions. Ce nombre de positions est contraint par le système visuel du robot. La mémoire spatiale est une structure qui prolonge cet ensemble de positions observables pour définir un ensemble de 97 x 114 positions. Les liens entre les positions sont définies comme pour l'agent simulé. Il faut toutefois noter que les déplacements du robot sont loin d'être aussi précis que ceux de l'agent simulé, ce qui rend les déplacements en mémoire spatiale approximatifs. Nous verrons cependant que cette imprécision n'a finalement que peu d'impact sur son comportement.

Je propose également de limiter l'ensemble des transformations T aux translations. Cette simplification est possible car les éléments disposés dans l'environnement ne disposent pas d'une orientation particulière. Nous ne considérerons, comme orientation des signatures, que l'orientation de l'agent. Cette simplification réduit considérablement le nombre de transformations à tester, et évite de devoir calculer la rotation d'une signature. L'espace des transformations T se résume donc à un espace à deux dimensions, et une transformation géométrique τ à un couple

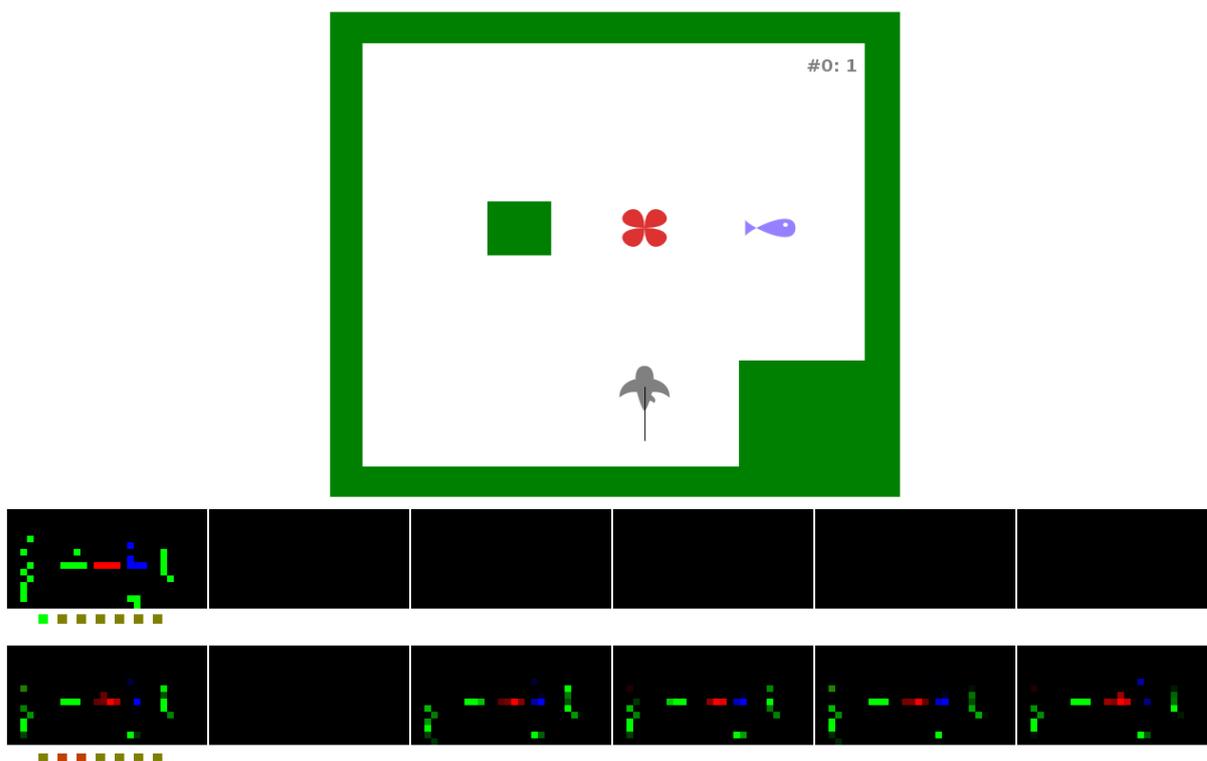


FIGURE 10.13 – Utilisation des signatures d'acte pour compléter le contexte interactionnel courant. En haut, l'agent dans son environnement. En bas, le contexte interactionnel courant E_t et le contexte augmenté \tilde{E}_t . Pour faciliter la lecture, nous ne tiendrons pas compte des prédictions négatives des actes visuels. Nous pouvons constater que pour chaque groupe d'acte visuel associé aux interactions primaires (excepté pour l'interaction *manger*, les actes visuels associés n'étant pas assez fiables), le contexte est reconnaissable.

(x, y) caractérisant la translation.

Le système de test est identique au système utilisé précédemment. Cependant, comme les actes visuels sont regroupés par position et couleur, et énoncés simultanément, l'agent ne peut pas les sélectionner pour les énoncer. L'expérimentation se déroule de la façon suivante : dans un premier temps, on laisse l'agent évoluer dans l'environnement, et apprendre ses signatures d'actes. Lorsque les signatures n'évoluent plus, et que le comportement de l'agent se stabilise, le comportement de l'agent est testé dans différentes situations permettant d'observer les différentes propriétés offertes par la mémoire spatiale. Ces expériences sont menées à la fois sur l'agent simulé et sur le robot.

L'agent est mu par deux mécanismes de sélections, selon le même principe que ceux utilisés pour la mémoire péri-personnelle. Un mécanisme d'apprentissage permet la vérification des signatures qui ne permettent pas de déterminer le résultat de l'acte associé avec une certitude suffisante. Ce mécanisme sélectionne l'acte pour lequel la certitude de succès est la plus faible, si celle-ci est inférieure à un certain seuil. Un seuil de 0.8 a été choisi.

Le mécanisme d'exploitation sélectionne l'acte, parmi ceux considérés comme possibles, qui dispose de la valeur de satisfaction globale la plus élevée. Ce mécanisme évalue le rapprochement global généré par chaque acte candidat, c'est-à-dire dont la certitude de succès est positive.

Les coefficients utilisés pour nos expérimentations ont été définis d'après des tests préliminaires. Ces coefficients permettent en effet d'obtenir des comportements permettant d'analyser comment l'agent exploite sa mémoire spatiale. Les expérimentations ont été effectuées avec les coefficients suivant :

-coefficient d'apprentissage α du réseau de neurone définissant les signatures d'interaction : 0.01 sur le système simulé, 0.02 sur le robot. Ce coefficient est plus élevé sur le robot pour réduire le temps d'apprentissage, l'autonomie de ce dernier ne permettant pas d'effectuer un apprentissage aussi long (en nombre de cycle de décision) que celui de l'agent simulé.

-coefficient d'influence γ : 0.4 pour l'agent simulé, 0.2 pour le robot. Cette différence est liée à la taille de leurs environnements respectifs : l'environnement du robot étant plus petit (par rapport à la distance parcourue dans ses déplacements), les objets sont plus proches d'un point de vue interactionnel.

-coefficient d'influence de la mémoire spatiale β : 0.05 pour l'agent simulé, 0.1 pour le robot.

L'environnement contient initialement un grand nombre de proies (jusqu'à 6 dans l'environnement simulé), de façon à augmenter les chances que l'agent interagisse avec l'une d'elles. En effet, l'agent ne connaît pas, initialement, les objets qui affordent ses actes, et ne cherchera donc pas à atteindre les proies. L'agent commence par tester ses actes dans des contextes environnementaux variés. Peu à peu, les signatures émergent et se stabilisent. L'agent commence à se diriger vers les proies. On réduit progressivement le nombre de proies, jusqu'à ce qu'il n'en reste plus que deux. Ceci permet de s'assurer que la capture des proies n'est pas liée au hasard.

Au bout de 2000 cycles de décision (1000 dans le cas du robot), le comportement devient stable : l'agent se dirige vers les proies et évite les murs. Le système d'apprentissage n'est plus

utilisé que ponctuellement, et une désactivation temporaire du mécanisme d'apprentissage ne perturbe pas le comportement de l'agent. Il faut cependant noter que ce mécanisme est parfois nécessaire : en effet, il arrive que l'agent se retrouve dans une situation où les actes *avancer*, *se cogner* et *manger* sont tous les trois considérés comme impossibles, avec, à chaque fois, l'un d'entre eux prédit avec une certitude très faible (en valeur absolue), mais négative. L'agent, qui cherche à se rapprocher d'une proie, alterne les rotations à droite et à gauche. Seul le mécanisme d'apprentissage lui permet de tester l'acte prédit comme un échec avec une faible certitude.

10.5.2 Analyse des signatures des actes

Les signatures des actes primaires sont affichées dans la Figure 10.14. On constate que les objets qui affordent les actes et leurs propriétés sont clairement considérées par l'agent. Les signatures sont similaires à celles observées dans l'expérience précédente. Les actes *avancer*, *se cogner* et *manger* sont tous les trois associés à un objet situé devant l'agent, de dimensions comparables à l'agent et d'une couleur qui caractérise le type de l'objet : vert pour *se cogner*, bleu pour *manger* et absence de vert et de bleu pour *avancer*. Notons également que la signature de l'acte *manger* comporte de petites régions liées à l'absence de vert (murs) de part et d'autre de la tache bleu caractérisant la proie : en effet, la présence partielle d'un mur entre l'agent et une proie située devant lui provoquera l'échec de cet acte. L'absence de mur à proximité de l'agent est donc perçue comme partie intégrante de l'objet qui afforde l'acte *manger*. Ceci montre que les objets intégrés par l'agent diffèrent des objets définis par un observateur externe. Nous pouvons également noter la faible influence des algues sur les prédictions : leur présence est en effet équivalente à l'absence de mur et de proie. Cependant, cette influence n'est pas nulle, et nous pouvons voir que l'algue afforde l'acte *avancer* (tache rougeâtre sur la signature de cet acte). Les actes *tourner* sont, quant à eux, fortement associés au biais de leur signature. Ces actes ne sont ainsi pas liés à un objet de l'environnement, mais à l'agent lui-même. On notera cependant que ces signatures sont liées, faiblement, à la présence d'un mur à proximité. L'agent a en effet tendance à tourner quand il se trouve face à un mur, ce qui explique que ce contexte soit considéré comme propice à l'énaction de ces actes.

Les signatures obtenues avec le robot sont proches de celles obtenues avec le système simulé : on reconnaît bien les caractéristiques des objets. En revanche, on peut observer une particularité : nous avons vu précédemment que l'agent pouvait voir une partie de son "corps" dans son champ visuel. Or, nous pouvons constater que la signature est vide dans la région correspondant à l'image de cette partie du corps. Ceci est dû au fait que la présence du corps de l'agent n'influe pas sur le résultat des actes. L'agent néglige ainsi la partie visible de son corps parce que non pertinente pour prédire le résultat de ses actes, un peu comme nous "oublions" la présence de notre nez, alors qu'il se trouve dans notre champ visuel. La différence avec les signatures obtenues avec le système simulé, utilisant pourtant le même système de décision, montre que le corps de l'agent a une grande influence sur la façon dont il construit ses objets au travers de ses actes.

Une fois les signatures apprises, un ensemble de tests est mis en œuvre pour évaluer les



FIGURE 10.14 – Signatures des sept actes primaires obtenues avec l'agent simulé au pas 2000 (en haut) et avec le robot au pas 1000 (en bas). Les poids associés aux actes visuels sont organisés topographiquement. Les huit carrés affichent les poids liés aux sept actes primaires et le biais de la signature. De gauche à droite, les signatures sont affichées dans l'ordre donné par le tableau 10.1. La signature de l'acte *avancer* définit un objet caractérisé par l'absence de vert et de bleu sur une région devant l'agent et d'une surface à peu près équivalente à celui-ci. L'acte *se cogner* est liée à un objet vert, et l'acte *manger* à un objet bleu. On remarquera que l'objet affordant l'acte *manger* est également caractérisé par l'absence de vert à proximité de l'agent. Les actes *avancer* et *manger* sont également liés à l'absence de l'acte *se cogner* dans le contexte interactionnel, tandis que l'acte *se cogner* est lié à lui-même. Les actes *tourner* sont quant à eux fortement liés au biais, caractérisant le fait que ces actes soient toujours possibles. Dans les signatures définies par le robot, on peut constater que la partie visible du robot (la caméra) n'est pas prise en compte dans les signatures, alors qu'elle est dans son champ visuel.

possibilités offertes par l'utilisation de la mémoire spatiale, c'est-à-dire la prise en compte du contexte environnemental dans son ensemble et les capacités d'adaptation à des changements environnementaux.

10.5.3 "Interprétation" des éléments disposés dans l'environnement

Lorsque l'agent a appris les signatures de ses actes, il se dirige vers les proies, qu'il considère comme attractives. Nous utilisons cette propriété pour amener l'agent à interagir avec les autres types d'éléments de l'environnement, et observer comment il interprète ces éléments par la façon dont il interagit avec eux. Dans cette expérience, nous proposons de masquer une proie vers laquelle l'agent se dirige, avec un autre élément, opaque, et d'observer le comportement de l'agent en présence de ce nouvel élément. Cette expérience est menée à la fois sur l'agent simulé et sur le robot. Étant donné que la mémoire spatiale utilisée implémente la topologie de l'espace et les déplacements des objets, on s'attend bien sûr à ce que l'objet masqué reste en mémoire, garantissant ainsi la permanence de l'objet. Nous pouvons cependant étudier comment l'agent génère ses comportements en présence de plusieurs objets présents dans sa mémoire spatiale.

Dans un premier temps, on laisse l'agent se diriger vers une proie, objet très attractif car affordant un acte doté d'une grande valeur de satisfaction. Avant qu'il ne l'atteigne, une algue et un mur sont tour à tour placés entre l'agent et la proie, de façon à ce qu'il ne puisse plus voir la proie, ni l'atteindre sans interagir avec le nouvel objet ou sans modifier sa trajectoire.

Dans le cas d'une algue, l'agent ne change pas sa trajectoire (Figure 10.15). En effet, l'algue permet d'énacter l'acte *avancer d'un pas*, qui permet de se rapprocher de la proie. Ceci montre que l'agent néglige cet élément, car il n'affecte pas les actes énoncés. L'agent interprète ainsi l'algue de la même façon qu'un espace vide.

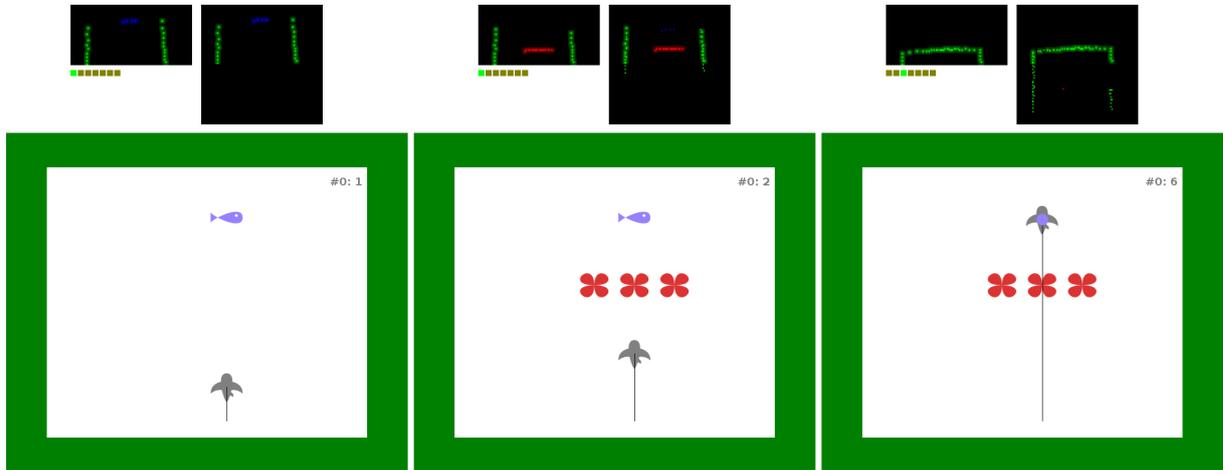


FIGURE 10.15 – Proie masquée par une algue. En haut, le contexte interactionnel et le contenu de la mémoire spatiale. En bas, l'agent dans son environnement. Après ajout de l'algue, l'agent continue de se diriger vers la proie sans tenir compte de l'"obstacle", celui-ci étant invisible d'un point de vue interactionnel, car affordant le même acte qu'un espace vide.

Dans le cas d'un mur, par contre, l'agent contourne l'obstacle (Figure 10.16). Ceci montre que l'agent considère les murs comme des éléments négatifs dont il doit s'éloigner. Cette expérience illustre le fonctionnement de la mémoire spatiale : au début, l'agent est fortement attiré par la proie, et modérément repoussé par le mur. Au fur et à mesure que l'agent s'approche du mur, son influence devient de plus en plus importante, jusqu'à ce que la valeur d'utilité attribuée à l'acte *avancer* rende cet acte moins intéressant qu'un acte *tourner*, qui permet à l'agent de s'éloigner du mur. L'agent contourne ensuite le mur, avant de se diriger à nouveau vers sa proie. Il faut noter que dans un premier temps l'agent tourne du côté qui permet d'éloigner le plus le mur, même si cela l'éloigne de la proie, mais choisit ensuite de le contourner par le côté qui laisse l'ouverture la plus large. Le mécanisme de la mémoire spatiale donne en effet une plus grande influence aux objets proches, avec lequel l'agent a le plus de chance d'interagir dans un avenir proche. Cette règle simple permet de générer des comportements adaptés au contexte, sans toutefois être optimal.

L'agent montre, par ses comportements, qu'il interprète les éléments qui composent son environnement, et peut tenir compte de plusieurs éléments contenus dans sa mémoire spatiale. Ces comportements ont été également observés avec le robot : la Figure 10.17 montre que le comportement de l'agent, face à un mur, est identique à celui de l'agent simulé, et ce, malgré la faible précision de la mémoire spatiale.

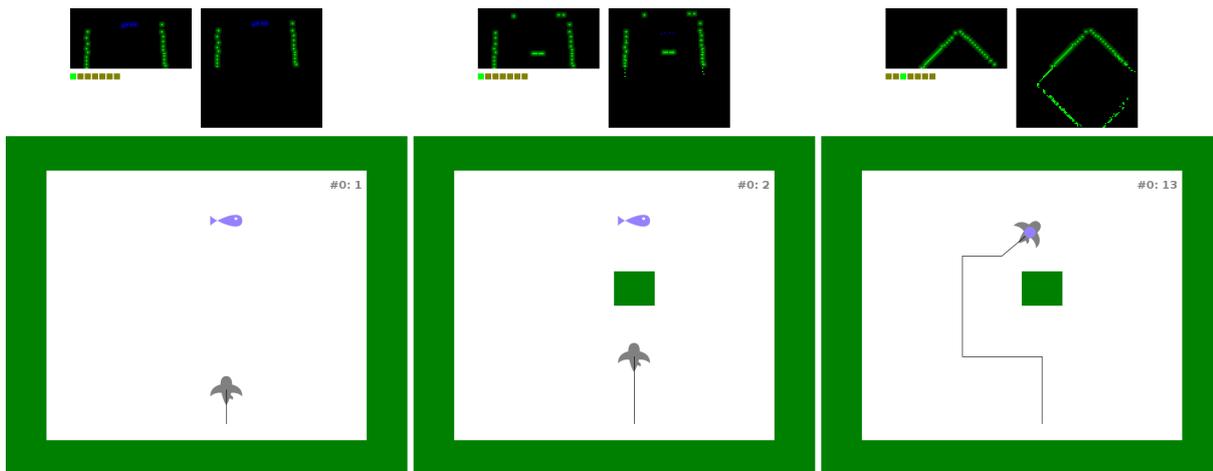


FIGURE 10.16 – Proie masquée par un mur. L'agent contourne l'obstacle, car celui-ci afforde un acte doté d'une valeur de satisfaction négative. En restant éloigné du mur, l'agent génère un comportement qui le pousse à contourner l'obstacle pour atteindre la proie dont il cherche à se rapprocher.

10.5.4 Influence des coefficients d'influence des objets et de la mémoire spatiale

Nous étudions ici l'impact du coefficient d'influence des objets γ , caractérisant l'attractivité des instances d'objets en fonction de leur distance, et du coefficient d'influence de la mémoire spatiale β sur les comportements de l'agent.

Le coefficient d'influence des objets est peut-être celui qui a le plus d'influence sur le comportement de l'agent. Si le coefficient est trop petit, l'influence des objets ne décroît pas assez vite avec la distance. De ce fait, l'influence des objets lointains peuvent avoir une influence supérieure à celle des objets plus proches. Dans le cas de notre système de test, l'agent reste au centre des régions vides et ne s'aventure pas dans une région contenant des murs à proximité, même si des proies sont présentes : les murs entourant l'environnement, même lointains, ont une influence répulsive trop importante pour que l'agent s'en approche. En revanche, si ce coefficient est trop grand, les objets distants auront une influence trop faible pour influencer les comportements de l'agent. Celui-ci ne tiendra alors compte que des objets les plus proches, voire uniquement des actes qu'il peut directement afforder. La Figure 10.18 montre le comportement de l'agent avec différents coefficients d'influences.

Le coefficient d'influence de la mémoire spatiale a un impact similaire, bien que moindre, sur le comportement : ce coefficient caractérise en effet l'influence des déplacements des objets provoqués par l'énaction des actes. Trop petit, les objets distants n'auront qu'une faible influence sur les décisions de l'agent. Trop grand, l'agent ne pourra pas s'éloigner des objets qui l'attirent, et peut rester bloqué face à un mur qui lui barre le chemin.

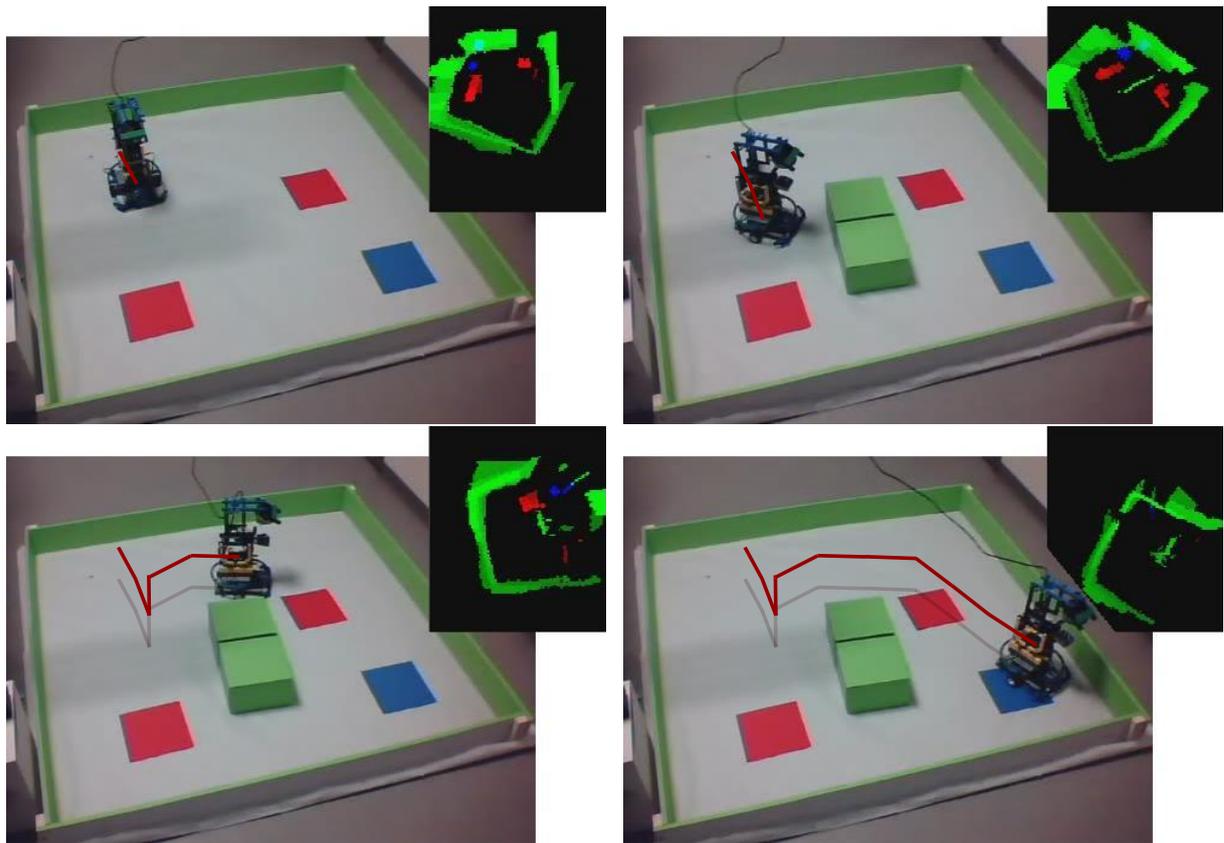


FIGURE 10.17 – Proie masquée par un mur dans le cas du robot. Comme dans le cas de l'agent simulé, le robot évite le mur qui apparaît devant lui pour atteindre la proie. Il passe également sur une algue sans en tenir compte.

10.5.5 Attractivité des objets en fonction de leur contexte

Cette expérimentation met en évidence l'influence des éléments de l'environnement sur l'attractivité des objets environnant. Elle montre également que l'agent n'a pas de but prédéfini. Au contraire, l'agent définit un but à atteindre, par le fait qu'il permettra, lorsque le contexte le permettra, d'énactier un acte doté d'une forte valeur de satisfaction. Pour cette expérience, deux proies sont disposées de façon symétrique à une certaine distance de l'agent. Autour de l'une des deux proies, un groupe de murs est ajouté, de façon à ce qu'ils soient visibles, sans toutefois masquer, même partiellement, la proie. On observe ensuite vers quelle proie l'agent se dirige.

On observe que, dans une grande majorité des cas, l'agent se dirige vers la proie qui n'est pas entourée de murs (Figure 10.19). La présence de ces objets a donc influencé la décision de l'agent, en ajoutant un poids négatif aux actes qui permettent de s'en rapprocher.

Nous pouvons remarquer que jusqu'à un certain point, l'agent se dirige vers une zone médiane entre les deux proies. L'agent est en effet attiré par les deux proies, qui peuvent toutes les deux se rapprocher en énantier l'acte *avancer d'un pas*. Au fur et à mesure que l'agent se rapproche des proies, cet acte devient de moins en moins intéressante, au profit des actes *tourner*. Arrivé

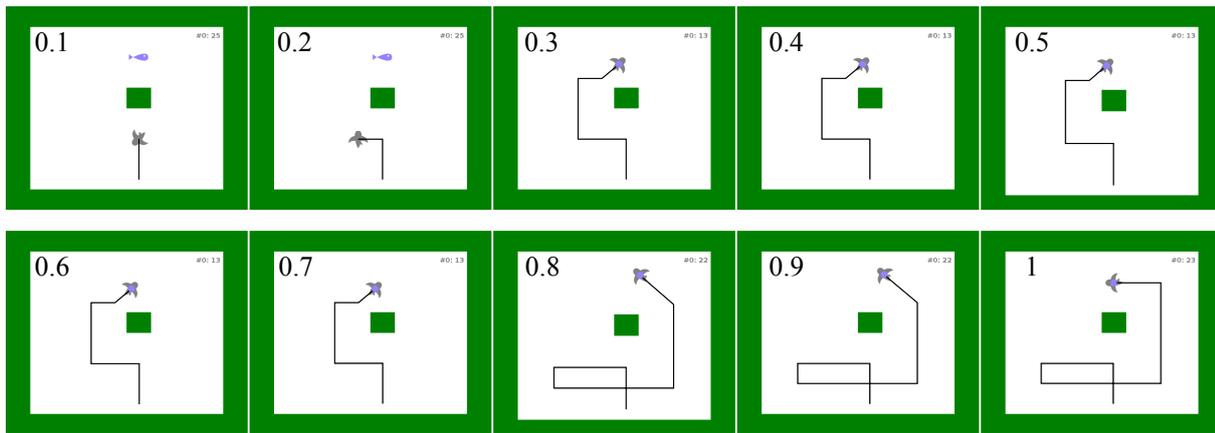


FIGURE 10.18 – Influence du coefficient γ sur le comportement de l'agent. De gauche à droite et haut en bas, le coefficient passe de 0.1 à 1 par pas de 0.1. Avec un coefficient trop faible (inférieur à 0.2), les objets, même lointains, ont une influence trop importante : l'agent ne s'aventure pas dans les régions où les murs sont trop rapprochés. Avec un coefficient de 0.3 à 0.7, l'agent contourne correctement l'obstacle. Avec un coefficient supérieur, la proie n'a plus assez d'influence. L'agent va alors circuler en se tenant à une distance raisonnable des murs, jusqu'à ce que la proie soit assez proche pour attirer l'agent. Notons que le chemin suivi montre que l'agent a une certaine tendance à éviter les murs en tournant à gauche, sans doute à cause d'une asymétrie dans ses signatures d'interaction.

à une certaine distance des proies, l'un des actes *tourner* devient plus intéressante que l'acte *avancer*. La différence entre ces actes est liée à la différence d'attractivité des régions à droite et à gauche de l'agent, atteignables par les actes *tourner*.

On notera toutefois que si l'asymétrie entre les deux configurations n'est pas assez marquée, l'agent sélectionnera la proie qui se trouve dans un côté préférentiel, qui peut différer à chaque apprentissage des signatures. Cette préférence pour l'un des cotés s'explique par la présence d'une asymétrie dans les signatures, qui génère une asymétrie dans l'attractivité des proies, même si elles sont placées de façon symétriques par rapport à l'agent.

10.5.6 Définition d'un objet à partir de son contexte et adaptation de la définition des objets à un changement environnemental

Cette expérience teste la possibilité, pour l'agent, de définir des objets qu'il ne peut détecter qu'indirectement, au travers d'un contexte qui caractérise leurs présences. Certaines propriétés des proies sont modifiées : elles sont maintenant invisibles et persistent quand l'agent interagit avec elles. Les proies sont placées à une distance constante d'algues, qui serviront de repères. Comme les proies sont persistantes, la mémoire spatiale n'efface plus son contenu autour de l'agent. Nous verrons que l'utilisation de cette propriété permet certaines observations sur la façon dont l'agent interagit avec les objets. Étant donné les dimensions de l'environnement et la

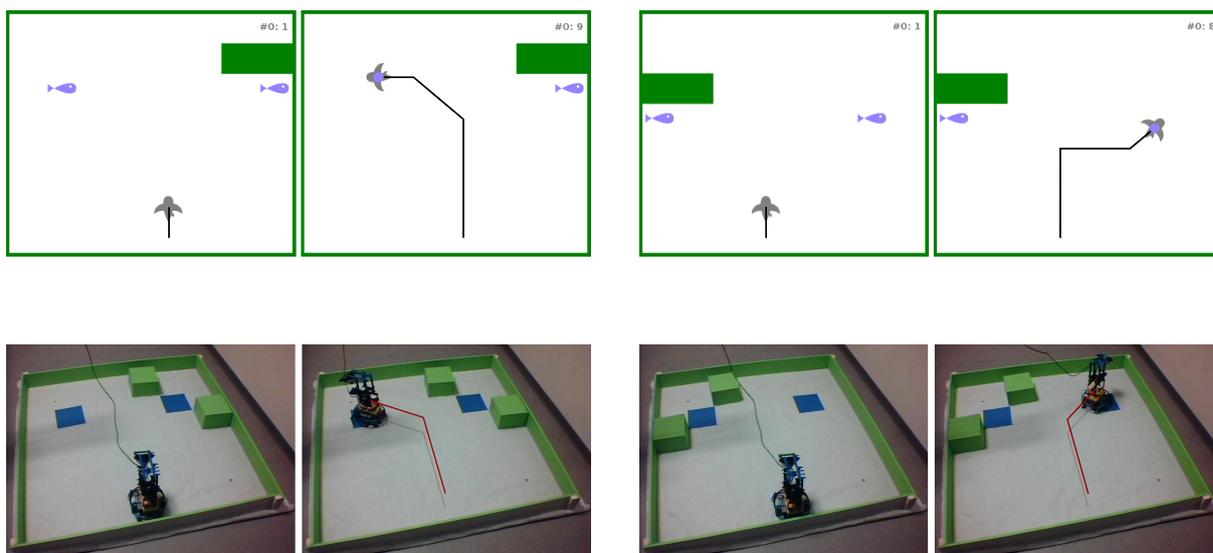


FIGURE 10.19 – Sélection d'une proie en fonction de son contexte, avec un agent simulé (en haut) et un robot (en bas). L'agent se dirige de préférence vers la proie qui n'est pas entourée de murs. On peut noter que l'agent se dirige vers le milieu des deux proies, jusqu'à ce qu'une différence dans l'attractivité des deux régions le pousse à choisir l'une des deux proies.

durée d'apprentissage, cette expérience sera seulement menée sur l'agent simulé.

On recommence l'apprentissage des signatures pour permettre à l'agent de définir les signatures de ses actes dans ce nouvel environnement. Il apparaît que l'apprentissage est plus long : il faut 5000 cycles de décision environ pour permettre un comportement stable. En observant la signature de l'acte *manger*, affiché dans la Figure 10.20, on se rend compte que l'agent a associé la présence d'une proie à la présence d'une algue à proximité (demi-cercle rouge sur la signature). Ainsi, l'objet qui afforde l'acte *manger* est composé d'un objet rouge à une certaine distance de l'agent. On observe également que l'acte *avancer* est liée à l'absence d'une algue à proximité de l'agent. Par contre, la signature de l'acte *se cogner* est similaire à celle observée précédemment.

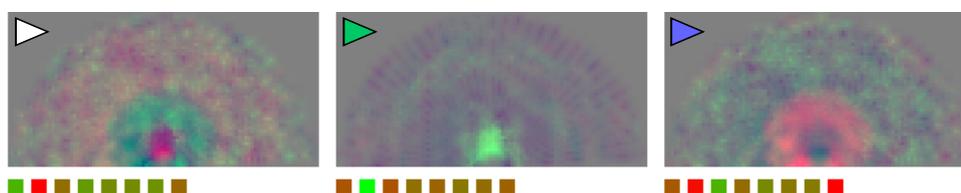


FIGURE 10.20 – Signatures des actes *avancer d'un pas*, *se cogner* et *manger une proie*. Nous pouvons voir que les objets des actes *avancer* et *manger* sont respectivement liées à l'absence et à la présence d'une algue (objet rouge) à une certaine distance, définie par le demi-cercle sur les signatures. La signature de l'acte *se cogner* est similaire à celle observée précédemment.

Ces signatures illustrent le fait qu'un objet est localisé par rapport à la position qui permet

d'énacter l'acte qu'il afforde : le demi-cercle rouge est en effet éloigné de la position de l'agent. Les instances de l'objet *manger* sont aussi localisées à une certaine distance des algues.

Le comportement de l'agent est également adapté à ces nouvelles proies : l'agent se dirige vers les algues de façon à passer à côté d'elles, et tourne autour d'elles, ce qui lui permet de *manger* des proies. Ce comportement montre que l'agent peut utiliser un contexte (ici, les algues) pour percevoir et interagir avec des objets qui échappent à son système sensoriel. La Figure 10.21 montre l'agent tournant autour d'une algue qui lui sert de repère.

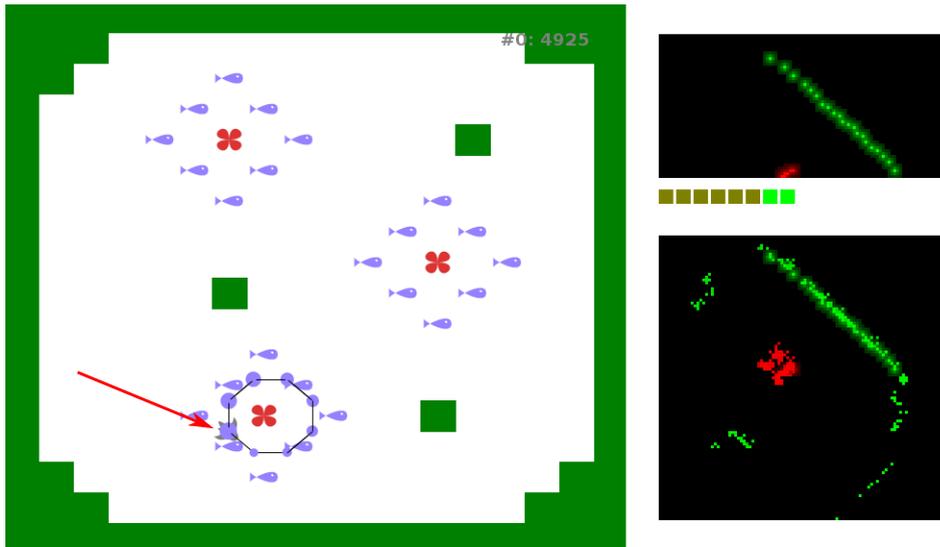


FIGURE 10.21 – Environnement de test pour l'expérience des proies invisibles. À droite, le contexte interactionnel et la mémoire spatiale. Après apprentissage des signatures, l'agent (en bas dans l'environnement) tourne autour d'une algue pour capturer les proies.

Afin d'étudier la façon dont l'agent se rapproche des instances d'objet, la fonctionnalité permettant d'oublier les éléments environnant quand l'agent mange une proie a été réactivée. Cette fonctionnalité n'est ici pas justifiée, puisque les proies ne disparaissent pas. Cependant, elle permet de pousser l'agent à quitter l'algue autour de laquelle il tourne, ce qui nous permet d'étudier la façon dont il s'approche des algues.

À chaque fois que l'agent *mange* une proie, il oublie l'algue située à proximité de lui, et se dirige vers l'algue la plus proche parmi celles qu'il peut voir (Figure 10.22). Il apparaît que l'agent s'approche des algues par le côté, montrant qu'il ne vise pas l'algue elle-même, mais la position à partir de laquelle la position de l'algue formera un contexte propice à l'énaction de l'acte *manger une proie*. Ce comportement lui permet de capturer un grand nombre de proies à chaque fois qu'il passe à proximité d'une algue. Ceci montre que l'agent localise bien une instance objet d'après la position qui permet d'interagir avec lui (dans notre cas, à côté de l'algue) et non de la position des éléments qui le caractérisent (un objet rouge).

Nous avons également effectué cette expérience en utilisant les signatures apprises avec des proies visibles, afin de tester la capacité de l'agent à s'adapter à de nouveaux objets. Dans un

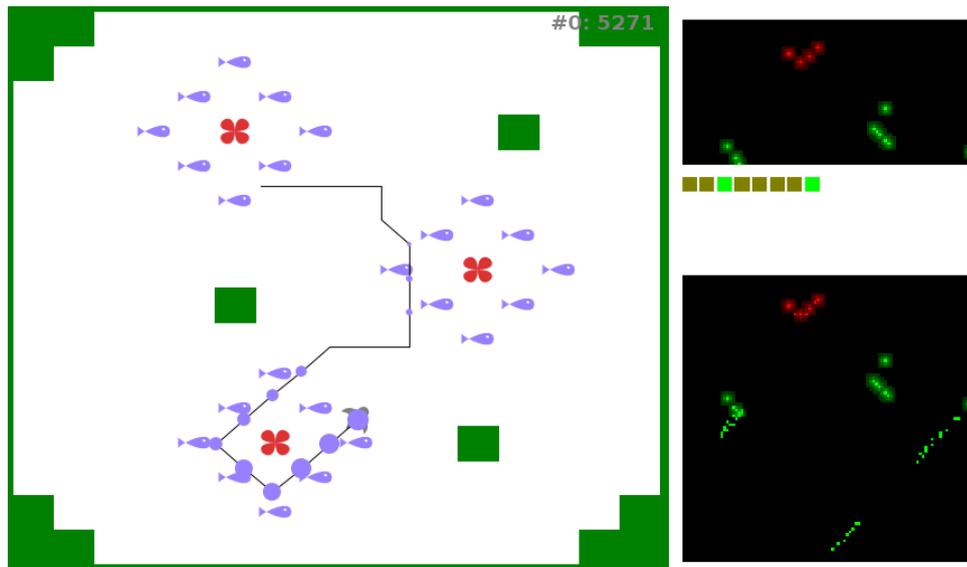


FIGURE 10.22 – Lorsque l'agent oublie l'algue autour de laquelle il tourne, il navigue d'une algue vers une autre. Nous pouvons observer que l'agent approche les algues par le côté, ce qui montre qu'il vise le centre d'un contexte défini par ses signatures, et non les éléments qui composent cette signature.

premier temps, l'agent agit comme si il n'y avait pas de proies, ce qui est normal puisqu'il ne voit pas les proies telles qu'il les a expérimentées. Il faut le "guider" à l'aide de blocs murs pour le faire interagir avec une proie. Au bout de quelques interactions avec les proies, le mécanisme d'apprentissage est à nouveau sollicité. Une fois les signatures à nouveau stabilisées, l'agent se comporte comme précédemment. L'apprentissage est toutefois moins long : 1500 cycles de décisions sont nécessaires. Cette rapidité vient probablement du fait que les signatures des autres actes étaient toujours valides dans cet environnement. En observant la signature de l'acte *manger*, affichée dans la Figure 10.23, on peut observer le demi-cercle caractérisant l'algue à une certaine distance de l'agent. Cependant, la tache bleue, caractérisant une proie visible, n'a pas disparu : l'agent n'a pas oublié les proies visibles.

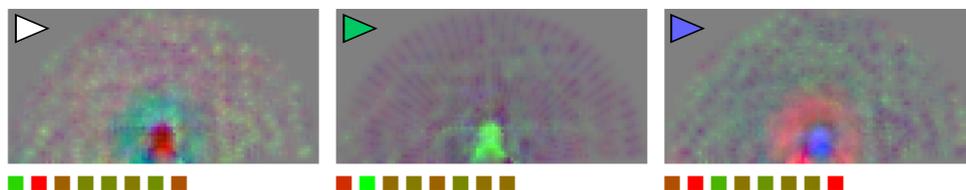


FIGURE 10.23 – De gauche à droite : signatures des actes *avancer*, *se cogner* et *manger*, issues de l'adaptation aux proies invisibles. La structure en arc de cercle apparaît sur les signatures des actes *avancer* et *manger*. Cependant, l'agent n'a pas oublié les proies bleues.

Cette dernière expérience montre que l'agent conserve, tout au long de sa "vie", la capacité à

s'adapter à des changements de son environnement. Cette capacité montre également que l'agent génère les modèles des objets en fonction de la façon dont il les expérimente par le biais de ses actes et de la façon dont il les considère. Si sa manière d'intégrer les objets est modifiée, les modèles des objets évoluent pour s'adapter.

10.6 Conclusion

Les expérimentations présentées dans ce chapitre ont permis d'étudier plus spécifiquement les propriétés des signatures d'interactions. Nous avons en effet pu observer que ces signatures caractérisent les objets qui affordent les interactions de l'agent d'après la façon dont il les expérimente. Ces signatures permettent également de constituer implicitement des *bundles* d'interaction permettant d'une part de regrouper les interactions affordées par un même objet, et d'autre part de lier des positions de l'espace par des interactions. Ces capacités permettent de caractériser la structure de l'espace observable, c'est-à-dire l'espace que l'agent peut expérimenter au travers de ses interactions, et de compléter le contexte interactionnel de l'agent.

Nous avons utilisé une mémoire spatiale codée en dur pour étudier les mécanismes d'exploitation de cette mémoire. Les expérimentations présentées dans ce chapitre ont montré que seules la distance et l'interaction permettant de se rapprocher d'une instance d'objet étaient nécessaires pour caractériser sa position, et permettre à l'agent de générer des comportements lui permettant de satisfaire sa motivation interactionnelle. Les observations de ces expérimentations ont été utilisées pour développer les mécanismes de construction de la mémoire spatiale (décrits au chapitre 6).

Chapitre 11

Expérimentation d'un dispositif de mémoire extra-personnelle : étude des mécanismes de construction d'une mémoire agnostique

Ce chapitre décrit les expérimentations menées pour tester et valider les mécanismes permettant de construire une mémoire spatiale, dans un environnement de test similaire à celui utilisé pour les expérimentations décrites au chapitre 10. Ces expérimentations ont pour but de démontrer que nos mécanismes permettent bien à l'agent de construire une structure caractérisant l'espace qui l'entoure sans se baser sur des préconceptions, en lui permettant de localiser et suivre les instances d'objet dans l'espace. Ces expérimentations doivent également montrer que les comportements de l'agent restent cohérents avec ceux observés avec une mémoire spatiale *codée en dur* (cf chapitre 10) de façon à montrer que la mémoire spatiale construite par l'agent dispose de propriétés similaires à celles de la mémoire *codée en dur*. Les dispositifs de test tiennent compte de simplifications basées sur les propriétés observées avec les expérimentations décrites au chapitre 10. Ces simplifications sont raisonnables car basées sur des propriétés qui ne dépendent pas de la mémoire spatiale.

11.1 Description du dispositif de test

Les mécanismes de la mémoire spatiale agnostique ont été testés sur un système simulé proche du système utilisé dans le chapitre précédent. Étant donnée la durée d'apprentissage des signatures de lieu et de présence, ces mécanismes n'ont pas été testés sur un robot. On utilise le même environnement, et le même ensemble d'interactions primaires, donné par la Table 10.1. Cependant, ce système diffère par l'absence de structure implémentant une mémoire de l'espace codée en dur. À la place, nous utilisons une structure simplifiée, limitée à l'espace observable, et

ne permettant pas le suivi des interactions préalablement énoncées. Cette structure s'inspire du *tectum optique* des vertébrés et génère une représentation topographique de l'espace observable, constitué de 100 x 50 positions dans lesquelles une interaction visuelle peut être énoncée. Les positions p de l'espace ne sont pas définies explicitement : chaque acte visuel est considéré comme une interaction primitive, indépendante des autres. Ce système se rapproche ainsi du modèle PRI.

Nous appliquons la simplification permettant de réduire le nombre d'interactions visuelles, fondée sur l'hypothèse 11 : nous ne considérons que trois types d'interactions visuelles, permettant de détecter une couleur parmi {rouge, vert, bleu}, en une position de l'espace, indépendamment de l'interaction primaire qui lui a donné naissance. Nous ne considérons donc que les sept interactions primaires.

Afin de simplifier l'analyse des résultats, nous proposons d'établir une relation entre les transformations produites par l'énoncé de séquences d'interactions et les transformations géométriques équivalentes. Nous limitons les transformations géométriques à des translations. Cette simplification est autorisée par le fait que les objets puissent être considérés indépendamment de leur orientation, et permet de considérer toute transformation géométrique comme un élément de \mathbb{R}^2 . Les différents paramètres liés aux transformations spatiales peuvent ainsi être affichés sur un plan. Cette correspondance entre les transformations liées aux séquences d'interactions et les transformations géométriques est définie *a priori* en utilisant les transformations théoriques des interactions primitives.

Les lieux sont définis grâce aux séquences d'interactions permettant d'atteindre les positions de l'espace observable qui les composent. Cependant, pour éviter d'utiliser des séquences trop longues, nécessaires pour atteindre certaines positions spatialement proches de l'agent, nous proposons une méthode fondée sur l'utilisation des signatures d'interaction et des images de signatures. Pour chaque signature d'interaction S considérée, chaque position p est considérée comme atteignable par la séquence dont l'image S' a le poids le plus élevé en p (en valeur absolue). La première interaction de cette séquence permet de définir l'interaction qui permet de se rapprocher le plus de p , et sa longueur caractérise sa distance. Les positions partageant des propriétés identiques forment alors un lieu. Afin de faciliter l'analyse des résultats, nous proposons d'utiliser la distance géométrique des positions plutôt que la longueur des séquences d'interactions qui permettent de les atteindre. Les lieux sont ainsi constitués de positions adjacentes, ce qui facilite la lecture des signatures de lieu.

Bien sûr, le découpage des lieux sera différent pour chaque signature. Cependant, comme les signatures des interactions primaires *avancer*, *se cogner* et *manger* désignent des objets dont la position et la taille sont similaires, les découpages seront suffisamment proches pour être considérés comme identiques. Nous utilisons, dans nos expérimentations, le découpage obtenu avec la signature de l'interaction *avancer*. Les découpages obtenus avec les signatures des interactions *avancer*, *se cogner* et *manger* sont montrés dans la Figure 11.1.

La longueur du path des lieux composites est limitée à 2. En effet, dans notre système, un path de longueur 1 est suffisant pour permettre l'observation de l'ensemble de l'espace global (H5) :

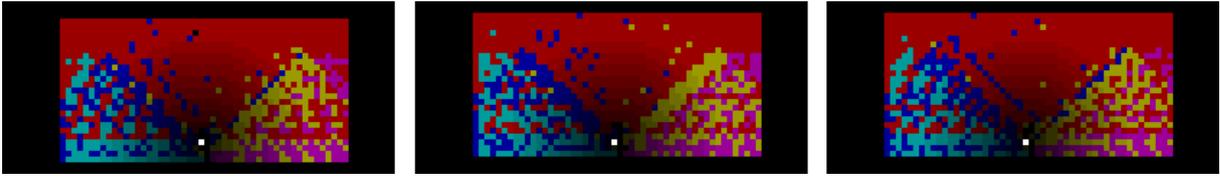


FIGURE 11.1 – Découpage des lieux primitifs obtenus avec les signatures des interactions *avancer* (à gauche), *se cogner* (au centre), *manger* (à droite). Le point blanc définit le centre (transformation neutre). Les couleurs montrent l'interaction qui permet de se rapprocher le plus de chaque point caractérisé par une transformation : rouge : avancer, bleu : tourner à gauche de 45° , jaune : tourner à droite de 45° , cyan : tourner à gauche de 90° , magenta : tourner à droite de 90° . Les espaces noirs sont des régions hors de l'espace observable ou pour lesquels aucune interaction ne permet de se rapprocher. Les dégradés montrent le découpage en fonction des distances. Le pas correspond à la distance parcourue par l'interaction *avancer d'un pas*. Comme nous pouvons le constater, les découpages sont similaires pour les trois interactions. Nous pouvons donc les considérer comme similaires et n'utiliser que l'un d'eux.

une rotation de 90° à droite ou à gauche permet, avec un champ de vision de 180° , de couvrir l'ensemble de l'espace. Avec un path de longueur 2, il devient possible de suivre, par mise à jour des lieux composites, toute position de l'espace global.

Comme pour le système utilisé pour tester la mémoire péri-personnelle, plusieurs niveaux de mémoires, caractérisant des niveaux de fiabilité différents, sont définies. Chaque objet est ainsi caractérisé par deux listes de lieux composites : la première ne relève, à la détection de l'objet, que les lieux composites dont la certitude dépasse un seuil élevé. La liste est ensuite mise à jour par suppression des premiers éléments du path quand ils sont éactés. Cette liste permet ainsi de définir un contexte de lieu fiable et donc favorable à un apprentissage. Cependant, la durée du suivi d'un objet ne peut pas excéder la longueur maximale du path des lieux composites. La seconde liste relève, à la détection d'un objet, les lieux composites reconnus avec une certitude plus faible que pour la première liste, afin de relever un plus grand nombre de lieux. La mise à jour de cette liste se fait par suppression des premiers éléments du path lorsqu'il est éacté, mais également par évocation des lieux par le biais des signatures de présence. Cette liste permet ainsi de suivre les objets jusqu'à épuisement des lieux composites. La fiabilité est bien sûr plus faible que pour la première liste.

11.2 Apprentissage des signatures

L'apprentissage nécessite d'apprendre les signatures d'interaction, de lieu et de présence. Étant donné que chaque type de signature dépend des précédents, ou d'informations obtenues par les précédents, je propose de segmenter l'apprentissage en en trois étapes :

Signatures d'interaction

Nous pouvons utiliser les signatures obtenues avec la mémoire codée en dur. En effet, ces signatures sont obtenues en se basant sur le contexte interactionnel et non sur la mémoire spatiale, et sont apprises grâce au système d'apprentissage, qui n'utilise pas la mémoire spatiale. L'obtention de ces signatures étant indépendant de la mémoire spatiale *codée en dur*, nous pouvons les réutiliser. Ces signatures permettent par la suite de définir les lieux primitifs. Les lieux, une fois construits, sont considérés comme définitifs.

Signatures de lieux

Durant cette étape, nous laissons l'agent évoluer dans son environnement, mû par son mécanisme d'apprentissage. Nous pouvons observer, sur la Figure 11.2, les signatures de lieu générées par l'agent. Le fait d'utiliser des transformations géométriques limitées aux translations nous permet de mettre en évidence les régions de l'espace qui appartiennent aux lieux : chaque translation est en effet associée à un point de l'espace. Nous pouvons voir que les signatures des lieux primitifs désignent bien des positions de l'espace observable correspondant aux positions qui définissent ces lieux. Nous pouvons également observer que les signatures des lieux composites correspondent aux régions désignées par leur lieu final, mais déplacé selon les transformations des interactions du path, dans la limite de l'espace observable. Ceci implique qu'une partie des régions désignées par les lieux composites peuvent faire partie de l'espace non-observable. Ces lieux seront donc utiles pour caractériser la présence des objets dans l'espace non-observable.

Signatures de présence

Nous rappelons que la signature de présence d'un lieu composite l caractérise les contextes de présence Ω permettant de prédire la présence d'une instance d'objet dans le lieu final de l si le path est énoncé avec succès. Un contexte de présence $\Omega_{\omega_k, t}$ est un ensemble de lieux caractérisant la position d'une instance d'objet ω_k dans l'espace (cf. chapitre 6). Cette dernière étape est exécutée lorsque les signatures de lieu se stabilisent. Le moment où l'apprentissage des signatures de lieux commence est cependant défini arbitrairement. Notons que pendant cette étape, l'agent continue de renforcer ses interactions de lieu.

Un contexte permettant l'apprentissage des signatures de position est défini pour chaque instance d'objet stockée en mémoire. Un contexte de présence est composé d'un vecteur donnant, à chaque cycle de décision, la présence ou l'absence d'une instance d'objet donnée dans chaque lieu composite. Le contexte se limitera aux lieux composites dont le path a une longueur maximale de 1. En effet, ces lieux composites sont suffisants pour caractériser l'ensemble des positions de l'espace. Afin de faciliter la lecture du contexte et des signatures de présence, les entrées du contexte et les poids des signatures sont organisés d'après leur path, l'interaction et la distance qui caractérisent le lieu final, comme affiché à la Figure 11.3.

L'exemple détaillé dans la Figure 11.4 illustre le principe de mise à jour de la position d'une instance d'objet à l'aide des signatures de position. Dans cet exemple, l'agent a intégré une

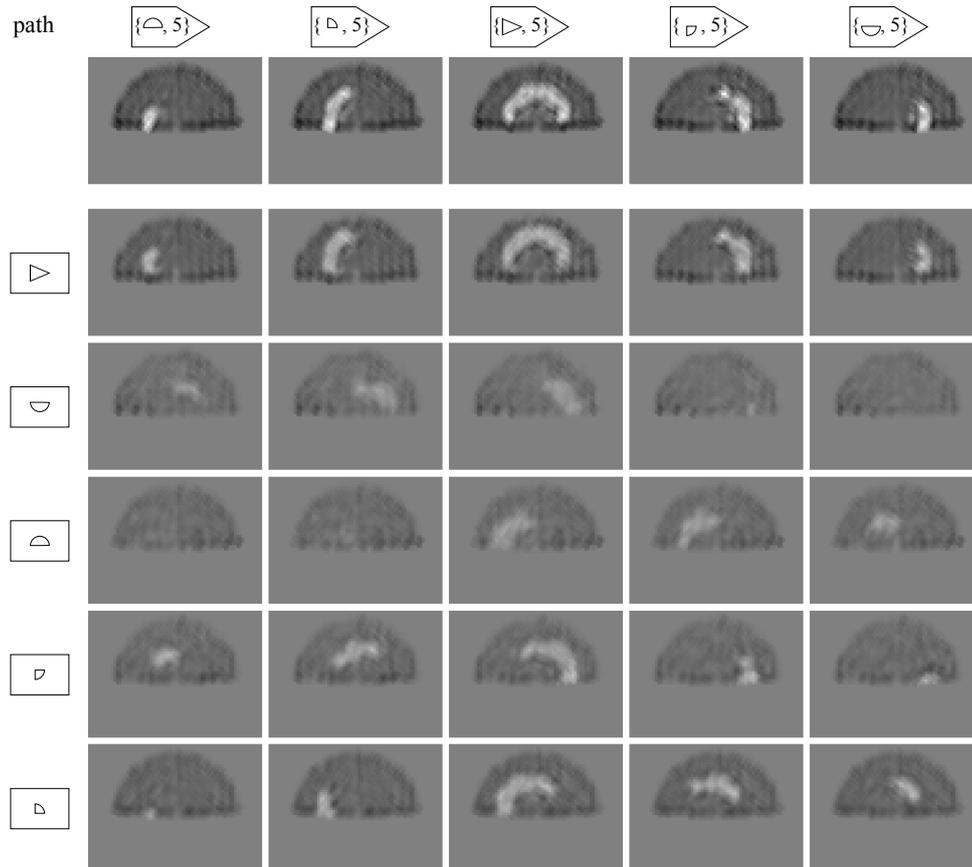


FIGURE 11.2 – Signatures de lieu des lieux composites dotées d'un path de longueur 0 et 1. En haut : les signatures de lieu des lieux primitifs associés à des régions situées à une distance de 5 unités, et pouvant être atteinte par l'interaction primitive (dans l'ordre suivant) : *tourner à gauche de 90°*, *tourner à gauche de 45°*, *avancer d'un pas*, *tourner à droite de 45°*, *tourner à droite de 90°*. Nous pouvons noter que les régions définies par les signatures sont limitées à l'espace visible. Les régions pour lesquelles une interaction *tourner* permet un rapprochement maximal se prolongent dans l'espace non-observable. En bas : les lieux composites dont le path a une longueur de 1. Chaque ligne correspond à un path composé d'une interaction primitive, avec, de haut en bas : *avancer d'un pas*, *tourner à droite de 90°*, *tourner à gauche de 90°*, *tourner à droite de 45°*, *tourner à gauche de 45°*. Les paths constitués des interactions *se cogner* et *manger* ne sont pas représentés. En effet, l'interaction *se cogner* ne génère pas d'interactions visuelles, et donc ne permet pas l'apprentissage des signatures de lieu. Un path constitué d'une interaction *manger* implique la présence d'une proie devant l'agent, qui masque une grande partie de l'espace observable, ce qui limite considérablement l'apprentissage des signatures de lieu. Nous pouvons remarquer que certaines signatures ne définissent pas de région dans l'espace observable. C'est le cas notamment des lieux $[tourner\ à\ droite\ de\ 90°][\{tourner\ à\ droite\ de\ 90°,5\}]$ et $[tourner\ à\ gauche\ de\ 90°][\{tourner\ à\ droite\ de\ 90°,5\}]$, liés à une région située derrière l'agent, dans l'espace non observable.

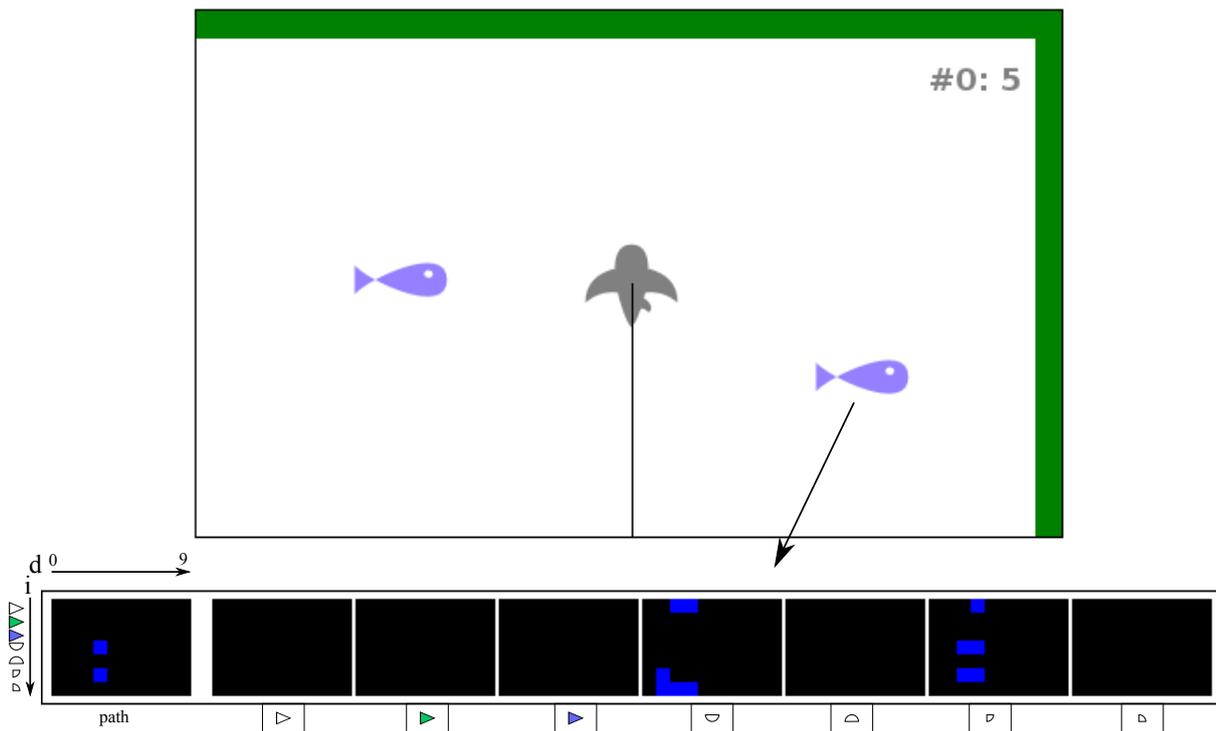


FIGURE 11.3 – Affichage du contexte d'une instance d'objet. Chaque groupe représente les entrées correspondant à des lieux composites constitués d'un même path. De gauche à droite, path de longueur nulle, path constitué de l'interaction *avancer*, *se cogner*, *manger*, *tourner à droite de 90°*, *à gauche de 90°*, *tourner à droite de 45°*, et *à gauche de 45°*. Pour chaque groupe, chaque ligne correspond à l'interaction composant le lieu final, et chaque colonne, une distance (de 0 à 9). Le contexte représenté ci-dessus représente la position de la proie à droite de l'agent.

instance d'objet correspondant à un élément proie. Après avoir avancé au cycle de décision suivant, l'agent ne détecte plus cette proie, mais continue de la suivre en mettant à jour les lieux composites qui caractérisent sa position. L'agent effectue ensuite une rotation à gauche. Comme aucun lieu composite associé à cette instance d'objet ne débute par cette interaction, l'agent devrait perdre la trace de cette instance. Cependant, la position définie par le contexte de présence de cette instance d'objet permet d'évoquer certains lieux composites. Un certain nombre de ces lieux composites débutent par l'interaction *tourner à gauche de 90°*, ce qui permet de continuer à suivre l'instance. Notons que dans cette configuration particulière, la proie, qui était à droite de l'agent, est maintenant localisée à une position pouvant être atteinte en tournant à gauche de 90°. La mise à jour de la position a ainsi permis le passage de la proie d'une région de l'espace non observable vers une autre région de l'espace non observable. La Figure 11.5 montre des exemples de signatures de présence de lieux composites dont le path commence par l'interaction *tourner à gauche de 90°*. Comme nous pouvons le constater, ces lieux sont évoqués par les contextes contenant des lieux composites dont le path est composé de l'interaction *tourner*

à droite de 45° , et dont la distance du lieu final est similaire.

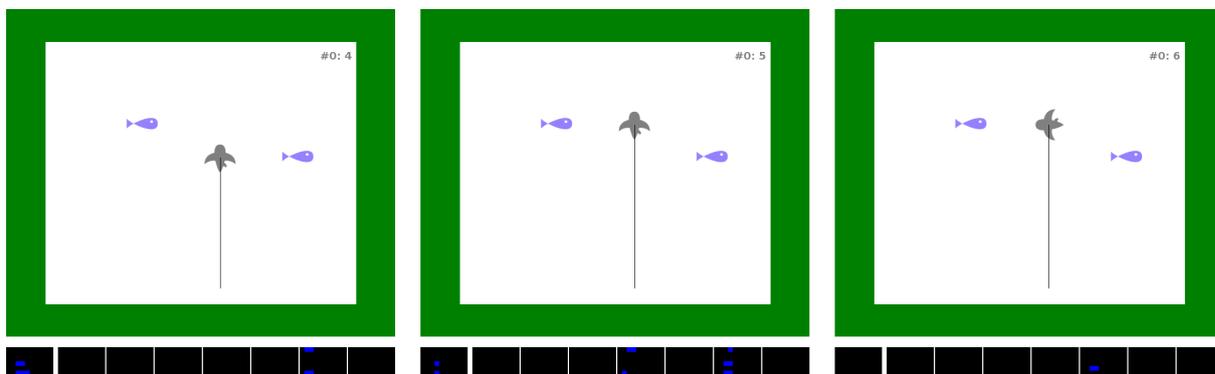


FIGURE 11.4 – Exemple de mise à jour de la position d’un objet en utilisant les signatures de position. Nous exploitons ici le fait que l’agent ait une préférence pour les proies situées à sa gauche. Au cycle de décision 4, l’agent intègre la position des instances d’objets qui l’entourent. Nous nous focaliserons plus particulièrement sur la proie située à droite de l’agent. Les contextes affichés sont ceux générés par cette instance d’objet. Comme nous pouvons le voir, la position de cette proie est désignée par des lieux composites caractérisant des régions atteignables en tournant à droite de 45° ou 90° , à une distance de 2 et 3, et par des lieux composites dont le path est constitué de l’interaction *tourner à droite de 45°* . Au cycle suivant, l’agent énonce l’interaction *avancer d’un pas*. La proie sort de l’espace observable, mais le mécanisme de mise à jour du path des lieux composites permettent de le suivre. La proie est toujours localisée dans une région pouvant être atteinte par une interaction *tourner à droite de 45° ou 90°* . Au cycle 6, l’agent tourne à gauche. Les lieux composites sont supprimés de la mémoire spatiale. Cependant, le contexte précédent permet d’évoquer un certain nombre de lieux composites, dont certains débutent par l’interaction *tourner à gauche de 90°* . Notons qu’après avoir effectué cette rotation, la proie, initialement à droite de l’agent, est désormais localisée dans une région de l’espace atteignable en tournant à gauche.

11.3 Expérimentations

Une fois l’apprentissage terminé, nous pouvons commencer l’étude des propriétés de la mémoire spatiale. Nous chercherons à observer la persistance de l’objet offerte par la mémoire spatiale, et la capacité de celle-ci à localiser les instances d’objets dans l’espace. Je propose d’utiliser certains des tests décrits pour tester le mécanisme des décisions de la mémoire spatiale *codée en dur*. Cependant, nous observerons plus attentivement le contenu de la mémoire spatiale, de façon à pouvoir observer l’emplacement estimé des objets et leur influence sur le comportement de l’agent.

Les comportements obtenus dépendent de la fonction d’influence des objets en fonction de leur distance. Je choisis d’utiliser les coefficients d’influence de la mémoire spatiale $\gamma = 1$ et

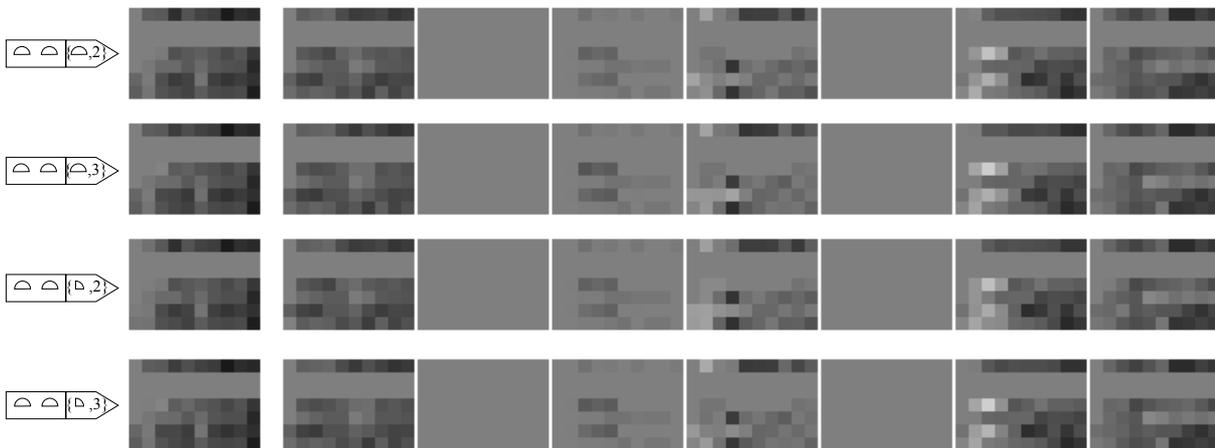


FIGURE 11.5 – Exemple de signatures de présence. Les contrastes ont été augmentés pour améliorer la lisibilité. Ces signatures sont très proches, car ces lieux composites caractérisent des régions de l'espace proche. Ces interactions peuvent être évoquées par un contexte de position comprenant les lieux composites $[tourner\ à\ gauche\ de\ 45^\circ][\{tourner\ à\ gauche\ de\ 90^\circ, 2\}]$ et $[tourner\ à\ gauche\ de\ 45^\circ][\{tourner\ à\ gauche\ de\ 45^\circ, 2\}]$.

$\beta = 15$, qui donnent pour chacune des expérimentations un comportement qui nous permettra d'étudier le fonctionnement de la mémoire spatiale. Ces coefficients sont plus élevés que pour la mémoire spatiale *codée en dur* en raison de l'échelle des distances différente.

11.3.1 Test de la *persistance de l'objet*

Comme au chapitre précédent, un objet attractif, ici, une proie, et masquée par différents types d'objets opaques. Cette expérience est menée dans un premier temps avec des algues, puis avec un mur. Nous observerons en détail la façon dont les instances d'objet sont suivies et localisées, et comment leur présence affectent le comportement de l'agent.

Test avec des algues

On laisse l'agent effectuer une première interaction, afin qu'il puisse détecter la proie. Cette proie est ensuite masquée par un ensemble d'algues. Nous pouvons constater que, comme avec la mémoire spatiale *codée en dur*, l'agent se dirige directement vers la proie sans tenir compte de l'algue (Figure 11.6). L'analyse de la mémoire spatiale, affichée dans la Figure 11.7, montre que la position de la proie est estimée avec une précision suffisante pour lui permettre de se diriger vers elle, jusqu'à ce qu'elle soit à nouveau visible. Le fait que l'instance d'objet étudiée reste dans l'espace visible nous permet de visualiser la région occupée par cette instance, à l'aide des signatures de lieu qui caractérisent la présence de cette instance. La région est affichée en additionnant les poids des signatures de lieu caractérisant la présence de la proie, pour chaque position de l'espace observable.

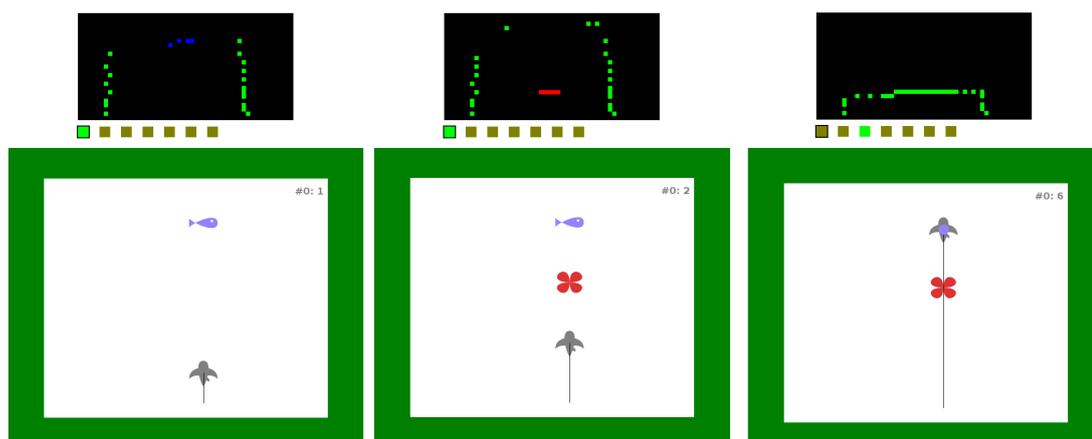


FIGURE 11.6 – Chemin utilisé par l’agent pour atteindre une proie. En haut, le contexte interactionnel, en bas, l’environnement. Un trait noir montre le chemin emprunté par l’agent. De gauche à droite : l’agent, après avoir effectué une première interaction, détecte les instances d’objets qui l’entourent. La proie est masquée avec une algue : le contexte interactionnel montre que l’agent ne la détecte plus. L’agent ne tient pas compte de l’algue et se dirige directement vers la proie.

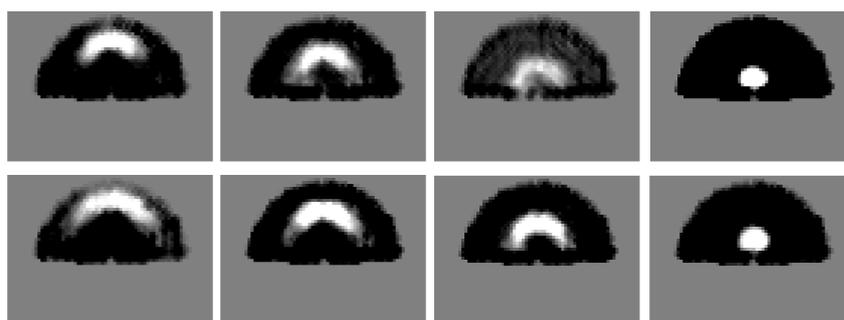


FIGURE 11.7 – Position estimée de la proie par la mémoire spatiale. En haut : le premier niveau de la mémoire, en bas, le second. Le fait que la proie soit dans l’espace observable nous permet d’observer la position estimée grâce aux signatures de lieu des lieux composites qui caractérisent cette position. Nous pouvons voir que la proie se rapproche à chaque pas, jusqu’à ce que la proie soit à nouveau visible au cycle de décision 4.

test avec des murs

Cette expérimentation est identique à la précédente, mais les proies sont masquées avec des murs (Figure 11.8). Nous pouvons observer que dans la plupart des cas, l’agent contourne l’obstacle, comme il le faisait avec la mémoire spatiale *codée en dur*. L’analyse de la position estimée montre que la mémoire spatiale permet une estimation suffisamment précise jusqu’à ce que la proie soit à nouveau visible (Figure 11.9). On notera que la première mémoire perd rapidement la position de l’instance d’objet. La seconde mémoire, bien que moins fiable, permet à l’agent de suivre la proie et de se diriger vers elle.

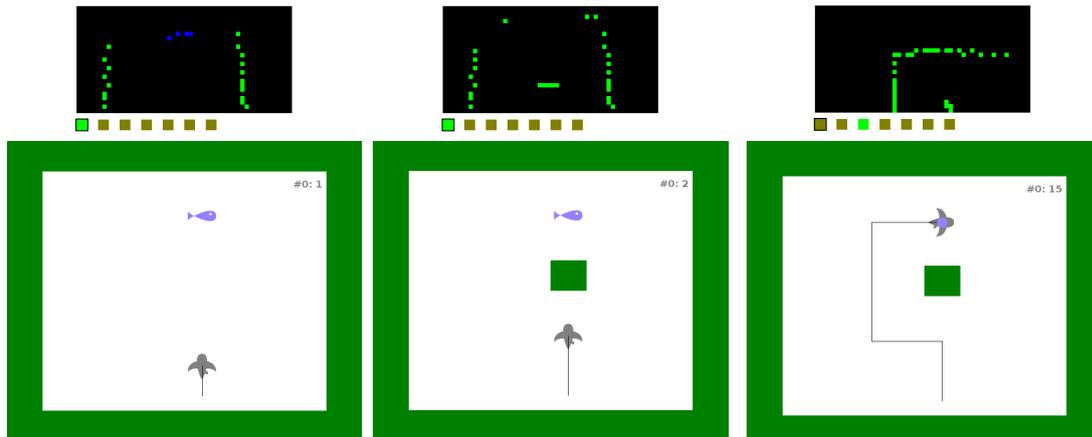


FIGURE 11.8 – Trajectoire utilisée par l'agent pour contourner un bloc mur. L'agent cherche à rester éloigné des instances d'objets qui affordent l'interaction *se cogner*.

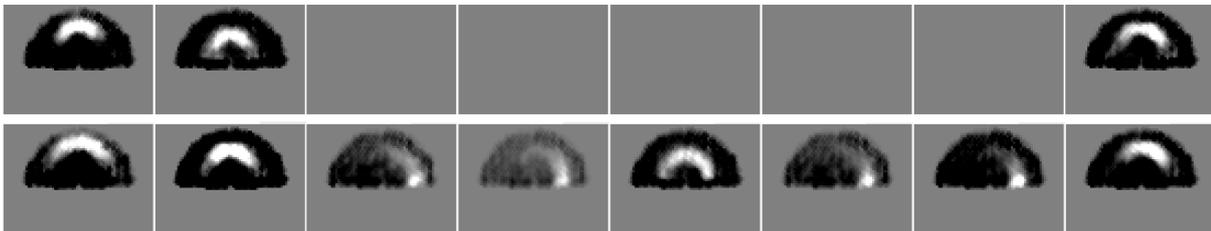


FIGURE 11.9 – Suivi de la proie par la mémoire spatiale jusqu'au pas 8, où l'agent perçoit à nouveau la proie. Le premier niveau de la mémoire spatiale fournit une meilleure précision aux pas 1 et 2, mais perd très vite la trace de la proie. Cependant, le second niveau de la mémoire continue de suivre cet élément.

11.3.2 Influence des coefficients d'influence de la mémoire spatiale

Le chemin utilisé par l'agent pour contourner le bloc mur dépend bien entendu du coefficient d'influence des objets. Nous pouvons observer une forte différence du comportement dans l'environnement présenté dans la Figure 11.10. L'agent a en effet une propension à tourner à gauche pour éviter un mur, dû certainement à une asymétrie dans les signatures des interactions. Le chemin utilisé par l'agent dépend d'un plus grand nombre de paramètres que pour la mémoire spatiale *codée en dur*, puisque les décisions de l'agent ne dépendent plus seulement de la distance des instances d'objet, mais également de l'estimation de leur position et la certitude de leur présence. Ainsi, avec un coefficient d'influence (γ) de 1, l'agent semble "hésiter" avant de s'engager dans le passage de gauche, c'est-à-dire qu'il est repoussé par les murs, trop proches, mais suffisamment attiré par la proie pour ne pas faire demi-tour. Après un certain nombre de mises à jour de la mémoire, les estimations sont suffisamment modifiées pour permettre à l'agent d'atteindre sa proie. Avec un coefficient d'influence de 1.4, le comportement diffère dès les premiers pas. Il en résulte un nombre d'instance d'objet intégrée en mémoire et une estimation

différente des distances. L'agent fait très vite demi-tour, puis passe par le côté droit de l'obstacle, plus large, pour atteindre la proie.

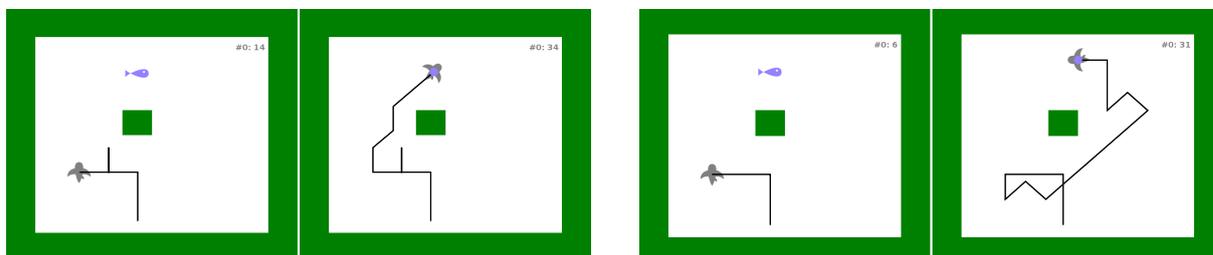


FIGURE 11.10 – À gauche, le chemin suivi par l'agent avec un coefficient d'influence de 1, à droite, avec un coefficient d'influence de 1,4. Nous pouvons observer que dans le premier cas, l'agent tente de passer plus tôt, avant de faire demi tour, et passer plus loin. Ce chemin modifie profondément les estimations de la mémoire spatiale, ce qui conduit l'agent à emprunter le passage à gauche. Dans le second cas, l'agent fait demi-tour et emprunte le chemin de droite, plus large.

Il faut toutefois noter que dans certaines configurations, l'agent ne parvient pas à trouver la proie. Cela s'explique par une faible fiabilité des signatures des lieux caractérisant certaines régions de l'espace : la position estimée de la proie peut alors être erronée, ce qui conduit l'agent à utiliser des interactions qui ne permettent pas de s'en rapprocher, ou perdent sa trace, si les signatures ne sont pas assez fiable pour qu'un lieu composite soit évoqué.

11.3.3 Prise en compte de plusieurs objets non visibles

L'expérience précédente permet d'observer comment les objets influent sur les décisions de l'agent. Cependant, les éléments étudiés restent visibles et sont donc intégrés et mis à jour dans la mémoire spatiale à chaque cycle de décision. Une seconde expérimentation est proposée pour observer comment des instances d'objets intégrées en mémoire peuvent influencer les décisions de l'agent. Cette expérimentation se déroule en deux parties. Dans la première, une proie est placée à une certaine distance de l'agent, puis on laisse celui-ci énoncer une interaction afin qu'il puisse détecter les instances d'objets peuplant son environnement. La proie est ensuite masquée avec un ensemble d'algues. Ce groupe d'algue permet de masquer la proie durant la totalité de l'approche. On observe ensuite le comportement de l'agent, notamment le chemin qu'il utilise pour se diriger vers la proie. Le chemin emprunté constitue le *chemin par défaut*.

La seconde partie de l'expérimentation est similaire à la première, et débute dans les mêmes conditions. On place une proie au même endroit que dans la première étape. On place également un bloc mur à proximité, sur le *chemin par défaut* suivi par l'agent dans la première étape. Une fois ces deux éléments détectés et intégrés, on ajoute un ensemble d'algues pour les masquer. On compare ensuite le *chemin par défaut* avec le chemin utilisé en présence du bloc mur.

Comportement en présence d'une proie seule

Une proie est placée dans l'environnement, à une certaine distance de l'agent. On laisse l'agent énoncer une interaction pour lui permettre de détecter les instances d'objets qui l'entourent. Une partie de l'environnement est ensuite remplie d'algues, de façon à masquer la proie durant toute la durée de l'approche. Comme le montre la Figure 11.11, l'agent se dirige tout droit dans les algues, en les ignorant, et, au niveau de la proie, tourne de 90° à gauche, avant de continuer à avancer jusqu'à ce qu'il atteigne la proie. Malgré la faible précision de la mémoire, l'agent est capable d'estimer la position de la proie de façon satisfaisante, et de se diriger vers elle en empruntant un chemin pertinent.

La Figure 11.12 montre la position estimée de la proie par rapport à l'agent. Nous pouvons voir que la proie se rapproche du côté gauche de l'agent, ce qui rend l'interaction *tourner à gauche de 90°* de plus en plus intéressante. L'agent énonce alors cette interaction au cycle de décision 5, quand la proie est exactement à gauche de l'agent.

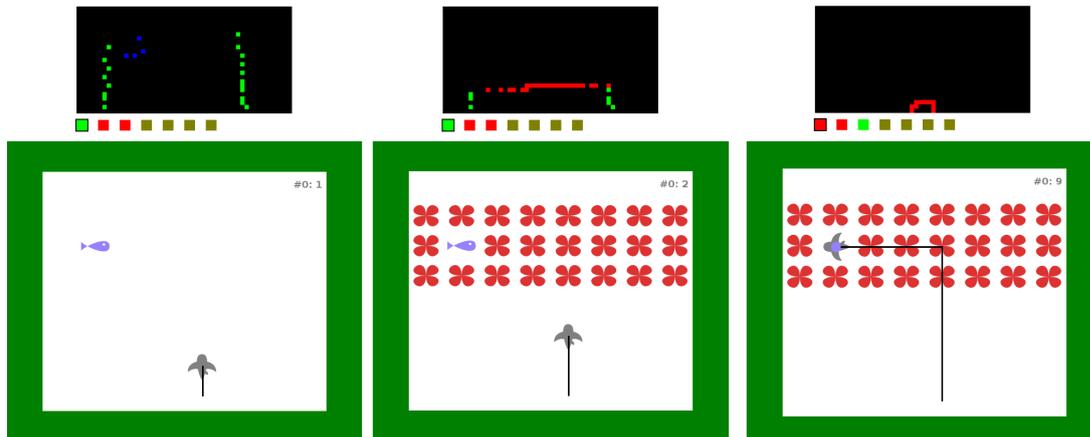


FIGURE 11.11 – Chemin utilisé par l'agent pour atteindre une proie. En haut, le contexte interactionnel, en bas, l'environnement. Un trait noir montre le chemin emprunté par l'agent. De gauche à droite : l'agent, après avoir effectué une première interaction, détecte et intègre les instances d'objets qui l'entourent. On masque la proie avec un groupe d'algues. Le contexte interactionnel montre que l'agent ne voit plus la proie. Le dernier environnement montre le chemin utilisé pour atteindre la proie, après un total de 9 interactions. Le chemin emprunté est relativement efficace du point de vue des valeurs de satisfaction des interactions, puisqu'il ne comporte qu'une seule interaction tourner. Nous pouvons constater que la précision de la position estimée de la proie est suffisante, puisque l'agent tourne à gauche au même niveau que la proie.

Nous pouvons observer sur la Figure 11.12 qu'après avoir mangé la proie, il subsiste un résidu de l'objet proie. En effet, au cours de l'apprentissage, il peut arriver que deux proies soient placés l'une à côté de l'autre, et considérées comme une même instance d'objet. Dans ce cas, une fois l'une des deux proies mangée, il reste une seconde proie à côté de l'agent, comme si une partie de l'objet subsistait. L'agent apprend donc qu'après avoir énoncé l'interaction *manger*, une partie

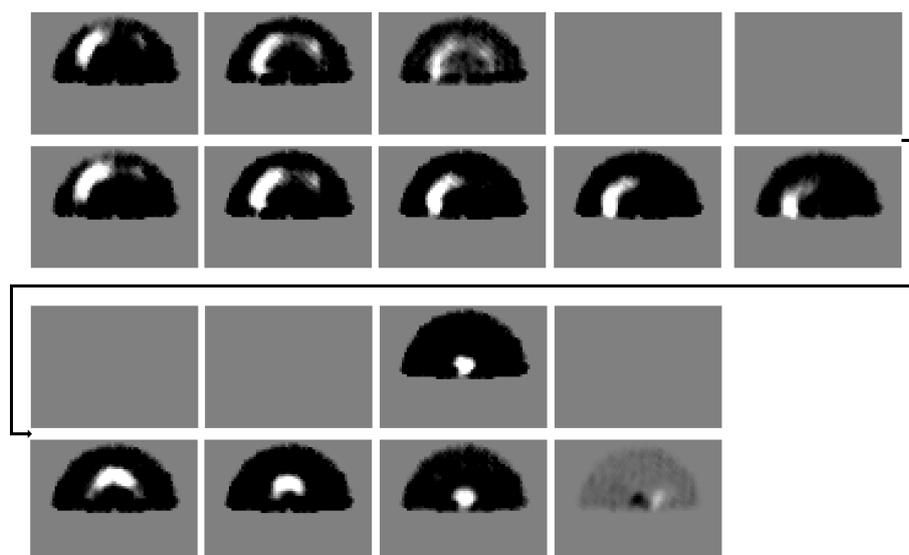


FIGURE 11.12 – Évolution de l'estimation de la position de la proie par la mémoire spatiale (à partir du cycle de décision 1). Ligne du haut : estimations du premier niveau de la mémoire spatiale. Ligne du bas : estimation du second niveau. Ces figures sont obtenues en effectuant la somme des signatures de position des lieux composites présents dans chacune des mémoires. Le premier niveau de la mémoire, qui met simplement à jour les lieux composites, perd l'instance d'objet après deux mises à jour. Le second niveau permet de suivre l'instance d'objet jusqu'à ce qu'il soit à nouveau visible. La précision de la position estimée reste à peu près constante, preuve que les conséquences de l'interaction *avancer* sont bien connues. Les différentes interactions énoncées sont reconnaissables : des cycles de décision 1 à 5, et des cycles de décision 6 à 8, nous pouvons voir que la région définissant la position estimée de la proie se rapproche. Des cycles de décision 5 à 6, l'agent effectue une rotation à gauche de 90° : l'instance d'objet, qui se trouvait à gauche, se retrouve devant l'agent une fois l'interaction énoncée.

de l'objet peut continuer d'exister. Une proie, ou groupe de proie, peut donc laisser un "résidu" selon la position initialement estimée.

Il faut cependant noter que la précision de la mémoire spatiale n'est pas homogène : en effet, pendant l'apprentissage, l'agent a tendance à apprendre les signatures de lieu dans un ordre qui dépend de l'ordre des interactions primitives dans l'ensemble I . À mesure qu'un comportement émerge, l'agent teste de moins en moins les lieux composites les moins utilisés. Dans l'exemple précédent, la précision dans la région à droite de l'agent est moins précise : l'agent a en effet tendance à capturer les proies situées à sa gauche, comme illustré dans la Figure 11.4, où l'agent avance d'un pas pour atteindre la proie à sa gauche, alors que la proie à sa droite est légèrement plus proche. En effectuant l'expérimentation précédente dans une configuration symétrique, on se rend compte que l'agent ne parvient pas à atteindre la proie (Figure 11.13). On peut toutefois remarquer que l'erreur d'estimation de la position de la proie est très faible, en considérant la distance parcourue par l'agent sans voir la proie.

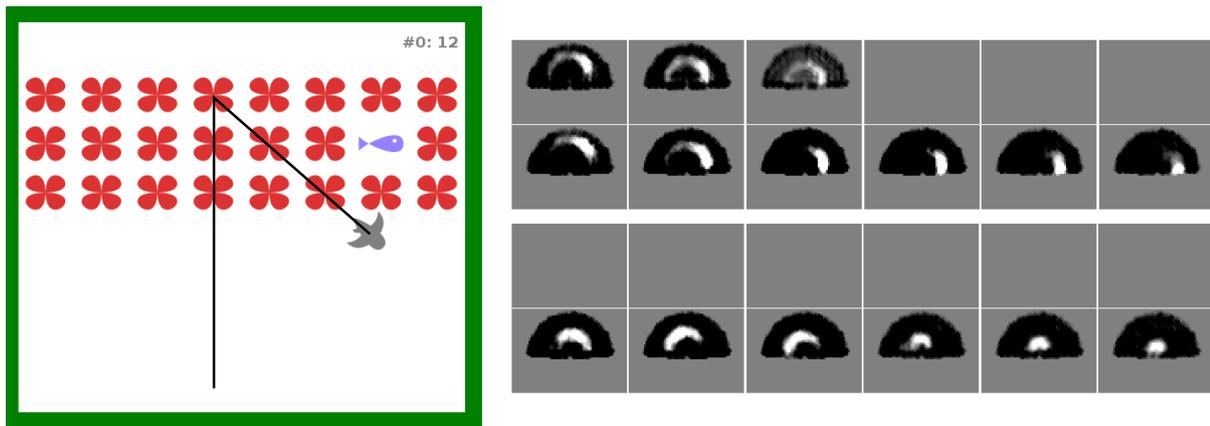


FIGURE 11.13 – Estimation de la position de la proie dans une configuration symétrique. La précision de la mémoire spatiale dans le coté droit de l'agent n'est plus suffisante pour se diriger vers la proie. Au cycle de décision 5, la proie est estimée à une position légèrement en avant, ce qui pousse l'agent à avancer une fois de plus, alors que la proie est à droite. Les estimations qui suivent s'écartent un peu plus à chaque cycle de décision : l'agent finit par perdre la position de la proie, bien que l'erreur soit faible.

Comportement en présence d'une proie et d'un mur

Nous procédons maintenant à la seconde partie du test. Les conditions initiales sont identiques à la précédente (mêmes signatures, même position de départ). Cependant, un bloc mur est ajouté sur le *chemin par défaut* observé précédemment. Comme nous pouvons le voir sur la Figure 11.14, l'agent utilise un autre chemin, ce qui lui permet d'éviter le bloc mur. Ce comportement montre que l'ensemble des instances d'objets intégrés dans la mémoire spatiale est bien pris en compte par le système de décision. On notera également que la position estimée des deux objets (Figures 11.15 et 11.16), même si la précision diminue à chaque cycle de décision, reste relativement proche de des positions réelles.

11.4 Conclusion

Comme nous avons pu l'observer, la mémoire spatiale permet à l'agent de conserver l'ensemble des instances d'objets qu'il perçoit dans son espace observable, et d'estimer leurs positions à chaque cycle de décision. La précision de ces estimations est faible, mais suffisante pour définir les interactions permettant de se rapprocher ou s'éloigner des objets.

Nous avons observé que la mémoire spatiale extra-personnelle perdait de son efficacité lorsque l'objet était trop proche. Il est raisonnable de penser que l'utilisation d'une mémoire péri-personnelle permettrait de prendre le relais, lorsque l'objet est proche, en donnant la séquence d'interaction exacte qui permettrait de l'atteindre, et non une estimation.

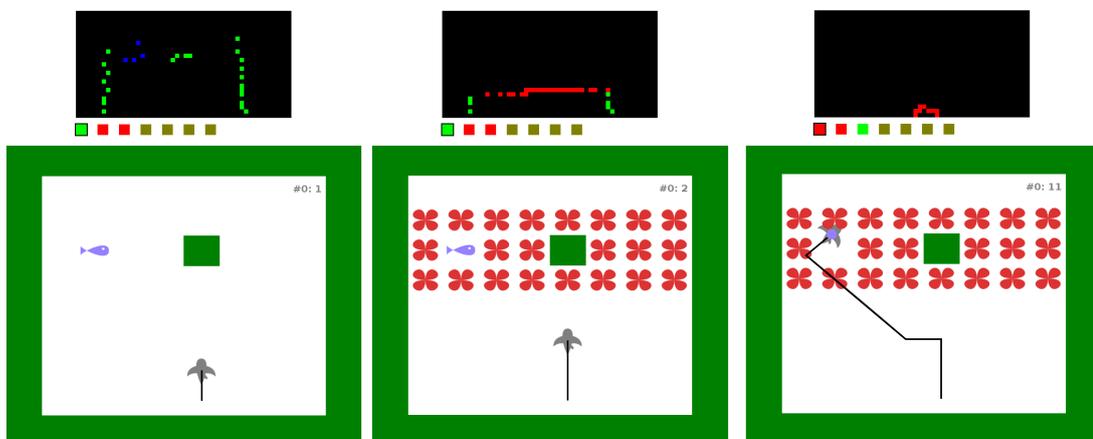


FIGURE 11.14 – Chemin utilisé par l’agent pour atteindre une proie en présence d’un mur. En haut, le contexte interactionnel, en bas, l’environnement. De gauche à droite : l’agent, après avoir effectué une première interaction, détecte et intègre les instances d’objets qui l’entourent. Les deux éléments sont ensuite masqués avec un groupe d’algues. Le contexte interactionnel montre que l’agent ne détecte plus les deux éléments. Enfin, l’agent atteint sa proie, après un total de 11 interactions. Le chemin emprunté est différent de celui de la première étape, ce qui démontre l’influence du mur, dont la présence n’est indiquée que par la mémoire spatiale.

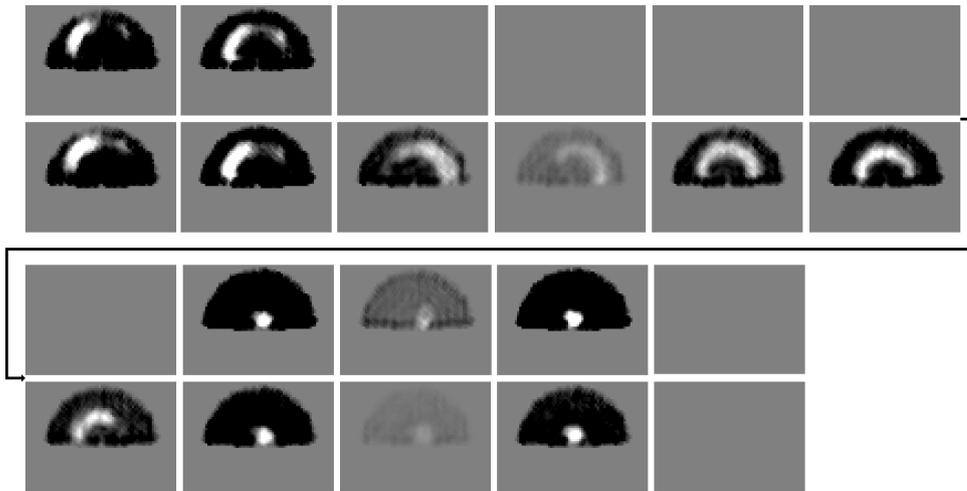


FIGURE 11.15 – Évolution de l'estimation de la position de l'élément proie par la mémoire spatiale (à partir du cycle de décision 1). On observe que la précision de la position estimée diminue rapidement après la rotation à gauche (cycle 3). On notera également que le premier niveau de la mémoire perd l'instance d'objet en seulement une mise à jour. Ceci montre que les conséquences de l'interaction tourner à gauche de 90° ne sont pas encore totalement connues. Notons cependant que malgré la faible précision, l'instance d'objet est toujours présente dans la région estimée. Les différentes interactions énoncées sont reconnaissables : *avancer* aux cycles de décision 1, 3 et 5 à 8, *tourner à gauche de 90° au pas 2*, *tourner à droite de 45° au cycle 4* et *tourner à droite de 90° au cycle 9*. Au cycle de décision 10, l'agent *mange* la proie. Comme il ne connaît aucune interaction permettant de la suivre, l'instance est supprimé de la mémoire.

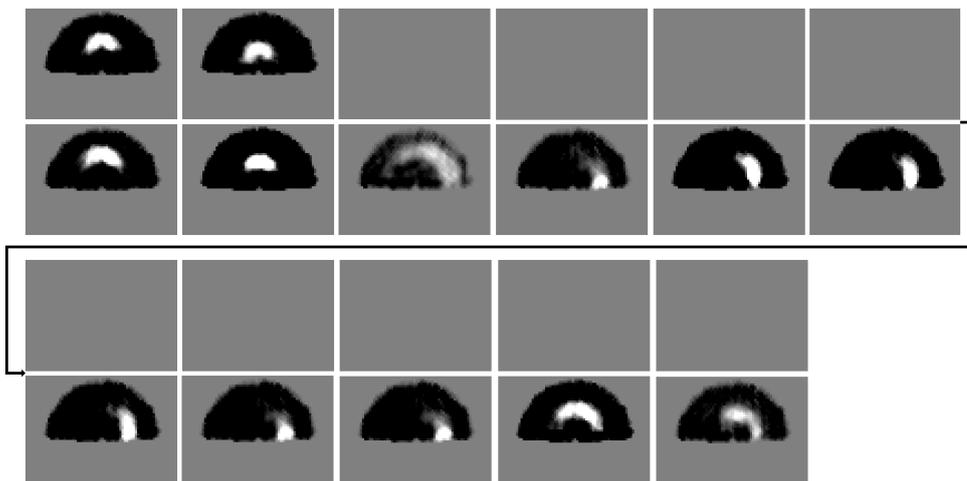


FIGURE 11.16 – Évolution de l'estimation de la position de l'élément mur par la mémoire spatiale. Tout comme avec la proie, la rotation génère une forte incertitude sur la position de l'objet. Il faut toutefois souligner que l'instance d'objet est suivi tout au long de la séquence des interactions énoncées par l'agent, avec une précision suffisante pour définir la région à éviter.

Quatrième partie

Synthèse, Discussion, Conclusion

Chapitre 12

Synthèse des résultats

Cette thèse nous a permis d'étudier comment exploiter le principe d'*Interactionnisme Radical* (*Radical Interactionism*) en construisant des mécanismes comportementaux le respectant. Ce type de mécanisme permet à un agent de caractériser, interpréter et exploiter le contenu *interagi* par un agent plongé dans un environnement spatial. Ces mécanismes ne sont basés que sur un ensemble de contingences sensorimotrices que nous appelons *interactions*. Ce type de mécanisme permet dans un premier temps de caractériser les éléments qui composent l'environnement de l'agent, en apprenant à intégrer les contextes expérimentés qui affordent chaque interaction. Ce simple principe guide la façon dont un agent se comporte dans son environnement : chaque interaction constitue un classifieur qui distingue les contextes qui affordent cette interaction de ceux qui ne l'affordent pas. Tout élément de l'environnement peut alors être caractérisé par l'ensemble des interactions qu'il afforde, indépendamment du nombre d'éléments constituant l'environnement (du point de vue d'un observateur extérieur). Dans nos expérimentations, nous avons pu constater que pour l'agent, les algues et les espaces vides sont intégrés comme des objets similaires qui affordent l'interaction *avancer*. Étant donné que certains éléments peuvent afforder plusieurs interactions, les mécanismes gèrent des *bundles* d'interactions. Notons toutefois que ces *bundles* sont implicites : deux interactions affordées par un même élément constitueront un *bundle* parce qu'elles seront considérées comme énonçables ou non énonçables de façon simultanée. Les objets tels qu'intégrés par l'agent, sous forme de signatures d'interaction, permettent de caractériser toutes les propriétés discriminantes des éléments qui affordent les interactions : taille, couleur et position par rapport à l'agent. Notons que cette dernière propriété est implicite, puisque la position *topographique* des interactions énoncées n'est pas connue. C'est en établissant des liens entre les interactions qu'il devient possible de caractériser les positions relatives des interactions par les séquences d'interactions qui permettent de passer d'une position à l'autre.

Cette dernière propriété est utilisée pour construire (par apprentissage) une structure reflétant les propriétés spatiales de l'environnement. Le sous-mécanisme dédié à la construction de cette structure permet dans un premier temps de définir des liens entre les interactions, en se basant sur les signatures d'interaction. En effet, la signature d'une interaction caractérise l'objet *pré-énoncé* qui afforde cette interaction, et, implicitement, sa position, par le biais des inter-

actions constituant sa signature. L'interaction elle-même caractérise l'objet *post-énaction* et sa position. Ainsi, en liant une interaction avec les interactions composant sa signature, on caractérise le déplacement (en référentiel égocentré) entre l'objet pré-énaction et l'objet post-énaction d'une interaction, produit par l'énaction de cette interaction. Ces liens permettent de définir une structure spatiale correspondant à l'espace observable.

Cette structure spatiale est ensuite segmentée en *lieux*, correspondant à des positions partageant des caractéristiques similaires. Nous avons en effet montré que pour considérer une instance d'objet lointaine, il est seulement nécessaire de connaître les interactions permettant de s'en rapprocher, et une distance pour caractériser l'influence de cet objet. Il n'est donc pas nécessaire de différencier des positions partageant les mêmes propriétés. Nous avons ensuite défini les lieux composites, qui permettent de caractériser l'ensemble de l'espace entourant l'agent. Toute région de l'espace peut alors être caractérisée par une liste de lieux (primitifs ou composites). Ainsi, la position d'une instance d'objet détectée et stockée en mémoire est estimée par l'ensemble des lieux qui contiennent cette instance, même si cette position est en dehors de l'espace observable. Cet ensemble de lieux constitue un contexte permettant de prédire la détection de cette instance dans l'espace observable. Nous avons défini la notion de *signature de présence* pour caractériser les contextes qui reflètent la région de l'espace caractérisée par un lieu composite. Ces signatures de présence ont deux fonctions : premièrement, elles permettent de lier les régions de l'espace par des interactions, comme nous avons pu le faire dans l'espace observable avec les signatures d'interaction. Deuxièmement, elles permettent d'évoquer les lieux composites caractérisant la position d'une instance d'objet, et ainsi prolonger le suivi de cette instance. Nous pouvons noter que le suivi des instances d'objet ne fait à aucun moment appel à une structure topographique représentant l'espace : la position estimée d'une instance est simplement mise à jour par le biais des interactions.

Nous avons développé deux mécanismes de sélection pour exploiter les structures construites par le mécanisme de construction de structures pour la prise de connaissance de l'environnement et le mécanisme de construction de la mémoire spatiale. Ces deux mécanismes ne sont pas exclusifs. Le premier repose sur des structures que nous avons appelées *interactions composites*. Une interaction composite caractérise une affordance, identifiée par son interaction finale, accessible en énant son *path*. Tout comme les interactions primitives, les interactions composites disposent d'une signature d'interaction. Un ensemble d'interactions composites permet ainsi de caractériser l'environnement sur une courte distance (la notion de *courte distance* est relative au nombre d'interactions) en respectant la notion d'espace représentatif de Poincaré : ce système ne repose en effet sur aucune structure topographique, les objets sont localisés uniquement par les interactions qui permettent de les atteindre. L'aspect séquentiel des interactions composites permet de suivre les objets qu'elles caractérisent sur de courtes durées (la notion de *courte durée* est relative à la complexité et l'importance du dispositif mémoire), et de compléter le contexte interactionnel de l'agent. Ces propriétés sont notamment utiles sur des systèmes qui ne peuvent expérimenter leur environnement que de façon partielle, en complétant progressivement le contexte interactionnel sans perdre les informations déjà acquises.

Le second mécanisme de sélection repose sur le contenu de la structure construite par le mécanisme de construction de la mémoire spatiale, et prend en considération les éléments constituant l'environnement plus lointain. Il permet à l'agent de satisfaire sa motivation interactionnelle à court et moyen terme, en tenant compte des interactions qui pourraient être énoncées dans un avenir proche. Pour cela, on affecte une valeur d'utilité aux interactions intentions candidates. La valeur d'utilité dépend du rapprochement des instances d'objet provoqué par l'énonciation d'une interaction intention candidate et de la valeur de satisfaction des interactions abordées par ces instances. La valeur d'utilité dépend également de la distance des instances d'objet : plus une instance d'objet est proche, plus il y a de chance que l'interaction qu'elle offre puisse être énoncée dans un avenir proche. Ce mécanisme se base sur les propriétés des lieux : chaque lieu primitif est caractérisé par l'interaction qui permet de se rapprocher des positions qui le composent, et du nombre d'interactions permettant d'atteindre ces positions. Les lieux composites, quant à eux, définissent les interactions permettant de se rapprocher d'une instance par le premier élément constituant leur *path*. Ceci permet de déterminer comment se rapprocher d'une instance située en dehors de l'espace observable.

Le mécanisme de la mémoire spatiale permet à l'agent de générer des comportements relativement complexes en prenant en compte les différents éléments constituant son environnement et leurs positions relatives. La façon dont l'agent agit en présence des différents éléments montre qu'il les a intégrés dans son comportement et est capable de les considérer avec une sémantique liée aux interactions qu'ils offrent. Nous pouvons remarquer, sur les signatures d'interaction, que les objets intégrés par l'agent peuvent naturellement différer de ceux considérés par un observateur extérieur comme éléments de l'environnement.

Chapitre 13

Discussion

13.1 Critiques globales

Le travail réalisé souffre de quelques limitations qu'il conviendra d'étudier et remédier. Il s'agit de perspectives très utiles pour continuer l'étude du paradigme de l'Interactionnisme Radical (Radical Interactionism).

Les critiques suivantes peuvent être faites :

- mécanisme de construction de structures pour la prise de connaissance de l'environnement : les interactions composites devraient être considérées comme points d'entrée des signatures. Les interactions composites stockées en mémoire peuvent en effet fournir des informations sur les éléments qui composent l'environnement immédiat et qui échappent au système sensoriel de l'agent. Non utilisée dans ce mécanisme, cette propriété reste proche du principe des signatures de présence utilisées par le mécanisme de construction de la mémoire spatiale, et qui permettent le suivi des instances d'objet dans l'espace non observable.
- mécanisme de construction de la mémoire spatiale : lors des expérimentations sur ce mécanisme (Chapitre 11), nous avons utilisé une structure topographique pour définir les régions de l'espace. La distance d'une instance d'objet est en effet définie par sa distance géométrique. Cette simplification a été effectuée pour des raisons de limitation de la complexité algorithmique, mais aussi parce qu'elle permet de faciliter la lecture des résultats en générant un espace de transformations spatiales continu à deux dimensions. Toutefois, nous avons démontré qu'il était possible d'apprendre à construire une structure non topologique, uniquement interactionnelle, équivalente fonctionnellement à cette structure topologique.
- mécanisme d'exploitation de la mémoire spatiale : nous n'avons pas utilisé les mécanismes caractérisant l'espace péri-personnel et l'espace extra-personnel de façon simultanée. Il est cependant raisonnable de penser que l'utilisation conjointe de ces mécanismes est possible : en effet, ces mécanismes ne sont pas incompatibles, et le mécanisme permettant de caractériser l'espace extra-personnel peut être adapté pour utiliser des interactions composites issues du mécanisme permettant de caractériser l'espace péri-personnel. Les propriétés des interactions primitives et composites sont en effet identiques pour le mécanisme carac-

térisant l'espace extra-personnel. Il s'agit là d'une perspective qu'il faudrait rapidement mettre en oeuvre.

13.2 Questions ouvertes

13.2.1 Éléments de réponse pour les questions posées par la recherche menée

Les différents mécanismes présentés dans ce manuscrit permettent de proposer des éléments de réponses aux questions et sous-questions de recherche présentées au chapitre 1. Ces questions sont essentielles pour préciser la portée et les limites de la recherche menée.

- Quelles sont les propriétés nécessaires et suffisantes de l'environnement pour permettre la construction d'un contexte spatio-temporel ?

L'intégration de l'espace s'effectue dans un référentiel égocentré. Il est donc nécessaire que les propriétés géométriques (c'est-à-dire les transformations relatives à l'agent quand il énonce une interaction) de l'environnement soient invariantes en tout point de l'espace. Le couplage agent-environnement doit également respecter certaines propriétés : en effet, pour pouvoir définir la signature d'une interaction, il est nécessaire que l'agent dispose d'organes sensori-moteurs, et donc d'un ensemble d'interactions permettant de détecter l'objet qui afforde cette interaction. Il est raisonnable de penser qu'un être vivant issu d'un processus d'évolution soit doté d'organes sensori-moteurs nécessaires pour prédire le résultat de ses interactions.

Ainsi, la complexité de l'environnement peut augmenter, tant que l'ensemble d'interactions de l'agent reste suffisant pour prédire le résultat de l'énonciation d'une interaction : tout élément de l'environnement pouvant alors être caractérisé par l'ensemble (fini) des interactions affordées et des interactions non affordées, et, de ce fait, se voir attribuer une sémantique liée à la valeur de satisfaction des interactions affordées. Bien que les environnements utilisés pour nos expérimentations soient très simples, dans le but de faciliter l'analyse des structures apprises par l'agent, ils nous ont permis de montrer que l'on pouvait considérer la complexité de l'environnement indépendamment de la complexité du système interactionnel. En effet, nous avons pu constater que les murs sont considérés de la même façon par l'agent quelle que soit leur forme (carré central ou bordure). Les algues sont considérées comme des espaces vides, car identiques d'un point de vue interactionnel.

- Quelles informations sont nécessaires et suffisantes pour permettre à un agent d'interagir avec son environnement en se basant sur le couplage entre l'agent et l'environnement ?

Nos différentes expérimentations ont montré que les informations nécessaires pour générer des comportements adaptés au contexte spatial de l'agent sont peu nombreuses : en effet, les objets sont définis par les interactions qu'ils affordent, et considérés par les interactions dont l'énonciation permet leur détection. La position d'une instance d'objet est considérée par une information à

propos de la distance de cette instance, et par les interactions qui permettent de s'en approcher. Les mécanismes d'apprentissage décrits dans ce manuscrit ont été développés dans l'unique but d'obtenir ces structures et ces informations.

- L'agent a-t-il besoin d'une organisation topographique de l'espace pour interagir avec son contexte spatial ?

Nos expérimentations successives semblent montrer que l'ensemble des informations nécessaires au fonctionnement de la mémoire spatiale et à l'émergence de comportements adaptés au contexte environnemental de l'agent peuvent être apprises sans se baser sur une structure topographique. Cependant, l'importante amélioration des capacités d'apprentissage et les simplifications permises par l'utilisation d'une structure topographique montre que l'existence de ce type de structure constitue un avantage significatif pour l'agent.

- Comment permettre un apprentissage efficace des régularités spatio-séquentielles ?

Nous proposons un mécanisme destiné à tester les structures apprises par l'agent dans le but de valider et/ou renforcer les signatures qui semblent peut fiable ou peu testées dans le contexte d'interaction actuel. L'expérimentation présentée au chapitre N montre que sans ce mécanisme, l'agent se base sur des coïncidences, ce qui peut l'amener à générer des comportements peu efficaces, voire dans certains cas, inadaptés à son contexte.

- Comment analyser le comportement de l'agent ?

Les expérimentations proposées dans ce manuscrit reposent sur l'analyse des interactions que l'agent a avec les différents éléments de son environnement. Ces expérimentations sont regroupées en plusieurs familles :

- Les expérimentations destinées à mesurer la valeur de satisfaction moyenne du comportement de l'agent, mis en oeuvre pour tester la mémoire péri-personnelle. Ces expérimentations sont les seules à utiliser une mesure quantitative, mais nécessite des conditions constantes pour être pertinentes. C'est pourquoi seule la mémoire péri-personnelle a été testée en utilisant ces expérimentations.

- Les expérimentations destinées à évaluer comment l'agent interagit avec les éléments de son environnement. On observe notamment le comportement de l'agent face à un élément spécifique (attraction, contournement, négligence), la façon dont il s'approche des éléments et le comportement en présence de plusieurs objets.

- Les expérimentations destinées à mesurer l'influence d'un élément sur le comportement de l'agent. Ces expérimentations consistent à observer et comparer le comportement de l'agent, en débutant l'expérimentation dans des conditions identiques, en présence ou en l'absence de l'élément à étudier.

- Comment un agent artificiel peut-il découvrir, organiser et exploiter des régularités spatio-temporelles issues du couplage entre l'agent et son environnement pour

exhiber des comportements "intelligents" ?

Les différents mécanismes présentés dans ce manuscrit permettent à l'agent de construire sa propre notion des objets sous une forme qu'il peut exploiter. Les propriétés spatiales de l'espace sont découvertes par l'agent au travers de ses interactions avec son environnement. Les connaissances générées par l'agent sont toutes basées sur les interactions.

13.2.2 Éléments de réponse aux problématiques abordées

Au chapitre 2, j'ai dégagé un ensemble de propriétés dont un agent artificiel doit disposer pour pouvoir intégrer son environnement et exhiber des comportements adaptés à son contexte environnemental. Ces propriétés sont rappelées dans le tableau 13.1. Ce tableau résume les propriétés observées dans le cas des mécanismes de la mémoire péri-personnelle (Chapitre 9) et de la mémoire extra-personnelle (Chapitres 10 et 11).

TABLE 13.1 – Problématiques abordées.

	Intrinsèquement motivé	Intégration autonome des objets	intégration autonome de l'espace proche	intégration autonome de l'espace lointain	émergence de comportements	exploitation des connaissances	apprentissage du <i>Body Schema</i>
Mécanisme intégrant l'espace péri-personnel (Chapitre 9)	*	*	*		-	*	-
extra-personnel (Chapitres 10 et 11)	*	*	*	*	*	*	-

- **intrinsèquement motivé** : le mécanisme de décision de l'agent repose sur le principe de motivation interactionnelle, qui repose sur la valeur de satisfaction associée aux interactions et que l'agent expérimente lorsqu'il énonce une interaction. L'agent n'a ainsi jamais accès aux états de l'environnement, et ne reçoit aucune information provenant de l'extérieur.

- **intégration autonome des objets** : l'agent génère ses propres "modèles" des objets qui affinent ses interactions d'après la façon dont il peut les expérimenter, c'est-à-dire au travers d'interactions. Les "modèles" ainsi générés dépendent de ses interactions et sont indépendants des modèles que pourrait construire un observateur extérieur à l'agent.

- **intégration de l'espace proche** : le mécanisme de la mémoire péri-personnelle permet, par le biais des interactions composites, de caractériser l'environnement "proche" (au sens interactionnel) de l'agent, en considérant la position des objets comme la séquence d'interaction permettant de l'atteindre. L'ensemble des interactions composites énonçables permet de définir l'espace avec lequel l'agent peut directement interagir. Le mécanisme de la mémoire extra-personnelle intègre également, par le biais des signatures des interactions primitives, l'espace immédiat. Bien que limité, cet espace constitue la première étape vers l'intégration de l'espace péri-personnel.

- **intégration de l'espace lointain** : le mécanisme de la mémoire extra-personnelle permet à l'agent de considérer l'ensemble de l'espace qui l'entoure (dans la limite, bien sûr, de ses capacités d'intégration). Le principe des signatures de lieu et de présence permet de déterminer la position des instances d'objet et de suivre ces positions sous une forme directement exploitable par l'agent.

- **émergence de comportements** : le mécanisme d'exploitation de la mémoire spatiale permet de considérer l'ensemble des éléments présents dans l'environnement pour générer des comportements. Ce mécanisme prend en considération les possibilités d'interactions immédiates, définies grâce aux signatures d'interaction, et futures, caractérisées par les instances d'objet, pour satisfaire sa motivation interactionnelle à court et moyen terme.

- **exploitation des connaissances** : les structures apprises par l'agent sont définies dans le but d'être exploitables par le mécanisme de sélection, et non dans le but d'établir des modèles explicites compréhensibles et utilisables par un observateur extérieur.

- **apprentissage du Body Schema** : les observations sur les signatures d'interactions ont montré que certaines propriétés du Body Schema pouvaient être observées : notamment la taille de l'agent et le volume (ou la surface dans un environnement 2D) qu'il occupe ou occupera dans l'espace.

13.3 Perspectives

Nos différentes expérimentations ont permis de tester la validité des différents aspects de nos modèles. Cependant, nous avons restreint nos investigations à des systèmes simples, évoluant dans des environnements limités. Dans cette section, nous proposons des voies à explorer pour permettre l'utilisation de nos modèles sur des systèmes plus sophistiqués.

13.3.1 Études complémentaires

Nous avons testé les différents mécanismes du modèle séparément. Une prochaine étape pourrait considérer le développement d'un système complet basé sur le modèle PRI. De plus, afin de se concentrer sur l'aspect spatial, nous avons ignoré l'aspect séquentiel étudié précédemment dans IDEAL. L'aspect séquentiel et spatial ne sont naturellement pas exclusifs. Nous proposons donc :

- d'utiliser simultanément des mécanismes permettant de gérer l'espace péri-personnel et l'espace extra-personnel. Nous avons vu que le mécanisme gérant l'espace péri-personnel, permet, sur la surface qu'il couvre, une meilleure précision et une plus forte réactivité, puisque aucun calcul n'est nécessaire pour se rapprocher d'un objet. L'utilisation des deux mécanismes serait ainsi profitable à un agent, notamment si celui-ci évolue dans un environnement dynamique.
- utiliser simultanément des mécanismes gérant les aspects spatiaux et séquentiels, par l'utilisation de séquences d'interactions en lieu et place des interactions primitives telles qu'elles

sont utilisées dans les mécanismes présentés dans ce manuscrit. En effet, une séquence d'interactions gérée par le mécanisme séquentiel constitue une interaction pouvant réussir ou échouer son éaction, ce qui la rend utilisable dans le mécanisme de gestion spatial proposé. L'utilisation du mécanisme séquentiel rend possible la construction d'interactions composites de complexité croissante, et d'implémenter le concept d'auto-programmation défini par le mécanisme séquentiel [45]. Notons que l'on peut simplifier la notation d'une interaction composite comme $\varphi = [\varphi_p, \varphi_f]$ avec φ_p et φ_f des interactions complexes (ou séquences d'interactions) représentant respectivement une position et un type d'objet.

13.3.2 Amélioration du mécanisme spatial

Nous avons développé notre mécanisme spatial dans un environnement statique. Cependant, le but à long terme est de pouvoir implémenter nos modèles dans des systèmes évoluant dans des environnements réels. Or, interagir avec un environnement réel implique de pouvoir interagir avec un environnement dynamique.

Les mécanismes que nous avons développés ne conviennent pas aux environnements dynamiques : en effet, ils nécessitent des environnements *prédictibles* du point de vue de l'agent ; il ne prend pas en compte le mouvement des objets. Une solution simple consisterait à doter l'agent d'un système sensoriel sensible au mouvement relatif. Une autre solution consisterait à utiliser un contexte incluant des informations à la fois spatiales et temporelles, c'est-à-dire un contexte incluant la séquence des ensembles éactés lors des cycles de décisions précédents. Un tel contexte contiendrait alors des informations sur les mouvements des objets constituant l'environnement.

Nous proposons également de prendre en compte les conséquences des interactions futures de l'agent sur son environnement dans les simulations de la mémoire spatiale. Il faudrait pouvoir simuler les effets d'une interaction à une position différente de celle de l'agent. Un tel agent pourrait réorganiser ses actes dans le but de modifier l'environnement actuel pour faire apparaître des affordances. Ce principe n'est pas très différent du système d'exploitation de la mémoire spatiale : en effet, dans le système tel que nous l'avons décrit dans ce manuscrit, on cherche à rapprocher l'environnement *interagi* d'une structure qui afforde une interaction, à partir des interactions éactables. Dans le système amélioré, on pourrait tenir compte des conséquences de l'éaction future d'une interaction, et chercher à rapprocher l'environnement d'un contexte où une instance d'objet serait présente. Ce système pourrait ainsi mesurer les conséquences d'une modification de l'environnement sur la possibilité d'éacter une interaction dans un futur proche.

13.3.3 Amélioration du modèle RI

Afin de concevoir des systèmes plus sophistiqués, nous proposons différentes voies pour étendre le modèle RI :

- utilisation d'un ensemble d'interactions *continues*. Les mouvements des êtres vivants sont en effet continus, dans l'ensemble des mouvements possibles et dans le temps, de même que l'activité des zones associées dans le cerveau, sous forme de cortices mobilisés pour

établir une continuité entre deux mouvements proches. Durant les travaux préliminaires à ceux présentés dans le manuscrit, une version d'un mécanisme gérant l'espace proche sur un agent doté d'un système visuel et doté d'une fovéa a été développée. Les interactions visuelles consistaient à considérer un élément d'une certaine couleur à une position définie par ses coordonnées cartésiennes en référentiel égocentré. Ces coordonnées étaient discrètes, mais faisaient partie d'un ensemble continu. Le système d'apprentissage des signatures utilisait cette continuité pour estimer la signature d'une interaction visuelle, où détecter d'éventuelles incohérences, à partir des interactions proches dans l'ensemble des coordonnées. Ce système permettait une amélioration significative de l'apprentissage des signatures. Toutefois, il supposait de prédéfinir la continuité entre les interactions.

- utilisation d'interactions conditionnelles. Nous avons vu au chapitre 9 que le comportement de l'agent peut se décrire par un organigramme où les interactions épistémiques forment les conditions. Une évolution du modèle RI pourrait générer et énoncer des interactions conditionnelles similaires à celles décrites dans les organigrammes issus de l'observation du comportement de l'agent, déléguant ainsi les décisions de bas niveau au système d'énonciation des interactions. Un tel agent pourrait ainsi énoncer des comportements complexes et non plus seulement des séquences d'interaction.
- définition dynamique de la valeur de satisfaction des interactions. Dans notre modèle, la valeur de satisfaction des interactions est fixe. La répétition d'un comportement peut donc se perpétuer indéfiniment. Dans le système décrit dans le chapitre 9, on évite la répétition d'interactions ne générant pas de mouvements en interdisant la sélection des interactions permettant la détection de leur propre affordance. L'utilisation d'une valeur de satisfaction dynamique éviterait le recours à de telles règles. On peut en effet définir une gestion de l'ennui, par exemple, en baissant la valeur de satisfaction d'une interaction trop souvent utilisée. La gestion dynamique des valeurs de satisfaction pourrait également rendre compte *d'états* observables de l'agent, comme la faim ou la fatigue. La modification de la valeur de satisfaction d'une interaction influe alors sur l'attractivité des instances de l'objet qui afforde cette interaction : une instance d'objet affordant l'interaction *manger* deviendrait alors attractive quand l'agent a faim et répulsive quand il est rassasié.
- possibilité d'énoncer des interactions en parallèles. Les êtres vivants agissent en activant plusieurs muscles simultanément. Une prochaine étape dans le développement du modèle RI pourrait être la possibilité d'énoncer plusieurs interactions intentions simultanément.

Chapitre 14

Conclusion

Dans ce manuscrit, nous nous sommes intéressés au développement d'une structure permettant à un agent artificiel d'appréhender l'environnement et l'espace qui l'entoure. Nous proposons une approche dans laquelle l'action et la perception d'un agent sont considérées comme indissociables, et forment une *interaction*. Cette approche, appelée Interactionnisme Radical (ou Radical Interactionism, RI), implémente les théories de la cognition qui stipulent que actions et perceptions ne sont pas prédéfinies *a priori* mais apprises par l'expérience. Ce modèle permet également une forte indépendance entre l'agent et l'environnement, appelée *agnosticisme environnemental*.

Nous nous sommes d'abord intéressés à la façon dont les êtres vivants représentent leur environnement. Ces observations montrent que les modèles de l'espace considèrent différentes régions fonctionnelles traitées par des zones différentes du cerveau. Certaines zones cérébrales forment une correspondance innée avec la topologie de l'espace, mais leur utilisation est limitée au champ visuel.

Nous avons dégagé trois mécanismes distincts bien que fortement interdépendants. Ces mécanismes ont été étudiés séparément, en utilisant des heuristiques qui ont été implémentées et validées dans les autres mécanismes.

Dans le chapitre 5, nous avons étudié le mécanisme de construction de structures pour la *prise de connaissance de l'environnement*, qui permet d'établir un lien implicite entre les éléments physiques de l'environnement et les interactions de l'agent, et permettant à celui-ci de construire une connaissance exploitable sur ces éléments. L'utilisation d'un ensemble fini d'interactions permet de définir et reconnaître des objets dans l'environnement immédiat de l'agent et de définir les interactions qu'il peut ou ne peut pas effectuer, sans qu'aucune connaissance *a priori* sur les éléments composant l'environnement ne soit nécessaire.

Dans le chapitre 6, nous nous sommes intéressés au mécanisme de construction d'une *mémoire spatiale*, permettant la construction d'une structure caractérisant le contenu de l'espace sur une distance plus importante que celle permise par le premier mécanisme. Cette structure s'est révélée nécessaire dans le cas d'un agent évoluant dans un environnement ouvert. Les expérimentations décrites au chapitre 10 sur le mécanisme d'exploitation de la mémoire spatiale ont permis de définir les informations minimales permettant à l'agent de naviguer dans un environnement

ouvert en tenant compte des affordances qu'il propose et de leur position relative par rapport à l'agent. Ces informations permettent de définir la position d'un objet en termes d'interaction et non en coordonnées, ce qui rend inutile le recours à une carte topographique de l'espace. Dans la première partie de ce chapitre, nous avons détaillé le mécanisme permettant à l'agent de reconnaître les objets distants qu'il peut détecter et stocker. Dans la seconde partie, nous avons détaillé les mécanismes permettant de suivre les objets pendant les déplacements de l'agent (nous sommes toujours en référentiel égo-centré) et d'exploiter le contenu de cette mémoire. Le système de suivi des objets ne repose sur aucune structure topographique de l'espace. Cependant, les expérimentations décrites au chapitre 11 montrent que l'agent se dirige naturellement vers les objets qui offrent des interactions attractives, même lorsqu'il ne peut plus les détecter, avec une précision suffisante pour les atteindre.

Dans le chapitre 7, nous nous sommes intéressés au mécanisme *d'exploitation de la mémoire spatiale*. Ce mécanisme ne repose pas sur l'utilisation d'algorithmes de type *path planning*. En effet, nous ne définissons pas de but à atteindre, ni de tâche à accomplir. Au lieu de cela, l'agent doit tenir compte de l'ensemble des affordances offertes par son environnement et agir de façon à maximiser sa satisfaction à court et moyen terme.

Deux approches sont proposées :

- la première approche permet de caractériser l'espace proche de l'agent. Elle repose sur l'utilisation de courtes séquences d'interactions permettant de définir des objets situés dans l'espace immédiatement accessible. L'utilisation de séquences d'interaction permet dans un premier temps un effet mémoire qui permet de suivre les objets préalablement détectés, même lorsqu'ils sont hors de portée du système sensoriel de l'agent, tout en donnant le moyen de les atteindre pour interagir avec eux. D'autre part, il devient possible d'établir un lien implicite entre des objets liés à un même élément de l'environnement et situés à deux emplacements différents (en référentiel égo-centré), et de définir l'interaction ou la séquence d'interaction permettant de lier ces deux emplacements. Cette approche ne repose pas sur une mémoire de l'espace topologique et ne permet pas à l'agent de naviguer dans un environnement ouvert. Cette approche est similaire à l'approche implémentant le mécanisme de construction de la mémoire spatiale, notamment dans l'utilisation de séquences d'interactions pour suivre les objets.
- la seconde approche propose d'observer le rapprochement des affordances provoqués par les interactions que l'agent peut énoncer. Le rapprochement d'une affordance liée à une interaction dotée d'une forte valeur de satisfaction a un impact positif puisqu'il permettra, dans un avenir proche, d'effectuer l'interaction attendue. Ce principe, bien que simple, permet à l'agent de démontrer des comportements spatiaux élaborés prenant en compte plusieurs objets de plusieurs types, comme le contournement d'obstacles. L'expérimentation avec un robot montre que le système reste fonctionnel dans un environnement bruité et malgré les imprécisions dans les déplacements du robot.

Ces différents mécanismes ont été testés par un ensemble d'expérimentations permettant de mettre en évidence les propriétés et connaissances qui en émergent. Les mécanismes de construction

pour la prise en compte de l'environnement et le mécanisme d'exploitation de la mémoire spatiale ont été testés dans les expérimentations décrites aux chapitres 9 et 10, ce qui a permis de tester ces mécanismes sur deux systèmes distincts. Le chapitre 9 est consacré à la première approche du mécanisme d'exploitation de la mémoire spatiale, et le chapitre 10 à la seconde. Le mécanisme de construction de la mémoire spatiale a été testé dans les expérimentations décrites au chapitre 11. Ces expérimentations se basent sur des propriétés observées et validées dans les expérimentations présentées aux chapitres 9 et 10.

Au final, nous avons montré que chaque mécanisme peut se passer de connaissances *a priori*. Les résultats obtenus sur les mécanismes de construction et d'exploitation d'une structure caractérisant l'environnement ne nécessite pas de structure respectant la topographie de l'espace, bien qu'une telle structure permette une réduction significative de la complexité du système.

14.1 Table des symboles

Symboles

t : cycle de décision

i : utilisé par défaut pour désigner une interaction primitive

I : ensemble des interactions primitives

s_i : valeur de satisfaction d'une interaction i

i_t : interaction intention, sélectionnée par l'agent en début de cycle

e_t : interaction énoncée, obtenue à la fin d'un cycle de décision

E_t : ensemble des interactions énoncées. Caractérise l'environnement tel qu'expérimenté par l'agent à la fin d'un cycle t

\mathcal{A}_i : groupe *alternatif* de l'interaction i

p : désigne une position de l'espace en référentiel égo-centré dans le modèle SRI

P, P' : ensemble des positions p de l'espace, respectivement de l'espace observable (i.e. détectable par le système sensoriel), et de l'espace global. Spécifique au modèle SRI

τ : transformation spatiale

τ_i : transformation produite par l'énoncé d'une interaction i .

T : ensemble des transformations spatiales

$E_{\tau,t}$: contexte image obtenu en appliquant τ au contexte interactionnel

a : utilisé par défaut pour désigner un acte (spécifique au modèle SRI)

A, A' : ensemble des actes, respectivement de l'espace observable et de l'espace global (spécifique au modèle SRI)

l : lieu (primitif ou composite)

L : ensemble des lieux primitifs

L' : ensemble des lieux composites

S_l : signature d'un lieu l

$S_{p,l}$: signature de présence d'un lieu l

O_i : objet physique qui afforde une interaction i

\hat{O}_i : objet physique dont la présence est caractérisée par l'énoncé de i

ω : instance d'un objet

Ω_ω : ensemble des lieux caractérisant la position de l'instance ω

$c_{l,\omega}$: certitude que le lieu l caractérise la présence de l'instance ω

S_i : signature de l'interaction i

φ : utilisé par défaut pour désigner une interaction composite.

Φ : ensemble des interactions composites

\mathcal{T} : liste des interactions composites considérées comme possibles

\mathcal{F} : liste des interactions composites considérées comme impossibles

$\phi_{i,t}$: fiabilité d'une interaction i au cycle de décision t

m : mouvement associé à une interaction primitive (hypothèse vestibulaire)

\mathcal{M} : ensemble des mouvements (hypothèse vestibulaire)

$\nu_{i,t}$: proximité globale des instances de l'objet défini par i

$\Delta\nu_i$: variation de proximité globale provoquée par l'énaction de i

s'_i : valeur de satisfaction globale de i

$c_{success}$: fonction de certitude de succès

$c_{failure}$: fonction de certitude d'échec

res : résultat (succès ou échec d'une interaction à un cycle de décision)

$c_{reverse}$: fonction de certitude d'énaction

c_{in} : fonction caractérisant la certitude de présence d'une instance d'objet dans un lieu l

c_{out} : fonction caractérisant la certitude d'absence d'une instance d'objet dans un lieu l

c_{pres} : fonction caractérisant la certitude de présence d'une instance dans un contexte Ω_ω

c_{abs} : fonction caractérisant la certitude d'absence d'une instance dans un contexte Ω_ω

Bibliographie

- [1] Ashraf Aboshosha et Andreas Zell, *Robust Mapping and Path Planning for Indoor Robots based on Sensor Integration of Sonar and a 2D Laser Range Finder*, IEEE 7th International Conference on Intelligent Engineering Systems, 2003.
- [2] Baris Akgun, Nilgun Dag, Tahir Bilal, Ilkay Atil, et Erol Sahin, , *Unsupervised learning of affordance relations on a humanoid robot*, International Symposium on Computer and Information Sciences (ISCIS), p. 254-259, Septembre 2009.
- [3] Michael L. Anderson, *Embodied cognition : A field guide*, Artificial Intelligence, vol. 149, p. 91–130, 2003.
- [4] John R. Anderson, *How can the human mind occur in the physical universe ?*, Oxford University Press, ISBN-13 : 9780195324259, 2007.
- [5] J. R. Andrews et N. Hogan, *Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator*, Proceedings of ASME Winter Meeting on Control of Manufacturing Processes and Robotic Systems, Boston, p. 243-251, 1983.
- [6] Dejanira Araiza-Illan et Tony J. Dodd, *Biologically Inspired Controller for the Autonomous Navigation of a Mobile Robot in an Evasion Task*, World Academy of Science, Engineering & Technology, vol. 4, n°8, p. 780, Août 2010.
- [7] Karl J. Åström, *Optimal control of Markov processes with incomplete state information*, Journal of Mathematical Analysis and Applications, vol. 10, n°1, p. 174–205, 1965.
- [8] Paul Bach-Y-Rita, *Brain mechanisms in sensory substitution*, New York : Academic Press, ISBN-13 : 9780120710409, 1972.
- [9] Tim Bailey et Hugh Durrant-Whyte, *Simultaneous Localisation and Mapping (SLAM) : Part II State of the Art*, IEEE Robotics and Automation Magazine, vol. 13, n°3, p. 108-117, Septembre 2006.
- [10] Ruzena Bajcsy, *Active perception*, Proceedings of the IEEE Special issue on Computer Vision, vol. 76, n°8, p. 996–1005, Août 1988.
- [11] José Baleia, Pedro Santana, et José Barata, *Self-Supervised Learning of Depth-Based Navigation Affordances from Haptic Cues*, Proceeding of IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) , p. 146-151 , Mai 2014.
- [12] Anna Berti et Francesca Frassinetti, *When far becomes near : Remapping of space by tool use*, Journal of Cognitive Neuroscience, vol. 12, n°3, p. 415–420, Mai 2000.

- [13] Douglas Blank, Deepak Kumar, Lisa Meeden, et James B. Marshall, *Bringing up robot : Fundamental mechanisms for creating a self-motivated, self-organizing architecture*, Cybernetics and Systems, vol. 36, n°2, p. 125–150, 2005.
- [14] Jona Boeddinghaus, Marco Ragni, Markus Knauff, et Bernhard Nebel, *Simulating spatial reasoning using ACT-R*, Proceedings of the Seventh International Conference on Cognitive Modeling, p. 62-67, 2006.
- [15] Véronique Boulenger, *Le langage et l'Action : Dynamique des liens fonctionnels unissant verbes d'action et contrôle moteur*, thèse de doctorat de Neuropsychologie, sous la direction de Tatjana Nazir, Université Lumière Lyon 2, Institut des Sciences Cognitives, 7 décembre 2006.
- [16] Rodney A. Brooks, *Intelligence without representation*, Artificial Intelligence, vol. 47, n°1-3, p. 139–159, Janvier 1991.
- [17] Thomas Buhrmann, Ezequiel A. Di Paolo, et Xabier Barandiaran, *A dynamical systems account of sensorimotor contingencies*, Frontiers in psychology, vol. 4(285), Mai 2013.
- [18] Vittorio Caggiano, Leonardo Fogassi, Giacomo Rizzolatti, Peter Thier, et Antonino Casile, *Mirror Neurons Differentially Encode the Peripersonal and Extrapersonal Space of Monkeys*, Science, vol. 324, n°5925, p. 403-406, Avril 2009.
- [19] Ken Caluwaerts, Mariacarla Staffa, Steve N'Guyen, Christophe Grand, Laurent Dollé, Antoine Favre-Félix, Benoît Girard, et Mehdi Khamassi, *A biologically inspired meta-control navigation system for the Psikharpax rat robot*, Bioinspiration & Biomimetics, vol. 7, n°2, p. 1-29, 2012.
- [20] Lucilla Cardinali, Claudio Brozzoli, et Alessandro Farnè, *Peripersonal Space and Body Schema : Two Labels for the Same Concept?*, Brain Topography, vol. 21, n°3-4, p. 252-260, Avril 2009.
- [21] Hande Çelikkanat, Erol Şahin, et Sinan Kalkan, *Recurrent Slow Feature Analysis for Developing Object Permanence in Robots*, IROS Workshop on Neuroscience and Robotics, 2013.
- [22] Anthony Chemero, *An outline of a theory of affordances*, Ecological Psychology, vol. 15, n°2, p. 181-195, 2003.
- [23] Eris Chinellato, Marco Antonelli, Beata J. Grzyb et Angel P. del Pobil, *Implicit mapping of the peripersonal space by gazing and reaching*, IEEE Transactions on Autonomous Mental Development, 2010.
- [24] William J. Clancey, *Situated Cognition : On Human Knowledge and Computer Representations*, Cambridge University Press, ISBN-13 : 9780521448710, 1997.
- [25] Ignasi Cos-Aguilera, Lola Cañamero, et Gillian Hayes, *Using a SOFM to learn Object Affordances*, Proceedings of the 5th Workshop of Physical Agents (WAF'04), p. 139-150, 2004.
- [26] Rodney M. Cotterill, *Cooperation of the basal ganglia, cerebellum, sensory cerebrum and hippocampus : possible implications for cognition, consciousness, intelligence and creativity*, Progress in Neurobiology, vol. 64, n°1, p. 1–33, Mai 2001.

-
- [27] Alan Cowey, M. Small, et S. Ellis, *Left visuo-spatial neglect can be worse in far than in near space*, *Neuropsychologia*, vol. 32, n° 9, p. 1059–1066, Septembre 1994.
- [28] Paul A. Crook et Gillian Hayes, *Learning in a state of confusion : Perceptual aliasing in grid world navigation*, *Proceedings of Towards Intelligent Mobile Robots, 4th British Conference on (Mobile) Robotics*, vol. 4, Août 2003.
- [29] James E. Cutting et Peter M. Vishton. Perceiving layout and knowing distances : the integration, relative potency and contextual use of different information about depth. In : *Handbook of perception and cognition, vol. 5 : Perception of space and motion*, p. 69-117, 1995.
- [30] Edward Deci et Richard M. Ryan, *Intrinsic Motivation and self-determination in human behavior*, New York : Plenum Press, ISBN-13 : 9780306420221, 1985.
- [31] Renaud Detry, Emre Baseski, Mila Popovic, Y. Touati, Norbert Kruger, Oliver Kroemer, Jan Peters, et Justus H. Piater, *Learning object-specific grasp affordance densities*, *Proceeding of IEEE 8th International Conference on Development and Learning*, p. 1-7, Juin 2009
- [32] Gary L. Drescher, *Made-up minds, a constructivist approach to artificial intelligence*, The MIT Press, ISBN-13 : 9780262517089, 2003.
- [33] Hubert L. Dreyfus, *Why Heideggerian AI failed et how fixing it would require making it more Heideggerian*, *Philosophical Psychology*, vol. 20, n°2, p. 247-268, Avril 2007.
- [34] Andreas K. Engel, Alexander Maye, Martin Kurthen, et Peter König, *Where's the action ? the pragmatic turn in cognitive science*, *Trends in Cognitive Sciences*, vol. 17, n°5, p. 202-209, Mai 2013.
- [35] Erdem Erdemir, Carl B. Frankel, Kazuhiko Kawamura, Stephen M. Gordon, Sean Thornton, et Baris Ulutas, *Towards a Cognitive Robot that Uses Internal Rehearsal to Learn Affordance Relations*, *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, p. 2016-2021, 2008.
- [36] Robert M. French, *Catastrophic forgetting in connectionist networks : Causes, consequences and solutions*, *Trends in Cognitive Sciences*, vol. 3, n°4, p. 138-135, 1999.
- [37] Sawa Fuke, Masaki Ogino, et Minoru Asada, *Body Image Constructed from Motor and Tactile Images with Visual Information*, *International Journal of Humanoid Robotics*, vol. 4, n°2, p. 347-364, Juin 2007.
- [38] Sawa Fuke, Masaki Ogino, et Minoru Asada, *Acquisition of the head-centered peri-personal spatial representation found in VIP neuron*, *IEEE Transactions on Autonomous Mental Development*, vol. 1, n°2, p. 131-140, 2009.
- [39] Vittorio Gallese, Luciano Fadiga, Leonardo Fogassi, et Giacomo Rizzolatti, *Action recognition in the premotor cortex*, *Brain*, vol. 119, n°2, p. 593-609, 1995.
- [40] Joseph Garnier, Olivier Georgeon, et Amelie Cordier, *Inferring Actions and Observations from Interactions*, *Second Workshop on Goal Reasoning at Advances in Cognitive Systems (ACS2013)*, p. 26-35, 2013.

- [41] Alberto Gatti, *On Distinguishing Sensorial and Eliciting Epistemic Actions and on the Relationship between Perceptive Structure of Body and Cognitive Processes*, XXVII Annual Conference of the Cognitive Science Society (CogSci2005), p. 773-778, 2005.
- [42] Simon Gay, Olivier Georgeon, et Jong Wook Kim, *Implementing Spatial Awareness in an Environment-Agnostic Agent*, 21st Annual Conference on Behavior Representation in Modeling and Simulation (BRIMS), p. 62-69, 2012.
- [43] Olivier L. Georgeon et David W. Aha, *The Radical Interactionism Conceptual Commitment*, Journal of Artificial General Intelligence, vol. 4, n°2, p. 31-36, 2013.
- [44] Olivier L. Georgeon, Mark A. Cohen, et Amélie V. Cordier, *A Model and Simulation of Early-stage Vision as a Developmental Sensorimotor Process*. Proceedings of the Conference on Artificial Intelligence Applications and Innovations (AIAI), p. 11-16, 2011.
- [45] Olivier L. Georgeon et James B. Marshall, *Demonstrating sensemaking emergence in artificial agents : A method and an example*, International Journal of Machine Consciousness, vol. 5, n°2, p. 131-144, 2013.
- [46] Olivier L. Georgeon et James B. Marshall, *The Small Loop Problem : A challenge for artificial emergent cognition*, Advances in Intelligent Systems and Computing, vol. 196, p. 137-144, 2013.
- [47] Olivier L. Georgeon, James B. Marshall, et Simon L. Gay, *Interactional Motivation in Artificial Systems : Between Extrinsic and Intrinsic Motivation*, Proceedings of the 2nd Joint IEEE International Conference on Development and Learning and on Epigenetic Robotics (EPIROB), p. 1-2, Novembre 2012.
- [48] Olivier L. Georgeon, James B. Marshall, et Riccardo Manzotti, *ECA : An enactivist cognitive architecture based on sensorimotor modeling*, Biologically Inspired Cognitive Architectures, vol. 6, p. 46-57, 2013.
- [49] Olivier L. Georgeon, Alain Mille, Thierry Bellet, Benoit Mathern, et Frank E. Ritter, *Supporting Activity Modeling from Activity Traces*, Expert Systems, vol. 29, n°3, p. 261-275, 2012.
- [50] Olivier L. Georgeon et Frank E. Ritter, *An intrinsically-motivated schema mechanism to model and simulate emergent cognition*, Cognitive Systems Research, vol. 15-16, p. 73-92, Mai 2011.
- [51] Olivier L. Georgeon et Ilias Sakellariou, *Designing Environment-Agnostic Agents*, Proceedings of ALA2012, Adaptive Learning Agents workshop, at AAMAS 2012, p. 25-32, 2012.
- [52] Olivier L. Georgeon, Christian Wolf, et Simon Gay, *An Enactive Approach to Autonomous Agent and Robot Learning*, IEEE Third Joint International Conference on Development and Learning and on Epigenetic Robotics (EPIROB2013), p. 1-6, 2013.
- [53] James J. Gibson. The theory of affordances. In *Perceiving, Acting, and Knowing : Toward an Ecological Psychology*, Eds. Robert Shaw and John Bransford, ISBN-13 : 9780470990148, 1977.

-
- [54] Michael S. A. Graziano, Charlotte S. R. Taylor, et Tirin Moore, *Complex movements evoked by microstimulation of precentral cortex*, *Neuron*, vol. 34, n°5, p. 841-851, Mai 2002.
- [55] Shane Griffith, Jivko Sinapov, Vladimir Sukhoy, et Alexander Stoytchev, *A Behavior-Grounded Approach to Forming Object Categories : Separating Containers From Non-containers*, *IEEE T. Autonomous Mental Development*, vol. 4, n°1, p. 54-69, Mars 2012.
- [56] Shane Griffith, Vladimir Sukhoy, Todd Wegter, et Alexander Stoytchev, *Object Categorization in the Sink : Learning Behavior-Grounded Object Categories with Water*, *Proceedings of the 2012 ICRA Workshop on Semantic Perception, Mapping, and Exploration (SPME)*, 2012.
- [57] Stephen Grossberg, Karen Roberts, Mario Aguilar, et Daniel Bullock, *A neural model of multimodal adaptive saccadic eye movement control by superior colliculus*, *Journal of Neuroscience*, vol. 17, p. 9706-9725, 1997.
- [58] Peter W. Halligan et John C. Marshall, *Left neglect for near but not far space in man*, *Nature*, vol. 350, p. 498-500, 1991.
- [59] Henry Head et Gordon Morgan Holmes, *Sensory disturbances from cerebral lesions*, *Brain*, vol. 34, n°2-3, p. 102-254, 1911.
- [60] Tucker Hermans, James M. Rehg, et Aaron F. Bobick, *Affordance Prediction via Learned Object Attributes*, *International Conference on Robotics and Automation (ICRA) : Workshop on Semantic Perception, Mapping, and Exploration*, Mai 2011.
- [61] Stephen M. Highstein, Richard R. Fay, et Arthur N. Popper, *The Vestibular System*, Springer Science & Business Media, ISBN-13 : 9780387983141, 2004.
- [62] Matej Hoffmann, Hugo Gravato Marques, Alejandro Hernandez Arieta, Hidenobu Sumioka, Max Lungarella, et Rolf Pfeifer, *Body Schema in Robotics : A Review*, *IEEE Transactions on Autonomous Mental Development*, vol. 2, n°4, p. 304-324, 2010.
- [63] Owen Holland, *The Future of Embodied Artificial Intelligence : Machine Consciousness ?*, *Embodied Artificial Intelligence*, *Lecture Notes in Computer Science*, vol. 3139, p. 37-53, 2004.
- [64] David Hume, *Treatise of human nature*, Oxford : David Fate Norton & Mary J. Norton, 1739, ed. 2000.
- [65] Edwin Hutchins, *Cognition in the wild*, MIT Press, ISBN-13 : 9780262082310, 1995.
- [66] Atsushi Iriki, Michio Tanaka, Shigeru Obayashi, et Yoshiaki Iwamura, *Self-images in the video monitor coded by monkey intraparietal neurons*, *Neuroscience Research*, vol. 40, n°2, p. 163-173, 2001.
- [67] Serena Ivaldi, Sao Mai Nguyen, Natalia Lyubova, Alain Droniou, Vincent Padois, David Filliat, Pierre-Yves Oudeyer, et Olivier Sigaud, *Object learning through active exploration*, *IEEE Transactions on Autonomous Mental Development*, vol. 6, n°1, p. 56-72, 2014.
- [68] Leslie P. Kaelbling, Michael L. Littman, et Anthony R. Cassandra, *Planning and acting in partially observable stochastic domains*, *Artificial Intelligence Journal*, vol. 101, n°1-2, p. 99-134, 1998.

- [69] Hans-Otto Karnath, Susanne Ferber, et Marc Himmelbach, *Spatial awareness is a function of the temporal not the posterior parietal lobe*, Nature, vol. 411, n°6840, p. 950-953, 2001.
- [70] Kazuhiko Kawamura, A.B. Koku, D.M. Wilkes, R.A. Peters II, et A. Sekmen, *Toward ego-centric navigation*, International Journal of Robotics and Automation, vol. 17, n°4, p. 135-145, 2002.
- [71] Oussama Khatib, *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*, Proceedings of IEEE International Conference on Robotics and Automation, vol. 2, p. 500-505, 1985.
- [72] Dongshin Kim, Jie Sun, Sang Min Oh, J. M. Rehg, et A. F. Bobick, *Traversability classification using unsupervised on-line visual learning for outdoor robot navigation*, Proceedings of 2006 IEEE International Conference on Robotics and Automation, p. 518-525, 2006.
- [73] Yoram Koren et Johann Borenstein, *Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation*, Proceedings of the IEEE Conference on Robotics and Automation, Sacramento, California, vol. 2, p. 1398-1404, 1991.
- [74] K. Kurata et J. Tanji, *Premotor cortex neurons in macaques : activity before distal and proximal forelimb movements*, Journal of Neuroscience, vol. 6, n°2, p. 403-11, 1986.
- [75] Michail G. Lagoudakis et Anthony S. Maida, *robot navigation with a polar neural map*, Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference, p. 965-, 1999.
- [76] Max Lungarella et Giorgio Metta, *Beyond Gazing, Pointing, and Reaching : A Survey of Developmental Robotics*, EPIROB, vol. 101, p. 81-89, 2003.
- [77] Max Lungarella, Giorgio Metta, Rolf Pfeifer, et Giulio Sandini, *Developmental robotics : a survey*, Connection Science, vol. 15, n°4, p. 151-190, Décembre 2003.
- [78] Monica Maranesi, Luca Bonini, et Leonardo Fogassi, *Cortical processing of object affordances for self and others' action*, Frontiers in Psychology, vol. 5 :538, 2014.
- [79] Angelo Maravita et Atsushi Iriki, *Tools for the body (schema)*, Trends in Cognitive Sciences, vol. 8, n°2, p. 79-86, 2004.
- [80] Denis Mareschal, Kim Plunkett, et Paul Harris, *Developing object permanence : A connectionist model*, Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society, p. 170-175, 1995.
- [81] Maja J. Mataric, edited by J. A. Meyer et S. W. Wilson, *Navigating With a Rat Brain : A Neurobiologically-Inspired Model for Robot Spatial Representation*, From Animals to Animats : Proceedings of the First International Conference on Simulation of Adaptive Behavior, p. 169-177, 1991.
- [82] Paul J. May, *The mammalian superior colliculus : laminar structure and connections*, Progress in Brain Research, vol. 151, p. 151-321, 2006.

-
- [83] Alexander Maye et Andreas K. Engel, *A discrete computational model of sensorimotor contingencies for object perception and control of behavior*, IEEE International Conference on Robotics and Automation (ICRA), p. 3810-3815, 2011.
- [84] Warren S. McCulloch et Walter Pitts, *A Logical Calculus of the Ideas Immanent in Nervous Activity*, Bulletin of Mathematical Biophysics, vol. 5, p. 115-133, 1943.
- [85] Jean-Arcady Meyer, Agnès Guillot, Benoît Girard, Mehdi Khamassi, Patrick Pirim, et Alain Berthoz, *The Psikharpax project : towards building an artificial rat*, Biomimetic Robotics, vol. 50, n°4, p. 211-223, 2005.
- [86] Francesco Mondada , Michael Bonani , Xavier Raemy , James Pugh , Christopher Cianci , Adam Klapotocz , Stéphane Magnenat , Jean-christophe Zufferey , Dario Floreano , et Alcherio Martinoli , *The e-puck, a robot designed for education in engineering*, Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, p. 59-65, 2009.
- [87] Luis Montesano et Manuel Lopes , *Learning grasping affordances from local visual descriptors*, IEEE 8TH International Conference on Development and Learning, p. 1-6, 2009.
- [88] Akira Murata, Luciano Fadiga, Leonardo Fogassi, Vittorio Gallese, Vassilis Raos, et Giacomo Rizzolatti, *Object Representation in the Ventral Premotor Cortex (Area F5) of the Monkey*, Journal of Neurophysiology, vol. 78, n°4, p. 2226-2230, 1997.
- [89] Allen Newell et Herbert A. Simon, *Computer science as empirical inquiry : symbols and search*, Communications of the ACM, vol. 19, n°3, p. 113-126, 1976.
- [90] Sao Mai Nguyen, Serena Ivaldi, Natalia Lyubova, Alain Droniou, Damien Gérardaux-Viret, David Filliat, Vincent Padois, Olivier Sigaud, et Pierre-Yves Oudeyer, *Learning to recognize objects through curiosity-driven manipulation with the iCub humanoid robot*, IEEE Third Joint International Conference on Development and Learning and Epigenetic Robotics (ICDL), p. 1-8, 2013.
- [91] Alva Noë, *Action in perception*, MIT Press, ISBN-13 : 9780262140881, 2004.
- [92] David P. M. Northmore. The Optic Tectum. In : *Encyclopedia of Fish Physiology : from Genome to Environment*, Academic Press, ISBN-13 : 9780123745453, 2011.
- [93] John O'Keefe et Jonathan Dostrovsky, *The hippocampus as a spatial map : preliminary evidence from unit activity in the freely moving rat*, Brain Research, vol. 34, n°1, p. 171-175, 1971.
- [94] John O'Keefe et Lynn Nadel, *The Hippocampus as a Cognitive Map*, Oxford University Press, ISBN-13 : 978-0198572060, 1978.
- [95] Douglas L. Oliver et Michael F. Huerta. Inferior and Superior Colliculi. In : *The Mammalian Auditory Pathway : Neuroanatomy*, New York : Springer-Verlag, ISBN-13 : 9780387976785, p. 168-221, 1992.
- [96] J. Kevin O'Regan, *Why Red Doesn't Sound Like a Bell : Understanding the feel of consciousness*, Oxford University Press USA, ISBN-13 : 9780199775224, 2011.

- [97] J. Kevin O'Regan, *How to Build a Robot that is Conscious and Feels*, Minds and Machines, vol. 22, n°2, p. 117–136, 2012.
- [98] J. Kevin O'Regan et A. Noë, *A sensorimotor account of vision and visual consciousness*, Behavioral and Brain Sciences, vol. 24, n°5, p. 939–1031, 2001.
- [99] Pierre-Yves Oudeyer, Frédéric Kaplan, et Verena V. Hafner, *Intrinsic Motivation Systems for Autonomous Mental Development*, IEEE Transactions on Evolutionary Computation, vol. 11, n°2, p. 265-286, 2007.
- [100] Rolf Pfeifer, *Building "Fungus Eaters" : Design Principles of Autonomous Agents*, Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior SAB96 (From Animals to Animats), p. 3-12, 1996.
- [101] Rolf Pfeifer, *Robots as cognitive tools*, International Journal of Cognition and Technology, vol. 1, n°1, p. 125–143, 2002.
- [102] Rolf Pfeifer et Josh C. Bongard, *How the Body Shapes the Way We Think : A New View of Intelligence*, MIT Press, ISBN-13 : 9780262162395, 2006.
- [103] Rolf Pfeifer et Christian Scheier , *From Perception to Action : The Right Direction ?*, Proceedings of From Perception to Action Conference, p. 1-11, 1994.
- [104] Rolf Pfeifer et Christian Scheier, *Sensory-motor coordination : The metaphor and beyond*, Robotics and Autonomous Systems, vol. 20, n°2, p. 157-178, Juin 1997.
- [105] Jean Piaget, *The construction of reality in the child*, Basic Books, New York, ISBN-13 : 978-0465014071, 1954.
- [106] David Pierce et Benjamin Kuipers, *Map learning with uninterpreted sensors and effectors*, Artificial Intelligence, vol. 92, n°1-2, p. 169-227, 1997.
- [107] Henri Poincaré, *La Science et l'Hypothèse*, Paris, éd. Flammarion, 1902.
- [108] Fred H. Previc, *The neuropsychology of 3-D space*, Psychological Bulletin, vol. 124, n°2, p. 123–164, Septembre 1998.
- [109] Jefferson Provost, Benjamin J. Kuipers, et Risto Miikkulainen, *Developing navigation behavior through self-organizing distinctive state abstraction*, Connection Science, vol. 18, n°2, p. 159-172, 2006.
- [110] Giacomo Rizzolatti, Maddalena Fabbri-Destro, et Luigi Cattaneo, *Mirror neurons and their clinical relevance*, Nature Clinical Practice Neurology, vol. 5, n°1, p. 24-34, 2009.
- [111] G. Rizzolatti, C. Scandolara, Maurizio Gentilucci, et Rosolino Camarda, *Response properties and behavioral modulation of 'mouth' neurons of the postarcuate cortex (area 6) in macaque monkeys*, Brain Research, vol. 225, n°2, p. 421-424, 1981.
- [112] Stuart Russell et Peter Norvig. Intelligent Agents. In : *Artificial Intelligence : A Modern Approach*, Prentice-Hall, ISBN-13 : 9780131038059, 1995.
- [113] Jürgen Schmidhuber, *Formal theory of creativity, fun, and intrinsic motivation*, IEEE Transactions on Autonomous Mental Development, vol. 2, n°3, p. 230-247, Septembre 2010.

-
- [114] Denis Sheynikhovich, Ricardo Chavarriaga, Thomas Strösslin, et Wolfram Gerstner, *Spatial Representation and Navigation in a Bio-inspired Robot*, Biomimetic Neural Learning for Intelligent Robots, Lecture Notes in Computer Science, Volume 3575, p. 245-264, 2005.
- [115] Satinder P. Singh, Andrew G. Barto, et Nuttapon Chentanez, *Intrinsically motivated reinforcement learning*. Advances in Neural Information Processing Systems, vol. 17, p. 1281-1288, 2005.
- [116] David M. Smith et Sheri J.Y. Mizumori, *Hippocampal Place Cells, Context, and Episodic Memory*, Hippocampus, vol. 16, n°9, p. 716-729, 2006.
- [117] Luc Steels, *The autotelic principle*, Embodied artificial intelligence, Lecture Notes in Computer Science, vol. 3139, p. 231-242, 2004.
- [118] John Stewart, Olivier Gapenne, et Ezequiel A. Di Paolo, *Enaction : Toward a new paradigm for cognitive science*, Bradford Books, ISBN-13 : 978-0262014601, 2010.
- [119] Thomas A. Stoffregen, *Affordances as properties of the animal environment system*, Ecological Psychology, vol. 15, n°2, p. 115-134, 2003.
- [120] Lucy A. Suchman, *Plans and situated actions : the problem of human-machine communication*, Cambridge University Press, 1987.
- [121] Ron Sun et C. Lee Giles, *Sequence Learning - Paradigms, Algorithms, and Applications*, Springer, ISBN-13 : 9783540415978, 2001.
- [122] Richard S. Sutton et A. Barto, *Reinforcement learning : An introduction*, Bradford Book, ISBN-13 : 978-0262193986, 1998.
- [123] Richard Sutton, Joseph Modayil, Michael Delp, Thomas Degris, Patrick M. Pilarski, Adam White, et Doina Precup, *Horde : A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction*, Proceeding of The 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), vol. 2, p. 761-768, 2011.
- [124] Jeffrey S. Taube, Robert U. Muller, et James B. Ranck Jr, *Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis*, The Journal of Neuroscience, vol.10, n°2, p. 420-436, Février 1990.
- [125] Kristinn R. Thórisson, Eric Nivel, Ricardo Sanz, et Pei Wang, *Approaches and Assumptions of Self-Programming in Achieving Artificial General Intelligence*, Journal of Artificial General Intelligence, vol. 3, n°3, p. 1-10, Février 2013.
- [126] Olivier Trullier, Sidney I. Wiener, Alain Berthoz, et Jean-Arcady Meyer, *Biologically-based artificial navigation systems : Review and prospects*, Progress in Neurobiology, vol. 51, n°5, p. 483-544, 1997.
- [127] Michael T. Turvey, *Affordances and prospective control : an outline of the ontology*, Ecological Psychology, vol. 4, n°3, p. 173-187, 1992.
- [128] Emre Uğur, Mehmet R. Doğar, Maya Çakmak, et Erol Şahin , *The learning and use of traversability affordance using range images on a mobile robot*, Proceeding of IEEE International Conference on Robotics and Automation, p. 1721-1726, 2007.

- [129] Francisco J. Varela, Evan Thompson, et Eleanor Rosch, *The embodied mind : Cognitive science and human experience*, MIT Press, ISBN-13 : 9780262720212, 1991.
- [130] David Vernon, Giorgio Metta, et Giulio Sandini, *A Survey of Artificial Cognitive Systems : Implications for the Autonomous Development of Mental Capabilities in Computational Agents*, IEEE transactions on evolutionary computation, vol. 11, n°2, p. 151-180, Avril 2007.

Cinquième partie

ANNEXES

Annexe A : résultats détaillés des expérimentations

1 Expérimentations du chapitre 9

Cette section donne le listing du contenu de la mémoire spatio-séquentielle dans les différentes configurations étudiées. Le tableau suivant donne la correspondance entre les symboles utilisés dans ce manuscrit et ceux manipulés par l'implémentation du système :

TABLE 1 – Correspondance des symboles

\square	= (-, :) (toucher devant un espace vide)
\blacksquare	= (-,) (toucher devant un mur)
\square	= (\, :) (toucher à droite un espace vide)
\blacktriangleright	= (\,) (toucher à droite un mur)
\triangleleft	= (/ , :) (toucher à gauche un espace vide)
\blacktriangleleft	= (/ ,) (toucher à gauche un mur)
\triangleright	= (>, 0) (avancer d'un pas)
\blacktriangleright	= (>, X) (se cogner contre un mur)
\cup	= (v, ') (tourner à droite de 90° face à un espace vide)
\blackcup	= (v, !) tourner à droite de 90° face un mur)
\triangle	= (^, ') (tourner à gauche de 90° face à un espace vide)
\blacktriangle	= (^, !) (tourner à gauche de 90° face à droite un mur)

mécanisme d'exploitation seul, suivi de la mémoire Spatio-séquentielle dans la configuration virage à gauche (Figure 9.13)

Cycle de décision 1 : (-, |)

===== final first level sequences

T : (-,)	T : (\, :) (-,)	T : (\,) (>, X)	T : (\,) (-,)	F : (-, :)
T : (>, X)	T : (-,) (-,)	T : (\, :) (v, ')	T : (\,) (^, ')	F : (>, 0)

===== final second level sequences

Annexe A : résultats détaillés des expérimentations

T : (-,|) T : (\, :)(-, |) T : (\, |)(>, X) T : (\, |)(-, |) F : (-, :)
T : (>, X) T : (-, |)(-, |) T : (\, :) (v, ') T : (\, |)(^, ') F : (>, 0)

===== final third level sequences

T : (-,|) T : (\, :)(-, |) T : (\, |)(>, X) T : (\, |)(-, |) F : (-, :)
T : (>, X) T : (-, |)(-, |) T : (\, :) (v, ') T : (\, |)(^, ') F : (>, 0)

=====

Cycle de décision 2 : (\, |)

===== final first level sequences

T : (v, !) T : (^, ')(-, :) T : (>, X) F : (v, ') F : (-, |)(\, :) F : (v, ')(>, 0)
T : (^, ') T : (^, ')(>, 0) T : (-, |) F : (^, !) F : (v, ')(-, :)

===== final second level sequences

T : (-, |) T : (\, :)(-, |) T : (\, :) (v, ') T : (^, ')(>, 0) F : (>, 0)
T : (v, !) T : (-, |)(-, |) T : (\, |)(-, |) F : (-, :) F : (-, |)(\, :)
T : (^, ') T : (\, |)(>, X) T : (\, |)(^, ') F : (v, ') F : (v, ')(-, :)
T : (>, X) T : (v, !)(-, |) T : (^, ')(-, :) F : (^, !) F : (v, ')(>, 0)

===== final third level sequences

T : (-, |) T : (\, :)(-, |) T : (\, :) (v, ') T : (^, ')(>, 0) F : (>, 0)
T : (v, !) T : (-, |)(-, |) T : (\, |)(-, |) F : (-, :) F : (-, |)(\, :)
T : (^, ') T : (\, |)(>, X) T : (\, |)(^, ') F : (v, ') F : (v, ')(-, :)
T : (>, X) T : (v, !)(-, |) T : (^, ')(-, :) F : (^, !) F : (v, ')(>, 0)

=====

Cycle de décision 3 : (^, ')

===== final first level sequences

T : (-, :) T : (>, 0)

===== final second level sequences

T : (-, :) F : (-, |) F : (\, :)(-, |) F : (\, |)(>, X) F : (\, |)(-, |)
T : (>, 0) F : (>, X) F : (-, |)(-, |) F : (\, :) (v, ') F : (\, |)(^, ')

===== final third level sequences

T : (-, :) F : (-, |) F : (\, :)(-, |) F : (\, |)(>, X) F : (\, |)(-, |)
T : (>, 0) F : (>, X) F : (-, |)(-, |) F : (\, :) (v, ') F : (\, |)(^, ')

=====

Cycle de décision 4 : (>, 0)

===== final first level sequences

===== final second level sequences

===== final third level sequences

=====

mécanisme d'exploitation seul, suivi de la mémoire Spatio-séquentielle dans la configuration voie sans issue (Figure 9.15)

Cycle de décision 1 : (-, |)

===== final first level sequences

T : (-, l)	T : (\, :) (-, l)	T : (\, l) (>, X)	T : (\, l) (-, l)	F : (-, :)
T : (>, X)	T : (-, l) (-, l)	T : (\, :) (v, ')	T : (/, :) (r, ')	F : (>, 0)

===== final second level sequences

T : (-, l)	T : (\, :) (-, l)	T : (\, l) (>, X)	T : (\, l) (-, l)	F : (-, :)
T : (>, X)	T : (-, l) (-, l)	T : (\, :) (v, ')	T : (/, :) (r, ')	F : (>, 0)

===== final third level sequences

T : (-, l)	T : (\, :) (-, l)	T : (\, l) (>, X)	T : (\, l) (-, l)	F : (-, :)
T : (>, X)	T : (-, l) (-, l)	T : (\, :) (v, ')	T : (/, :) (r, ')	F : (>, 0)

=====

Cycle de décision 2 : (\, l)

===== final first level sequences

T : (v, !)	T : (/, :) (r, ')	T : (/, l) (v, !)	F : (v, ')	F : (-, l) (/, l)
T : (r, ') (-, :)	T : (r, !) (\, l)	T : (>, X)	F : (-, l) (\, :)	F : (v, ' (>, 0)
T : (r, !) (-, l)	T : (-, l) (/, :)	T : (-, l)	F : (v, ' (-, :)	

===== final second level sequences

T : (-, l)	T : (-, l) (-, l)	T : (\, l) (-, l)	T : (r, !) (\, l)	F : (v, ')	F : (-, l) (/, l)
T : (v, !)	T : (\, l) (>, X)	T : (r, ' (-, :)	T : (-, l) (/, :)	F : (>, 0)	F : (v, ' (>, 0)
T : (>, X)	T : (v, !) (-, l)	T : (r, !) (-, l)	T : (/, l) (v, !)	F : (-, l) (\, :)	
T : (\, :) (-, l)	T : (\, :) (v, ')	T : (/, :) (r, ')	F : (-, :)	F : (v, ' (-, :)	

===== final third level sequences

T : (-, l)	T : (-, l) (-, l)	T : (\, l) (-, l)	T : (r, !) (\, l)	F : (v, ')	F : (-, l) (/, l)
T : (v, !)	T : (\, l) (>, X)	T : (r, ' (-, :)	T : (-, l) (/, :)	F : (>, 0)	F : (v, ' (>, 0)
T : (>, X)	T : (v, !) (-, l)	T : (r, !) (-, l)	T : (/, l) (v, !)	F : (-, l) (\, :)	
T : (\, :) (-, l)	T : (\, :) (v, ')	T : (/, :) (r, ')	F : (-, :)	F : (v, ' (-, :)	

=====

Cycle de décision 3 : (/, !)

===== final first level sequences

T : (v, !)	T : (r, !)	T : (v, !) (-, l)	F : (v, ')	F : (r, ')	F : (r, ' (>, 0)
------------	------------	-------------------	------------	------------	------------------

===== final second level sequences

T : (v, !)	T : (r, ' (-, :)	T : (r, !) (\, l)	F : (v, ')	F : (v, ' (-, :)	F : (r, ' (>, 0)
T : (r, !)	T : (r, !) (-, l)	T : (-, l) (/, :)	F : (r, ')	F : (-, l) (/, l)	
T : (v, !) (-, l)	T : (/, :) (r, ')	T : (/, l) (v, !)	F : (-, l) (\, :)	F : (v, ' (>, 0)	

===== final third level sequences

T : (v, !)	T : (r, ' (-, :)	T : (r, !) (\, l)	F : (v, ')	F : (v, ' (-, :)	F : (r, ' (>, 0)
T : (r, !)	T : (r, !) (-, l)	T : (-, l) (/, :)	F : (r, ')	F : (-, l) (/, l)	
T : (v, !) (-, l)	T : (/, :) (r, ')	T : (/, l) (v, !)	F : (-, l) (\, :)	F : (v, ' (>, 0)	

=====

Cycle de décision 4 : (v, !)

===== final first level sequences

Annexe A : résultats détaillés des expérimentations

T : (-, l)

===== final second level sequences

T : (-, l)	T : (\, :) (-, l)	T : (\, :) (v, ')	F : (>, 0)
T : (>, X)	T : (-, l) (-, l)	F : (-, :)	

===== final third level sequences

T : (-, l)	T : (\, :) (-, l)	T : (\, :) (v, ')	F : (>, 0)
T : (>, X)	T : (-, l) (-, l)	F : (-, :)	

=====

Cycle de décision 5 : (\, :)

===== final first level sequences

T : (v, ')	T : (v, ') (-, :)	T : (v, ') (>, 0)	F : (*, :) (-, :)	F : (/ , :) (*, ')	F : (-, l) (/ , :)
T : (-, l) (\, :)	T : (-, l) (/ , l)	F : (v, !)	F : (*, !) (-, l)	F : (*, !) (\, l)	F : (/ , l) (v, !)

===== final second level sequences

T : (v, ')	T : (-, l) (/ , l)	F : (v, !)	F : (/ , :) (*, ')	F : (/ , l) (v, !)
T : (-, l) (\, :)	T : (v, ') (>, 0)	F : (*, !) (-, :)	F : (*, !) (\, l)	
T : (v, ') (-, :)	T : (-, l)	F : (*, !) (-, l)	F : (-, l) (/ , :)	

===== final third level sequences

T : (-, l)	T : (-, l) (\, :)	T : (v, ') (-, :)	F : (-, :)	F : (*, !) (-, :)	F : (-, l) (/ , :)
T : (v, ')	T : (\, :) (-, l)	T : (-, l) (/ , l)	F : (v, !)	F : (*, !) (-, l)	F : (/ , l) (v, !)
T : (*, ')	T : (-, l) (-, l)	T : (v, ') (>, 0)	F : (*, !)	F : (/ , :) (*, ')	
T : (>, X)	T : (\, :) (v, ')	T : (*, ') (>, 0)	F : (>, 0)	F : (*, !) (\, l)	

=====

Cycle de décision 6 : (v, ')

===== final first level sequences

T : (-, :)	T : (>, 0)
------------	------------

===== final second level sequences

T : (-, :)	F : (-, l)	F : (\, :) (-, l)	F : (\, l) (>, X)	F : (\, l) (-, l)
T : (>, 0)	F : (>, X)	F : (-, l) (-, l)	F : (\, :) (v, ')	F : (/ , :) (*, ')

===== final third level sequences

T : (-, :)	F : (-, l)	F : (\, :) (-, l)	F : (\, l) (>, X)	F : (\, l) (-, l)
T : (>, 0)	F : (>, X)	F : (-, l) (-, l)	F : (\, :) (v, ')	F : (/ , :) (*, ')

=====

Cycle de décision 7 : (>, 0)

===== final first level sequences

===== final second level sequences

===== final third level sequences

=====

utilisation d'un mécanisme d'apprentissage avec des interactions composites de longueur 2, suivi de la mémoire Spatio-séquentielle dans la configuration virage à droite (Figure 9.20)

Cycle de décision 1 : (-, l)

===== final first level sequences

T : (-, l)	T : (-, l) (>, X)	T : (\, l) (-, l)	F : (-, :)	F : (/, l) (-, :)	F : (\, l) (>, 0)
T : (>, X)	T : (/, l) (-, l)	T : (*, !)\, l)	F : (>, 0)	F : (/, :) (-, :)	F : (-, :) (>, 0)
T : (/, l) (>, X)	T : (v, ')\, l)	T : (-, l) (-, l)	F : (-, :) (-, :)	F : (\, l) (-, :)	F : (/, l) (>, 0)
T : (>, X) (-, l)	T : (v, !)\, l)	T : (\, l) (>, X)	F : (v, !)\, l)	F : (*, !)\, l)	

===== final second level sequences

T : (-, l)	T : (\, :) (>, X)	T : (*, !)\, l)	F : (-, :) (-, :)	F : (/, :) (-, :)	F : (\, :) (>, 0)
T : (>, X)	T : (/, l) (-, l)	T : (\, :) (-, l)	F : (\, :) (-, :)	F : (\, l) (-, :)	
T : (*, ')\, l)	T : (v, ')\, l)	T : (-, l) (-, l)	F : (v, !)\, l)	F : (*, !)\, l)	
T : (/, l) (>, X)	T : (/, :) (-, l)	T : (\, l) (>, X)	F : (-, :) (-, :)	F : (\, l) (>, 0)	
T : (>, X) (-, l)	T : (v, !)\, l)	F : (-, :)	F : (/, l) (-, :)	F : (-, :) (>, 0)	
T : (-, l) (>, X)	T : (\, l) (-, l)	F : (>, 0)	F : (v, ')\, l)	F : (/, l) (>, 0)	

===== final third level sequences

T : (-, l)	T : (\, :) (>, X)	T : (*, !)\, l)	F : (-, :) (-, :)	F : (/, :) (-, :)	F : (\, :) (>, 0)
T : (>, X)	T : (/, l) (-, l)	T : (\, :) (-, l)	F : (\, :) (-, :)	F : (\, l) (-, :)	
T : (*, ')\, l)	T : (v, ')\, l)	T : (-, l) (-, l)	F : (v, !)\, l)	F : (*, !)\, l)	
T : (/, l) (>, X)	T : (/, :) (-, l)	T : (\, l) (>, X)	F : (-, :) (-, :)	F : (\, l) (>, 0)	
T : (>, X) (-, l)	T : (v, !)\, l)	F : (-, :)	F : (/, l) (-, :)	F : (-, :) (>, 0)	
T : (-, l) (>, X)	T : (\, l) (-, l)	F : (>, 0)	F : (v, ')\, l)	F : (/, l) (>, 0)	

=====

Cycle de décision 2 : (/, l)

===== final first level sequences

T : (/, l)	T : (>, X) (/, l)	T : (\, l) (*, !)	F : (/, :) (*, ')	F : (-, :) (/, :)	F : (\, :) (/, :)
T : (*, !)	T : (-, l) (/, l)	T : (>, X)	F : (*, ')\, l)	F : (-, l) (/, :)	F : (*, ')\, l)
T : (/, l) (/, l)	T : (\, :) (/, l)	T : (-, l)	F : (-, :) (*, ')	F : (\, l) (/, :)	F : (-, :)
T : (/, l) (*, !)	T : (-, l) (*, !)	F : (/, :)	F : (-, l) (*, ')	F : (-, l) (/, :)	F : (>, 0)
T : (*, !)\, l)	T : (\, l) (/, l)	F : (*, ')	F : (\, l) (*, ')	F : (>, X) (/, :)	

===== final second level sequences

T : (-, l)	T : (/, l) (>, X)	T : (v, !)\, l)	F : (-, :)	F : (*, ')\, l)	F : (*, !)\, l)
T : (/, l)	T : (>, X) (-, l)	T : (\, l) (-, l)	F : (/, :)	F : (-, l) (*, ')	F : (/, :) (/, :)
T : (*, !)	T : (-, l) (>, X)	T : (-, l) (*, !)	F : (*, ')	F : (\, l) (*, ')	F : (\, l) (>, 0)
T : (>, X)	T : (\, :) (>, X)	T : (*, !)\, l)	F : (>, 0)	F : (/, l) (-, :)	F : (-, :) (>, 0)
T : (-, :) (/, l)	T : (>, X) (/, l)	T : (-, :) (*, !)	F : (-, :) (-, :)	F : (v, ')\, l)	F : (/, l) (>, 0)
T : (/, l) (/, l)	T : (/, l) (-, l)	T : (\, :) (-, l)	F : (\, :) (-, :)	F : (/, :) (-, :)	F : (>, X) (/, :)
T : (\, :) (*, !)	T : (v, ')\, l)	T : (\, l) (/, l)	F : (v, !)\, l)	F : (-, :) (/, :)	F : (\, :) (/, :)
T : (/, l) (*, !)	T : (-, l) (/, l)	T : (-, l) (*, !)	F : (/, :) (*, ')	F : (-, l) (/, :)	F : (\, :) (*, ')
T : (*, !)\, l)	T : (\, :) (/, l)	T : (-, l) (-, l)	F : (*, ')\, l)	F : (\, l) (/, :)	F : (\, :) (>, 0)
T : (*, ')\, l)	T : (/, :) (-, l)	T : (\, l) (>, X)	F : (-, :) (*, ')	F : (\, l) (-, :)	F : (*, ')\, l)

===== final third level sequences

T : (-, l)	T : (/, l) (>, X)	T : (v, !)\, l)	F : (-, :)	F : (*, ')\, l)	F : (*, !)\, l)
T : (/, l)	T : (>, X) (-, l)	T : (\, l) (-, l)	F : (/, :)	F : (-, l) (*, ')	F : (/, :) (/, :)
T : (*, !)	T : (-, l) (>, X)	T : (-, l) (*, !)	F : (*, ')	F : (\, l) (*, ')	F : (\, l) (>, 0)
T : (>, X)	T : (\, :) (>, X)	T : (*, !)\, l)	F : (>, 0)	F : (/, l) (-, :)	F : (-, :) (>, 0)
T : (-, :) (/, l)	T : (>, X) (/, l)	T : (-, :) (*, !)	F : (-, :) (-, :)	F : (v, ')\, l)	F : (/, l) (>, 0)
T : (/, l) (/, l)	T : (/, l) (-, l)	T : (\, :) (-, l)	F : (\, :) (-, :)	F : (/, :) (-, :)	F : (>, X) (/, :)
T : (\, :) (*, !)	T : (v, ')\, l)	T : (\, l) (/, l)	F : (v, !)\, l)	F : (-, :) (/, :)	F : (\, :) (/, :)
T : (/, l) (*, !)	T : (-, l) (/, l)	T : (-, l) (*, !)	F : (/, :) (*, ')	F : (-, l) (/, :)	F : (\, :) (*, ')
T : (*, !)\, l)	T : (\, :) (/, l)	T : (-, l) (-, l)	F : (*, ')\, l)	F : (\, l) (/, :)	F : (\, :) (>, 0)
T : (*, ')\, l)	T : (/, :) (-, l)	T : (\, l) (>, X)	F : (-, :) (*, ')	F : (\, l) (-, :)	F : (*, ')\, l)

=====

Cycle de décision 3 : (\, :)

===== final first level sequences

Annexe A : résultats détaillés des expérimentations

T : (\, :)	T : (\, :) (v, !)	T : (-, :) (\, !)	F : (v, !)	F : (\, !) (\, !)
T : (v, !)	T : (>, X) (\, :)	T : (/ , :) (\, :)	F : (v, !) (-, !)	F : (-, !) (\, !)
T : (-, :) (\, :)	T : (-, !) (v, !)	T : (v, !) (>, 0)	F : (-, !) (v, !)	F : (/ , !) (\, !)
T : (/ , !) (v, !)	T : (/ , !) (\, :)	T : (/ , !)	F : (\, !) (v, !)	F : (>, X) (\, !)
T : (v, !) (-, :)	T : (\, :) (\, :)	F : (\, !)	F : (/ , :) (v, !)	F : (/ , :)

===== final second level sequences

T : (\, :)	T : (/ , !) (^, !)	T : (-, :) (^, !)	F : (v, !)	F : (-, :) (/ , :)	F : (>, X) (\, !)
T : (/ , !)	T : (^, !) (-, !)	T : (-, !) (\, :)	F : (^, !)	F : (/ , :) (v, !)	F : (>, X) (/ , :)
T : (v, !)	T : (>, X) (\, :)	T : (\, !) (/ , !)	F : (-, :) (v, !)	F : (-, !) (/ , :)	F : (\, :) (/ , :)
T : (^, !)	T : (>, X) (/ , !)	T : (\, !) (^, !)	F : (v, !) (-, !)	F : (\, !) (/ , :)	F : (\, !) (^, !)
T : (-, :) (\, :)	T : (-, !) (v, !)	T : (/ , :) (\, :)	F : (-, !) (v, !)	F : (\, !) (\, !)	F : (^, !) (>, 0)
T : (-, :) (/ , !)	T : (-, :) (v, !)	T : (/ , :) (v, !)	F : (\, !) (v, !)	F : (-, :) (\, !)	F : (-, :)
T : (/ , !) (/ , !)	T : (-, !) (/ , !)	T : (v, !) (>, 0)	F : (/ , :) (^, !)	F : (/ , :) (\, !)	F : (>, 0)
T : (/ , !) (v, !)	T : (\, :) (/ , !)	T : (>, X)	F : (^, !) (-, :)	F : (-, !) (\, !)	
T : (v, !) (-, :)	T : (-, !) (^, !)	T : (-, !)	F : (-, :) (^, !)	F : (/ , :) (/ , :)	
T : (\, :) (v, !)	T : (/ , !) (\, :)	F : (\, !)	F : (-, !) (^, !)	F : (/ , !) (\, !)	
T : (\, :) (^, !)	T : (\, :) (\, :)	F : (/ , :)	F : (\, !) (^, !)	F : (/ , !) (v, !)	

===== final third level sequences

T : (-, !)	T : (^, !) (\, !)	T : (\, !) (-, !)	F : (-, :)	F : (-, :) (^, !)	F : (^, !) (\, :)
T : (\, :)	T : (/ , !) (>, X)	T : (-, !) (^, !)	F : (\, !)	F : (^, !) (\, :)	F : (-, !) (\, !)
T : (/ , !)	T : (>, X) (-, !)	T : (^, !) (\, !)	F : (/ , :)	F : (-, !) (^, !)	F : (/ , :) (/ , :)
T : (v, !)	T : (-, !) (>, X)	T : (/ , !) (\, :)	F : (v, !)	F : (\, !) (^, !)	F : (/ , !) (\, !)
T : (^, !)	T : (>, X) (\, :)	T : (\, !) (\, :)	F : (^, !)	F : (/ , !) (-, :)	F : (/ , !) (v, !)
T : (>, X)	T : (\, :) (>, X)	T : (-, :) (^, !)	F : (>, 0)	F : (v, !) (/ , :)	F : (\, !) (>, 0)
T : (-, :) (\, :)	T : (>, X) (/ , !)	T : (\, !) (-, !)	F : (-, :) (-, :)	F : (/ , :) (-, :)	F : (>, X) (\, !)
T : (-, :) (/ , !)	T : (/ , !) (-, !)	T : (-, !) (\, :)	F : (\, :) (-, :)	F : (-, !) (/ , :)	F : (-, :) (>, 0)
T : (/ , !) (/ , !)	T : (-, !) (v, !)	T : (\, !) (/ , !)	F : (-, :) (v, !)	F : (/ , :) (v, !)	F : (/ , !) (>, 0)
T : (/ , !) (v, !)	T : (v, !) (/ , !)	T : (\, !) (^, !)	F : (v, !) (-, !)	F : (-, !) (/ , :)	F : (>, X) (/ , :)
T : (v, !) (-, :)	T : (-, :) (v, !)	T : (-, !) (-, !)	F : (\, !) (v, !)	F : (\, !) (/ , :)	F : (\, :) (/ , :)
T : (\, :) (v, !)	T : (-, !) (/ , !)	T : (/ , :) (\, :)	F : (\, !) (v, !)	F : (\, !) (-, :)	F : (\, :) (^, !)
T : (\, :) (^, !)	T : (\, :) (/ , !)	T : (\, !) (>, X)	F : (v, !) (/ , :)	F : (\, !) (\, !)	F : (\, :) (>, 0)
T : (/ , !) (^, !)	T : (/ , :) (-, !)	T : (/ , :) (v, !)	F : (/ , :) (^, !)	F : (-, :) (^, !)	F : (^, !) (>, 0)
T : (^, !) (-, !)	T : (v, !) (/ , !)	T : (v, !) (>, 0)	F : (^, !) (-, :)	F : (/ , :) (\, !)	

=====

Cycle de décision 4 : (v, !)

===== final first level sequences

T : (-, :)	T : (>, 0)
------------	------------

===== final second level sequences

T : (-, :)	T : (/ , !) (-, :)	T : (-, :) (>, 0)	F : (/ , !) (>, X)	F : (/ , :) (-, !)	F : (-, !) (-, !)
T : (>, 0)	T : (v, !) (/ , :)	T : (/ , !) (>, 0)	F : (>, X) (-, !)	F : (v, !) (/ , !)	F : (\, !) (>, X)
T : (-, :) (-, :)	T : (/ , :) (-, :)	T : (\, :) (>, 0)	F : (-, !) (>, X)	F : (-, !) (-, !)	F : (\, !) (-, !)
T : (\, :) (-, :)	T : (\, !) (-, :)	F : (-, !)	F : (\, :) (>, X)	F : (^, !) (\, !)	
T : (v, !) (/ , :)	T : (^, !) (\, :)	F : (>, X)	F : (/ , !) (-, !)	F : (\, :) (-, !)	
T : (^, !) (\, :)	T : (\, !) (>, 0)	F : (^, !) (\, !)	F : (v, !) (/ , !)	F : (/ , :) (>, X)	

===== final third level sequences

T : (-, :)	T : (/ , !) (-, :)	T : (-, :) (>, 0)	F : (^, !) (\, !)	F : (v, !) (/ , !)	F : (/ , :) (>, X)
T : (>, 0)	T : (v, !) (/ , :)	T : (/ , !) (>, 0)	F : (/ , !) (>, X)	F : (/ , :) (-, !)	F : (-, !) (-, !)
T : (-, :) (-, :)	T : (/ , :) (-, :)	T : (\, :) (>, 0)	F : (>, X) (-, !)	F : (v, !) (/ , !)	F : (\, !) (>, X)
T : (\, :) (-, :)	T : (\, !) (-, :)	T : (/ , !)	F : (-, !) (>, X)	F : (\, !) (-, !)	F : (/ , :)
T : (v, !) (/ , :)	T : (^, !) (\, :)	F : (-, !)	F : (\, :) (>, X)	F : (^, !) (\, !)	
T : (^, !) (\, :)	T : (\, !) (>, 0)	F : (>, X)	F : (/ , !) (-, !)	F : (\, :) (-, !)	

=====

Cycle de décision 5 : (>, 0)

===== final first level sequences

===== final second level sequences

===== final third level sequences

=====

utilisation d'un mécanisme d'apprentissage avec des interactions composites de longueur 2, suivi de la mémoire Spatio-séquentielle dans la configuration voie sans issue (Figure 9.22)

Cycle de décision 1 : (-,|)

===== final first level sequences

T : (-,)	T : (-,) (>, X)	T : (\,) (-,)	F : (-, :)	F : (/,) (-, :)	F : (\,) (>, 0)
T : (>, X)	T : (/,) (-,)	T : (*, !) (\,)	F : (>, 0)	F : (/, :) (-, :)	F : (-, :) (>, 0)
T : (/,) (>, X)	T : (v, ') (/,)	T : (-,) (-,)	F : (v, !) (-, :)	F : (\,) (-, :)	F : (/,) (>, 0)
T : (>, X) (-,)	T : (v, !) (/,)	T : (\,) (>, X)	F : (v, !) (/, :)	F : (*, !) (\, :)	

===== final second level sequences

T : (-,)	T : (\, :) (>, X)	T : (*, !) (\,)	F : (-, :) (-, :)	F : (/, :) (-, :)	F : (\, :) (>, 0)
T : (>, X)	T : (/,) (-,)	T : (\, :) (-,)	F : (\, :) (-, :)	F : (\,) (-, :)	
T : (*, ') (\,)	T : (v, ') (/,)	T : (-,) (-,)	F : (v, !) (/, :)	F : (*, !) (\, :)	
T : (/,) (>, X)	T : (/, :) (-,)	T : (\,) (>, X)	F : (*, ') (\, :)	F : (\,) (>, 0)	
T : (>, X) (-,)	T : (v, !) (/,)	F : (-, :)	F : (/,) (-, :)	F : (-, :) (>, 0)	
T : (-,) (>, X)	T : (\,) (-,)	F : (>, 0)	F : (v, ') (/, :)	F : (/,) (>, 0)	

===== final third level sequences

T : (-,)	T : (\, :) (>, X)	T : (*, !) (\,)	F : (-, :) (-, :)	F : (/, :) (-, :)	F : (\, :) (>, 0)
T : (>, X)	T : (/,) (-,)	T : (\, :) (-,)	F : (\, :) (-, :)	F : (\,) (-, :)	
T : (*, ') (\,)	T : (v, ') (/,)	T : (-,) (-,)	F : (v, !) (/, :)	F : (*, !) (\, :)	
T : (/,) (>, X)	T : (/, :) (-,)	T : (\,) (>, X)	F : (*, ') (\, :)	F : (\,) (>, 0)	
T : (>, X) (-,)	T : (v, !) (/,)	F : (-, :)	F : (/,) (-, :)	F : (-, :) (>, 0)	
T : (-,) (>, X)	T : (\,) (-,)	F : (>, 0)	F : (v, ') (/, :)	F : (/,) (>, 0)	

Cycle de décision 2 : (/,|)

===== final first level sequences

T : (/,)	T : (>, X) (/,)	T : (\,) (*, !)	F : (/, :) (*, ')	F : (-, :) (/, :)	F : (\, :) (/, :)
T : (*, !)	T : (-,) (/,)	T : (>, X)	F : (*, ') (-, :)	F : (-,) (/, :)	F : (*, ') (>, 0)
T : (/,) (/,)	T : (\, :) (/,)	T : (-,)	F : (-, :) (*, ')	F : (\,) (/, :)	F : (-, :)
T : (/,) (*, !)	T : (-,) (*, !)	F : (/, :)	F : (-,) (*, ')	F : (/, :) (/, :)	F : (>, 0)
T : (*, !) (-,)	T : (\,) (/,)	F : (*, ')	F : (\,) (*, ')	F : (>, X) (/, :)	

===== final second level sequences

T : (-,)	T : (/,) (>, X)	T : (v, !) (/,)	F : (-, :)	F : (*, ') (\, :)	F : (*, !) (\, :)
T : (/,)	T : (>, X) (-,)	T : (\,) (-,)	F : (/, :)	F : (-,) (*, ')	F : (/, :) (/, :)
T : (*, !)	T : (-,) (>, X)	T : (-,) (*, !)	F : (*, ')	F : (\,) (*, ')	F : (\,) (>, 0)
T : (>, X)	T : (\, :) (>, X)	T : (*, !) (\,)	F : (>, 0)	F : (/,) (-, :)	F : (-, :) (>, 0)
T : (-, :) (/,)	T : (>, X) (/,)	T : (-, :) (*, !)	F : (-, :) (-, :)	F : (v, ') (/, :)	F : (/,) (>, 0)
T : (/,) (/,)	T : (/,) (-,)	T : (\, :) (-,)	F : (\, :) (-, :)	F : (/, :) (-, :)	F : (>, X) (/, :)
T : (\, :) (*, !)	T : (v, ') (/,)	T : (\,) (/,)	F : (v, !) (/, :)	F : (-, :) (/, :)	F : (\, :) (/, :)
T : (/,) (*, !)	T : (-,) (/,)	T : (\,) (*, !)	F : (/, :) (*, ')	F : (-,) (/, :)	F : (\, :) (*, ')
T : (*, !) (-,)	T : (\, :) (/,)	T : (-,) (-,)	F : (*, ') (-, :)	F : (\,) (/, :)	F : (\,) (>, 0)
T : (*, ') (\,)	T : (/, :) (-,)	T : (\,) (>, X)	F : (-, :) (*, ')	F : (\,) (-, :)	F : (*, ') (>, 0)

===== final third level sequences

T : (-,)	T : (/,) (>, X)	T : (v, !) (/,)	F : (-, :)	F : (*, ') (\, :)	F : (*, !) (\, :)
T : (/,)	T : (>, X) (-,)	T : (\,) (-,)	F : (/, :)	F : (-,) (*, ')	F : (/, :) (/, :)
T : (*, !)	T : (-,) (>, X)	T : (-,) (*, !)	F : (*, ')	F : (\,) (*, ')	F : (\,) (>, 0)
T : (>, X)	T : (\, :) (>, X)	T : (*, !) (\,)	F : (>, 0)	F : (/,) (-, :)	F : (-, :) (>, 0)
T : (-, :) (/,)	T : (>, X) (/,)	T : (-, :) (*, !)	F : (-, :) (-, :)	F : (v, ') (/, :)	F : (/,) (>, 0)
T : (/,) (/,)	T : (/,) (-,)	T : (\, :) (-,)	F : (\, :) (-, :)	F : (/, :) (-, :)	F : (>, X) (/, :)
T : (\, :) (*, !)	T : (v, ') (/,)	T : (\,) (/,)	F : (v, !) (/, :)	F : (-, :) (/, :)	F : (\, :) (/, :)
T : (/,) (*, !)	T : (-,) (/,)	T : (\,) (*, !)	F : (/, :) (*, ')	F : (-,) (/, :)	F : (\, :) (*, ')
T : (*, !) (-,)	T : (\, :) (/,)	T : (-,) (-,)	F : (*, ') (-, :)	F : (\,) (/, :)	F : (\,) (>, 0)
T : (*, ') (\,)	T : (/, :) (-,)	T : (\,) (>, X)	F : (-, :) (*, ')	F : (\,) (-, :)	F : (*, ') (>, 0)

Annexe A : résultats détaillés des expérimentations

Cycle de décision 3 : (\,|)

===== final first level sequences

T : (\,)	T : (/,:)(v,!)	T : (>,X)(\,)	F : (-,:)(\,)	F : (-,)(v,')	F : (v,')(>,0)
T : (v,!)	T : (\,)(\,)	T : (/,)	F : (/,)(v,')	F : (/,)(\,)	F : (-,')
T : (v,!)(-,)	T : (-,)(\,)	T : (-,)	F : (v,')(-,:)	F : (\,)(\,)	F : (/,:)
T : (-,)(v,!)	T : (/,)(\,)	F : (\,)	F : (\,)(v,')	F : (-,)(\,)	
T : (\,)(v,!)	T : (/,)(v,!)	F : (v,')	F : (>,X)(\,)	F : (/,:)(\,)	

===== final second level sequences

T : (\,)	T : (/,:)(-,)	T : (-,)(\,)	F : (-,')	F : (-,)(v,')	F : (\,)(/,:)
T : (/,)	T : (-,)(-,)	T : (\,)(/,:)	F : (-,:)(\,)	F : (-,:)(v,')	F : (\,)(-,)
T : (v,')	T : (>,X)(/,:)	T : (/,)(\,)	F : (/,)(v,')	F : (-,)(/,:)	F : (/,:)(v,')
T : (-,)	T : (/,:)(v,')	T : (/,)(v,')	F : (v,')(-,:)	F : (-,)(/,:)	F : (-,)(>,0)
T : (-,:)(/,:)	T : (-,)(/,:)	T : (\,)(-,)	F : (\,)(v,')	F : (\,)(/,:)	F : (v,')(>,0)
T : (/,)(/,:)	T : (\,)(/,:)	T : (>,X)(\,)	F : (/,:)(-,)	F : (/,)(\,)	F : (-,)
T : (-,)(v,')	T : (-,)(-,)	T : (-,)	F : (-,)(-,)	F : (\,)(\,)	F : (>,0)
T : (v,')(-,)	T : (\,)(\,)	T : (>,X)	F : (-,)(-,)	F : (-,)(\,)	
T : (-,)(v,')	T : (-,)(\,)	F : (\,)	F : (-,)(-,)	F : (/,:)(/,:)	
T : (\,)(v,')	T : (/,:)(\,)	F : (/,:)	F : (\,)(-,)	F : (>,X)(/,:)	
T : (\,)(-,)	T : (-,)(-,)	F : (v,')	F : (>,X)(\,)	F : (/,:)(\,)	

===== final third level sequences

T : (-,)	T : (-,')(\,)	T : (-,')(\,)	F : (\,)	F : (-,')(\,)	F : (-,')(\,)
T : (\,)	T : (/,)(>,X)	T : (\,)(\,)	F : (/,:)	F : (-,)(-,)	F : (-,)(\,)
T : (/,)	T : (>,X)(-,)	T : (-,')(\,)	F : (v,')	F : (\,)(-,)	F : (/,:)(/,:)
T : (v,')	T : (-,)(>,X)	T : (/,)(\,)	F : (-,')	F : (>,X)(\,)	F : (\,)(>,0)
T : (-,)	T : (\,)(>,X)	T : (-,')(-,)	F : (>,0)	F : (-,)(v,')	F : (-,')(>,0)
T : (>,X)	T : (>,X)(/,:)	T : (\,)(-,)	F : (-,')(-,)	F : (/,)(-,)	F : (/,)(>,0)
T : (-,')(/,:)	T : (/,)(-,)	T : (-,)(\,)	F : (-,)(\,)	F : (-,')(\,)	F : (-,')(\,)
T : (/,)(/,:)	T : (v,')(/,:)	T : (\,)(/,:)	F : (\,)(-,)	F : (v,')(/,:)	F : (/,:)(\,)
T : (-,')(\,)	T : (/,:)(v,')	T : (/,)(\,)	F : (/,)(v,')	F : (/,:)(-,)	F : (\,)(/,:)
T : (v,')(-,)	T : (-,)(/,:)	T : (/,)(v,')	F : (v,')(-,)	F : (-,')(/,:)	F : (-,')(-,)
T : (-,)(v,')	T : (\,)(/,:)	T : (\,)(-,)	F : (\,)(v,')	F : (-,)(/,:)	F : (\,)(>,0)
T : (\,)(v,')	T : (/,:)(-,)	T : (>,X)(\,)	F : (v,')(/,:)	F : (\,)(/,:)	F : (/,:)(v,')
T : (\,)(-,)	T : (v,')(/,:)	T : (-,)(-,)	F : (/,:)(-,)	F : (\,)(-,)	F : (-,')(>,0)
T : (/,)(-,)	T : (\,)(-,)	T : (\,)(>,X)	F : (-,')(-,)	F : (/,)(\,)	F : (/,)(\,)
T : (-,')(-,)	T : (-,)(-,)	F : (-,')	F : (-,')(-,)	F : (\,)(\,)	F : (v,')(>,0)

Cycle de décision 4 : (v,!)

===== final first level sequences

T : (-,|)

===== final second level sequences

T : (-,)	T : (\,)(>,X)	T : (-,')(\,)	F : (-,')(-,)	F : (/,:)(-,)	F : (\,)(>,0)
T : (>,X)	T : (/,)(-,)	T : (\,)(-,)	F : (\,)(-,)	F : (\,)(-,)	
T : (-,')(\,)	T : (v,')(/,:)	T : (-,)(-,)	F : (v,')(/,:)	F : (-,)(\,)	F : (-,')(\,)
T : (/,)(>,X)	T : (/,:)(-,)	T : (\,)(>,X)	F : (-,')(\,)	F : (\,)(>,0)	
T : (>,X)(-,)	T : (v,')(/,:)	F : (-,')	F : (/,)(-,)	F : (-,')(>,0)	
T : (-,)(>,X)	T : (\,)(-,)	F : (>,0)	F : (v,')(/,:)	F : (/,)(>,0)	

===== final third level sequences

T : (-,)	T : (\,)(>,X)	T : (-,')(\,)	F : (>,0)	F : (v,')(/,:)	F : (/,)(>,0)
T : (>,X)	T : (/,)(-,)	T : (\,)(-,)	F : (-,')(-,)	F : (/,:)(-,)	F : (\,)(>,0)
T : (-,')(\,)	T : (v,')(/,:)	T : (-,)(-,)	F : (\,)(-,)	F : (\,)(-,)	F : (/,:)
T : (/,)(>,X)	T : (/,:)(-,)	T : (\,)(>,X)	F : (v,')(/,:)	F : (-,')(\,)	
T : (>,X)(-,)	T : (v,')(/,:)	T : (/,)	F : (-,')(\,)	F : (\,)(>,0)	
T : (-,)(>,X)	T : (\,)(-,)	F : (-,')	F : (/,)(-,)	F : (-,')(>,0)	

=====

Cycle de décision 5 : ($\setminus, :$)

===== final first level sequences

T : ($\setminus, :$)	T : ($v, ')$ ($-, :$)	T : ($/, $) ($\setminus, :$)	T : ($v, ')$ ($>, 0$)	F : ($-, $) ($v, !$)	F : ($-, $) ($\setminus, $)
T : ($v, ')$	T : ($\setminus, :$) ($v, ')$	T : ($\setminus, :$) ($\setminus, :$)	F : ($\setminus, $)	F : ($\setminus, $) ($v, !$)	F : ($/, $) ($\setminus, $)
T : ($-, :$) ($\setminus, :$)	T : ($>, X$) ($\setminus, :$)	T : ($-, $) ($\setminus, :$)	F : ($v, !$)	F : ($/, :$) ($v, !$)	F : ($/, $) ($v, !$)
T : ($/, $) ($v, ')$	T : ($-, $) ($v, ')$	T : ($/, :$) ($\setminus, :$)	F : ($v, !$) ($-, $)	F : ($\setminus, $) ($\setminus, $)	F : ($>, X$) ($\setminus, $)

===== final second level sequences

T : ($\setminus, :$)	T : ($>, X$) ($\setminus, :$)	T : ($/, :$) ($\setminus, :$)	F : ($v, !$)	F : ($\setminus, $) ($\setminus, $)	F : ($>, X$) ($\setminus, $)
T : ($v, ')$	T : ($-, $) ($v, ')$	T : ($/, :$) ($v, ')$	F : ($-, :$) ($v, !$)	F : ($-, :$) ($\setminus, $)	F : ($-, :$)
T : ($-, :$) ($\setminus, :$)	T : ($-, :$) ($v, ')$	T : ($v, ')$ ($>, 0$)	F : ($v, !$) ($-, $)	F : ($/, :$) ($\setminus, $)	F : ($>, 0$)
T : ($/, $) ($v, ')$	T : ($/, $) ($\setminus, :$)	T : ($>, X$)	F : ($-, $) ($v, !$)	F : ($-, $) ($\setminus, $)	
T : ($v, ')$ ($-, :$)	T : ($\setminus, :$) ($\setminus, :$)	T : ($-, $)	F : ($\setminus, $) ($v, !$)	F : ($/, $) ($\setminus, $)	
T : ($\setminus, :$) ($v, ')$	T : ($-, $) ($\setminus, :$)	F : ($\setminus, $)	F : ($/, :$) ($v, !$)	F : ($/, $) ($v, !$)	

===== final third level sequences

T : ($-, $)	T : ($>, X$) ($-, $)	T : ($\setminus, $) ($-, $)	T : ($v, ')$ ($>, 0$)	F : ($\setminus, $) ($v, !$)	F : ($/, :$) ($\setminus, $)
T : ($\setminus, :$)	T : ($-, $) ($>, X$)	T : ($^-, !$) ($\setminus, $)	F : ($-, :$)	F : ($v, !$) ($/, :$)	F : ($^-, !$) ($\setminus, :$)
T : ($v, ')$	T : ($>, X$) ($\setminus, :$)	T : ($/, $) ($\setminus, :$)	F : ($\setminus, $)	F : ($^-, !$) ($\setminus, :$)	F : ($-, $) ($\setminus, $)
T : ($>, X$)	T : ($\setminus, :$) ($>, X$)	T : ($\setminus, :$) ($\setminus, :$)	F : ($v, !$)	F : ($/, $) ($-, :$)	F : ($/, $) ($\setminus, $)
T : ($-, :$) ($\setminus, :$)	T : ($/, $) ($-, $)	T : ($\setminus, :$) ($-, $)	F : ($>, 0$)	F : ($v, ')$ ($/, :$)	F : ($/, $) ($v, !$)
T : ($/, $) ($v, ')$	T : ($-, $) ($v, ')$	T : ($-, $) ($\setminus, :$)	F : ($-, :$) ($-, :$)	F : ($/, :$) ($-, :$)	F : ($\setminus, $) ($>, 0$)
T : ($v, ')$ ($-, :$)	T : ($v, ')$ ($/, $)	T : ($-, $) ($-, $)	F : ($\setminus, :$) ($-, :$)	F : ($/, :$) ($v, !$)	F : ($>, X$) ($\setminus, $)
T : ($\setminus, :$) ($v, ')$	T : ($-, :$) ($v, ')$	T : ($/, :$) ($\setminus, :$)	F : ($-, :$) ($v, !$)	F : ($\setminus, $) ($-, :$)	F : ($-, :$) ($>, 0$)
T : ($^-, !$) ($\setminus, $)	T : ($/, :$) ($-, $)	T : ($\setminus, $) ($>, X$)	F : ($v, !$) ($-, $)	F : ($\setminus, $) ($\setminus, $)	F : ($/, $) ($>, 0$)
T : ($/, $) ($>, X$)	T : ($v, !$) ($/, $)	T : ($/, :$) ($v, ')$	F : ($-, $) ($v, !$)	F : ($-, :$) ($\setminus, $)	F : ($\setminus, :$) ($>, 0$)

=====

Cycle de décision 6 : ($v, ')$)

===== final first level sequences

T : ($-, :$)	T : ($>, 0$)
----------------	----------------

===== final second level sequences

T : ($-, :$)	T : ($/, $) ($-, :$)	T : ($-, :$) ($>, 0$)	F : ($/, $) ($>, X$)	F : ($/, :$) ($-, $)	F : ($-, $) ($-, $)
T : ($>, 0$)	T : ($v, ')$ ($/, :$)	T : ($/, $) ($>, 0$)	F : ($>, X$) ($-, $)	F : ($v, !$) ($/, $)	F : ($\setminus, $) ($>, X$)
T : ($-, :$) ($-, :$)	T : ($/, :$) ($-, :$)	T : ($\setminus, :$) ($>, 0$)	F : ($-, $) ($>, X$)	F : ($\setminus, $) ($-, $)	
T : ($\setminus, :$) ($-, :$)	T : ($\setminus, $) ($-, :$)	F : ($-, $)	F : ($\setminus, :$) ($>, X$)	F : ($^-, !$) ($\setminus, $)	
T : ($v, !$) ($/, :$)	T : ($^-, !$) ($\setminus, :$)	F : ($>, X$)	F : ($/, $) ($-, $)	F : ($\setminus, :$) ($-, $)	
T : ($^-, !$) ($\setminus, :$)	T : ($\setminus, $) ($>, 0$)	F : ($^-, !$) ($\setminus, $)	F : ($v, ')$ ($/, $)	F : ($/, :$) ($>, X$)	

===== final third level sequences

T : ($-, :$)	T : ($/, $) ($-, :$)	T : ($-, :$) ($>, 0$)	F : ($^-, !$) ($\setminus, $)	F : ($v, ')$ ($/, $)	F : ($/, :$) ($>, X$)
T : ($>, 0$)	T : ($v, ')$ ($/, :$)	T : ($/, $) ($>, 0$)	F : ($/, $) ($>, X$)	F : ($/, :$) ($-, $)	F : ($-, $) ($-, $)
T : ($-, :$) ($-, :$)	T : ($/, :$) ($-, :$)	T : ($\setminus, :$) ($>, 0$)	F : ($>, X$) ($-, $)	F : ($v, !$) ($/, $)	F : ($\setminus, $) ($>, X$)
T : ($\setminus, :$) ($-, :$)	T : ($\setminus, $) ($-, :$)	T : ($/, $)	F : ($-, $) ($>, X$)	F : ($\setminus, $) ($-, $)	F : ($/, :$)
T : ($v, !$) ($/, :$)	T : ($^-, !$) ($\setminus, :$)	F : ($-, $)	F : ($\setminus, :$) ($>, X$)	F : ($^-, !$) ($\setminus, $)	
T : ($^-, !$) ($\setminus, :$)	T : ($\setminus, $) ($>, 0$)	F : ($>, X$)	F : ($/, $) ($-, $)	F : ($\setminus, :$) ($-, $)	

=====

Cycle de décision 7 : ($>, 0$)

===== final first level sequences

===== final second level sequences

===== final third level sequences

=====

utilisation d'un mécanisme d'apprentissage avec des interactions composites de longueur 3, suivi de la mémoire Spatio-séquentielle dans la configuration virage à droite (Figure 9.25)

Cycle de décision 1 : (-,|)

===== final first level sequences

T : (-,)	T : (*,!) \,)	T : (\, :) (/,) (-,)	F : (>, 0)	F : (/, :) (>, 0)	F : (\, :) (\, :) (-, :)
T : (>, X)	T : (\, :) (-,)	T : (/,) (\, :) (-,)	F : (-, :) (-, :)	F : (\,) (>, 0)	F : (\, :) (*, !) (\, :)
T : (/,) (>, X)	T : (v, !) (-,) (*, !)	T : (v, !) (-,) (/,)	F : (\, :) (-, :)	F : (-, :) (*, !) (/,)	F : (v, !) (-,) (*, !)
T : (>, X) (-,)	T : (*, !) (-,) (v, !)	T : (\, :) (\, :) (>, X)	F : (v, !) (/, :)	F : (\, :) (/,) (-, :)	F : (/,) (/,) (>, 0)
T : (-,) (>, X)	T : (\, :) (v, !) (/,)	T : (/,) (/,) (-,)	F : (*, !) (\, :)	F : (-, :) (>, 0)	F : (v, !) (-, :) (*, !)
T : (\, :) (>, X)	T : (*, !) (-,) (\,)	T : (\, :) (*, !) (\,)	F : (/,) (-, :)	F : (/,) (>, 0)	F : (*, !) (v, !)
T : (/,) (-,)	T : (/, :) (\,) (-,)	T : (\, :) (*, !) (/, :)	F : (v, !) (/, :)	F : (\,) (*, !) (/,)	F : (\, :) (*, !) (v, !)
T : (v, !) (/,)	T : (\, :) (/,) (>, X)	T : (v, !) (>, X) (/,)	F : (/, :) (-, :)	F : (\, :) (*, !) (/,)	F : (*, !) (>, X) (\, :)
T : (/,) (v, !) (\, :)	T : (/, :) (/,) (>, X)	T : (*, !) (v, !)	F : (v, !) (-,) (/, :)	F : (/,) (\, :) (-, :)	F : (/,) (*, !) (-, :)
T : (/, :) (-,)	T : (-,) (-,)	T : (\, :) (*, !) (v, !)	F : (/,) (v, !) (\,)	F : (*, !) (-,) (\, :)	F : (\,) (v, !) (*, !)
T : (/,) (v, !) (/,)	T : (\,) (*, !) (/, :)	T : (v, !) (*, !)	F : (\, :) (v, !) (/, :)	F : (/,) (/,) (-, :)	
T : (\,) (-,)	T : (\, :) (\, :) (-,)	F : (-, :)	F : (*, !) (\, :)	F : (\, :) (/,) (>, 0)	

===== final second level sequences

T : (-,)	T : (*, !) (-,) (v, !)	T : (\, :) (*, !) (\,)	F : (/, :) (-, :)	F : (/,) (\, :) (-, :)	F : (*, !) (-, :) (\, :)
T : (>, X)	T : (\, :) (v, !) (/,)	T : (\, :) (*, !) (/, :)	F : (v, !) (-,) (/, :)	F : (*, !) (/,) (\, :)	F : (v, !) (\,) (*, !)
T : (*, !) \,)	T : (*, !) (-,) (\,)	T : (/, :) (/, :) (-,)	F : (/,) (v, !) (\,)	F : (\, :) (>, 0)	F : (v, !) (-, :) (*, !)
T : (/,) (>, X)	T : (/, :) (\,) (-,)	T : (v, !) (>, X) (/,)	F : (\, :) (v, !) (/, :)	F : (*, !) (-,) (\, :)	F : (*, !) (/,) (v, !)
T : (>, X) (-,)	T : (v, !) (>, X)	T : (/,) (/,) (-,)	F : (\,) (*, !) (\, :)	F : (/,) (/,) (-, :)	F : (/,) (/,) (>, X) (/, :)
T : (-,) (>, X)	T : (\, :) (/,) (>, X)	T : (\, :) (*, !) (v, !)	F : (\,) (-, :)	F : (\, :) (/,) (>, 0)	F : (*, !) (v, !)
T : (\, :) (>, X)	T : (/,) (/,) (>, X)	T : (*, !) (>, X) (\,)	F : (*, !) (\, :)	F : (/,) (v, !) (/, :)	F : (\, :) (*, !) (v, !)
T : (/,) (-,)	T : (-,) (-,)	T : (*, !) (-,)	F : (/, :) (-, :)	F : (\, :) (\, :) (\, :)	F : (\, :) (\, :) (\, :)
T : (v, !) (/,)	T : (\,) (*, !) (/, :)	T : (v, !) (*, !)	F : (/, :) (>, 0)	F : (v, !) (\, :) (/, :)	F : (*, !) (*, !)
T : (/,) (v, !) (\, :)	T : (\, :) (\, :) (-,)	F : (-, :)	F : (\,) (v, !) (/, :)	F : (/, :) (\,) (>, 0)	F : (v, !) (*, !)
T : (/, :) (-,)	T : (\, :) (/,) (-,)	F : (>, 0)	F : (>, 0)	F : (\, :) (*, !) (\, :)	F : (\, :) (*, !) (v, !)
T : (v, !) (/,)	T : (\,) (>, X)	F : (-, :) (-, :)	F : (-, :) (*, !) (/,)	F : (*, !) (/, :) (\, :)	F : (v, !) (v, !) (\,)
T : (/,) (v, !) (/,)	T : (/,) (\, :) (-,)	F : (\, :) (-, :)	F : (\, :) (/,) (-, :)	F : (v, !) (-,) (*, !)	
T : (\,) (-,)	T : (v, !) (-,) (/,)	F : (-, :) (>, 0)	F : (-, :) (>, 0)	F : (/,) (/,) (>, 0)	
T : (*, !) \,)	T : (/,) (\, :) (>, X)	F : (*, !) \, :)	F : (/,) (>, 0)	F : (/, :) (/, :) (-, :)	
T : (\, :) (-,)	T : (\, :) (\, :) (>, X)	F : (/,) (-, :)	F : (\,) (*, !) (/,)	F : (v, !) (\, :) (*, !)	
T : (v, !) (-,) (*, !)	T : (/,) (/,) (-,)	F : (v, !) (/, :)	F : (\, :) (*, !) (/,)	F : (/, :) (/, :) (>, 0)	

===== final third level sequences

T : (-,)	T : (\, :) (-,) (v, !)	T : (*, !) (-, :) (\,)	F : (/,) (v, !) (\,)	F : (\, :) (*, !) (/,)	F : (*, !) (/, :) (v, !)
T : (>, X)	T : (\, :) (v, !) (/,)	T : (\, :) (*, !) (/, :)	F : (\,) (/, :) (-, :)	F : (/,) (\, :) (-, :)	F : (v, !) (\, :) (*, !)
T : (*, !) \,)	T : (*, !) (-,) (\,)	T : (*, !) (-, :) (v, !)	F : (\, :) (v, !) (/, :)	F : (v, !) (\,) (/, :)	F : (/, :) (/, :) (>, 0)
T : (/,) (>, X)	T : (\,) (/, :) (-,)	T : (/, :) (/, :) (-,)	F : (/, :) (v, !) (\,)	F : (*, !) (/,) (\, :)	F : (*, !) (-, :) (\, :)
T : (>, X) (-,)	T : (/, :) (\,) (-,)	T : (/, :) (\,) (>, X)	F : (\,) (*, !) (\, :)	F : (\, :) (>, 0)	F : (v, !) (\,) (*, !)
T : (-,) (>, X)	T : (/, :) (>, X)	T : (\,) (\,) (>, X)	F : (\,) (-, :)	F : (*, !) (-,) (\, :)	F : (v, !) (-, :) (*, !)
T : (\, :) (>, X)	T : (\, :) (/,) (>, X)	T : (v, !) (>, X) (/,)	F : (*, !) (\, :)	F : (/,) (/,) (-, :)	F : (*, !) (-,) (v, !)
T : (/,) (-,)	T : (/,) (/,) (>, X)	T : (*, !) (v, !)	F : (/,) (*, !) (\, :)	F : (\, :) (\, :) (>, 0)	F : (*, !) (/,) (v, !)
T : (v, !) (/,)	T : (-,) (-,)	T : (\, :) (*, !) (v, !)	F : (/, :) (*, !) (\, :)	F : (\, :) (/,) (>, 0)	F : (/,) (\, :) (>, 0)
T : (/,) (v, !) (\, :)	T : (\,) (*, !) (/, :)	T : (*, !) (>, X) (\,)	F : (-, :) (*, !) (\, :)	F : (/,) (v, !) (/, :)	F : (v, !) (>, X) (/, :)
T : (/, :) (-,)	T : (/, :) (*, !) (\,)	T : (*, !) (*, !)	F : (v, !) (-, :) (/, :)	F : (\, :) (\, :) (-, :)	F : (*, !) (v, !)
T : (v, !) (/,)	T : (\, :) (\, :) (-,)	T : (/,) (*, !) (v, !)	F : (/, :) (>, 0)	F : (-, :) (\, :) (>, 0)	F : (\, :) (*, !) (v, !)
T : (v, !) (-, :) (/,)	T : (\, :) (/,) (-,)	T : (v, !) (*, !)	F : (\,) (v, !) (/, :)	F : (/, :) (\,) (-, :)	F : (/,) (*, !) (v, !)
T : (/, :) (v, !) (\, :)	T : (\,) (>, X)	F : (-, :)	F : (\,) (>, 0)	F : (v, !) (\, :) (/, :)	F : (*, !) (>, X) (\, :)
T : (/,) (v, !) (/,)	T : (/, :) (/, :) (>, X)	F : (>, 0)	F : (-, :) (\,) (-, :)	F : (\, :) (>, 0) (\,)	F : (\, :) (>, 0) (\,)
T : (\,) (-,)	T : (/,) (\, :) (-,)	F : (-, :) (-, :)	F : (-, :) (*, !) (/,)	F : (/, :) (\,) (>, 0)	F : (v, !) (*, !)
T : (*, !) \,)	T : (\,) (\,) (-,)	F : (\, :) (-, :)	F : (\, :) (/,) (-, :)	F : (\,) (\,) (>, 0)	F : (\,) (\,) (*, !)
T : (\,) (*, !) \,)	T : (v, !) (-,) (/,)	F : (v, !) (/, :)	F : (-, :) (>, 0)	F : (\, :) (*, !) (\, :)	F : (v, !) (v, !) (\,)
T : (\, :) (-,)	T : (/,) (\, :) (>, X)	F : (*, !) (\, :)	F : (\,) (\,) (-, :)	F : (*, !) (/, :) (\, :)	
T : (v, !) (-,) (*, !)	T : (\, :) (\, :) (>, X)	F : (/,) (-, :)	F : (/,) (>, 0)	F : (v, !) (-,) (*, !)	
T : (*, !) (-,) (v, !)	T : (/,) (/,) (-,)	F : (v, !) (/, :)	F : (\,) (*, !) (/,)	F : (*, !) (-, :) (v, !)	
T : (/,) (*, !) \,)	T : (\, :) (*, !) (\,)	F : (\, :) (-, :)	F : (\,) (-, :) (\,)	F : (\,) (/,) (>, 0)	
T : (\,) (v, !) (/,)	T : (v, !) (-, :) (*, !)	F : (v, !) (-,) (/, :)	F : (/,) (-, :) (*, !)	F : (/, :) (/, :) (-, :)	

Cycle de décision 2 : (/,|)

===== final first level sequences

Annexe A : résultats détaillés des expérimentations

F : (-, l)	F : (/, l)(v, ')(\, :)	F : (\, :)(v, ')(/, l)	F : (\, :)(/, l)(-, l)	F : (\, :)(-, ')(/, :)	F : (-, ')(>, X)(\, l)
F : (>, X)	F : (/, :)(-, l)	F : (-, ')(-, l)(\, l)	F : (\, l)(>, X)	F : (-, ')(/, l)(\, l)	F : (-, ')(-, ')
F : (-, ')(\, l)	F : (v, ')(/, l)	F : (/, :)(\, l)(-, l)	F : (v, ')(\, :)(/, l)	F : (/, :)(/, :)(-, l)	F : (v, ')(-, ')
F : (/, l)(>, X)	F : (/, l)(v, ')(/, l)	F : (/, :)(>, X)	F : (/, l)(\, :)(-, l)	F : (v, ')(\, l)(-, ')	F : (\, l)(v, ')(-, ')
F : (>, X)(-, l)	F : (\, l)(-, l)	F : (\, :)(-, l)(>, X)	F : (v, ')(-, l)(/, l)	F : (v, ')(\, :)(-, ')	F : (v, ')(v, ')(\, :)
F : (-, l)(>, X)	F : (-, ')(\, l)	F : (/, l)(/, l)(>, X)	F : (/, l)(\, :)(>, X)	F : (-, ')(/, l)(v, ')	F : (/, :)
F : (\, :)(>, X)	F : (\, :)(-, l)	F : (-, l)(-, l)	F : (\, :)(\, :)(>, X)	F : (v, ')(>, X)(/, l)	F : (/, l)(\, l)
F : (/, l)(-, l)	F : (v, ')(-, l)(-, ')	F : (\, l)(-, ')(/, :)	F : (/, l)(/, l)(-, l)	F : (-, ')(-, ')(v, ')	F : (/, l)(v, ')
F : (v, ')(/, l)	F : (-, ')(-, l)(v, ')	F : (\, :)(\, :)(-, l)	F : (\, :)(-, ')(\, l)	F : (\, :)(-, ')(v, ')	F : (-, :)(-, ')

===== final third level sequences

T : (-, :)	T : (\, l)(>, 0)	T : (/, :)(\, l)(-, :)	F : (-, l)	F : (/, l)(v, ')	F : (-, l)(-, ')(/, :)
T : (\, :)	T : (-, :)(-, :)(/, l)	T : (v, ')(\, :)(/, :)	F : (\, l)	F : (/, l)(-, :)(v, ')	F : (/, l)(/, l)(v, ')
T : (/, l)	T : (-, :)(v, ')(\, :)	T : (-, :)(-, l)(\, :)	F : (/, :)	F : (\, :)(-, l)(-, ')	F : (\, :)(-, l)(\, l)
T : (v, ')	T : (-, l)(-, ')(/, l)	T : (\, :)(>, 0)(\, l)	F : (v, ')	F : (-, :)(-, :)(/, :)	F : (-, l)(\, :)(-, ')
T : (-, ')	T : (-, :)(\, l)(-, :)	T : (-, :)(-, :)(-, ')	F : (-, ')	F : (/, :)(\, l)(-, l)	F : (\, :)(-, ')(\, l)
T : (>, 0)	T : (-, :)(-, ')(/, l)	T : (/, :)(\, l)(>, 0)	F : (>, X)	F : (/, :)(>, X)	F : (>, X)(-, l)(\, l)
T : (-, :)(-, :)	T : (/, l)(-, :)(\, :)	T : (\, l)(\, l)(>, 0)	F : (-, :)(v, ')	F : (>, X)(\, l)	F : (-, ')(-, l)(\, :)
T : (-, :)(\, :)	T : (\, :)(/, l)(-, :)	T : (-, l)(\, :)(-, ')	F : (v, ')(-, l)	F : (\, l)(-, :)(-, ')	F : (v, ')(-, :)(-, ')
T : (\, :)(-, :)	T : (-, :)(-, :)(\, :)	T : (\, :)(-, ')(\, l)	F : (-, l)(v, ')	F : (\, l)(-, :)(-, ')	F : (-, :)(-, :)(-, ')
T : (-, :)(/, l)	T : (\, :)(\, :)(>, 0)	T : (-, :)(\, :)(/, l)	F : (\, l)(v, ')	F : (\, :)(/, l)(>, X)	F : (\, :)(/, l)(\, :)
T : (/, l)(/, l)	T : (\, l)(\, l)(-, :)	T : (/, l)(/, l)(>, 0)	F : (/, :)(-, ')	F : (-, l)(-, l)	F : (-, ')(\, :)(/, :)
T : (/, l)(v, ')	T : (\, l)(-, :)(/, l)	T : (/, l)(/, l)(\, :)	F : (-, ')(-, ')	F : (>, X)(/, :)	F : (/, l)(/, l)(\, l)
T : (v, ')(-, :)	T : (/, l)(>, 0)	T : (\, l)(\, l)(-, ')	F : (-, :)(-, ')	F : (/, :)(-, l)(\, l)	F : (-, :)(/, l)(\, l)
T : (\, :)(v, ')	T : (>, X)(-, l)(/, l)	T : (\, l)(\, l)(-, ')	F : (-, l)(-, ')	F : (\, l)(-, ')(/, :)	F : (-, :)(/, l)(\, l)
T : (v, ')(/, :)	T : (-, l)(>, X)(\, :)	T : (\, l)(\, l)(-, ')	F : (-, ')(\, l)	F : (\, l)(-, ')(/, :)	F : (v, ')(\, l)(/, l)
T : (-, ')(\, :)	T : (\, l)(-, ')(/, l)	T : (\, l)(\, l)(-, ')	F : (\, l)(\, l)	F : (-, :)(\, l)(/, :)	F : (-, :)(-, :)(\, l)
T : (\, :)(-, ')	T : (/, :)(\, :)	T : (/, :)(\, :)(\, :)	F : (/, l)(>, X)	F : (\, :)(/, :)	F : (-, :)(/, :)
T : (/, l)(-, ')	T : (\, l)(-, :)(\, l)	T : (-, :)(\, l)(/, l)	F : (>, X)(-, l)	F : (/, l)(-, :)(\, l)	F : (-, l)(>, X)(/, :)
T : (-, ')(-, l)	T : (\, l)(-, :)(-, ')	T : (v, ')(-, l)(-, ')	F : (-, l)(>, X)	F : (/, :)(-, ')(\, l)	F : (-, ')(-, :)(-, ')
T : (>, X)(\, :)	T : (-, ')(\, l)(\, l)	T : (-, :)(\, l)(\, :)	F : (-, :)(>, X)	F : (-, l)(-, ')(\, l)	F : (-, l)(-, ')(-, ')
T : (-, l)(v, ')	T : (/, l)(-, :)(-, ')	T : (-, ')(-, :)(v, ')	F : (/, l)(-, l)	F : (\, :)(\, :)(-, l)	F : (\, :)(-, ')(/, :)
T : (-, l)(\, :)	T : (-, l)(\, :)(/, l)	T : (v, ')(/, :)(\, :)	F : (v, ')(/, l)	F : (\, :)(/, l)(-, l)	F : (-, :)(-, ')(/, :)
T : (-, :)(v, ')	T : (\, :)(\, :)(/, l)	T : (/, :)(/, l)(>, 0)	F : (-, l)(/, :)	F : (/, :)(/, :)(>, X)	F : (\, :)(\, :)(/, :)
T : (v, ')(/, :)	T : (-, l)(-, l)(\, :)	T : (/, :)(\, :)(-, :)	F : (-, :)(-, ')	F : (>, X)(-, l)(-, ')	F : (-, ')(-, :)(v, ')
T : (/, l)(-, :)	T : (-, l)(>, X)(/, l)	T : (v, ')(>, 0)	F : (/, l)(v, ')(\, :)	F : (>, X)(-, l)(\, :)	F : (>, X)(-, l)(\, l)
T : (-, l)(/, l)	T : (-, ')(/, :)(>, X)	T : (v, ')(/, :)(v, ')	F : (/, :)(-, l)	F : (\, :)(>, X)(/, :)	F : (/, :)(/, :)(-, l)
T : (\, :)(/, l)	T : (/, :)(>, X)(\, :)	T : (>, X)(\, :)(-, ')	F : (/, :)(-, l)(v, ')	F : (-, l)(-, l)(/, :)	F : (/, :)(\, l)(>, X)
T : (v, ')(-, l)(/, :)	T : (-, l)(/, :)(\, :)	T : (v, ')(\, :)(-, ')	F : (v, ')(/, l)	F : (\, :)(-, ')	F : (>, X)(\, :)(-, ')
T : (/, l)(v, ')(\, l)	T : (v, ')(/, l)(\, :)	T : (/, :)(/, l)(>, 0)	F : (-, ')(-, :)(/, l)	F : (-, l)(-, l)(\, l)	F : (-, l)(-, l)(\, :)
T : (\, l)(/, :)(-, :)	T : (-, l)(-, l)(/, l)	T : (-, ')(-, :)(\, :)	F : (\, l)(/, :)	F : (-, l)(/, :)(\, l)	F : (v, ')(\, l)(-, ')
T : (\, :)(v, ')(/, :)	T : (/, l)(\, :)(-, :)	T : (v, ')(\, l)(-, ')	F : (-, :)(-, :)(v, ')	F : (\, l)(\, l)(/, :)	F : (v, ')(\, :)(-, ')
T : (-, l)(-, ')	T : (\, :)(>, X)(/, l)	T : (v, ')(-, :)(-, ')	F : (/, :)(v, ')(\, :)	F : (-, l)(>, X)(\, l)	F : (v, ')(/, l)(\, l)
T : (/, :)(v, ')(\, l)	T : (-, l)(-, l)(-, ')	T : (-, ')(-, l)(v, ')	F : (/, l)(v, ')(/, l)	F : (v, ')(\, :)(/, l)	F : (v, ')(/, l)(v, ')
T : (\, l)(-, ')(\, :)	T : (v, ')(\, l)(/, :)	T : (-, :)(\, l)(-, ')	F : (-, :)(v, ')(\, l)	F : (/, l)(\, :)(-, l)	F : (\, l)(\, l)(>, X)
T : (\, l)(-, :)	T : (/, l)(/, l)(v, ')	T : (-, :)(\, l)(-, ')	F : (\, l)(-, l)	F : (-, l)(-, l)(-, ')	F : (v, ')(/, :)(-, ')
T : (/, l)(\, :)	T : (/, l)(/, l)(v, ')	T : (>, X)(-, l)(-, ')	F : (-, l)(v, ')(/, l)	F : (-, ')(\, l)(/, :)	F : (-, ')(/, l)(v, ')
T : (\, :)(\, :)	T : (\, :)(>, 0)	T : (/, :)(\, :)(v, ')	F : (-, ')(\, l)	F : (\, l)(\, l)(-, l)	F : (>, X)(\, l)(>, X)
T : (-, l)(v, ')(\, :)	T : (-, ')(-, l)(\, :)	T : (/, :)(>, 0)(>, X)	F : (\, l)(-, ')(\, l)	F : (-, :)(\, l)(-, ')	F : (-, :)(\, l)(\, :)
T : (-, :)(-, ')	T : (>, X)(/, :)(\, :)	T : (>, 0)(>, X)(/, :)	F : (\, l)(\, l)	F : (\, l)(\, l)(-, ')	F : (>, 0)(>, X)(/, l)
T : (-, ')(\, :)	T : (/, :)(-, l)(\, :)	T : (-, :)(-, :)(v, ')	F : (-, :)(\, l)	F : (/, :)(\, :)(>, X)	F : (/, :)(>, 0)(>, 0)
T : (\, :)(-, l)(-, ')	T : (/, :)(v, ')	T : (/, l)(\, :)(>, 0)	F : (/, :)(\, l)	F : (\, :)(-, l)(/, :)	F : (v, ')(>, X)
T : (-, l)(\, :)	T : (/, :)(-, l)(v, ')	T : (v, ')(>, X)(/, :)	F : (\, :)(-, l)	F : (>, X)(\, :)(/, :)	F : (>, X)(\, :)(/, l)
T : (>, X)(-, l)(\, :)	T : (/, l)(/, l)(-, :)	T : (-, ')(v, ')	F : (v, ')(-, l)(-, ')	F : (-, l)(\, :)(/, :)	F : (-, ')(v, ')(\, l)
T : (/, l)(-, ')(\, :)	T : (-, l)(-, l)(v, ')	T : (-, ')(>, X)	F : (-, l)(\, l)	F : (/, :)(/, :)(v, ')	F : (-, ')(v, ')
T : (/, l)(-, :)(v, ')	T : (\, :)(\, :)(>, 0)	T : (-, ')(v, ')(\, :)	F : (-, ')(-, l)(v, ')	F : (v, ')(-, l)(/, l)	F : (\, :)(-, ')(v, ')
T : (/, :)(-, ')(\, :)	T : (\, :)(/, l)(>, 0)	T : (\, :)(-, ')(v, ')	F : (/, l)(-, ')(\, l)	F : (-, l)(/, :)(v, ')	F : (-, ')(>, X)(\, l)
T : (-, :)(-, ')(\, :)	T : (>, X)(\, :)(/, l)	T : (/, l)(-, ')(v, ')	F : (\, l)(v, ')(/, l)	F : (/, l)(\, :)(>, X)	F : (-, ')(-, ')
T : (\, :)(-, l)(/, l)	T : (v, ')(/, l)(v, ')	T : (-, ')(>, X)(\, :)	F : (\, :)(-, l)(v, ')	F : (\, :)(\, :)(>, X)	F : (/, l)(-, ')(v, ')
T : (v, ')(-, :)(/, :)	T : (/, l)(v, ')(/, :)	T : (-, ')(-, ')	F : (/, :)(/, :)	F : (>, X)(-, l)(v, ')	F : (v, ')(-, ')
T : (/, :)(>, 0)	T : (>, X)(-, l)(v, ')	T : (v, ')(-, ')	F : (\, :)(v, ')(/, l)	F : (/, l)(/, l)(-, l)	F : (v, ')(-, ')(/, :)
T : (\, l)(/, l)	T : (\, :)(\, :)(-, :)	T : (\, l)(v, ')(-, ')	F : (-, ')(-, l)(\, l)	F : (/, :)(>, X)(\, l)	F : (\, l)(v, ')(-, ')
T : (\, l)(v, ')(/, :)	T : (-, :)(/, l)(v, ')	T : (v, ')(-, ')(/, l)	F : (\, l)(/, :)(-, l)	F : (/, :)(/, :)(\, l)	F : (v, ')(v, ')(\, :)
T : (\, l)(-, ')	T : (-, :)(\, :)(>, 0)	T : (v, ')(v, ')(\, l)	F : (/, l)(\, l)	F : (v, ')(/, :)(\, l)	

=====

Cycle de décision 4 : (v, ')

===== final first level sequences

===== final second level sequences

===== final third level sequences

Annexe A : résultats détaillés des expérimentations

F : (°,°) (-, :) (v, °) F : (v, °) (\, |) (/, :) F : (v, °) (\, |) (°, °) F : (°, °) (/, |) (v, °) F : (\, :) (°, °) (v, °) F : (v, °) (°, °)
 F : (/, |) (/, |) (>, 0) F : (v, °) (\, :) (°, °) F : (v, °) (-, :) (°, °) F : (/, |) (\, :) (>, 0) F : (/, |) (°, °) (v, °) F : (/, |) (v, °) (°, °)
 F : (/, :) (/, :) (-, :) F : (/, :) (/, :) (>, 0) F : (°, °) (-, |) (v, °) F : (v, °) (>, X) (/, :) F : (°, °) (>, X) (\, :) F : (\, |) (v, °) (°, °)
 F : (°, °) (/, :) (v, °) F : (°, °) (-, :) (\, :) F : (v, °) (/, :) (>, X) F : (°, °) (v, °) F : (\, |) (°, °) (v, °) F : (/, :) (v, °) (v, °)

=====

Cycle de décision 2 : (/, |)

===== final first level sequences

T : (/,)	T : (-, :) (-, :) (/,)	T : (-,)	F : (-, :) (°, °)	F : (\,) (\,) (/, :)	F : (v, °) (/, :)
T : (°, °)	T : (>, X) (-,) (/,)	T : (v, °) (/,)	F : (-,) (°, °)	F : (-,) (-,) (°, °)	F : (\, :) (-, :)
T : (-, :) (/,)	T : (-,) (\, :) (/,)	T : (°, °) (\,)	F : (\,) (°, °)	F : (\,) (\,) (°, °)	F : (/,) (-, :)
T : (/,) (/,)	T : (\, :) (\, :) (/,)	T : (/,) (>, X)	F : (-, :) (/, :)	F : (\, :) (-,) (/, :)	F : (v, °) (/, :)
T : (/,) (°, °)	T : (-,) (-,) (/,)	T : (\, :) (-,) (/,)	F : (>,) (/, :)	F : (>, X) (\, :) (/, :)	F : (/,) (-,) (>, 0)
T : (°, °) (-,)	T : (-,) (-,) (°, °)	T : (/,) (-,)	F : (\,) (/, :)	F : (-,) (\, :) (/, :)	F : (°, °) (v, °)
T : (-,) (/,)	T : (>, X) (\, :) (/,)	T : (°, °) (v, °)	F : (/, :) (/, :)	F : (°, °) (>, 0)	F : (v, °) (°, °)
T : (\, :) (/,)	T : (\,) (\,) (/,)	T : (\,) (-,)	F : (\, :) (-,) (°, °)	F : (>, X) (-,) (°, °)	F : (>, X) (-,) (°, °)
T : (-,) (°, °)	T : (>, X) (-,) (°, °)	F : (/, :)	F : (-, :) (-, :) (/, :)	F : (\, :) (\, :) (/, :)	F : (-, :)
T : (\, :) (-,) (°, °)	T : (°, °) (>, X)	F : (°, °)	F : (\, :) (/, :)	F : (-, :)	F : (°, °)
T : (\, :) (-,) (/,)	T : (\,) (-,) (/,)	F : (/, :) (°, °)	F : (>, X) (-,) (/, :)	F : (°, °) (\, :)	F : (°, °) (\, :)
T : (\,) (/,)	T : (>, X)	F : (°, °) (-, :)	F : (-,) (-,) (/, :)	F : (>, 0)	

===== final second level sequences

T : (-,)	T : (v, °) (-,) (°, °)	T : (\,) (\,) (/,)	F : (°, °) (-, :)	F : (-, :) (>, 0)	F : (°, °) (/, :) (\, :)
T : (/,)	T : (°, °) (-,) (v, °)	T : (°, °) (/,) (\,)	F : (-, :) (°, °)	F : (\,) (\,) (-, :)	F : (°, °) (>, 0)
T : (°, °)	T : (/,) (°, °) (\,)	T : (/, :) (/, :) (-,)	F : (°, °) (\, :)	F : (/,) (>, 0)	F : (v, °) (-,) (°, °)
T : (>, X)	T : (\, :) (-,) (/,)	T : (>, X) (-,) (°, °)	F : (-,) (°, °)	F : (>, X) (/, :)	F : (>, X) (-,) (°, °)
T : (-, :) (/,)	T : (\, :) (v, °) (/,)	T : (°, °) (-,) (v, °)	F : (\,) (°, °)	F : (/,) (v, °) (/, :)	F : (/,) (v, °) (-, :)
T : (/,) (/,)	T : (°, °) (-,) (\,)	T : (°, °) (v, °)	F : (/,) (-, :)	F : (\, :) (/, :)	F : (°, °) (-, :) (v, °)
T : (\,) (°, °)	T : (\,) (/, :) (-,)	T : (°, °) (>, X)	F : (v, °) (/, :)	F : (\,) (/,) (-, :)	F : (/,) (/,) (>, 0)
T : (/,) (°, °)	T : (\,) (/,)	T : (\,) (\,) (\,)	F : (°, °) (-, :)	F : (>, X) (-,) (/, :)	F : (°, °) (/, :) (-, :)
T : (°, °) (-,)	T : (\,) (°, °)	T : (/,) (°, °) (v, °)	F : (-, :) (/, :)	F : (-,) (-,) (/, :)	F : (v, °) (\,) (/, :)
T : (°, °) (\,)	T : (-, :) (-, :) (/,)	T : (v, °) (°, °)	F : (-,) (/, :)	F : (\, :) (°, °)	F : (v, °) (/,) (\,)
T : (/,) (>, X)	T : (/, :) (>, X)	T : (v, °) (°, °) (/,)	F : (v, °) (-,) (/, :)	F : (\,) (\,) (/, :)	F : (v, °) (\,) (°, °)
T : (>, X) (-,)	T : (\,) (-, :) (/,)	T : (v, °) (v, °) (\,)	F : (\,) (/, :)	F : (/,) (\, :) (-, :)	F : (°, °) (-,) (\, :)
T : (-,) (>, X)	T : (\, :) (/,) (>, X)	T : (/, :) (v, °) (v, °)	F : (\,) (/, :) (-, :)	F : (-,) (-,) (°, °)	F : (v, °) (-, :) (°, °)
T : (\, :) (>, X)	T : (/,) (/,) (>, X)	T : (°, °) (/,) (v, °)	F : (\, :) (v, °) (/, :)	F : (\,) (\,) (°, °)	F : (v, °) (/, :) (v, °)
T : (>, X) (/,)	T : (>, X) (-,) (/,)	T : (°, °) (/,) (\,)	F : (/, :) (v, °) (\,)	F : (°, °) (v, °) (\,)	F : (°, °) (-,) (v, °)
T : (/,) (-,)	T : (-,) (-,)	T : (\,) (/,) (-,)	F : (\,) (°, °) (\, :)	F : (°, °) (/,) (v, °)	F : (°, °) (/,) (v, °)
T : (v, °) (/,)	T : (-,) (\, :) (/,)	T : (\,) (-,) (°, °)	F : (\,) (-, :)	F : (\, :) (>, 0)	F : (°, °) (v, °)
T : (-,) (/,)	T : (\,) (-, :) (/,)	T : (-,) (\,) (/,)	F : (°, °) (\, :)	F : (°, °) (-,) (\, :)	F : (/,) (-,) (v, °)
T : (\, :) (/,)	T : (\, :) (\, :) (-,)	T : (\,) (/,) (>, X)	F : (/,) (°, °) (\, :)	F : (\, :) (-,) (/, :)	F : (°, °) (>, X) (\, :)
T : (/, :) (-,)	T : (\, :) (/,) (-,)	T : (/,) (\,) (-,)	F : (/, :) (/, :)	F : (>, X) (\, :) (/, :)	F : (v, °) (°, °)
T : (v, °) (/,)	T : (\,) (>, X)	T : (\,) (>, X) (/,)	F : (\,) (-,) (\, :)	F : (-,) (\, :) (°, °)	F : (-,) (\, :)
T : (v, °) (-, :) (/,)	T : (v, °) (/,) (\, :)	T : (v, °) (v, °) (\,)	F : (v, °) (-, :) (/, :)	F : (/,) (/,) (-, :)	F : (\,) (v, °) (°, °)
T : (/, :) (v, °) (\, :)	T : (-,) (-,) (/,)	F : (-, :)	F : (/, :) (>, 0)	F : (\, :) (/,) (>, 0)	F : (v, °) (°, °) (/, :)
T : (/,) (v, °) (/,)	T : (v, °) (\, :) (/,)	F : (/, :)	F : (\,) (v, °) (/, :)	F : (/,) (v, °) (/, :)	F : (v, °) (v, °) (°, °)
T : (\,) (-,)	T : (/,) (\, :) (-,)	F : (°, °)	F : (\,) (>, 0)	F : (/, :) (v, °) (/, :)	F : (/, :) (v, °) (\, :)
T : (-,) (°, °)	T : (-,) (-,) (°, °)	F : (>, 0)	F : (\, :) (-,) (°, °)	F : (\, :) (\, :)	F : (v, °) (v, °) (\, :)
T : (°, °) (\,)	T : (\,) (\,) (-,)	F : (-, :) (-, :)	F : (-, :) (-, :) (/, :)	F : (v, °) (\, :) (/, :)	
T : (-, :) (°, °)	T : (°, °) (-,) (/,)	F : (\,) (-, :)	F : (\,) (-, :) (/, :)	F : (\,) (\,) (>, 0)	
T : (\, :) (-,)	T : (>, X) (\, :) (/,)	F : (v, °) (/, :)	F : (\,) (-, :) (°, °)	F : (-, :) (-, :) (°, °)	
T : (\, :) (-,) (°, °)	T : (/,) (/,) (-,)	F : (/, :) (°, °)	F : (\, :) (/,) (-, :)	F : (v, °) (\,) (/,)	

===== final third level sequences

T : (-,)	T : (>, X) (/,)	T : (-, :) (°, °)	T : (/, :) (\,) (-,)	T : (°, °) (/, :) (\,)	T : (>, X) (\, :) (/,)
T : (/,)	T : (/,) (-,)	T : (\, :) (-,)	T : (/, :) (>, X)	T : (\,) (>, X)	T : (/,) (/,) (-,)
T : (°, °)	T : (v, °) (/,)	T : (\,) (-,) (°, °)	T : (\,) (-, :) (/,)	T : (°, °) (/, :) (>, X)	T : (/, :) (v, °) (/,)
T : (>, X)	T : (-,) (/,)	T : (v, °) (-,) (°, °)	T : (\, :) (/,) (>, X)	T : (/, :) (/, :) (>, X)	T : (-, :) (-, :) (°, °)
T : (-, :) (/,)	T : (\, :) (/,)	T : (°, °) (-,) (v, °)	T : (/,) (/,) (>, X)	T : (v, °) (/,) (\, :)	T : (\, :) (v, °) (\,)
T : (/,) (/,)	T : (/, :) (-,)	T : (/,) (°, °) (\,)	T : (>, X) (-,) (/,)	T : (-,) (-,) (/,)	T : (-,) (\, :) (°, °)
T : (\, :) (°, °)	T : (v, °) (/,)	T : (\,) (v, °) (/,)	T : (-,) (-,)	T : (v, °) (\, :) (/,)	T : (v, °) (-,) (\, :)
T : (/,) (°, °)	T : (v, °) (-, :) (/,)	T : (\, :) (-,) (/,)	T : (\,) (-,) (°, °)	T : (/,) (\, :)	T : (\, :) (\, :)
T : (°, °) (-,)	T : (/, :) (v, °) (\, :)	T : (\,) (v, °) (/,)	T : (/, :) (°, °) (\,)	T : (\,) (>, X) (/,)	T : (\,) (\,) (°, °)
T : (°, °) (\,)	T : (/,) (v, °) (/,)	T : (-,) (-,) (\,)	T : (-,) (\, :) (/,)	T : (-,) (-,) (°, °)	T : (-,) (-,) (\, :)
T : (/,) (>, X)	T : (\,) (-,)	T : (\,) (/, :) (-,)	T : (\,) (\, :) (/,)	T : (\,) (\,) (-,)	T : (\,) (\,) (/,)
T : (>, X) (-,)	T : (-,) (°, °)	T : (\,) (/,)	T : (\, :) (\, :) (-,)	T : (v, °) (-,) (/,)	T : (-, :) (\,) (/,)
T : (-,) (>, X)	T : (°, °) (\,)	T : (°, °) (\,)	T : (\, :) (/,) (-,)	T : (/,) (\, :) (>, X)	T : (/,) (\, :) (/, :)
T : (\,) (>, X)	T : (\,) (°, °) (\,)	T : (-, :) (-, :) (/,)	T : (-,) (>, X) (/,)	T : (\,) (\, :) (>, X)	T : (°, °) (/,) (\,)

T : (ε, ε) (-, ε) (ε, ε)	T : (λ, l) (/, l) (-, l)	F : (λ, ε) (ε, ε) (/, ε)	F : (λ, l) (-, ε) (λ, l)	F : (-, ε) (λ, ε) (>, 0)	F : (/, ε) (/, ε) (>, 0)
T : (/, ε) (/, ε) (-, l)	T : (λ, l) (-, l) (ε, ε)	F : (/, ε) (ε, ε) (λ, l)	F : (λ, l) (/, l) (-, ε)	F : (/, ε) (λ, l) (-, ε)	F : (ε, ε) (-, ε) (λ, ε)
T : (λ, l) (/, ε) (>, X)	T : (-, l) (λ, l) (/, l)	F : (λ, l) (ε, ε) (λ, ε)	F : (/, l) (-, ε) (ε, ε)	F : (ε, ε) (λ, ε) (/, ε)	F : (ε, ε) (-, ε) (ε, ε)
T : (/, ε) (λ, l) (>, X)	T : (λ, l) (/, l) (>, X)	F : (λ, l) (-, ε)	F : (-, l) (ε, ε) (λ, l)	F : (λ, ε) (>, 0) (λ, l)	F : (ε, ε) (/, ε) (ε, ε)
T : (ε, ε) (/, ε) (ε, ε)	T : (/, l) (λ, l) (-, l)	F : (ε, ε) (λ, ε)	F : (ε, ε) (λ, ε)	F : (-, l) (λ, ε) (ε, ε)	F : (ε, ε) (-, l) (ε, ε)
T : (>, X) (λ, ε) (ε, ε)	T : (λ, l) (-, l) (/, l)	F : (/, l) (ε, ε) (λ, ε)	F : (>, X) (-, l) (/, ε)	F : (/, ε) (λ, l) (>, 0)	F : (ε, ε) (/, ε) (>, X)
T : (ε, ε) (λ, ε) (ε, ε)	T : (ε, ε) (ε, ε) (λ, l)	F : (/, ε) (/, ε)	F : (λ, ε) (>, X) (/, ε)	F : (λ, l) (λ, l) (>, 0)	F : (ε, ε) (/, l) (ε, ε)
T : (λ, l) (λ, l) (>, X)	F : (-, ε)	F : (/, ε) (ε, ε) (λ, ε)	F : (-, l) (-, l) (/, ε)	F : (λ, ε) (ε, ε) (λ, ε)	F : (/, l) (λ, ε) (>, 0)
T : (-, ε) (λ, l) (ε, ε)	F : (/, ε)	F : (-, ε) (ε, ε) (λ, ε)	F : (λ, ε) (ε, ε)	F : (-, ε) (-, ε) (ε, ε)	F : (ε, ε) (>, X) (/, ε)
T : (>, X) (-, l) (ε, ε)	F : (ε, ε)	F : (ε, ε) (-, ε) (/, ε)	F : (λ, l) (λ, l) (/, ε)	F : (λ, l) (/, ε) (>, 0)	F : (ε, ε) (ε, ε)
T : (ε, ε) (/, l) (ε, ε)	F : (>, 0)	F : (/, ε) (>, 0)	F : (/, l) (λ, ε) (-, ε)	F : (λ, ε) (λ, ε) (ε, ε)	F : (λ, ε) (ε, ε) (ε, ε)
T : (ε, ε) (ε, ε) (>, X) (/, l)	F : (-, ε) (-, ε)	F : (λ, l) (ε, ε) (/, ε)	F : (-, l) (-, l) (ε, ε)	F : (λ, l) (ε, ε) (λ, ε)	F : (/, l) (ε, ε) (ε, ε)
T : (ε, ε) (ε, ε)	F : (λ, ε) (-, ε)	F : (λ, l) (>, 0)	F : (-, ε) (λ, l) (ε, ε)	F : (ε, ε) (λ, l) (/, l)	F : (ε, ε) (>, X) (λ, ε)
T : (ε, ε) (>, X)	F : (ε, ε) (/, ε)	F : (λ, ε) (-, l) (ε, ε)	F : (λ, l) (λ, l) (ε, ε)	F : (ε, ε) (/, ε) (λ, ε)	F : (λ, l) (ε, ε) (ε, ε)
T : (λ, ε) (ε, ε) (ε, ε)	F : (ε, ε) (ε, ε)	F : (-, ε) (-, ε) (/, ε)	F : (ε, ε) (ε, ε) (λ, ε)	F : (-, l) (>, X) (/, ε)	F : (ε, ε) (ε, ε)
T : (/, l) (ε, ε) (ε, ε)	F : (ε, ε) (λ, ε)	F : (λ, l) (-, ε) (ε, ε)	F : (λ, ε) (>, 0)	F : (ε, ε) (λ, ε) (/, ε)	F : (/, l) (ε, ε) (ε, ε)
T : (ε, ε) (>, X) (λ, l)	F : (-, ε) (ε, ε)	F : (λ, l) (-, ε) (/, ε)	F : (/, ε) (λ, ε) (>, X)	F : (λ, ε) (λ, ε) (/, ε)	F : (ε, ε) (ε, ε) (/, ε)
T : (/, l) (ε, ε)	F : (-, l) (ε, ε)	F : (λ, l) (-, ε) (ε, ε)	F : (/, ε) (λ, ε) (>, X)	F : (λ, ε) (λ, ε) (/, ε)	F : (ε, ε) (ε, ε) (/, ε)
T : (ε, ε) (ε, ε)	F : (λ, l) (ε, ε)	F : (λ, ε) (/, l) (-, ε)	F : (λ, ε) (-, l) (/, ε)	F : (ε, ε) (-, ε) (ε, ε)	F : (ε, ε) (ε, ε) (ε, ε)
T : (ε, ε) (ε, ε) (λ, l)	F : (/, l) (-, ε)	F : (-, ε) (>, 0)	F : (>, X) (λ, ε) (/, ε)	F : (/, l) (/, l) (>, 0)	F : (ε, ε) (ε, ε) (λ, ε)
T : (/, ε) (ε, ε) (ε, ε)	F : (ε, ε) (/, ε)	F : (λ, l) (λ, l) (-, ε)	F : (/, l) (λ, ε) (/, ε)	F : (/, ε) (λ, ε) (-, ε)	F : (/, l) (ε, ε) (ε, ε)
T : (ε, ε) (/, l) (ε, ε)	F : (/, ε) (-, ε)	F : (/, l) (>, 0)	F : (/, l) (/, l) (-, ε)	F : (>, X) (λ, ε) (ε, ε)	F : (ε, ε) (ε, ε) (λ, ε)
T : (-, l) (λ, l) (ε, ε)	F : (-, ε) (/, ε)	F : (>, X) (/, ε)	F : (λ, ε) (λ, ε) (>, 0)	F : (ε, ε) (/, ε) (ε, ε)	F : (-, ε) (ε, ε)
T : (ε, ε) (/, l) (λ, l)	F : (-, l) (/, ε)	F : (λ, ε) (-, l) (λ, l)	F : (λ, ε) (/, l) (>, 0)	F : (ε, ε) (/, l) (/, ε)	F : (ε, ε) (ε, ε)
T : (ε, ε) (ε, ε) (ε, ε)	F : (ε, ε) (-, l) (/, ε)	F : (-, ε) (λ, l) (/, ε)	F : (/, l) (ε, ε) (/, ε)	F : (ε, ε) (λ, ε) (ε, ε)	F : (ε, ε) (λ, l) (λ, l)
T : (/, l) (ε, ε) (ε, ε)	F : (λ, l) (/, ε)	F : (/, l) (ε, ε) (/, ε)	F : (/, ε) (ε, ε) (/, ε)	F : (ε, ε) (/, l) (λ, l)	F : (ε, ε) (ε, ε)
T : (ε, ε) (ε, ε) (/, l)	F : (λ, l) (/, ε) (-, ε)	F : (λ, ε) (/, ε)	F : (λ, ε) (λ, ε) (-, ε)	F : (ε, ε) (λ, l) (ε, ε)	F : (ε, ε) (ε, ε)

=====

Cycle de décision 3 : (λ, l)

===== final first level sequences

T : (λ, l)	T : (λ, l) (-, l) (λ, l)	T : (/, l) (-, l) (λ, l)	F : (>, X) (λ, ε)	F : (/, ε) (λ, ε)	F : (/, ε) (/, ε) (λ, ε)
T : (ε, ε)	T : (/, ε) (-, l) (λ, l)	T : (/, l) (>, X) (λ, l)	F : (-, l) (ε, ε)	F : (-, l) (-, l) (λ, ε)	F : (ε, ε) (>, 0)
T : (ε, ε) (-, l)	T : (-, l) (-, l) (λ, l)	T : (/, l)	F : (-, ε) (ε, ε)	F : (-, l) (/, ε) (λ, ε)	F : (-, ε) (-, ε) (ε, ε)
T : (-, l) (ε, ε)	T : (-, l) (>, X) (λ, l)	T : (λ, l) (/, l)	F : (λ, l) (λ, ε)	F : (/, l) (-, l) (λ, ε)	F : (ε, ε)
T : (λ, l) (ε, ε)	T : (-, l) (/, ε) (ε, ε)	T : (-, l) (/, l)	F : (λ, ε) (λ, ε)	F : (/, l) (/, l) (ε, ε)	F : (/, ε)
T : (/, ε) (ε, ε)	T : (/, l) (/, l) (ε, ε)	T : (-, l)	F : (-, l) (λ, ε)	F : (/, ε) (-, l) (λ, ε)	F : (λ, l) (/, ε)
T : (/, ε) (-, l) (ε, ε)	T : (>, X) (-, l) (λ, l)	T : (λ, ε)	F : (>, X) (-, l) (λ, ε)	F : (/, ε) (-, l) (ε, ε)	F : (λ, l) (ε, ε)
T : (λ, l) (λ, l)	T : (/, l) (/, l) (λ, l)	F : (ε, ε)	F : (/, l) (-, ε) (ε, ε)	F : (-, l) (-, l) (ε, ε)	F : (λ, l) (ε, ε)
T : (-, l) (λ, l)	T : (-, ε) (/, l) (ε, ε)	F : (-, ε) (λ, ε)	F : (/, l) (-, ε) (λ, ε)	F : (-, l) (/, l) (λ, ε)	F : (ε, ε) (ε, ε)
T : (/, l) (λ, l)	T : (ε, ε) (>, X)	F : (/, l) (ε, ε)	F : (-, ε) (-, ε) (λ, ε)	F : (>, X) (-, l) (ε, ε)	F : (ε, ε) (ε, ε)
T : (/, l) (ε, ε)	T : (-, l) (/, l) (λ, l)	F : (ε, ε) (-, ε)	F : (/, l) (>, X) (λ, ε)	F : (-, ε) (/, l) (ε, ε)	F : (ε, ε) (ε, ε)
T : (>, X) (λ, l)	T : (-, l) (/, l) (ε, ε)	F : (λ, ε) (ε, ε)	F : (-, l) (>, X) (λ, ε)	F : (/, l) (/, l) (λ, ε)	F : (ε, ε) (ε, ε)

===== final second level sequences

T : (-, l)	T : (-, l) (/, l)	T : (λ, l) (/, ε) (-, l)	T : (ε, ε) (λ, ε) (/, l)	T : (ε, ε) (>, X)	F : (λ, ε)
T : (λ, l)	T : (λ, ε) (/, l)	T : (λ, l) (/, l)	T : (/, l) (λ, ε) (-, l)	T : (ε, ε) (>, X) (λ, l)	F : (/, ε)
T : (/, l)	T : (/, ε) (-, l)	T : (/, l) (λ, l)	T : (-, l) (-, l) (ε, ε)	T : (/, l) (ε, ε) (ε, ε)	F : (ε, ε)
T : (ε, ε)	T : (/, ε) (-, l) (ε, ε)	T : (/, l) (ε, ε)	T : (λ, l) (λ, l) (-, l)	T : (ε, ε) (ε, ε)	F : (ε, ε)
T : (ε, ε)	T : (ε, ε) (/, l)	T : (λ, l) (ε, ε)	T : (/, ε) (/, ε) (ε, ε)	T : (ε, ε) (ε, ε) (/, l)	F : (>, 0)
T : (>, X)	T : (ε, ε) (-, ε) (/, l)	T : (-, ε) (-, ε) (/, l)	T : (ε, ε) (-, l) (/, l)	T : (ε, ε) (ε, ε) (λ, l)	F : (-, ε) (-, ε)
T : (-, ε) (/, l)	T : (/, ε) (ε, ε) (λ, ε)	T : (/, ε) (>, X)	T : (-, l) (/, ε) (ε, ε)	T : (/, ε) (ε, ε) (ε, ε)	F : (-, ε) (λ, ε)
T : (/, l) (/, l)	T : (/, l) (ε, ε) (/, l)	T : (>, X) (λ, l)	T : (>, X) (λ, ε) (/, l)	T : (-, l) (/, l) (λ, l)	F : (λ, ε) (-, ε)
T : (-, ε) (ε, ε)	T : (λ, l) (-, l)	T : (λ, l) (-, l) (λ, l)	T : (>, X) (-, l) (ε, ε)	T : (ε, ε) (ε, ε) (λ, l)	F : (/, l) (ε, ε)
T : (ε, ε) (-, l)	T : (-, l) (ε, ε)	T : (λ, l) (-, ε) (/, l)	T : (λ, l) (/, l) (-, l)	T : (ε, ε) (ε, ε) (/, l)	F : (ε, ε) (-, ε)
T : (-, l) (ε, ε)	T : (ε, ε) (λ, l)	T : (λ, ε) (/, l) (>, X)	T : (/, l) (/, l) (ε, ε)	T : (ε, ε) (/, l) (ε, ε)	F : (λ, ε) (ε, ε)
T : (λ, l) (ε, ε)	T : (λ, l) (λ, l)	T : (/, l) (/, l) (>, X)	T : (>, X) (-, l) (λ, l)	T : (ε, ε) (/, l) (λ, l)	F : (ε, ε) (/, ε)
T : (λ, ε) (ε, ε)	T : (-, ε) (λ, l)	T : (>, X) (-, l) (/, l)	T : (-, l) (-, l) (ε, ε)	T : (ε, ε) (ε, ε) (/, l)	F : (/, ε) (ε, ε)
T : (/, l) (ε, ε)	T : (/, ε) (λ, l)	T : (-, l) (-, l)	T : (/, l) (/, l) (λ, l)	T : (-, l) (/, l) (ε, ε)	F : (-, ε) (-, ε)
T : (ε, ε) (-, l)	T : (-, ε) (ε, ε)	T : (/, ε) (-, l) (λ, l)	T : (-, ε) (/, l) (λ, l)	T : (/, l) (-, l) (λ, l)	F : (-, ε) (ε, ε)
T : (ε, ε) (λ, l)	T : (λ, ε) (-, l)	T : (-, l) (λ, ε) (/, l)	T : (λ, l) (λ, l) (/, l)	T : (λ, l) (/, l) (-, l)	F : (ε, ε) (λ, ε)
T : (/, l) (ε, ε)	T : (/, ε) (λ, l)	T : (λ, l) (-, l) (>, X)	T : (ε, ε) (λ, l) (λ, l)	T : (λ, l) (-, l) (ε, ε)	F : (-, l) (ε, ε)
T : (ε, ε) (-, l)	T : (ε, ε) (ε, ε)	T : (λ, ε) (λ, ε) (-, l)	T : (/, ε) (/, ε) (-, l)	T : (-, l) (λ, l) (/, l)	F : (λ, l) (ε, ε)
T : (>, X) (-, l)	T : (ε, ε) (-, l) (ε, ε)	T : (λ, l) (-, l) (ε, l)	T : (-, ε) (/, l) (ε, ε)	T : (/, l) (>, X) (λ, l)	F : (>, X) (λ, ε)
T : (-, l) (>, X)	T : (ε, ε) (-, l) (ε, ε)	T : (λ, l) (>, X)	T : (λ, l) (-, l) (ε, ε)	T : (>, X) (-, l) (ε, ε)	F : (-, l) (ε, ε)
T : (>, X) (/, l)	T : (/, l) (ε, ε) (λ, l)	T : (-, l) (-, l) (λ, l)	T : (ε, ε) (/, l) (ε, ε)	T : (/, l) (λ, l) (-, l)	F : (/, l) (-, ε)
T : (/, l) (-, l)	T : (λ, ε) (-, l) (/, l)	T : (ε, ε) (/, l) (λ, ε)	T : (ε, ε) (>, X)	T : (λ, l) (-, l) (/, l)	F : (-, ε) (ε, ε)
T : (ε, ε) (/, l)	T : (λ, ε) (ε, ε) (/, l)	T : (-, l) (ε, ε) (/, l)	T : (ε, ε) (ε, ε) (λ, l)	T : (ε, ε) (ε, ε) (λ, l)	F : (ε, ε) (/, ε)
T : (/, ε) (ε, ε)	T : (ε, ε) (-, l) (λ, l)	T : (-, l) (>, X) (λ, l)	T : (ε, ε) (ε, ε)	F : (-, ε)	F : (/, ε) (-, ε)

F : (c, l) (v, ')	F : (c, l) (>X) (\, :)	F : (v, l) (c, ')	F : (v, l) (c, ')/ (, :)	F : (v, l) (v, l) (\, :)
F : (c, l) (v, ')/ (\, :)	F : (c, l) (c, ')/ (, :)	F : (v, l) (c, ')/ (, :)	F : (v, l) (c, ')/ (c, ')	F : (-, :)/ (v, l)
F : (\, :)/ (c, l) (v, ')	F : (\, l) (c, l) (v, ')	F : (l, l) (v, l) (c, ')	F : (v, l) (v, ')/ (\, :)	F : (-, :)/ (\, l)
F : (l, l) (c, l) (v, ')	F : (c, l) (v, l) (\, :)	F : (\, l) (v, l) (c, ')	F : (l, :)/ (v, l) (c, ')	

=====

Cycle de décision 4 : (v, l)

===== final first level sequences

T : (-, l)	T : (>, X)
------------	------------

===== final second level sequences

T : (-, l)	T : (c, l) (-, l) (\, l)	T : (v, l) (c, l)	F : (l, l) (-, :)	F : (\, l) (l, l) (-, :)	F : (c, l) (-, l) (v, ')
T : (>, X)	T : (\, l) (l, :)/ (-, l)	T : (l, :)/ (v, l) (v, ')	F : (v, ')/ (l, :)	F : (l, l) (\, :)/ (-, :)	F : (c, ')/ (l, l) (v, ')
T : (c, ')/ (\, l)	T : (l, :)/ (>, X)	T : (c, l) (l, l) (v, l)	F : (l, :)/ (-, :)	F : (c, ')/ (l, l) (\, :)	F : (c, l) (v, ')
T : (l, l) (>, X)	T : (\, :)/ (l, l) (>, X)	T : (c, l) (l, l) (\, l)	F : (v, l) (-, l) (l, :)	F : (c, l) (l, l) (v, ')	F : (l, l) (c, l) (v, ')
T : (>, X) (-, l)	T : (l, l) (l, l) (>, X)	T : (\, l) (l, l) (-, l)	F : (\, l) (l, :)/ (-, :)	F : (\, :)/ (>, 0)	F : (c, l) (>, X) (\, :)
T : (-, l) (>, X)	T : (-, l) (-, l)	T : (\, l) (l, l) (>, X)	F : (\, :)/ (v, ')/ (l, :)	F : (c, l) (-, l) (\, :)	F : (v, l) (c, ')
T : (\, :)/ (>, X)	T : (\, :)/ (\, :)/ (-, l)	T : (l, l) (\, l) (-, l)	F : (l, :)/ (v, l) (\, l)	F : (l, l) (l, l) (-, :)	F : (l, l) (v, l) (c, ')
T : (l, l) (-, l)	T : (\, :)/ (l, l) (-, l)	T : (l, l)	F : (\, l) (c, ')/ (\, :)	F : (\, :)/ (l, l) (>, 0)	F : (\, l) (v, l) (c, ')
T : (v, ')/ (l, l)	T : (\, l) (>, X)	T : (-, l) (c, l)	F : (\, l) (-, :)	F : (l, l) (v, ')/ (l, :)	F : (l, :)/ (v, l) (v, l)
T : (l, :)/ (-, l)	T : (v, l) (\, :)/ (l, l)	T : (\, :)/ (l, l)	F : (c, l) (\, :)	F : (l, :)/ (v, l) (l, :)	F : (l, :)
T : (v, l) (l, l)	T : (l, l) (\, :)/ (-, l)	T : (-, l) (l, l)	F : (l, l) (c, l) (\, :)	F : (\, :)/ (\, :)/ (-, :)	F : (-, l) (l, :)
T : (v, ')/ (l, l)	T : (\, l) (\, l) (-, l)	T : (c, l)	F : (l, :)/ (c, ')/ (\, :)	F : (v, l) (\, :)/ (l, :)	F : (\, :)/ (l, :)
T : (l, :)/ (v, l) (\, :)	T : (v, l) (-, l) (l, l)	T : (c, ')/ (l, l)	F : (\, l) (c, ')/ (-, :)	F : (\, l) (\, l) (>, 0)	F : (-, l) (c, ')
T : (l, l) (v, ')/ (l, l)	T : (l, l) (l, l) (-, l)	T : (v, ')/ (\, l)	F : (l, :)/ (>, 0)	F : (c, l) (l, :)/ (\, :)	F : (\, l) (l, :)
T : (\, l) (-, l)	T : (c, ')/ (l, l) (\, l)	T : (v, l) (\, l)	F : (\, l) (v, l) (l, :)	F : (v, l) (-, l) (c, ')	F : (\, l) (c, ')
T : (c, l) (\, l)	T : (l, :)/ (l, :)/ (-, l)	F : (-, :)	F : (\, l) (>, 0)	F : (c, ')/ (c, :)/ (v, ')	F : (-, l) (v, l)
T : (\, :)/ (-, l)	T : (c, l) (l, :)/ (v, l)	F : (>, 0)	F : (\, :)/ (l, l) (-, :)	F : (l, l) (l, l) (>, 0)	F : (c, ')
T : (v, l) (-, l) (c, l)	T : (c, ')/ (l, l) (v, l)	F : (-, :)/ (-, :)	F : (-, :)/ (>, 0)	F : (l, :)/ (l, :)/ (-, :)	F : (c, ')/ (l, :)
T : (c, l) (-, l) (v, l)	T : (c, l) (v, l)	F : (\, :)/ (-, :)	F : (\, l) (\, l) (>, 0)	F : (v, l) (\, l) (l, :)	F : (v, l) (\, :)
T : (l, l) (c, l) (\, l)	T : (c, l) (>, X) (\, l)	F : (v, l) (l, :)	F : (l, l) (>, 0)	F : (c, ')/ (-, :)/ (\, :)	F : (v, l) (\, :)
T : (\, :)/ (v, ')/ (l, l)	T : (l, l) (c, l) (v, l)	F : (c, ')/ (\, :)	F : (l, l) (v, l) (l, :)	F : (v, ')/ (-, :)/ (c, ')	

===== final third level sequences

T : (-, l)	T : (-, l) (c, l)	T : (l, :)/ (\, :)	T : (>, X) (\, :)/ (l, l)	T : (>, X) (-, l) (c, l)	T : (c, l) (l, l)
T : (\, :)	T : (c, l) (\, l)	T : (\, l) (-, :)/ (c, l)	T : (l, l) (l, l) (-, l)	T : (c, ')/ (l, l) (v, l)	T : (v, l) (\, l)
T : (l, l)	T : (\, l) (c, ')/ (\, l)	T : (l, :)/ (c, ')/ (\, l)	T : (l, :)/ (v, l) (l, l)	T : (>, X) (l, :)/ (v, ')	F : (-, :)
T : (v, ')	T : (l, l) (\, :)	T : (-, l) (\, :)/ (l, l)	T : (>, X) (-, l) (v, ')	T : (-, :)/ (-, :)/ (v, ')	F : (\, l)
T : (c, l)	T : (\, :)/ (\, :)	T : (\, :)/ (\, :)/ (l, l)	T : (-, :)/ (l, l) (v, ')	T : (v, l) (>, X) (l, l)	F : (l, :)
T : (>, X)	T : (-, :)/ (c, l)	T : (\, :)/ (\, :)/ (-, l)	T : (-, l) (l, :)/ (v, ')	T : (c, l) (v, l)	F : (v, l)
T : (-, :)/ (\, :)	T : (\, :)/ (-, l)	T : (\, :)/ (l, l) (-, l)	T : (-, :)/ (-, :)/ (c, l)	T : (c, l) (>, X)	F : (c, ')
T : (-, :)/ (l, l)	T : (\, :)/ (-, l) (c, l)	T : (-, l) (-, l) (\, :)	T : (\, :)/ (c, l) (\, l)	T : (c, l) (v, ')/ (\, :)	F : (>, 0)
T : (l, l) (l, l)	T : (v, l) (-, l) (c, l)	T : (-, l) (>, X) (l, l)	T : (-, l) (\, :)/ (c, l)	T : (\, :)/ (c, l) (v, l)	F : (-, :)/ (-, :)
T : (l, l) (v, ')	T : (-, l) (\, :)	T : (c, l) (l, :)/ (\, l)	T : (c, l) (c, :)/ (c, l)	T : (c, l) (>, X) (\, l)	F : (-, :)/ (-, :)
T : (v, ')/ (-, :)	T : (c, l) (-, l) (v, l)	T : (\, l) (>, X)	T : (\, :)/ (\, :)/ (c, l)	T : (c, l) (c, l) (l, :)	F : (-, :)/ (v, l)
T : (\, :)/ (v, ')	T : (l, l) (c, ')/ (\, l)	T : (l, :)/ (l, :)/ (>, X)	T : (l, l) (l, l) (\, :)	T : (c, l) (v, l) (\, :)	F : (v, l) (c, l)
T : (\, :)/ (v, ')	T : (\, l) (v, l) (l, l)	T : (l, :)/ (l, :)/ (>, X)	T : (l, :)/ (l, :)/ (\, :)	T : (l, l) (v, l) (v, l)	F : (-, l) (v, l)
T : (\, :)/ (c, l)	T : (>, X) (-, l) (\, :)	T : (l, :)/ (>, X) (\, :)	T : (\, l) (\, l) (c, l)	T : (v, l) (c, l)	F : (\, l) (v, l)
T : (c, l) (-, l)	T : (l, l) (-, :)/ (v, ')	F : (\, :)/ (-, :)	T : (c, l) (-, :)/ (\, l)	T : (v, l) (c, ')/ (l, l)	F : (v, l) (l, :)
T : (c, ')/ (\, l)	T : (\, :)/ (-, l) (l, l)	T : (v, ')/ (l, l) (\, :)	T : (\, l) (\, l) (l, l)	T : (\, l) (v, l) (c, l)	F : (l, :)/ (c, ')
T : (l, l) (>, X)	T : (\, :)/ (v, ')/ (l, l)	T : (-, l) (-, l) (l, l)	T : (v, l) (v, ')/ (\, :)	T : (c, l) (v, ')/ (c, l)	F : (c, ')/ (c, :)
T : (>, X) (-, l)	T : (c, l) (-, l) (\, l)	T : (l, l) (-, l) (\, :)	T : (-, :)/ (\, l) (l, l)	T : (l, :)/ (v, l) (v, ')	F : (-, :)/ (c, ')
T : (-, l) (>, X)	T : (\, l) (l, :)/ (-, l)	T : (v, l) (\, :)/ (l, l)	T : (\, :)/ (c, l) (l, l)	T : (c, l) (l, l) (v, l)	F : (c, ')/ (\, :)
T : (>, X) (\, :)	T : (\, l) (l, l)	T : (l, l) (\, :)/ (-, l)	T : (c, ')/ (-, :)/ (v, l)	T : (-, l) (\, l) (c, l)	F : (-, l) (c, ')
T : (\, :)/ (>, X)	T : (\, l) (c, l)	T : (\, :)/ (>, X) (l, l)	T : (-, :)/ (l, l) (l, l)	T : (c, l) (l, l) (\, l)	F : (\, l) (c, ')
T : (>, X) (l, l)	T : (-, :)/ (-, :)/ (l, l)	T : (-, l) (-, l) (c, l)	T : (c, ')/ (-, :)/ (v, l)	T : (v, l) (c, l) (c, l)	F : (l, l) (-, :)
T : (-, l) (-, l)	T : (l, :)/ (\, l) (-, l)	T : (\, l) (\, l) (c, l)	T : (v, l) (c, ')/ (l, l)	T : (\, l) (v, l) (c, l)	F : (v, ')/ (l, :)
T : (-, l) (v, ')	T : (l, :)/ (>, X)	T : (-, :)/ (c, l) (l, l)	T : (\, l) (l, :)/ (>, X)	T : (v, l) (c, l) (l, l)	F : (l, :)/ (-, :)
T : (v, ')/ (l, l)	T : (l, l) (-, :)/ (\, :)	T : (l, l) (l, l) (v, ')	T : (l, :)/ (\, l) (>, X)	T : (\, l) (l, l) (-, l)	F : (-, :)/ (l, :)
T : (-, :)/ (v, ')	T : (-, :)/ (-, :)/ (\, :)	T : (>, X) (l, :)/ (\, :)	T : (c, l) (\, :)/ (c, l)	T : (\, l) (-, l) (c, l)	F : (l, :)/ (v, l)
T : (-, l) (l, l)	T : (\, l) (-, :)/ (l, l)	T : (l, :)/ (>, X) (\, :)	T : (v, ')/ (>, 0)	T : (-, l) (\, l) (l, l)	F : (-, l) (l, :)
T : (\, :)/ (l, l)	T : (l, l) (>, X) (\, :)	T : (l, :)/ (v, ')	T : (v, l) (l, :)/ (v, ')	T : (\, l) (l, l) (>, X)	F : (v, l) (-, l) (l, :)
T : (l, :)/ (-, l)	T : (\, :)/ (l, l) (>, X)	T : (l, :)/ (-, l) (v, ')	T : (>, X) (\, :)/ (c, l)	T : (l, l) (\, l) (-, l)	F : (l, :)/ (-, l) (v, l)
T : (v, l) (l, l)	T : (l, l) (l, l) (>, X)	T : (-, l) (l, l) (v, ')	T : (v, l) (\, :)/ (c, l)	T : (\, l) (-, l) (c, l)	F : (\, l) (l, :)
T : (v, ')/ (-, :)/ (l, l)	T : (>, X) (-, l) (l, l)	T : (-, l) (l, l) (\, :)	T : (c, l) (l, :)/ (v, l)	T : (v, l) (v, l) (\, l)	F : (\, l) (l, :)/ (-, :)
T : (l, :)/ (v, l) (\, :)	T : (-, l) (-, l)	T : (v, l) (-, l) (l, l)	T : (\, l) (\, l) (>, X)	T : (c, ')/ (l, l)	F : (-, :)/ (-, :)/ (v, l)
T : (l, l) (v, ')/ (l, l)	T : (-, l) (>, X) (\, :)	T : (l, l) (\, :)/ (>, X)	T : (v, l) (\, l) (c, l)	T : (-, :)/ (\, l) (c, l)	F : (\, :)/ (v, ')/ (l, :)
T : (\, l) (-, l)	T : (>, X) (l, l) (\, :)	T : (\, :)/ (\, :)/ (>, X)	T : (c, ')/ (\, l) (c, l)	T : (c, l) (c, l)	F : (-, l) (v, l) (l, l)

Annexe A : résultats détaillés des expérimentations

F : (/,:)(v,!)(\,l)	F : (\,l)(-,:)(^,')	F : (/,l)(\,,:)(-,:)	F : (v,!)(\,,:)(/,:)	F : (/,l)(/,:)(>,0)	F : (\,l)(^,')(\,,:)
F : (\,l)(^,')(\,,:)	F : (\,,:)(/,:)(-,:)	F : (-,l)(-,:)(^,')	F : (\,,:)(>,0)(\,l)	F : (/,:)(/,:)(-,:)	F : (v,!)(^,')
F : (\,l)(-,:)	F : (-,:(>,0)	F : (-,:(\,l)(^,')	F : (-,:(\,,:)(^,')	F : (>,X)(\,,:)(^,')	F : (v,!)(^,')(/,:)
F : (\,l)(\,l)	F : (\,l)(\,l)(-,:)	F : (\,l)(\,l)(^,')	F : (/,:)(\,l)(>,0)	F : (-,:(/,:)(v,!)	F : (/,:)(v,!)(^,')
F : (-,:(\,l)	F : (/,:)(>,0)	F : (^,')(/,l)(\,,:)	F : (\,l)(\,l)(>,0)	F : (^,')(/,:(v,!)	F : (\,l)(v,!)(^,')
F : (/,:)(\,l)	F : (>,X)(/,:)	F : (^,')(/,l)(v,!)	F : (-,:(\,l)(>,X)	F : (v,!)(\,l)(/,:)	F : (v,!)(^,')(/,:)
F : (^,')(\,,:)	F : (/,:)(-,:)(\,l)	F : (\,,:)(>,0)	F : (>,X)(-,:)(\,l)	F : (v,!)(\,,:)(^,')	F : (v,!)(^,')(^,')
F : (-,l)(\,l)	F : (/,:)(-,:)(\,l)	F : (^,')(-,:(\,l)(\,,:)	F : (^,')(\,l)(^,')	F : (v,!)(/,:)(\,l)	F : (v,!)(v,!)(\,,:)
F : (/,:)(^,')(\,,:)	F : (/,:)(\,l)(/,:)	F : (/,:)(\,,:)(>,X)	F : (\,,:)(^,')(\,,:)	F : (v,!)(\,l)(^,')	F : (/,:)(v,!)(v,!)
F : (/,:)(/,:)	F : (/,:)(v,!)(/,:)	F : (\,,:)(-,:)(/,:)	F : (-,:(-,:)(^,')	F : (/,:)(/,:)(>,0)	F : (v,!)(v,!)(\,,:)
F : (/,:)(^,')(\,,:)	F : (\,,:)(/,:)	F : (>,X)(\,,:)(/,:)	F : (-,l)(-,:)(v,!)	F : (^,')(-,:(\,,:)	F : (-,l)(/,:)(\,l)
F : (-,:(^,')(\,,:)	F : (/,:)(-,:)(\,l)	F : (/,:)(\,,:)(/,:)	F : (\,l)(/,:)(>,0)	F : (v,!)(-,:)(^,')	F : (^,')(-,:(v,!)(\,l)
F : (v,!)(-,:)(/,:)	F : (\,l)(-,:)(\,l)	F : (/,:)(/,:)(-,:)	F : (\,,:)(\,,:)(^,')	F : (v,!)(/,:)(v,!)	F : (^,')(^,')(/,l)
F : (/,:)(>,0)	F : (\,l)(/,:)(-,:)	F : (\,,:)(\,,:)(>,0)	F : (\,l)(^,')(\,,:)	F : (^,')(-,:(v,!)	F : (-,l)(/,:)(v,!)
F : (/,:)(\,l)	F : (/,:)(-,:)(^,')	F : (\,l)(-,:)(>,0)	F : (/,:)(/,:)(\,l)	F : (v,!)(/,:)(>,X)	F : (/,:)(-,:(\,l)(\,l)
F : (\,l)(v,!)(/,:)	F : (-,l)(^,')(\,l)	F : (/,:)(/,:)(v,!)	F : (-,:(/,:)(\,l)	F : (>,X)(\,l)(>,X)	F : (/,:)(>,X)(\,l)
F : (/,:)(v,!)	F : (\,,:)(^,')(/,l)	F : (-,l)(/,:)(v,!)	F : (^,')(/,:(\,,:)	F : (>,X)(/,:)(v,!)	F : (>,X)(/,:)(\,l)
F : (\,l)(>,0)	F : (>,X)(/,:)(\,l)	F : (/,:)(v,!)(/,:)	F : (-,:(-,:)(\,l)	F : (^,')(/,l)(v,!)	F : (>,X)(/,:)(v,!)
F : (/,:)(-,:)(v,!)	F : (>,X)(-,:)(/,:)	F : (/,:)(v,!)(/,:)	F : (-,l)(>,X)(/,:)	F : (/,:)(\,,:)(>,0)	F : (/,:)(>,X)
F : (\,,:)(-,:)(^,')	F : (\,,:)(>,X)(/,:)	F : (>,X)(-,:)(v,!)	F : (^,')(>,0)	F : (v,!)(>,X)	F : (^,')(/,:)
F : (-,:(-,:)(/,:)	F : (-,l)(-,:)(/,:)	F : (/,:)(>,X)(\,l)	F : (v,!)(-,:)(^,')	F : (v,!)(>,X)(/,:)	F : (^,')(/,:)
F : (\,l)(-,:)(v,!)	F : (/,:)(^,')(\,l)	F : (/,:)(/,:)(\,l)	F : (^,')(/,:(\,,:)	F : (^,')(\,,:)(^,')	F : (^,')(\,,:)
F : (>,X)(\,l)	F : (-,l)(-,:)(\,l)	F : (\,,:)(\,,:)(-,:)	F : (-,:(^,')(/,:(^,')	F : (^,')(\,,:)(\,l)	F : (v,!)(\,,:)
F : (\,l)(-,:)(\,l)	F : (-,l)(/,:)(\,l)	F : (-,:(\,,:)(>,0)	F : (>,X)(-,:)(^,')	F : (\,,:)(^,')(\,,:)	F : (v,!)(\,,:)
F : (\,l)(-,:)(/,:)	F : (/,:)(\,l)(/,:)	F : (/,:)(\,,:)(-,:)	F : (\,,:)(\,,:)(/,:)	F : (/,:)(^,')(\,,:)	
F : (-,:(\,l)(-,:)	F : (-,l)(>,X)(\,l)	F : (/,:)(/,:)(v,!)	F : (^,')(-,:(v,!)	F : (^,')(>,X)(\,,:)	

=====

Cycle de décision 5 : (v, :)

===== final first level sequences

===== final second level sequences

T : (/,:)	T : (\,l)	F : (-,:(/,:)	F : (\,,:)
T : (-,:(/,:)	F : (/,:)	F : (-,:(^,')	

===== final third level sequences

T : (\,l)	T : (>,X)(\,l)	T : (-,:(/,:)(\,l)	T : (-,:(^,')	F : (/,:)(>,X)(\,,:)
T : (/,:)	T : (\,l)(-,:)(\,l)	T : (\,l)(\,l)(^,')	T : (/,:)(\,,:)	F : (-,l)(>,X)(\,,:)
T : (v,!)	T : (\,l)(-,:)(/,:)	T : (-,:(-,:)(\,l)	T : (>,0)	F : (>,X)(/,:)(\,,:)
T : (^,')	T : (>,X)(-,:)(/,:)	T : (\,l)(\,l)(/,:)	F : (\,,:)	F : (>,X)(/,:)
T : (-,:(/,:)	T : (/,:)(-,:)(\,l)	T : (-,:(\,l)(\,l)	F : (/,:)	F : (-,:(\,l)(\,l)
T : (/,:)(/,:)	T : (/,:)(-,:)(\,l)	T : (^,')(\,,:)(^,')	F : (v,!)	F : (-,:(\,l)(/,:)
T : (-,:(v,!)	T : (\,l)(-,:)(^,')	T : (-,:(^,')(/,:(^,')	F : (^,')	F : (/,:)(\,,:)
T : (v,!)(-,:)(/,:)	T : (-,:(\,l)(/,:)	T : (-,:(-,:)(v,!)	F : (-,:(\,,:)	F : (-,l)(^,')(\,l)
T : (-,l)(v,!)	T : (\,,:)(\,,:)(/,:)	T : (v,!)(/,:)(v,!)	F : (/,:)(v,!)	F : (-,l)(-,:)(\,l)
T : (\,l)(v,!)	T : (-,l)(>,X)(/,:)	T : (>,X)(\,,:)(^,')	F : (v,!)(-,:)	F : (-,l)(-,:)(\,l)
T : (\,,:)(^,')	T : (>,X)(/,:)(\,l)	T : (-,:(\,l)(^,')	F : (\,,:)(v,!)	F : (>,X)(-,:)(/,:)
T : (/,:)(^,')	T : (^,')(/,:(>,X)	T : (^,')(/,:(^,')	F : (/,:)(^,')	F : (/,:)(>,X)(\,,:)
T : (-,:(-,:)(/,:)	T : (-,l)(-,:)(\,l)	T : (>,X)(\,l)(>,X)	F : (^,')(-,:(^,')	F : (-,l)(/,:)(\,,:)
T : (>,X)(/,:)	T : (v,!)(/,:)(\,,:)	T : (>,X)(/,:)(v,!)	F : (-,:(^,')	F : (\,,:)(>,X)(/,:)
T : (/,:)(v,!)	T : (-,l)(-,:)(/,:)	T : (v,!)(>,X)	F : (-,l)(^,')	F : (-,l)(-,:)(/,:)
T : (-,l)(/,:)	T : (-,l)(/,:)(\,l)	T : (^,')(\,,:)(\,l)	F : (\,l)(^,')	F : (\,,:)(^,')
T : (\,,:)(/,:)	T : (-,l)(>,X)(\,l)	T : (^,')(>,X)	F : (>,X)(\,,:)	F : (/,:)(-,:)(\,,:)
T : (/,:)(-,:)(v,!)	T : (\,,:)(>,X)(/,:)	T : (v,!)(^,')(/,l)	F : (-,l)(v,!)	F : (\,l)(\,l)(/,:)
T : (-,:(-,:)(v,!)	T : (/,:)(/,:)(v,!)	T : (-,l)(/,:)(\,l)	F : (-,:(/,:)	F : (-,l)(-,:)(^,')
T : (-,l)(v,!)(/,:)	T : (-,l)(/,:)(v,!)	T : (^,')(\,,:)(\,l)	F : (-,l)(/,:)	F : (-,:(\,l)(^,')
T : (-,l)(^,')	T : (>,X)(\,,:)(/,:)	T : (^,')(^,')(/,l)	F : (\,l)(/,:)	F : (-,:(^,')(/,l)
T : (\,l)(\,l)	T : (>,X)(-,:)(v,!)	T : (-,l)(\,l)(^,')	F : (/,:)(\,,:)	F : (/,:)(/,:)(v,!)
T : (-,:(\,l)	T : (/,:)(>,X)(\,l)	T : (v,!)(^,')(\,,:)	F : (\,,:)(\,,:)	F : (>,X)(/,:)(\,,:)
T : (/,:)(\,l)	T : (/,:)(/,:)(\,l)	T : (-,l)(/,:)(v,!)	F : (-,l)(v,!)	F : (/,:)(-,:)(>,X)
T : (-,:(-,:)(/,:)	T : (-,:(-,:)(^,')	T : (v,!)(^,')(/,l)	F : (>,X)(-,:)(\,,:)	F : (\,,:)(-,:)(/,:)
T : (\,,:)(-,:)(^,')	T : (/,:)(/,:)(v,!)	T : (/,:)(-,:)(\,l)	F : (/,:)(-,:)(v,!)	F : (/,:)(-,:)(\,,:)
T : (-,l)(\,l)	T : (-,l)(-,:)(^,')	T : (/,:)(-,:)(\,l)	F : (/,:)(-,:)(^,')	F : (/,:)(-,:)(\,,:)
T : (\,,:)(-,:)(/,:)	T : (-,l)(\,l)(>,X)	T : (\,l)(-,:)(^,')	F : (/,:)(-,:)	F : (>,X)(\,,:)(/,:)
T : (\,l)(/,:)	T : (>,X)(-,:)(\,l)	T : (-,l)(\,l)(/,:)	F : (\,,:)(-,:)(^,')	F : (/,:)(v,!)
T : (/,:)(\,l)	T : (-,l)(\,,:)(^,')	T : (/,:)(>,X)(\,l)	F : (-,:(-,:)(/,:)	F : (-,l)(\,,:)(/,:)
T : (/,:)(v,!)	T : (>,X)(\,l)(^,')	T : (>,X)(/,:)(\,l)	F : (\,l)(-,:)(/,:)	F : (/,:)(-,:)(v,!)
T : (\,l)(^,')	T : (\,,:)(\,,:)(^,')	T : (\,l)(-,:)(/,:)	F : (\,l)(-,:)(^,')	F : (-,l)(-,:)(v,!)
T : (-,:(-,:)(/,:)	T : (-,l)(-,:)(v,!)	T : (>,X)(/,:)(v,!)	F : (/,:)(-,:)(\,,:)	F : (-,l)(/,:)(\,,:)
T : (/,:)(-,:)(v,!)	T : (/,:)(/,:)(\,l)	T : (v,!)(v,!)(\,l)	F : (-,:(-,:)(\,,:)	F : (>,X)(-,:)(v,!)

F : (-, :) (/, l) (v, ')	F : (v, ')(\, l) (/, l)	F : (' , !)(\, :) (' , !)	F : (' , !)(\, l) (' , !)	F : (v, !)(', ')(/, :)
F : (-, l) (/, :) (v, ')	F : (/, :) (/, :) (\, :)	F : (>, X)(\, :) (' , ')	F : (>, X) (/, :) (v, ')	F : (v, !)(', ')(', ')
F : (-, l) (\, :) (' , ')	F : (-, l) (>, X) (/, :)	F : (v, ')(>, O)	F : (-, :) (-, :) (v, ')	F : (v, !)(v, ')(\, :)
F : (-, :) (-, :) (' , ')	F : (' , ')(>, O)	F : (v, ')(/, l) (\, l)	F : (' , !)(v, ')(\, :)	F : (v, !)(v, !)(\, :)
F : (/, l) (/, l) (\, :)	F : (>, X) (-, l) (' , ')	F : (v, !)(/, l) (\, l)	F : (' , !)(', ')(/, :)	F : (/ , l) (\, l)
F : (\, :) (\, :) (' , ')	F : (\, :) (\, :) (/, :)	F : (' , !)(/, :) (v, !)	F : (' , !)(v, !)(\, :)	
F : (/, :) (/, :) (v, ')	F : (-, :) (/ , l) (\, :)	F : (v, !)(/, :) (v, !)	F : (v, !)(', ')(/, :)	

=====

Cycle de décision 6 : (>,O)

===== final first level sequences

===== final second level sequences

===== final third level sequences

=====

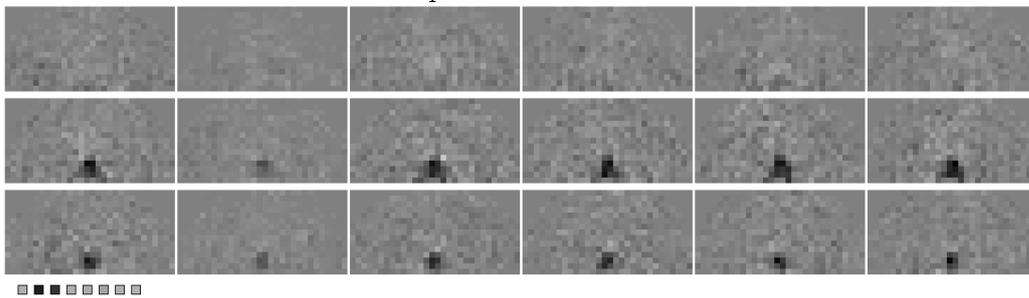
2 Expérimentations du chapitre 10

Affichage détaillé des signatures des interactions primaires obtenues dans les expérimentations du chapitre 10.

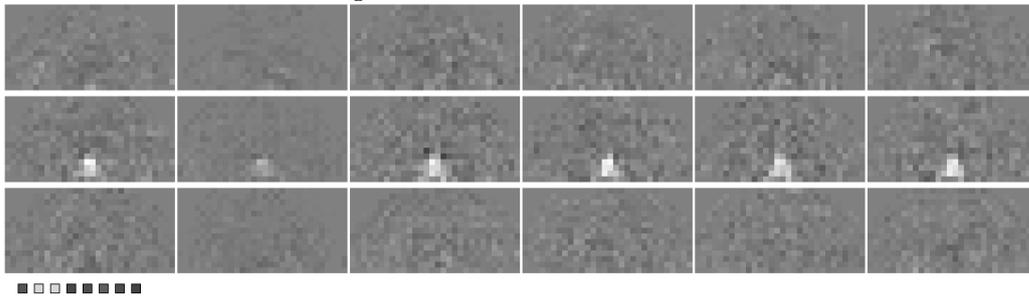
2.1 Système interactionnel complet

Nous affichons ici les signatures données en Figure 10.5. Les composantes RGB sont séparées afin d'observer plus spécifiquement les poids associés à chaque interaction. De haut en bas, sont affichées les groupes d'interaction dans l'ordre rouge, vert, bleu.

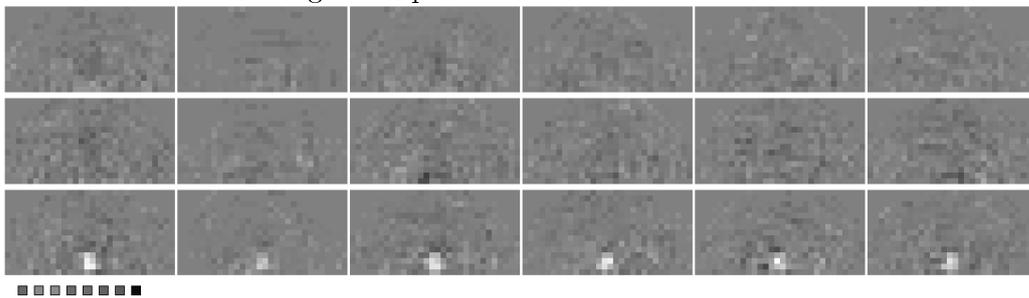
- Avancer d'un pas :



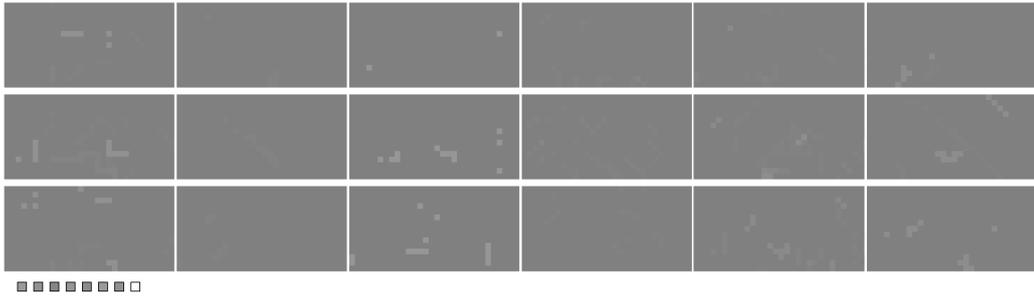
- Se cogner :



- Manger une proie :



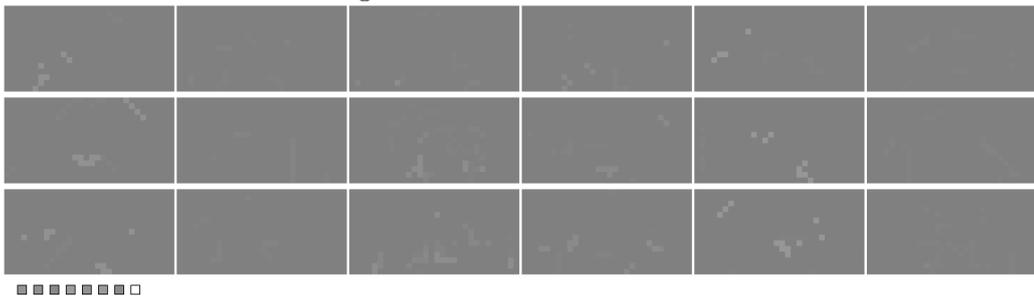
- tourner à gauche de 90°



- tourner à droite de 90°



- tourner à gauche de 45°



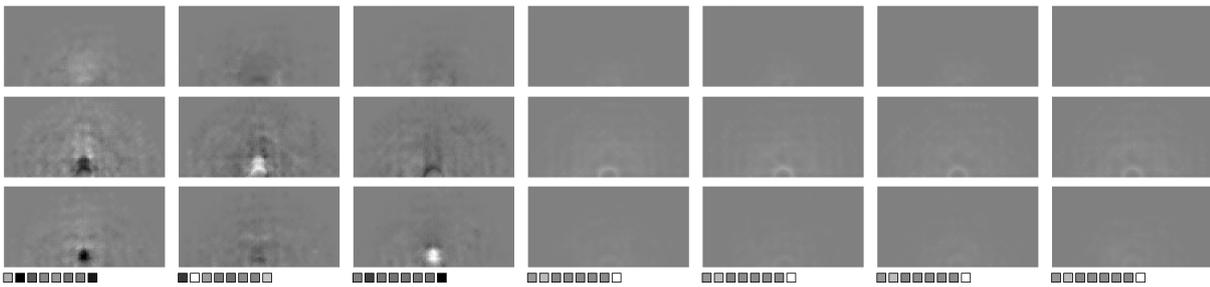
- tourner à droite de 45°



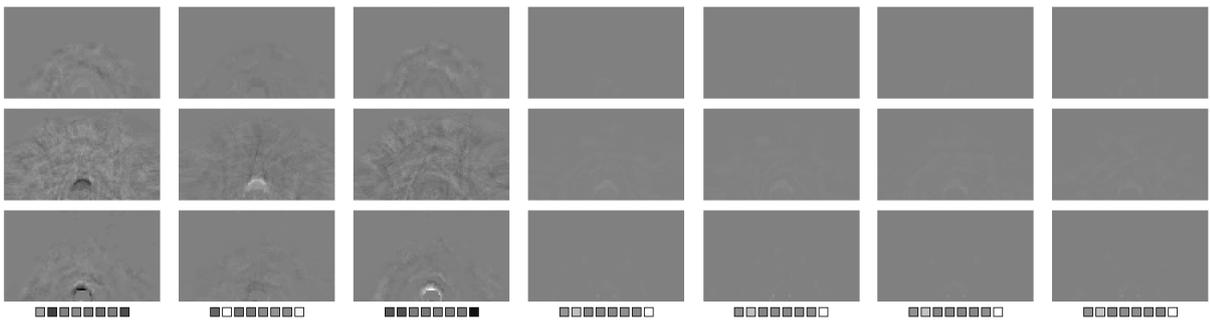
2.2 Système interactionnel simplifié

Nous affichons ici les signatures obtenues avec un système interactionnel simplifié (Figure 10.14), dans le cas d'un agent simulé et d'un robot. Les composantes RGB sont séparées afin d'observer plus spécifiquement les poids associés à chaque interaction. De gauche à droite, les signatures des interactions *avancer*, *se cogner*, *manger*, *tourner à gauche de 90°*, *tourner à droite de 90°*, *tourner à gauche de 45°*, *tourner à droite de 45°*. De haut en bas, sont affichées les groupes d'interaction dans l'ordre rouge, vert, bleu.

- signatures obtenues avec l'agent simulé

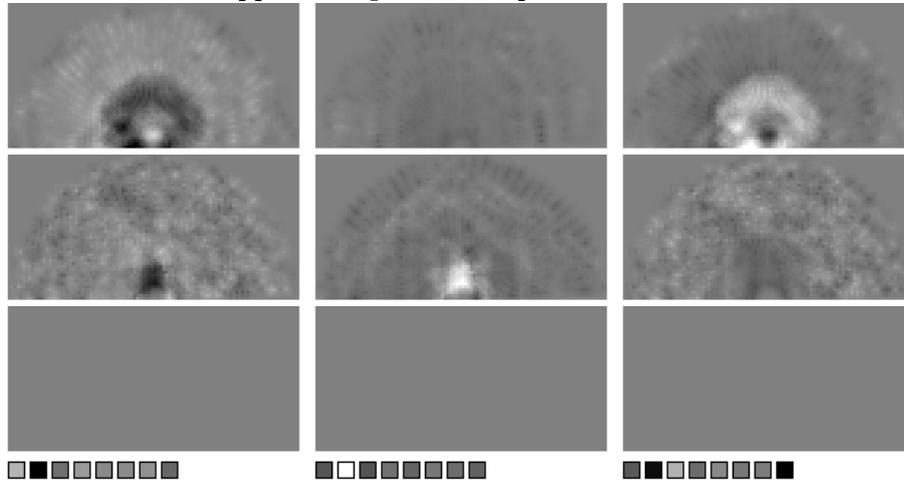


- signatures obtenues avec le robot

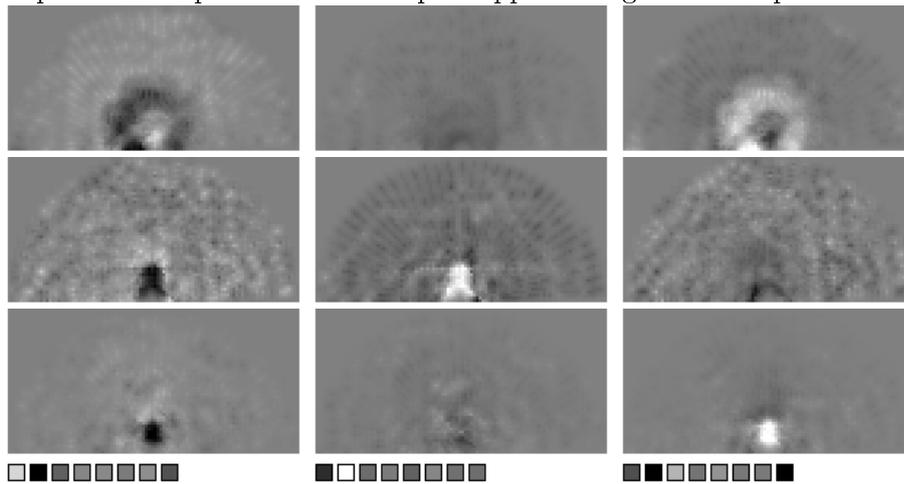


2.3 Signatures obtenues dans l'expérience des proies invisibles

- Apprentissage avec des proies invisibles.



- Adaptation aux proies invisibles après apprentissage avec des proies visibles.



3 Expérimentations du chapitre 11

Cette section fournit la listes des lieux composites permettant d'estimer la position de la proie et du mur central dans les expériences de suivi des objets par la mémoire spatiale, au chapitre 11. Nous affichons, à chaque cycle de décision, les lieux composites du second mécanisme de suivi. Chaque lieu est affiché avec la certitude que l'objet soit contenu par la région définissant ce lieu. Le tableau suivant donne la correspondance entre les symboles utilisés dans ce manuscrit et ceux manipulés par l'implémentation du système :

TABLE 2 – Correspondance des symboles

	=	>O (avancer d'un pas)
	=	>X (se cogner contre un mur)
	=	>F (manger une proie)
	=	^O (tourner à gauche de 90°)
	=	vO (tourner à droite de 90°)
	=	/O (tourner à gauche de 45°)
	=	\O (tourner à droite de 45°)

3.1 Expérimentation avec une proie seule

Suivi de la proie :

Cycle de décision 1 : >O

>0, [>0, 6], 0.8	>0, /0, [>0, 5], 0.73	/0, /0, [\0, 8], 0.79	>x, /0, [>0, 6], 0.68	\0, /0, [>0, 6], 0.77
>0, [>0, 7], 0.8	>0, /0, [>0, 6], 0.77	/0, /0, [\0, 9], 0.55	>x, /0, [>0, 7], 0.51	\0, /0, [/0, 7], 0.22
>0, [^0, 6], 0.8	v0, ^0, [>0, 8], 0.51	>0, [>0, 5], 0.32	v0, ^0, [>0, 6], 0.67	\0, /0, [/0, 8], 0.26
>0, [/0, 5], 0.8	v0, ^0, [/0, 8], 0.76	>0, [/0, 7], 0.8	v0, ^0, [>0, 7], 0.59	/0, >0, [>0, 4], 0.42
>0, [/0, 6], 0.8	^0, v0, [>0, 8], 0.55	\0, [>0, 9], 0.57	v0, ^0, [>0, 9], 0.42	/0, v0, [/0, 9], 0.28
^0, [\0, 9], 0.57	^0, v0, [>0, 9], 0.52	\0, [/0, 9], 0.71	v0, ^0, [/0, 7], 0.35	/0, \0, [>0, 8], 0.42
\0, [^0, 9], 0.71	^0, v0, [/0, 8], 0.56	/0, [>0, 6], 0.72	v0, ^0, [/0, 9], 0.76	/0, \0, [/0, 7], 0.51
/0, [>0, 7], 0.76	\0, ^0, [>0, 7], 0.54	>0, >0, [^0, 4], 0.76	^0, >0, [>0, 9], 0.76	/0, /0, [>0, 7], 0.48
/0, [>0, 8], 0.76	\0, /0, [>0, 7], 0.7	>0, >0, [^0, 5], 0.41	^0, v0, [>0, 7], 0.64	/0, /0, [>0, 8], 0.33
>0, >0, [>0, 5], 0.76	\0, /0, [>0, 8], 0.67	>0, >0, [/0, 7], 0.52	^0, \0, [>0, 9], 0.37	/0, /0, [>0, 9], 0.57
>0, >0, [>0, 6], 0.76	/0, >0, [>0, 5], 0.81	>0, ^0, [>0, 5], 0.23	\0, ^0, [>0, 5], 0.36	
>0, >0, [^0, 6], 0.76	/0, >0, [>0, 6], 0.65	>0, ^0, [>0, 6], 0.51	\0, ^0, [>0, 6], 0.4	
>0, >0, [/0, 5], 0.76	/0, /0, [v0, 8], 0.52	>0, /0, [>0, 4], 0.49	\0, ^0, [>0, 8], 0.36	
>0, >0, [/0, 6], 0.64	/0, /0, [\0, 7], 0.61	>x, /0, [>0, 5], 0.27	\0, ^0, [>0, 9], 0.73	

Cycle de décision 2 : >O

[>0, 6], 0.8	[/0, 7], 0.8	v0, ^0, [/0, 8], 0.52	/0, \0, [/0, 6], 0.73	\0, >F, [/0, 7], 0.71
[>0, 7], 0.8	>0, [^0, 4], 0.76	^0, v0, [>0, 7], 0.79	/0, \0, [/0, 7], 0.73	\0, >F, [/0, 8], 0.72
[^0, 6], 0.8	>0, [^0, 5], 0.41	^0, v0, [>0, 8], 0.79	>0, /0, [>0, 4], 0.46	\0, ^0, [>0, 6], 0.76
[/0, 5], 0.8	>0, [/0, 7], 0.52	^0, v0, [>0, 9], 0.79	>0, /0, [/0, 4], 0.37	\0, /0, [>0, 6], 0.41
[/0, 6], 0.8	^0, [>0, 5], 0.23	^0, v0, [/0, 8], 0.79	>x, ^0, [>0, 5], 0.38	/0, >0, [>0, 3], 0.67
>0, [>0, 5], 0.81	^0, [>0, 6], 0.51	^0, \0, [>0, 6], 0.55	>x, /0, [>0, 5], 0.45	/0, >0, [>0, 4], 0.59
>0, [>0, 6], 0.76	/0, [>0, 4], 0.49	\0, ^0, [>0, 7], 0.44	v0, ^0, [>0, 8], 0.43	/0, >0, [/0, 3], 0.42
>0, [^0, 6], 0.76	/0, [>0, 7], 0.46	\0, /0, [>0, 7], 0.7	^0, v0, [>0, 5], 0.52	/0, \0, [>0, 5], 0.76
>0, [/0, 5], 0.76	>0, /0, [>0, 5], 0.37	\0, /0, [>0, 8], 0.67	^0, v0, [>0, 6], 0.61	/0, \0, [>0, 8], 0.76
>0, [/0, 6], 0.72	>x, ^0, [>0, 6], 0.48	\0, /0, [/0, 7], 0.67	^0, \0, [>0, 5], 0.79	/0, \0, [^0, 6], 0.64
/0, [>0, 5], 0.8	v0, ^0, [>0, 6], 0.55	/0, >0, [>0, 5], 0.45	^0, \0, [>0, 7], 0.55	/0, \0, [/0, 5], 0.4
/0, [>0, 6], 0.77	v0, ^0, [>0, 7], 0.52	/0, \0, [>0, 6], 0.76	\0, >F, [^0, 7], 0.8	/0, /0, [>0, 6], 0.73
[>0, 5], 0.32	v0, ^0, [/0, 7], 0.56	/0, \0, [>0, 7], 0.81	\0, >F, [^0, 8], 0.57	

Cycle de décision 3 : >O

[>0, 5], 0.81	>0, [/0, 5], 0.77	>0, [-0, 6], 0.76	>x, /0, [>0, 5], 0.81	\0, /0, [/0, 5], 0.55
[>0, 6], 0.77	>0, /0, [>0, 4], 0.8	>0, [/0, 6], 0.64	v0, ~0, [>0, 6], 0.52	/0, >0, [/0, 4], 0.75
[-0, 6], 0.8	>0, /0, [>0, 5], 0.61	~0, [>0, 5], 0.4	v0, ~0, [/0, 7], 0.6	/0, \0, [>0, 6], 0.59
[/0, 5], 0.8	>0, /0, [/0, 4], 0.71	/0, [>0, 6], 0.73	~0, v0, [>0, 5], 0.67	/0, \0, [-0, 6], 0.64
[/0, 6], 0.8	/0, >0, [>0, 3], 0.75	/0, [/0, 5], 0.77	~0, \0, [>0, 5], 0.65	/0, \0, [/0, 7], 0.5
[-0, 4], 0.76	/0, >0, [>0, 4], 0.76	>0, ~0, [>0, 4], 0.64	~0, \0, [>0, 6], 0.51	/0, /0, [>0, 4], 0.77
[-0, 5], 0.41	/0, >0, [/0, 2], 0.75	>0, /0, [/0, 5], 0.61	\0, ~0, [>0, 4], 0.78	/0, /0, [>0, 5], 0.56
[/0, 7], 0.52	/0, >0, [/0, 3], 0.75	>x, ~0, [>0, 4], 0.5	\0, ~0, [>0, 5], 0.72	
/0, [>0, 5], 0.8	/0, \0, [/0, 5], 0.59	>x, ~0, [>0, 5], 0.53	\0, ~0, [>0, 6], 0.5	
/0, [>0, 4], 0.77	/0, \0, [/0, 6], 0.5	>x, ~0, [>0, 6], 0.78	\0, /0, [>0, 5], 0.69	
/0, [/0, 4], 0.8	>0, [-0, 5], 0.76	>x, /0, [>0, 4], 0.72	\0, /0, [>0, 6], 0.53	

Cycle de décision 4 : >O

[/0, 5], 0.77	/0, [/0, 6], 0.72	>x, /0, [/0, 4], 0.66	\0, ~0, [/0, 4], 0.22	/0, >0, [/0, 4], 0.51
/0, [>0, 4], 0.8	>0, ~0, [>0, 4], 0.66	>x, /0, [/0, 5], 0.37	\0, ~0, [/0, 5], 0.22	/0, \0, [>0, 5], 0.32
/0, [>0, 5], 0.76	>0, /0, [>0, 4], 0.47	v0, ~0, [-0, 6], 0.72	\0, ~0, [/0, 6], 0.22	/0, \0, [-0, 6], 0.79
/0, [/0, 4], 0.71	>0, /0, [>0, 5], 0.76	~0, ~0, [/0, 7], 0.55	\0, /0, [>0, 5], 0.22	/0, \0, [/0, 5], 0.54
[-0, 5], 0.76	>0, /0, [-0, 4], 0.8	~0, >0, [>0, 3], 0.45	\0, /0, [>0, 6], 0.55	/0, \0, [/0, 6], 0.28
[-0, 6], 0.76	>0, /0, [/0, 4], 0.66	~0, >0, [>0, 5], 0.4	\0, /0, [-0, 6], 0.64	/0, /0, [>0, 3], 0.5
[/0, 6], 0.64	>0, /0, [/0, 5], 0.78	~0, v0, [>0, 5], 0.82	\0, /0, [/0, 5], 0.36	/0, /0, [>0, 4], 0.28
~0, [>0, 4], 0.67	>x, ~0, [>0, 3], 0.94	~0, \0, [>0, 5], 0.75	/0, >0, [>0, 2], 0.72	/0, /0, [>0, 5], 0.51
/0, [/0, 5], 0.72	>x, ~0, [>0, 4], 0.51	~0, /0, [>0, 6], 0.5	/0, >0, [>0, 3], 0.48	/0, /0, [>0, 6], 0.74
[-0, 5], 0.55	>x, ~0, [>0, 5], 0.26	~0, /0, [\0, 5], 0.72	/0, >0, [>0, 4], 0.51	/0, /0, [\0, 5], 0.32
~0, [>0, 5], 0.67	>x, ~0, [>0, 6], 0.64	\0, ~0, [>0, 4], 0.24	/0, >0, [-0, 3], 0.63	
~0, [>0, 6], 0.59	>x, /0, [>0, 4], 0.66	\0, ~0, [>0, 5], 0.55	/0, >0, [/0, 2], 0.71	
~0, [\0, 5], 0.44	>x, /0, [>0, 5], 0.56	\0, ~0, [>0, 6], 0.29	/0, >0, [/0, 3], 0.59	

Cycle de décision 5 : >O

~0, [>0, 4], 0.66	>0, /0, [>0, 5], 0.81	>x, /0, [>0, 5], 0.7	~0, /0, [>0, 4], 0.75	/0, >0, [>0, 2], 0.86
/0, [>0, 4], 0.64	>0, /0, [-0, 4], 0.59	>x, /0, [-0, 4], 0.84	~0, /0, [>0, 5], 0.54	/0, >0, [>0, 3], 0.35
/0, [>0, 5], 0.76	>0, /0, [-0, 5], 0.72	>x, /0, [>0, 4], 0.68	~0, /0, [\0, 5], 0.58	/0, >0, [/0, 2], 0.67
/0, [-0, 4], 0.8	>0, /0, [-0, 6], 0.66	>x, /0, [/0, 5], 0.87	\0, ~0, [>0, 4], 0.4	/0, >0, [/0, 3], 0.39
/0, [/0, 4], 0.66	>0, /0, [/0, 4], 0.47	~0, >0, [>0, 2], 0.71	\0, ~0, [>0, 5], 0.24	/0, /0, [>0, 3], 0.41
/0, [/0, 5], 0.78	>0, /0, [/0, 5], 0.78	~0, >0, [>0, 3], 0.69	\0, ~0, [>0, 6], 0.69	/0, /0, [>0, 4], 0.93
~0, [>0, 5], 0.64	>0, /0, [/0, 6], 0.94	~0, >0, [/0, 2], 0.8	\0, ~0, [/0, 4], 0.22	/0, /0, [>0, 5], 0.88
>0, ~0, [>0, 4], 0.61	>x, ~0, [>0, 4], 0.51	~0, >0, [/0, 3], 0.53	\0, ~0, [/0, 5], 0.43	
>0, /0, [>0, 4], 0.5	>x, ~0, [>0, 5], 0.68	~0, \0, [>0, 5], 0.54	\0, ~0, [/0, 6], 0.58	

Cycle de décision 6 : ~O

[>0, 4], 0.79	\0, [/0, 5], 0.26	>0, /0, [\0, 4], 0.74	v0, /0, [-0, 6], 0.87	/0, >0, [v0, 5], 0.43
[>0, 5], 0.79	/0, [>0, 6], 0.2	>0, /0, [/0, 4], 0.65	v0, /0, [/0, 6], 0.71	/0, >0, [v0, 6], 0.21
>0, [>0, 2], 0.79	/0, [\0, 6], 0.58	>x, /0, [>0, 5], 0.3	~0, v0, [>0, 4], 0.69	/0, >0, [\0, 5], 0.65
>0, [>0, 3], 0.79	>0, >0, [>0, 1], 0.3	>F, \0, [>0, 4], 0.7	~0, v0, [>0, 5], 0.69	/0, >0, [\0, 6], 0.89
>0, [/0, 2], 0.8	>0, >0, [>0, 2], 0.28	>F, \0, [>0, 5], 0.3	~0, v0, [\0, 5], 0.93	/0, v0, [>0, 5], 0.89
>0, [/0, 3], 0.79	>0, >0, [\0, 0], 0.3	>F, \0, [>0, 6], 0.49	~0, \0, [>0, 5], 0.88	/0, v0, [/0, 5], 0.24
\0, [>0, 5], 0.79	>0, >0, [\0, 1], 0.35	>F, \0, [/0, 3], 0.48	\0, >0, [/0, 4], 0.79	/0, >0, [>0, 3], 0.79
/0, [>0, 4], 0.75	>0, >0, [/0, 1], 0.4	>F, \0, [/0, 4], 0.45	\0, ~0, [>0, 5], 0.38	/0, \0, [>0, 4], 0.55
/0, [>0, 5], 0.54	>0, >0, [/0, 2], 0.23	>F, \0, [/0, 5], 0.57	\0, ~0, [>0, 6], 0.73	/0, \0, [>0, 5], 0.2
/0, [\0, 5], 0.79	>0, \0, [>0, 3], 0.56	v0, ~0, [>0, 4], 0.35	\0, ~0, [\0, 5], 0.23	/0, \0, [>0, 6], 0.57
>F, [>0, 3], 0.79	>0, \0, [>0, 4], 0.65	v0, ~0, [>0, 5], 0.39	\0, ~0, [\0, 6], 0.29	/0, \0, [/0, 3], 0.43
>F, [>0, 4], 0.36	>0, /0, [>0, 3], 0.49	v0, ~0, [>0, 7], 0.34	\0, /0, [>0, 4], 0.76	/0, \0, [/0, 4], 0.38
>F, [/0, 3], 0.26	>0, /0, [>0, 4], 0.42	v0, ~0, [/0, 5], 0.48	\0, /0, [>0, 5], 0.9	
>F, [/0, 4], 0.22	>0, /0, [\0, 3], 0.27	v0, /0, [>0, 6], 0.68	\0, /0, [>0, 6], 0.7	

Cycle de décision 7 : >O

[>0, 2], 0.79	/0, [>0, 3], 0.49	/0, [\0, 5], 0.21	~0, \0, [>0, 4], 0.38	\0, ~0, [\0, 4], 0.55
[>0, 3], 0.79	/0, [>0, 4], 0.42	>0, >0, [-0, 1], 0.31	~0, \0, [\0, 4], 0.27	\0, ~0, [\0, 5], 0.53
[/0, 2], 0.8	/0, [\0, 3], 0.31	>0, >0, [/0, 1], 0.47	~0, \0, [\0, 5], 0.35	\0, ~0, [/0, 4], 0.28
[/0, 3], 0.79	/0, [\0, 4], 0.74	>0, /0, [>0, 2], 0.39	\0, >0, [>0, 4], 0.34	\0, /0, [>0, 2], 0.67
>0, [>0, 1], 0.3	/0, [/0, 4], 0.65	>0, /0, [\0, 2], 0.22	\0, >0, [-0, 3], 0.82	\0, /0, [>0, 3], 0.6
>0, [>0, 2], 0.38	\0, [-0, 3], 0.38	>0, /0, [\0, 3], 0.36	\0, >0, [-0, 4], 0.6	\0, /0, [>0, 4], 0.24
>0, [\0, 0], 0.3	\0, [-0, 4], 0.68	v0, ~0, [>0, 3], 0.66	\0, >0, [-0, 5], 0.33	\0, /0, [/0, 3], 0.52
>0, [\0, 1], 0.35	\0, [-0, 5], 0.39	v0, ~0, [/0, 2], 0.69	\0, >0, [/0, 4], 0.62	\0, /0, [/0, 4], 0.65
>0, [/0, 1], 0.4	\0, [/0, 4], 0.4	v0, ~0, [/0, 3], 0.31	\0, >0, [/0, 5], 0.76	/0, >0, [v0, 4], 0.89
>0, [/0, 2], 0.4	\0, [/0, 5], 0.23	~0, v0, [>0, 3], 0.44	\0, ~0, [>0, 3], 0.44	/0, >0, [\0, 4], 0.89
\0, [>0, 3], 0.56	/0, [>0, 5], 0.22	~0, v0, [>0, 4], 0.32	\0, ~0, [>0, 4], 0.21	/0, >0, [\0, 5], 0.79
\0, [>0, 4], 0.65	/0, [v0, 4], 0.39	~0, \0, [>0, 3], 0.59	\0, ~0, [>0, 5], 0.45	/0, v0, [>0, 3], 0.55

Annexe A : résultats détaillés des expérimentations

/0, v0, [>0, 4], 0.46	/0, v0, [/0, 4], 0.34	/0, \0, [>0, 3], 0.36	/0, \0, [/0, 3], 0.46
/0, v0, [>0, 5], 0.49	/0, v0, [/0, 5], 0.36	/0, \0, [>0, 4], 0.2	/0, \0, [/0, 4], 0.46
/0, v0, [^0, 4], 0.22	/0, \0, [>0, 2], 0.67	/0, \0, [/0, 2], 0.56	

Cycle de décision 8 : >O (l'agent voit à nouveau la proie)

\0, [^0, 1], 0.88	\0, /0, [\0, 1], 0.79	v0, ^0, [\0, 0], 0.72	\0, >F, [/0, 8], 0.41	\0, /0, [^0, 1], 0.28
\0, [^0, 2], 0.88	\0, /0, [/0, 0], 0.59	v0, ^0, [/0, 1], 0.9	\0, ^0, [v0, 1], 0.24	/0, v0, [v0, 2], 0.26
\0, [\0, 0], 0.88	\0, /0, [/0, 1], 0.85	v0, ^0, [/0, 2], 0.27	\0, ^0, [v0, 3], 0.23	/0, v0, [\0, 2], 0.47
\0, [/0, 0], 0.88	/0, \0, [^0, 1], 0.57	v0, /0, [^0, 2], 0.76	\0, ^0, [\0, 0], 0.28	/0, v0, [/0, 1], 0.3
\0, [/0, 1], 0.9	/0, \0, [\0, 0], 0.65	v0, /0, [/0, 2], 0.53	\0, ^0, [\0, 1], 0.22	/0, \0, [v0, 1], 0.24
/0, [v0, 1], 0.76	/0, \0, [/0, 1], 0.74	^0, v0, [/0, 2], 0.6	\0, ^0, [\0, 2], 0.3	/0, \0, [\0, 1], 0.47
/0, [v0, 2], 0.68	^0, [v0, 1], 0.22	^0, \0, [v0, 1], 0.74	\0, ^0, [\0, 4], 0.24	/0, \0, [/0, 0], 0.39
/0, [\0, 1], 0.68	^0, [/0, 1], 0.24	^0, \0, [\0, 2], 0.35	\0, ^0, [/0, 1], 0.24	
/0, [\0, 2], 0.74	\0, [/0, 2], 0.88	\0, >F, [^0, 7], 0.27	\0, \0, [^0, 1], 0.24	
\0, /0, [>0, 1], 0.7	>0, \0, [/0, 1], 0.68	\0, >F, [^0, 8], 0.25	\0, \0, [/0, 1], 0.24	
\0, /0, [\0, 0], 0.75	v0, >F, [v0, 2], 0.6	\0, >F, [/0, 7], 0.44	\0, /0, [v0, 1], 0.27	

Cycle de décision 9 : >F

>F, [>0, 3], 0.28	>F, [/0, 3], 0.28	>X, \0, [>0, 2], 0.04	^0, v0, [\0, 3], 0.04	/0, \0, [v0, 3], 0.04
>F, [v0, 2], 0.7	>0, v0, [>0, 1], 0.28	>X, \0, [>0, 3], 0.04	^0, v0, [\0, 4], 0.04	/0, \0, [v0, 4], 0.04
>F, [v0, 3], 0.47	>0, v0, [\0, 1], 0.57	>X, \0, [\0, 2], 0.04	^0, v0, [\0, 5], 0.04	/0, \0, [v0, 5], 0.04
>F, [v0, 4], 0.46	>0, v0, [/0, 1], 0.65	>X, \0, [\0, 3], 0.04	\0, /0, [v0, 3], 0.04	/0, \0, [v0, 6], 0.04
>F, [v0, 5], 0.75	>0, v0, [/0, 2], 0.34	^0, v0, [>0, 3], 0.04	\0, /0, [\0, 3], 0.04	/0, \0, [\0, 4], 0.04
>F, [\0, 2], 0.79	>0, \0, [v0, 4], 0.44	^0, v0, [>0, 4], 0.04	/0, v0, [>0, 2], 0.04	/0, \0, [\0, 5], 0.04
>F, [\0, 3], 0.59	>0, \0, [\0, 4], 0.74	^0, v0, [>0, 5], 0.04	/0, v0, [>0, 3], 0.04	/0, \0, [\0, 6], 0.04
>F, [\0, 4], 0.85	>X, v0, [>0, 1], 0.04	^0, v0, [v0, 3], 0.04	/0, \0, [>0, 3], 0.04	/0, \0, [\0, 7], 0.04
>F, [\0, 5], 0.65	>X, v0, [/0, 1], 0.04	^0, v0, [v0, 4], 0.04	/0, \0, [>0, 4], 0.04	/0, \0, [/0, 3], 0.04
>F, [\0, 6], 0.65	>X, v0, [/0, 2], 0.04	^0, v0, [v0, 5], 0.04	/0, \0, [>0, 5], 0.04	

3.2 Expérimentation avec une proie seule (à droite)

Suivi de la proie :

Cycle de décision 1 : >O

>0, [>0, 5], 0.86	>0, >0, [>0, 4], 0.71	>0, [v0, 6], 0.86	v0, >0, [/0, 8], 0.62	\0, >0, [>0, 5], 0.6
>0, [>0, 6], 0.92	>0, >0, [>0, 5], 0.79	>0, [/0, 6], 0.92	v0, >0, [/0, 9], 0.4	\0, >0, [/0, 5], 0.81
>0, [>0, 7], 0.86	>0, >0, [v0, 5], 0.55	v0, [>0, 8], 0.51	v0, ^0, [>0, 8], 0.64	\0, ^0, [>0, 8], 0.42
>0, [\0, 5], 0.92	>0, >0, [>0, 4], 0.6	v0, [>0, 9], 0.32	v0, /0, [>0, 6], 0.62	\0, ^0, [>0, 9], 0.51
>0, [\0, 6], 0.92	>0, >0, [\0, 5], 0.81	v0, [^0, 8], 0.68	v0, /0, [>0, 7], 0.67	\0, /0, [\0, 6], 0.68
v0, [^0, 9], 0.8	>0, \0, [>0, 5], 0.51	v0, [/0, 8], 0.8	v0, /0, [>0, 8], 0.62	/0, >0, [v0, 8], 0.32
\0, [>0, 6], 0.71	>0, \0, [>0, 6], 0.68	v0, [/0, 9], 0.37	v0, /0, [/0, 8], 0.78	/0, >0, [v0, 9], 0.23
\0, [>0, 7], 0.62	v0, ^0, [>0, 6], 0.64	\0, [>0, 8], 0.31	^0, v0, [>0, 9], 0.62	/0, >0, [\0, 8], 0.22
/0, [>0, 8], 0.64	v0, ^0, [>0, 7], 0.64	>0, >0, [>0, 6], 0.39	^0, v0, [/0, 8], 0.71	/0, >0, [\0, 9], 0.25
/0, [>0, 9], 0.62	^0, v0, [>0, 7], 0.67	>0, >0, [v0, 4], 0.46	^0, \0, [>0, 8], 0.79	/0, v0, [>0, 5], 0.57
/0, [v0, 8], 0.67	^0, v0, [>0, 8], 0.62	>0, >0, [v0, 6], 0.55	^0, \0, [v0, 9], 0.28	/0, v0, [>0, 7], 0.5
/0, [v0, 9], 0.62	\0, /0, [>0, 6], 0.68	>0, >0, [\0, 6], 0.23	^0, \0, [\0, 8], 0.27	/0, \0, [>0, 7], 0.26
/0, [\0, 8], 0.78	\0, /0, [>0, 7], 0.75	>0, /0, [\0, 7], 0.45	^0, \0, [\0, 9], 0.54	/0, \0, [/0, 6], 0.42
/0, [\0, 9], 0.62	/0, \0, [>0, 6], 0.52	v0, >0, [^0, 8], 0.71	\0, >0, [>0, 4], 0.2	

Cycle de décision 2 : >O

[>0, 5], 0.86	\0, [>0, 6], 0.68	v0, ^0, [>0, 5], 0.67	/0, v0, [>0, 5], 0.8	v0, /0, [>0, 7], 0.52
[>0, 6], 0.92	[v0, 6], 0.86	v0, ^0, [>0, 6], 0.52	/0, \0, [>0, 5], 0.71	^0, v0, [>0, 4], 0.44
[>0, 7], 0.86	[/0, 6], 0.92	v0, ^0, [>0, 7], 0.4	/0, \0, [>0, 6], 0.71	^0, v0, [>0, 7], 0.42
[\0, 5], 0.92	>0, [>0, 6], 0.39	^0, v0, [>0, 5], 0.42	/0, \0, [>0, 7], 0.75	^0, v0, [\0, 6], 0.86
[\0, 6], 0.92	>0, [v0, 4], 0.46	^0, v0, [\0, 4], 0.86	/0, \0, [\0, 6], 0.54	\0, >0, [/0, 4], 0.92
>0, [>0, 4], 0.71	>0, [v0, 6], 0.55	^0, v0, [\0, 5], 0.86	/0, \0, [/0, 6], 0.54	\0, /0, [>0, 7], 0.51
>0, [>0, 5], 0.79	>0, [\0, 6], 0.44	\0, >0, [>0, 3], 0.92	>0, \0, [>0, 6], 0.41	\0, /0, [\0, 6], 0.68
>0, [v0, 5], 0.55	/0, [\0, 7], 0.45	\0, >0, [>0, 4], 0.92	>0, \0, [\0, 5], 0.44	/0, v0, [>0, 7], 0.8
>0, [\0, 4], 0.67	/0, [v0, 7], 0.4	\0, >0, [/0, 3], 0.92	v0, ^0, [>0, 8], 0.4	/0, \0, [v0, 6], 0.71
>0, [\0, 5], 0.81	>0, \0, [>0, 4], 0.41	\0, /0, [>0, 6], 0.51	v0, /0, [>0, 5], 0.28	/0, \0, [\0, 5], 0.79
\0, [>0, 5], 0.92	>0, \0, [>0, 5], 0.41	\0, /0, [\0, 5], 0.68	v0, /0, [>0, 6], 0.3	

Cycle de décision 3 : >O

>0, [4], 0.71	\0, [0, 3], 0.68	>0, [v0, 6], 0.78	v0, ~0, [0, 7], 0.51	\0, \0, [0, 4], 0.54
>0, [5], 0.92	\0, [\0, 3], 0.86	>0, [\0, 6], 0.83	v0, ~0, [\0, 5], 0.51	\0, \0, [0, 5], 0.81
[v0, 5], 0.55	\0, [\0, 4], 0.92	v0, [0, 4], 0.56	v0, ~0, [\0, 6], 0.86	\0, \0, [/0, 3], 0.8
\0, [4], 0.67	\0, [/0, 3], 0.59	v0, [0, 5], 0.59	v0, \0, [0, 5], 0.53	\0, \0, [/0, 4], 0.37
\0, [5], 0.81	\0, [/0, 4], 0.68	v0, [0, 6], 0.52	v0, /0, [0, 5], 0.57	\0, /0, [0, 5], 0.53
>0, [6], 0.68	>0, \0, [0, 3], 0.8	>0, v0, [0, 3], 0.59	v0, /0, [0, 6], 0.69	\0, /0, [v0, 5], 0.44
[v0, 4], 0.46	>0, \0, [0, 4], 0.71	>0, v0, [0, 4], 0.52	~0, v0, [0, 5], 0.52	\0, /0, [v0, 6], 0.49
[v0, 6], 0.55	>0, \0, [/0, 4], 0.79	>0, v0, [0, 5], 0.71	~0, v0, [v0, 4], 0.52	\0, /0, [\0, 6], 0.27
\0, [6], 0.81	~0, v0, [v0, 5], 0.55	>0, \0, [0, 5], 0.56	~0, v0, [v0, 6], 0.55	/0, v0, [0, 4], 0.61
\0, [0, 4], 0.68	~0, v0, [\0, 4], 0.6	>0, \0, [0, 6], 0.65	~0, v0, [\0, 6], 0.5	/0, v0, [0, 5], 0.21
\0, [0, 5], 0.92	~0, v0, [\0, 5], 0.81	>0, \0, [\0, 5], 0.53	~0, v0, [\0, 7], 0.49	/0, \0, [0, 6], 0.37
\0, [0, 6], 0.65	\0, >0, [0, 3], 0.94	>X, \0, [0, 4], 0.6	\0, >0, [0, 2], 0.86	/0, \0, [v0, 5], 0.55
\0, [\0, 5], 0.53	\0, >0, [\0, 2], 0.85	>F, \0, [0, 5], 0.65	\0, >0, [0, 4], 0.94	/0, \0, [v0, 6], 0.42
>0, [0, 4], 0.54	\0, >0, [/0, 3], 0.59	v0, >0, [0, 3], 0.55	\0, >0, [v0, 3], 0.67	/0, \0, [\0, 6], 0.7
>0, [v0, 4], 0.6	\0, /0, [\0, 5], 0.62	v0, >0, [/0, 3], 0.5	\0, >0, [\0, 3], 0.85	
>0, [v0, 5], 0.81	/0, \0, [0, 5], 0.61	v0, ~0, [0, 5], 0.5	\0, >0, [\0, 4], 0.5	
>0, [\0, 5], 0.83	/0, \0, [\0, 5], 0.7	v0, ~0, [0, 6], 0.62	\0, \0, [0, 3], 0.36	

Cycle de décision 4 :

>0, [4], 0.72	\0, [\0, 3], 0.37	>X, v0, [0, 5], 0.52	v0, /0, [0, 5], 0.85	\0, /0, [v0, 5], 0.21
[v0, 4], 0.72	\0, [\0, 4], 0.6	>X, \0, [0, 3], 0.52	v0, /0, [0, 6], 0.6	\0, /0, [v0, 6], 0.92
[v0, 5], 0.81	\0, [/0, 3], 0.67	>X, \0, [0, 4], 0.55	~0, v0, [v0, 5], 0.6	\0, /0, [\0, 5], 0.95
\0, [5], 0.83	>0, v0, [0, 2], 0.62	>X, \0, [0, 5], 0.5	~0, v0, [v0, 6], 0.54	\0, /0, [\0, 6], 0.97
\0, [0, 3], 0.8	>0, v0, [0, 3], 0.31	>X, \0, [\0, 4], 0.49	~0, v0, [\0, 4], 0.55	/0, v0, [0, 3], 0.72
\0, [0, 4], 0.72	>0, v0, [0, 4], 0.38	>F, \0, [0, 5], 0.67	~0, v0, [\0, 5], 0.54	/0, v0, [0, 4], 0.66
\0, [/0, 4], 0.79	>0, v0, [0, 5], 0.93	>F, \0, [/0, 3], 0.85	~0, v0, [\0, 6], 0.57	/0, v0, [0, 5], 0.8
[v0, 6], 0.78	>0, v0, [/0, 2], 0.68	v0, >0, [0, 2], 0.5	\0, >0, [0, 3], 0.74	/0, v0, [0, 6], 0.83
\0, [6], 0.83	>0, v0, [/0, 4], 0.49	v0, >0, [0, 3], 0.59	\0, >0, [0, 4], 0.5	/0, v0, [\0, 4], 0.91
v0, [0, 3], 0.6	>0, \0, [0, 3], 0.23	v0, >0, [/0, 3], 0.64	\0, >0, [0, 3], 0.58	/0, v0, [\0, 5], 0.92
v0, [0, 4], 0.72	>0, \0, [0, 4], 0.7	v0, ~0, [0, 5], 0.66	\0, >0, [\0, 3], 0.7	/0, v0, [/0, 4], 0.37
v0, [0, 5], 0.71	>0, \0, [0, 5], 0.46	v0, ~0, [0, 6], 0.75	\0, >0, [\0, 4], 0.57	/0, \0, [v0, 5], 0.49
\0, [0, 5], 0.72	>0, \0, [0, 6], 0.59	v0, ~0, [\0, 5], 0.86	\0, >0, [/0, 3], 0.56	/0, \0, [v0, 6], 0.45
\0, [0, 6], 0.65	>0, \0, [v0, 5], 0.49	v0, ~0, [\0, 6], 0.89	\0, v0, [0, 5], 0.48	/0, \0, [\0, 5], 0.22
\0, [\0, 5], 0.72	>0, \0, [\0, 4], 0.62	v0, \0, [0, 4], 0.8	\0, \0, [0, 3], 0.77	/0, \0, [\0, 6], 0.92
>0, [v0, 6], 0.72	>0, \0, [\0, 5], 0.61	v0, \0, [0, 5], 0.54	\0, \0, [0, 4], 0.74	
>0, [\0, 6], 0.5	>0, \0, [/0, 4], 0.56	v0, \0, [0, 6], 0.5	\0, \0, [0, 5], 0.77	
v0, [0, 6], 0.25	>X, v0, [0, 3], 0.52	v0, \0, [~0, 4], 0.54	\0, \0, [/0, 3], 0.31	
\0, [v0, 5], 0.61	>X, v0, [0, 4], 0.59	v0, \0, [~0, 6], 0.86	\0, \0, [/0, 4], 0.61	

Cycle de décision 5 : >O

[v0, 6], 0.72	>0, v0, [0, 4], 0.66	>X, \0, [0, 5], 0.59	~0, v0, [v0, 6], 0.59	/0, >F, [v0, 2], 0.48
\0, [6], 0.5	>0, v0, [0, 5], 0.55	>X, \0, [\0, 4], 0.61	~0, v0, [\0, 5], 0.31	/0, >F, [v0, 3], 0.77
v0, [0, 2], 0.62	>0, v0, [/0, 2], 0.52	>X, \0, [v0, 5], 0.47	~0, v0, [0, 6], 0.81	/0, >F, [\0, 1], 0.74
v0, [0, 3], 0.83	>0, \0, [0, 4], 0.2	>F, \0, [/0, 3], 0.24	\0, >0, [0, 4], 0.52	/0, >F, [\0, 2], 0.77
v0, [0, 4], 0.83	>0, \0, [0, 5], 0.39	v0, >0, [0, 2], 0.28	\0, >0, [v0, 4], 0.25	/0, >F, [\0, 3], 0.49
v0, [0, 5], 0.93	>0, \0, [0, 6], 0.23	v0, >0, [0, 3], 0.91	\0, >0, [v0, 5], 0.67	/0, v0, [0, 3], 0.92
v0, [/0, 2], 0.68	>0, \0, [v0, 5], 0.4	v0, >0, [\0, 2], 0.71	\0, >0, [\0, 3], 0.39	/0, v0, [0, 4], 0.83
v0, [/0, 4], 0.49	>0, \0, [v0, 6], 0.53	v0, ~0, [v0, 6], 0.44	\0, >0, [\0, 4], 0.43	/0, v0, [0, 5], 0.66
\0, [0, 3], 0.23	>0, \0, [\0, 4], 0.3	v0, ~0, [\0, 6], 0.54	\0, >0, [\0, 5], 0.73	/0, v0, [0, 6], 0.55
\0, [0, 4], 0.83	>0, \0, [\0, 5], 0.2	v0, ~0, [\0, 7], 0.68	\0, v0, [0, 5], 0.69	/0, v0, [\0, 4], 0.27
\0, [0, 5], 0.83	>0, \0, [\0, 6], 0.2	v0, \0, [0, 4], 0.33	\0, \0, [0, 3], 0.74	/0, v0, [\0, 5], 0.34
\0, [0, 6], 0.83	>X, v0, [0, 3], 0.23	v0, \0, [0, 5], 0.68	\0, \0, [0, 4], 0.59	/0, v0, [/0, 4], 0.21
\0, [v0, 5], 0.83	>X, v0, [0, 4], 0.52	v0, \0, [0, 6], 0.38	\0, \0, [0, 5], 0.64	/0, \0, [v0, 5], 0.59
\0, [\0, 4], 0.83	>X, v0, [0, 5], 0.3	v0, \0, [~0, 4], 0.57	\0, \0, [/0, 4], 0.66	/0, \0, [v0, 6], 0.77
\0, [\0, 5], 0.83	>X, v0, [/0, 3], 0.21	v0, \0, [~0, 6], 0.24	\0, /0, [v0, 6], 0.75	/0, \0, [\0, 5], 0.35
\0, [/0, 4], 0.83	>X, v0, [/0, 4], 0.33	v0, \0, [0, 5], 0.62	/0, >F, [0, 2], 0.6	/0, \0, [\0, 6], 0.66
>0, v0, [0, 2], 0.83	>X, \0, [0, 3], 0.53	v0, /0, [0, 5], 0.5	/0, >F, [0, 3], 0.6	
>0, v0, [0, 3], 0.24	>X, \0, [0, 4], 0.4	v0, /0, [\0, 5], 0.24	/0, >F, [v0, 1], 0.54	

Cycle de décision 6 : >O

v0, [0, 2], 0.83	\0, [\0, 5], 0.92	>0, \0, [0, 4], 0.31	>X, v0, [/0, 4], 0.52	v0, \0, [0, 5], 0.21
v0, [0, 3], 0.92	\0, [\0, 6], 0.92	>0, \0, [0, 5], 0.68	>X, \0, [0, 3], 0.88	v0, \0, [0, 6], 0.21
v0, [0, 4], 0.92	v0, [0, 6], 0.92	>0, \0, [0, 6], 0.79	>X, \0, [0, 4], 0.49	v0, /0, [0, 5], 0.45
v0, [0, 5], 0.92	v0, [\0, 5], 0.87	>0, \0, [v0, 5], 0.7	>X, \0, [0, 5], 0.22	v0, /0, [v0, 6], 0.2
v0, [/0, 2], 0.92	v0, [/0, 4], 0.34	>0, \0, [v0, 6], 0.44	>X, \0, [\0, 4], 0.42	v0, /0, [\0, 5], 0.24
\0, [0, 4], 0.92	\0, [0, 3], 0.25	>0, \0, [\0, 4], 0.35	>X, \0, [\0, 5], 0.23	v0, /0, [\0, 6], 0.23
\0, [0, 5], 0.92	>0, v0, [0, 2], 0.39	>0, \0, [\0, 5], 0.24	v0, >0, [0, 2], 0.52	\0, >0, [v0, 3], 0.27
\0, [0, 6], 0.92	>0, v0, [0, 3], 0.83	>0, \0, [\0, 6], 0.92	v0, >0, [0, 3], 0.83	\0, >0, [v0, 4], 0.33
\0, [v0, 5], 0.92	>0, v0, [0, 4], 0.82	>X, v0, [0, 3], 0.88	v0, >0, [\0, 2], 0.38	\0, >0, [v0, 5], 0.26
\0, [v0, 6], 0.92	>0, v0, [0, 5], 0.59	>X, v0, [0, 4], 0.48	v0, >0, [/0, 3], 0.5	\0, >0, [\0, 3], 0.36
\0, [\0, 4], 0.92	>0, v0, [/0, 2], 0.78	>X, v0, [0, 5], 0.47	v0, \0, [0, 4], 0.2	\0, >0, [\0, 4], 0.33

Annexe A : résultats détaillés des expérimentations

$\setminus 0, \forall 0, [> 0, 5], 0.27$	$\setminus 0, \setminus 0, [> 0, 5], 0.29$	$/ 0, \forall 0, [> 0, 6], 0.34$	$/ 0, \forall 0, [\setminus 0, 5], 0.78$
$\setminus 0, \setminus 0, [> 0, 3], 0.29$	$/ 0, \forall 0, [> 0, 4], 0.53$	$/ 0, \forall 0, [\forall 0, 5], 0.64$	$/ 0, \setminus 0, [\forall 0, 6], 0.79$
$\setminus 0, \setminus 0, [> 0, 4], 0.6$	$/ 0, \forall 0, [> 0, 5], 0.6$	$/ 0, \forall 0, [\setminus 0, 4], 0.62$	

Cycle de décision 7 : $\forall 0$

$[> 0, 2], 0.83$	$\setminus 0, [> 0, 4], 0.7$	$> 0, \setminus 0, [> 0, 4], 0.76$	$\sim 0, \forall 0, [> 0, 5], 0.28$	$\setminus 0, / 0, [> 0, 6], 0.88$
$[> 0, 3], 0.92$	$\setminus 0, [> 0, 5], 0.7$	$> 0, \setminus 0, [/ 0, 3], 0.87$	$\sim 0, \forall 0, [\setminus 0, 5], 0.25$	$/ 0, \forall 0, [> 0, 5], 0.48$
$[> 0, 4], 0.92$	$\setminus 0, [> 0, 6], 0.7$	$> 0, \setminus 0, [/ 0, 4], 0.76$	$\setminus 0, > 0, [> 0, 3], 0.34$	$/ 0, \forall 0, [/ 0, 5], 0.47$
$[> 0, 5], 0.92$	$/ 0, [> 0, 5], 0.45$	$> F, \setminus 0, [> 0, 4], 0.4$	$\setminus 0, > 0, [> 0, 4], 0.54$	$/ 0, \setminus 0, [> 0, 3], 0.52$
$[/ 0, 2], 0.92$	$/ 0, [\forall 0, 6], 0.2$	$> F, \setminus 0, [> 0, 5], 0.57$	$\setminus 0, > 0, [/ 0, 3], 0.59$	$/ 0, \setminus 0, [> 0, 4], 0.88$
$[> 0, 6], 0.92$	$/ 0, [\setminus 0, 5], 0.43$	$> F, \setminus 0, [/ 0, 4], 0.52$	$\setminus 0, > 0, [/ 0, 4], 0.74$	$/ 0, \setminus 0, [> 0, 5], 0.49$
$[\setminus 0, 5], 0.87$	$/ 0, [\setminus 0, 6], 0.23$	$\forall 0, \sim 0, [> 0, 5], 0.51$	$\setminus 0, \sim 0, [\setminus 0, 5], 0.79$	$/ 0, \setminus 0, [> 0, 6], 0.25$
$[/ 0, 4], 0.34$	$> 0, [> 0, 4], 0.35$	$\forall 0, \sim 0, [> 0, 6], 0.67$	$\setminus 0, \sim 0, [\setminus 0, 6], 0.7$	$/ 0, \setminus 0, [/ 0, 4], 0.6$
$> 0, [> 0, 2], 0.52$	$\setminus 0, [/ 0, 4], 0.7$	$\forall 0, \sim 0, [> 0, 7], 0.46$	$\setminus 0, \setminus 0, [\sim 0, 6], 0.68$	
$> 0, [> 0, 3], 0.83$	$\setminus 0, [/ 0, 5], 0.77$	$\forall 0, / 0, [> 0, 5], 0.46$	$\setminus 0, \setminus 0, [/ 0, 5], 0.79$	
$> 0, [\setminus 0, 2], 0.38$	$/ 0, [> 0, 6], 0.29$	$\forall 0, / 0, [> 0, 6], 0.66$	$\setminus 0, / 0, [> 0, 4], 0.7$	
$> 0, [/ 0, 3], 0.7$	$> 0, \setminus 0, [> 0, 3], 0.43$	$\sim 0, \forall 0, [> 0, 4], 0.84$	$\setminus 0, / 0, [> 0, 5], 0.92$	

Cycle de décision 8 : $\setminus 0$

$[> 0, 4], 0.88$	$/ 0, [> 0, 5], 0.92$	$> 0, / 0, [> 0, 3], 0.5$	$\forall 0, \sim 0, [> 0, 6], 0.64$	$/ 0, > 0, [> 0, 3], 0.3$
$[> 0, 5], 0.74$	$/ 0, [> 0, 6], 0.88$	$> 0, / 0, [> 0, 4], 0.67$	$\forall 0, \sim 0, [> 0, 7], 0.84$	$/ 0, > 0, [> 0, 4], 0.29$
$[> 0, 6], 0.7$	$> 0, [> 0, 5], 0.73$	$> 0, / 0, [> 0, 5], 0.85$	$\forall 0, / 0, [/ 0, 7], 0.35$	$/ 0, > 0, [/ 0, 3], 0.32$
$[/ 0, 4], 0.75$	$> 0, [/ 0, 5], 0.76$	$> 0, / 0, [/ 0, 3], 0.54$	$\sim 0, \forall 0, [> 0, 4], 0.62$	$/ 0, \forall 0, [> 0, 5], 0.23$
$[/ 0, 5], 0.77$	$> F, [> 0, 3], 0.2$	$> 0, / 0, [/ 0, 4], 0.75$	$\sim 0, \forall 0, [> 0, 5], 0.36$	$/ 0, \forall 0, [\sim 0, 6], 0.46$
$> 0, [> 0, 3], 0.88$	$> F, [> 0, 4], 0.23$	$> X, / 0, [> 0, 5], 0.58$	$\sim 0, \forall 0, [> 0, 6], 0.22$	$/ 0, \forall 0, [/ 0, 5], 0.21$
$> 0, [> 0, 4], 0.54$	$> F, [/ 0, 3], 0.67$	$> X, / 0, [> 0, 6], 0.47$	$\sim 0, \forall 0, [> 0, 7], 0.26$	$/ 0, \setminus 0, [> 0, 4], 0.42$
$> 0, [/ 0, 3], 0.88$	$> F, [/ 0, 4], 0.43$	$> F, \setminus 0, [> 0, 4], 0.93$	$\sim 0, \setminus 0, [> 0, 5], 0.52$	$/ 0, \setminus 0, [> 0, 5], 0.32$
$> 0, [/ 0, 4], 0.76$	$\setminus 0, [> 0, 6], 0.88$	$> F, \setminus 0, [> 0, 5], 0.59$	$\sim 0, \setminus 0, [> 0, 6], 0.22$	$/ 0, \setminus 0, [> 0, 6], 0.43$
$\sim 0, [\setminus 0, 5], 0.79$	$/ 0, [\setminus 0, 5], 0.53$	$> F, \setminus 0, [> 0, 6], 0.63$	$\setminus 0, \sim 0, [> 0, 5], 0.24$	$/ 0, \setminus 0, [/ 0, 5], 0.56$
$\sim 0, [\setminus 0, 6], 0.7$	$/ 0, [\setminus 0, 6], 0.36$	$> F, \setminus 0, [/ 0, 3], 0.68$	$\setminus 0, \sim 0, [> 0, 6], 0.33$	
$\setminus 0, [\sim 0, 6], 0.68$	$> 0, > 0, [> 0, 2], 0.56$	$> F, \setminus 0, [/ 0, 4], 0.5$	$\setminus 0, / 0, [> 0, 5], 0.24$	
$\setminus 0, [/ 0, 5], 0.79$	$> 0, > 0, [/ 0, 2], 0.74$	$> F, \setminus 0, [/ 0, 5], 0.3$	$\setminus 0, / 0, [> 0, 6], 0.31$	
$/ 0, [> 0, 4], 0.7$	$> 0, \sim 0, [> 0, 4], 0.69$	$\forall 0, \sim 0, [> 0, 5], 0.56$	$\setminus 0, / 0, [/ 0, 5], 0.44$	

Cycle de décision 9 : > 0

$[> 0, 3], 0.88$	$/ 0, [> 0, 5], 0.85$	$> X, / 0, [/ 0, 4], 0.62$	$\setminus 0, \sim 0, [> 0, 6], 0.52$	$/ 0, \forall 0, [/ 0, 5], 0.56$
$[> 0, 4], 0.54$	$/ 0, [/ 0, 3], 0.54$	$> F, \setminus 0, [/ 0, 4], 0.43$	$\setminus 0, \sim 0, [/ 0, 4], 0.58$	$/ 0, \setminus 0, [> 0, 3], 0.74$
$[/ 0, 3], 0.88$	$/ 0, [/ 0, 4], 0.84$	$\forall 0, \sim 0, [> 0, 4], 0.35$	$\setminus 0, / 0, [> 0, 4], 0.77$	$/ 0, \setminus 0, [> 0, 4], 0.69$
$[/ 0, 4], 0.76$	$> F, [> 0, 3], 0.44$	$\forall 0, \sim 0, [> 0, 5], 0.22$	$\setminus 0, / 0, [> 0, 5], 0.21$	$/ 0, \setminus 0, [> 0, 5], 0.5$
$[> 0, 5], 0.73$	$> F, [/ 0, 3], 0.36$	$\forall 0, \sim 0, [/ 0, 4], 0.22$	$\setminus 0, / 0, [/ 0, 4], 0.39$	$/ 0, \setminus 0, [> 0, 3], 0.67$
$[/ 0, 5], 0.76$	$/ 0, [\setminus 0, 5], 0.52$	$\forall 0, \sim 0, [/ 0, 5], 0.22$	$/ 0, > 0, [> 0, 2], 0.39$	$/ 0, \setminus 0, [/ 0, 4], 0.85$
$> 0, [> 0, 2], 0.56$	$> 0, / 0, [> 0, 3], 0.84$	$\sim 0, \forall 0, [> 0, 4], 0.22$	$/ 0, > 0, [> 0, 3], 0.48$	$/ 0, \setminus 0, [/ 0, 5], 0.54$
$> 0, [/ 0, 2], 0.74$	$> 0, / 0, [> 0, 4], 0.36$	$\sim 0, \forall 0, [> 0, 5], 0.22$	$/ 0, > 0, [/ 0, 2], 0.34$	
$\sim 0, [> 0, 4], 0.69$	$> 0, / 0, [/ 0, 3], 0.45$	$\sim 0, \forall 0, [/ 0, 4], 0.22$	$/ 0, > 0, [/ 0, 3], 0.63$	
$/ 0, [> 0, 3], 0.5$	$> 0, / 0, [/ 0, 4], 0.77$	$\setminus 0, \sim 0, [> 0, 4], 0.22$	$/ 0, \forall 0, [> 0, 5], 0.54$	
$/ 0, [> 0, 4], 0.67$	$> X, / 0, [> 0, 5], 0.83$	$\setminus 0, \sim 0, [> 0, 5], 0.22$	$/ 0, \forall 0, [/ 0, 4], 0.56$	

Cycle de décision 10 : > 0

$[> 0, 2], 0.56$	$> 0, [> 0, 1], 0.75$	$/ 0, [\setminus 0, 3], 0.76$	$\setminus 0, > 0, [\sim 0, 4], 0.31$	$\setminus 0, / 0, [> 0, 4], 0.31$
$[/ 0, 2], 0.74$	$> 0, [> 0, 2], 0.58$	$/ 0, [\setminus 0, 4], 0.72$	$\setminus 0, > 0, [/ 0, 4], 0.61$	$\setminus 0, / 0, [/ 0, 4], 0.39$
$/ 0, [> 0, 3], 0.84$	$> 0, [/ 0, 1], 0.47$	$> 0, > 0, [\sim 0, 1], 0.2$	$\setminus 0, \sim 0, [> 0, 3], 0.31$	$/ 0, \setminus 0, [> 0, 2], 0.7$
$/ 0, [> 0, 4], 0.75$	$> 0, [/ 0, 2], 0.93$	$> 0, > 0, [\setminus 0, 1], 0.2$	$\setminus 0, \sim 0, [> 0, 4], 0.32$	$/ 0, \setminus 0, [> 0, 3], 0.33$
$/ 0, [/ 0, 3], 0.75$	$/ 0, [\forall 0, 3], 0.59$	$> 0, > 0, [/ 0, 1], 0.33$	$\setminus 0, \sim 0, [> 0, 5], 0.65$	$/ 0, \setminus 0, [/ 0, 2], 0.7$
$/ 0, [/ 0, 4], 0.77$	$/ 0, [\forall 0, 4], 0.57$	$> X, / 0, [> 0, 3], 0.26$	$\setminus 0, \sim 0, [/ 0, 4], 0.56$	$/ 0, \setminus 0, [/ 0, 3], 0.58$

Cycle de décision 11 : > 0

$[> 0, 1], 0.75$	$\setminus 0, [\sim 0, 4], 0.44$	$\forall 0, \sim 0, [/ 0, 1], 0.69$	$\sim 0, \setminus 0, [> 0, 4], 0.27$	$\setminus 0, / 0, [> 0, 3], 0.49$
$[> 0, 2], 0.58$	$\setminus 0, [/ 0, 2], 0.51$	$\forall 0, \sim 0, [/ 0, 2], 0.36$	$\setminus 0, > 0, [\sim 0, 2], 0.43$	$\setminus 0, / 0, [\setminus 0, 1], 0.22$
$[/ 0, 1], 0.47$	$/ 0, [> 0, 2], 0.62$	$\forall 0, \sim 0, [/ 0, 3], 0.39$	$\setminus 0, \sim 0, [> 0, 3], 0.21$	$\setminus 0, / 0, [/ 0, 2], 0.21$
$[/ 0, 2], 0.93$	$/ 0, [> 0, 3], 0.29$	$\forall 0, / 0, [/ 0, 2], 0.24$	$\setminus 0, \sim 0, [> 0, 4], 0.44$	$/ 0, \setminus 0, [> 0, 2], 0.42$
$> 0, [\sim 0, 1], 0.26$	$/ 0, [\setminus 0, 2], 0.36$	$\forall 0, / 0, [/ 0, 3], 0.28$	$\setminus 0, \sim 0, [\forall 0, 3], 0.6$	$/ 0, \setminus 0, [> 0, 3], 0.37$
$> 0, [\setminus 0, 1], 0.26$	$/ 0, [\setminus 0, 3], 0.65$	$\sim 0, \forall 0, [> 0, 2], 0.22$	$\setminus 0, \sim 0, [\setminus 0, 2], 0.21$	$/ 0, \setminus 0, [/ 0, 2], 0.28$
$> 0, [/ 0, 1], 0.33$	$/ 0, [\setminus 0, 4], 0.2$	$\sim 0, \forall 0, [/ 0, 1], 0.24$	$\setminus 0, \sim 0, [\setminus 0, 3], 0.68$	$/ 0, \setminus 0, [/ 0, 3], 0.57$
$\forall 0, [\sim 0, 4], 0.26$	$> 0, \setminus 0, [/ 0, 2], 0.48$	$\sim 0, \forall 0, [/ 0, 2], 0.2$	$\setminus 0, \sim 0, [\setminus 0, 4], 0.64$	$/ 0, / 0, [\setminus 0, 3], 0.76$
$\setminus 0, [> 0, 3], 0.81$	$> 0, / 0, [/ 0, 1], 0.24$	$\sim 0, \forall 0, [/ 0, 3], 0.28$	$\setminus 0, \sim 0, [/ 0, 4], 0.28$	
$\setminus 0, [\sim 0, 3], 0.6$	$\forall 0, \sim 0, [> 0, 2], 0.27$	$\sim 0, \setminus 0, [> 0, 3], 0.22$	$\setminus 0, / 0, [> 0, 2], 0.37$	

Cycle de décision 12 : >O

[^0, 1], 0.26	[/0, 2], 0.69	>0, \0, [/0, 1], 0.39	\0, ^0, [\0, 1], 0.57	\0, /0, [/0, 2], 0.55
\0, 1], 0.72	\0, [>0, 1], 0.39	v0, ^0, [/0, 1], 0.57	\0, ^0, [\0, 2], 0.76	/0, \0, [\0, 1], 0.35
[/0, 1], 0.69	\0, [^0, 2], 0.57	v0, ^0, [/0, 2], 0.76	\0, ^0, [/0, 1], 0.72	/0, \0, [/0, 1], 0.7
\0, [/0, 2], 0.76	\0, [/0, 1], 0.76	v0, /0, [/0, 2], 0.72	\0, /0, [>0, 1], 0.69	/0, \0, [/0, 2], 0.86
/0, [/0, 1], 0.39	/0, [>0, 1], 0.72	^0, v0, [/0, 2], 0.69	\0, /0, [\0, 1], 0.39	
\0, 2], 0.72	/0, [\0, 2], 0.69	\0, ^0, [v0, 3], 0.39	\0, /0, [/0, 1], 0.88	

3.3 Expérimentation avec une proie et un bloc mur

Suivi de la proie :

Cycle de décision 1 : >O

>0, [>0, 6], 0.76	>0, /0, [>0, 6], 0.76	>0, [>0, 5], 0.26	v0, ^0, [>0, 6], 0.44	\0, /0, [/0, 7], 0.38
>0, [^0, 6], 0.58	v0, ^0, [/0, 8], 0.59	>0, [>0, 7], 0.76	v0, ^0, [>0, 7], 0.61	\0, /0, [/0, 8], 0.22
>0, [/0, 5], 0.73	^0, v0, [>0, 8], 0.55	>0, [/0, 7], 0.43	v0, ^0, [>0, 8], 0.47	/0, >0, [>0, 4], 0.36
>0, [/0, 6], 0.72	^0, v0, [>0, 9], 0.52	\0, [>0, 9], 0.58	v0, ^0, [>0, 9], 0.27	/0, v0, [/0, 9], 0.49
^0, [\0, 9], 0.58	^0, v0, [/0, 8], 0.56	/0, [>0, 9], 0.73	v0, ^0, [/0, 7], 0.2	/0, \0, [/0, 7], 0.46
\0, [^0, 9], 0.73	\0, ^0, [>0, 7], 0.51	>0, >0, [>0, 6], 0.72	v0, ^0, [/0, 9], 0.59	/0, \0, [/0, 8], 0.3
\0, [/0, 9], 0.73	\0, /0, [>0, 7], 0.68	>0, >0, [^0, 4], 0.34	^0, >0, [>0, 9], 0.67	/0, \0, [/0, 9], 0.55
/0, [>0, 7], 0.73	\0, /0, [>0, 8], 0.67	>0, >0, [^0, 5], 0.52	^0, v0, [>0, 7], 0.48	/0, /0, [>0, 7], 0.23
/0, [>0, 8], 0.73	/0, >0, [>0, 5], 0.75	>0, >0, [/0, 6], 0.35	^0, \0, [>0, 9], 0.29	/0, /0, [>0, 8], 0.21
>0, >0, [>0, 5], 0.72	/0, >0, [>0, 6], 0.61	>0, ^0, [>0, 5], 0.54	\0, ^0, [>0, 6], 0.33	/0, /0, [>0, 9], 0.48
>0, >0, [^0, 6], 0.59	/0, \0, [>0, 8], 0.53	>0, ^0, [>0, 6], 0.5	\0, ^0, [>0, 8], 0.21	/0, /0, [v0, 8], 0.55
>0, >0, [/0, 5], 0.67	/0, /0, [\0, 8], 0.71	>0, /0, [>0, 4], 0.66	\0, ^0, [>0, 9], 0.63	/0, /0, [\0, 7], 0.52
>0, /0, [>0, 5], 0.63	/0, /0, [\0, 9], 0.56	>x, /0, [>0, 7], 0.58	\0, /0, [>0, 6], 0.76	

Cycle de décision 2 : >O

[>0, 6], 0.76	>0, [^0, 4], 0.34	^0, v0, [>0, 7], 0.68	/0, \0, [/0, 7], 0.61	\0, >F, [/0, 8], 0.59
[^0, 6], 0.58	>0, [^0, 5], 0.52	^0, v0, [>0, 8], 0.67	>0, /0, [>0, 4], 0.42	\0, ^0, [>0, 6], 0.67
[/0, 5], 0.73	>0, [/0, 6], 0.56	^0, v0, [>0, 9], 0.58	>0, /0, [/0, 4], 0.4	\0, /0, [>0, 6], 0.48
[/0, 6], 0.72	^0, [>0, 5], 0.54	^0, v0, [/0, 8], 0.58	>x, ^0, [>0, 5], 0.43	/0, >0, [>0, 3], 0.63
>0, [>0, 5], 0.72	^0, [>0, 6], 0.5	^0, \0, [>0, 6], 0.73	>x, /0, [>0, 5], 0.71	/0, >0, [>0, 4], 0.76
>0, [^0, 6], 0.76	/0, [>0, 4], 0.66	\0, ^0, [>0, 7], 0.75	v0, ^0, [>0, 8], 0.54	/0, >0, [/0, 3], 0.76
>0, [/0, 5], 0.68	/0, [>0, 7], 0.56	\0, /0, [>0, 7], 0.63	^0, v0, [>0, 5], 0.76	/0, \0, [>0, 5], 0.43
/0, [>0, 5], 0.68	>0, /0, [>0, 5], 0.4	\0, /0, [>0, 8], 0.63	^0, v0, [>0, 6], 0.43	/0, \0, [>0, 8], 0.58
/0, [>0, 6], 0.76	>x, ^0, [>0, 6], 0.71	\0, /0, [/0, 7], 0.63	^0, \0, [>0, 5], 0.58	/0, \0, [^0, 6], 0.73
[>0, 5], 0.26	v0, ^0, [>0, 6], 0.54	/0, >0, [>0, 5], 0.76	^0, \0, [>0, 7], 0.73	/0, \0, [/0, 5], 0.72
[>0, 7], 0.76	v0, ^0, [>0, 7], 0.54	/0, \0, [>0, 6], 0.58	\0, >F, [^0, 7], 0.72	/0, /0, [>0, 6], 0.61
[/0, 7], 0.43	v0, ^0, [/0, 7], 0.76	/0, \0, [>0, 7], 0.58	\0, >F, [^0, 8], 0.61	
>0, [>0, 6], 0.72	v0, ^0, [/0, 8], 0.76	/0, \0, [/0, 6], 0.61	\0, >F, [/0, 7], 0.47	

Cycle de décision 3 : ^O

[>0, 5], 0.54	v0, [/0, 8], 0.75	\0, [>0, 7], 0.73	v0, ^0, [>0, 6], 0.63	\0, \0, [>0, 8], 0.53
[>0, 6], 0.5	\0, [>0, 6], 0.81	>0, \0, [>0, 5], 0.47	v0, /0, [>0, 7], 0.76	/0, v0, [>0, 7], 0.78
v0, [>0, 7], 0.81	v0, [>0, 5], 0.76	>0, \0, [>0, 6], 0.59	\0, >0, [>0, 5], 0.64	/0, v0, [>0, 8], 0.72
v0, [>0, 8], 0.83	v0, [>0, 6], 0.47	>x, v0, [>0, 6], 0.67	\0, >0, [>0, 6], 0.61	/0, \0, [>0, 6], 0.81
v0, [>0, 9], 0.81	\0, [>0, 5], 0.58	>x, v0, [>0, 7], 0.48	\0, \0, [>0, 7], 0.5	

Cycle de décision 4 : >O

\0, [>0, 5], 0.52	>0, \0, [>0, 4], 0.52	>0, \0, [>0, 6], 0.67
\0, [>0, 6], 0.59	>0, \0, [>0, 5], 0.6	v0, >0, [>0, 3], 0.65

Cycle de décision 5 : \0

[>0, 5], 0.52	>0, [>0, 6], 0.72	\0, [>0, 6], 0.55	^0, v0, [>0, 5], 0.5	\0, /0, [>0, 6], 0.67
[>0, 6], 0.59	>0, [\0, 5], 0.5	>0, /0, [>0, 4], 0.75	\0, >0, [>0, 3], 0.77	\0, \0, [>0, 5], 0.59
>0, [>0, 4], 0.51	>0, [/0, 5], 0.69	>0, /0, [>0, 5], 0.59	\0, >0, [/0, 3], 0.56	/0, \0, [>0, 6], 0.44
>0, [>0, 5], 0.78	\0, [>0, 5], 0.53	v0, ^0, [>0, 6], 0.64	\0, /0, [>0, 5], 0.55	

Cycle de décision 6 : >O

Annexe A : résultats détaillés des expérimentations

>0, [4], 0.72	/0, [>0, 4], 0.75	>0, /0, [>0, 4], 0.83	v0, ~0, [>0, 6], 0.51	/0, v0, [>0, 5], 0.67
>0, [5], 0.78	/0, [>0, 6], 0.59	>0, /0, [>0, 5], 0.86	~0, v0, [>0, 5], 0.32	/0, \0, [>0, 5], 0.29
>0, [6], 0.72	>0, [>0, 4], 0.72	>0, /0, [/0, 4], 0.65	\0, ~0, [>0, 6], 0.66	/0, \0, [>0, 6], 0.73
\0, [5], 0.72	\0, [>0, 5], 0.66	>x, /0, [>0, 5], 0.78	\0, /0, [>0, 5], 0.66	
/0, [5], 0.69	>0, /0, [>0, 3], 0.47	v0, ~0, [>0, 5], 0.94	\0, /0, [>0, 6], 0.55	

Cycle de décision 7 : >O

>0, [4], 0.72	>0, [>0, 2], 0.49	>0, /0, [/0, 4], 0.45	\0, ~0, [>0, 4], 0.22	\0, /0, [/0, 5], 0.56
/0, [>0, 3], 0.47	>0, [/0, 2], 0.5	>x, ~0, [>0, 4], 0.66	\0, ~0, [>0, 5], 0.22	/0, \0, [>0, 4], 0.41
/0, [>0, 4], 0.83	>0, >0, [/0, 1], 0.44	>x, /0, [>0, 5], 0.31	\0, ~0, [>0, 6], 0.22	
/0, [>0, 5], 0.86	>0, /0, [>0, 3], 0.77	>x, /0, [/0, 4], 0.58	\0, /0, [>0, 4], 0.22	
/0, [/0, 4], 0.65	>0, /0, [>0, 4], 0.68	~0, >0, [>0, 3], 0.22	\0, /0, [>0, 5], 0.51	

Cycle de décision 8 : >O (l'agent voit à nouveau la proie)

\0, [v0, 1], 0.65	\0, /0, [\0, 1], 0.61	/0, [\0, 2], 0.86	v0, ~0, [\0, 0], 0.24	~0, \0, [v0, 1], 0.32
\0, [~0, 1], 0.65	\0, /0, [/0, 1], 0.57	>0, v0, [v0, 1], 0.8	v0, ~0, [\0, 2], 0.47	~0, \0, [\0, 2], 0.29
\0, [\0, 0], 0.86	/0, v0, [v0, 2], 0.81	>0, v0, [\0, 1], 0.73	v0, ~0, [/0, 1], 0.31	\0, \0, [>0, 1], 0.39
\0, [\0, 1], 0.8	/0, v0, [\0, 1], 0.81	>0, \0, [v0, 1], 0.91	v0, ~0, [/0, 2], 0.45	\0, \0, [\0, 1], 0.2
\0, [/0, 0], 0.73	/0, v0, [\0, 2], 0.81	>0, \0, [v0, 2], 0.33	v0, /0, [~0, 1], 0.42	\0, /0, [>0, 1], 0.23
\0, [/0, 1], 0.91	/0, v0, [/0, 1], 0.81	>0, \0, [\0, 1], 0.24	v0, /0, [~0, 2], 0.25	\0, /0, [v0, 2], 0.41
\0, \0, [~0, 1], 0.81	/0, \0, [/0, 1], 0.67	>0, \0, [\0, 2], 0.28	v0, /0, [/0, 1], 0.22	\0, /0, [\0, 0], 0.24
\0, \0, [~0, 2], 0.74	v0, [~0, 2], 0.31	>x, \0, [>0, 3], 0.2	v0, /0, [/0, 2], 0.25	\0, /0, [\0, 2], 0.25
\0, \0, [\0, 0], 0.76	v0, [~0, 3], 0.33	>x, \0, [v0, 2], 0.39	~0, v0, [v0, 1], 0.25	/0, v0, [v0, 1], 0.31
\0, \0, [/0, 0], 0.66	v0, [/0, 2], 0.38	>x, \0, [v0, 3], 0.48	~0, v0, [v0, 2], 0.24	/0, \0, [v0, 1], 0.81
\0, \0, [/0, 1], 0.85	v0, [/0, 3], 0.35	>x, \0, [\0, 2], 0.41	~0, v0, [v0, 3], 0.21	/0, \0, [\0, 0], 0.74
\0, \0, [/0, 2], 0.56	/0, [v0, 1], 0.65	>x, \0, [\0, 3], 0.27	~0, v0, [\0, 1], 0.24	/0, \0, [\0, 1], 0.76
\0, /0, [v0, 1], 0.66	/0, [v0, 2], 0.62	v0, >F, [v0, 2], 0.26	~0, v0, [\0, 2], 0.23	

Cycle de décision 9 : >O

v0, [~0, 2], 0.32	\0, >0, [~0, 1], 0.2	\0, \0, [~0, 1], 0.22	\0, /0, [\0, 2], 0.23
\0, [/0, 1], 0.3	\0, >0, [/0, 1], 0.27	\0, \0, [/0, 1], 0.21	

Cycle de décision 10 : vO

\0, [~0, 1], 0.84	\0, /0, [\0, 1], 0.77	>x, \0, [\0, 3], 0.66	\0, >F, [/0, 7], 0.41	\0, /0, [~0, 1], 0.23
\0, [~0, 2], 0.84	\0, /0, [/0, 0], 0.52	v0, >F, [v0, 2], 0.41	\0, >F, [/0, 8], 0.39	/0, v0, [v0, 2], 0.48
\0, [\0, 0], 0.84	\0, /0, [/0, 1], 0.83	v0, ~0, [\0, 0], 0.24	\0, ~0, [v0, 1], 0.21	/0, v0, [\0, 1], 0.21
\0, [/0, 0], 0.84	/0, \0, [\0, 0], 0.64	v0, ~0, [/0, 1], 0.57	\0, ~0, [\0, 0], 0.24	/0, v0, [\0, 2], 0.45
\0, [/0, 1], 0.84	/0, \0, [/0, 1], 0.65	v0, ~0, [/0, 2], 0.27	\0, ~0, [\0, 1], 0.21	/0, v0, [/0, 1], 0.45
/0, [v0, 1], 0.84	/0, [v0, 2], 0.84	v0, /0, [~0, 2], 0.22	\0, ~0, [\0, 2], 0.22	/0, \0, [v0, 1], 0.32
/0, [\0, 2], 0.57	/0, [\0, 1], 0.55	v0, /0, [/0, 2], 0.26	\0, ~0, [/0, 1], 0.22	/0, \0, [~0, 1], 0.35
\0, /0, [>0, 1], 0.64	>x, \0, [v0, 2], 0.54	~0, \0, [\0, 2], 0.21	\0, \0, [~0, 1], 0.22	/0, \0, [\0, 1], 0.64
/0, [\0, 5], 0.61	>x, [v0, 1], 0.52	\0, >F, [~0, 7], 0.27	\0, \0, [/0, 0], 0.22	/0, \0, [/0, 0], 0.52
\0, /0, [\0, 0], 0.71	>x, \0, [\0, 2], 0.84	\0, >F, [~0, 8], 0.2	\0, \0, [/0, 1], 0.22	

Cycle de décision 11 : >F (L'objet disparaît)

Suivi du mur :

Cycle de décision 1 : >O

>0, [>0, 3], 0.88	v0, /0, [/0, 7], 0.51	>F, [/0, 3], 0.65	>x, \0, [>0, 6], 0.8	v0, /0, [/0, 6], 0.32
>0, [>0, 4], 0.92	~0, v0, [>0, 4], 0.83	>F, [/0, 4], 0.34	>F, \0, [>0, 4], 0.23	~0, v0, [>0, 3], 0.32
>0, [/0, 3], 0.88	~0, v0, [>0, 5], 0.86	\0, [>0, 5], 0.32	>F, \0, [>0, 5], 0.61	~0, v0, [>0, 6], 0.3
/0, [>0, 5], 0.53	\0, /0, [>0, 5], 0.8	\0, [>0, 6], 0.34	>F, \0, [>0, 6], 0.33	~0, v0, [>0, 7], 0.33
/0, [>0, 6], 0.82	\0, /0, [>0, 6], 0.5	\0, [/0, 5], 0.28	>F, \0, [/0, 3], 0.59	~0, v0, [\0, 5], 0.3
/0, [\0, 5], 0.61	/0, >0, [v0, 5], 0.55	/0, [>0, 7], 0.36	>F, \0, [/0, 4], 0.41	~0, \0, [>0, 5], 0.3
/0, [\0, 6], 0.8	/0, >0, [v0, 6], 0.68	/0, [v0, 6], 0.35	>F, \0, [/0, 5], 0.41	~0, \0, [>0, 6], 0.31
>0, >0, [>0, 1], 0.7	/0, >0, [\0, 5], 0.59	>0, >0, [>0, 3], 0.25	v0, ~0, [>0, 4], 0.42	\0, >0, [>0, 3], 0.63
>0, >0, [>0, 2], 0.8	/0, >0, [\0, 6], 0.58	>0, >0, [/0, 1], 0.53	v0, ~0, [>0, 6], 0.61	\0, >0, [/0, 4], 0.3
>0, >0, [\0, 1], 0.61	/0, v0, [>0, 5], 0.61	>0, \0, [>0, 3], 0.82	v0, ~0, [>0, 7], 0.23	/0, >0, [>0, 5], 0.39
>0, >0, [/0, 2], 0.82	/0, \0, [>0, 4], 0.68	>0, \0, [>0, 4], 0.21	v0, ~0, [/0, 5], 0.35	/0, >0, [>0, 6], 0.26
>0, \0, [/0, 4], 0.61	/0, \0, [>0, 5], 0.74	>0, \0, [/0, 3], 0.35	v0, /0, [>0, 6], 0.34	/0, v0, [~0, 6], 0.26
>0, /0, [\0, 4], 0.8	/0, \0, [>0, 6], 0.71	>0, /0, [>0, 3], 0.61	v0, /0, [>0, 7], 0.54	/0, v0, [/0, 5], 0.25
>0, /0, [\0, 5], 0.8	>F, [>0, 3], 0.88	>0, /0, [>0, 4], 0.8	v0, /0, [~0, 6], 0.53	/0, \0, [/0, 4], 0.41
v0, ~0, [>0, 5], 0.63	>F, [>0, 4], 0.92	>0, /0, [v0, 5], 0.7	v0, /0, [~0, 7], 0.2	

Cycle de décision 2 : >O

>0, [3], 0.88	/0, [0, 5], 0.8	\0, /0, [/0, 4], 0.69	>0, >0, [\0, 2], 0.61	~0, v0, [0, 5], 0.65
>0, [4], 0.92	/0, [/0, 4], 0.83	/0, >0, [0, 3], 0.74	>0, >0, [/0, 0], 0.54	~0, v0, [/0, 3], 0.83
/0, [3], 0.88	>0, >0, [\0, 0], 0.86	/0, >0, [v0, 4], 0.71	>0, v0, [0, 3], 0.53	~0, v0, [/0, 4], 0.71
>0, [0, 1], 0.94	>0, >0, [\0, 1], 0.61	/0, >0, [\0, 4], 0.69	>0, v0, [~0, 3], 0.89	~0, \0, [0, 4], 0.57
>0, [0, 2], 0.88	>0, >0, [/0, 1], 0.53	/0, >0, [\0, 5], 0.88	>0, v0, [~0, 4], 0.9	~0, \0, [\0, 5], 0.53
>0, [\0, 1], 0.84	>0, /0, [\0, 2], 0.88	/0, v0, [0, 4], 0.92	>0, v0, [/0, 2], 0.85	\0, >0, [0, 3], 0.7
>0, [/0, 2], 0.84	v0, ~0, [0, 3], 0.94	/0, v0, [0, 5], 0.84	>0, v0, [/0, 3], 0.86	\0, >0, [0, 4], 0.73
\0, [/0, 4], 0.79	v0, ~0, [/0, 3], 0.66	/0, v0, [/0, 4], 0.81	>0, v0, [/0, 4], 0.92	\0, >0, [~0, 5], 0.53
/0, [\0, 4], 0.9	~0, v0, [0, 3], 0.65	/0, v0, [0, 5], 0.81	>0, \0, [0, 2], 0.54	\0, \0, [0, 3], 0.86
/0, [\0, 5], 0.84	~0, v0, [0, 4], 0.65	/0, \0, [0, 2], 0.81	>0, \0, [0, 3], 0.89	\0, >0, [/0, 5], 0.77
>0, [0, 3], 0.59	\0, >0, [~0, 3], 0.8	/0, \0, [0, 3], 0.81	>0, \0, [0, 2], 0.53	\0, ~0, [\0, 5], 0.68
>0, [0, 1], 0.59	\0, >0, [~0, 4], 0.53	/0, \0, [0, 4], 0.81	>0, \0, [0, 3], 0.84	\0, \0, [~0, 5], 0.62
\0, [0, 3], 0.82	\0, >0, [/0, 4], 0.77	/0, \0, [/0, 2], 0.81	>0, /0, [0, 2], 0.84	\0, \0, [/0, 5], 0.62
\0, [0, 4], 0.61	\0, ~0, [0, 4], 0.68	/0, \0, [0, 3], 0.81	>0, /0, [0, 3], 0.79	\0, /0, [0, 5], 0.55
\0, [/0, 3], 0.35	\0, ~0, [0, 5], 0.68	/0, \0, [/0, 4], 0.88	>0, /0, [\0, 3], 0.88	/0, v0, [0, 3], 0.69
/0, [0, 3], 0.61	\0, ~0, [\0, 4], 0.68	>0, [\0, 2], 0.92	>x, \0, [0, 4], 0.9	/0, v0, [~0, 4], 0.84
/0, [0, 4], 0.8	\0, ~0, [/0, 4], 0.62	>0, [0, 3], 0.65	>F, \0, [0, 4], 0.57	/0, v0, [/0, 3], 0.75
/0, [v0, 5], 0.7	\0, /0, [0, 2], 0.58	\0, [0, 5], 0.7	v0, ~0, [0, 5], 0.94	
\0, [~0, 3], 0.36	\0, /0, [0, 3], 0.61	>0, >0, [0, 1], 0.8	v0, ~0, [0, 2], 0.58	
\0, [~0, 4], 0.8	\0, /0, [0, 4], 0.55	>0, >0, [v0, 1], 0.61	v0, /0, [0, 4], 0.66	
\0, [/0, 5], 0.8	\0, /0, [/0, 3], 0.69	>0, >0, [~0, 1], 0.59	v0, /0, [0, 4], 0.68	

Cycle de décision 3 : ~O

v0, [0, 3], 0.87	>0, \0, [0, 4], 0.5	>x, \0, [0, 3], 0.97	\0, >0, [\0, 4], 0.82	/0, v0, [\0, 5], 0.88
v0, [0, 4], 0.81	>0, \0, [0, 5], 0.61	>x, \0, [0, 4], 0.45	\0, >0, [\0, 5], 0.75	/0, v0, [/0, 4], 0.49
v0, [0, 5], 0.81	>0, \0, [0, 6], 0.53	>x, \0, [0, 5], 0.93	\0, v0, [0, 5], 0.85	/0, \0, [~0, 4], 0.21
v0, [/0, 3], 0.83	>0, \0, [v0, 5], 0.74	>x, \0, [\0, 4], 0.21	\0, \0, [0, 3], 0.62	/0, \0, [v0, 5], 0.95
v0, [/0, 4], 0.71	>0, \0, [v0, 6], 0.53	>x, \0, [\0, 5], 0.83	\0, \0, [0, 4], 0.24	/0, \0, [v0, 6], 0.33
\0, [0, 4], 0.81	>0, \0, [\0, 4], 0.55	~0, v0, [0, 6], 0.97	\0, \0, [0, 5], 0.95	/0, \0, [\0, 4], 0.47
\0, [\0, 5], 0.77	>0, \0, [\0, 5], 0.87	~0, v0, [\0, 5], 0.23	\0, \0, [\0, 5], 0.74	/0, \0, [\0, 5], 0.31
\0, [\0, 4], 0.81	>0, \0, [\0, 6], 0.87	~0, v0, [\0, 6], 0.33	\0, \0, [0, 4], 0.51	/0, \0, [\0, 6], 0.26
>0, v0, [0, 3], 0.77	>x, v0, [0, 3], 0.5	\0, >0, [0, 4], 0.9	/0, v0, [0, 3], 0.63	
>0, v0, [0, 4], 0.57	>x, v0, [0, 4], 0.84	\0, >0, [v0, 4], 0.35	/0, v0, [0, 4], 0.67	
>0, v0, [0, 5], 0.7	>x, v0, [0, 5], 0.74	\0, >0, [v0, 5], 0.68	/0, v0, [0, 5], 0.36	
>0, v0, [/0, 2], 0.57	>x, v0, [/0, 4], 0.94	\0, >0, [\0, 3], 0.81	/0, v0, [\0, 4], 0.88	

Cycle de décision 4 : >O

v0, [0, 3], 0.77	v0, [\0, 5], 0.26	>x, v0, [0, 3], 0.21	v0, \0, [0, 4], 0.5	\0, \0, [0, 4], 0.46
v0, [0, 4], 0.57	v0, [/0, 4], 0.42	>x, v0, [0, 4], 0.39	v0, \0, [0, 5], 0.87	\0, \0, [0, 5], 0.81
v0, [0, 5], 0.7	>0, v0, [0, 3], 0.53	>x, v0, [0, 5], 0.25	v0, \0, [0, 6], 0.46	/0, v0, [0, 4], 0.86
v0, [/0, 2], 0.57	>0, v0, [0, 4], 0.58	>x, v0, [/0, 4], 0.21	v0, /0, [0, 5], 0.22	/0, v0, [0, 5], 0.35
\0, [0, 4], 0.5	>0, v0, [0, 5], 0.54	>x, \0, [0, 3], 0.59	v0, /0, [v0, 6], 0.82	/0, v0, [0, 6], 0.7
\0, [0, 5], 0.61	>0, v0, [/0, 2], 0.37	>x, \0, [0, 4], 0.81	v0, /0, [0, 5], 0.37	/0, v0, [v0, 5], 0.81
\0, [0, 6], 0.53	>0, \0, [0, 4], 0.37	>x, \0, [0, 5], 0.58	v0, /0, [\0, 6], 0.58	/0, v0, [\0, 4], 0.32
\0, [v0, 5], 0.74	>0, \0, [0, 5], 0.77	>x, \0, [\0, 4], 0.22	\0, >0, [v0, 4], 0.65	/0, v0, [\0, 5], 0.57
\0, [v0, 6], 0.53	>0, \0, [0, 6], 0.54	>x, \0, [\0, 5], 0.27	\0, >0, [v0, 5], 0.48	/0, v0, [/0, 4], 0.42
\0, [\0, 4], 0.58	>0, \0, [v0, 5], 0.27	v0, >0, [0, 2], 0.22	\0, >0, [\0, 3], 0.53	/0, v0, [\0, 6], 0.48
\0, [\0, 5], 0.87	>0, \0, [v0, 6], 0.31	v0, >0, [0, 3], 0.3	\0, >0, [\0, 4], 0.4	/0, \0, [v0, 6], 0.48
\0, [\0, 6], 0.87	>0, \0, [\0, 5], 0.48	v0, >0, [\0, 2], 0.2	\0, v0, [0, 5], 0.39	
v0, [0, 6], 0.32	>0, \0, [\0, 6], 0.43	v0, >0, [/0, 3], 0.31	\0, \0, [0, 3], 0.43	

Cycle de décision 5 : \O

[0, 4], 0.5	>0, [\0, 6], 0.22	>x, \0, [0, 4], 0.7	~0, v0, [v0, 4], 0.36	\0, /0, [v0, 5], 0.62
[0, 5], 0.61	v0, [0, 4], 0.77	>F, \0, [0, 4], 0.73	~0, v0, [v0, 5], 0.27	\0, /0, [\0, 5], 0.32
[0, 6], 0.53	v0, [0, 6], 0.61	>F, \0, [0, 5], 0.59	~0, v0, [v0, 6], 0.27	\0, /0, [\0, 6], 0.29
[v0, 5], 0.74	v0, [/0, 5], 0.44	>F, \0, [/0, 3], 0.81	~0, v0, [\0, 4], 0.46	/0, v0, [0, 3], 0.26
[v0, 6], 0.53	\0, [0, 6], 0.91	>F, \0, [/0, 4], 0.58	~0, v0, [\0, 5], 0.4	/0, v0, [0, 4], 0.41
[\0, 4], 0.58	\0, [\0, 4], 0.91	v0, >0, [0, 3], 0.7	~0, v0, [\0, 6], 0.29	/0, v0, [0, 5], 0.33
[\0, 5], 0.87	\0, [\0, 5], 0.29	v0, ~0, [0, 5], 0.73	\0, >0, [0, 2], 0.25	/0, v0, [/0, 4], 0.39
[\0, 6], 0.87	\0, [/0, 3], 0.87	v0, ~0, [0, 6], 0.59	\0, >0, [0, 3], 0.48	/0, \0, [0, 4], 0.26
>0, [v0, 4], 0.91	\0, [/0, 4], 0.81	v0, ~0, [0, 7], 0.81	\0, >0, [0, 4], 0.23	/0, \0, [0, 5], 0.46
>0, [v0, 5], 0.81	>0, >0, [v0, 6], 0.7	v0, ~0, [\0, 5], 0.58	\0, >0, [\0, 2], 0.49	/0, \0, [0, 6], 0.42
>0, [\0, 3], 0.53	>0, >0, [\0, 5], 0.73	v0, ~0, [\0, 6], 0.7	\0, >0, [\0, 3], 0.26	/0, \0, [v0, 5], 0.31
>0, [\0, 4], 0.81	>0, v0, [0, 3], 0.59	v0, >0, [0, 5], 0.73	\0, >0, [/0, 3], 0.58	/0, \0, [v0, 6], 0.67
v0, [0, 5], 0.73	>0, v0, [0, 4], 0.81	v0, /0, [0, 4], 0.59	\0, \0, [0, 4], 0.62	/0, \0, [\0, 4], 0.31
\0, [0, 3], 0.91	>0, v0, [0, 5], 0.58	v0, /0, [0, 5], 0.81	\0, \0, [0, 5], 0.23	/0, \0, [\0, 5], 0.37
\0, [0, 4], 0.77	>0, \0, [0, 3], 0.7	v0, /0, [0, 6], 0.58	\0, \0, [0, 6], 0.37	/0, \0, [\0, 6], 0.74
\0, [0, 5], 0.91	>0, \0, [0, 4], 0.73	v0, /0, [/0, 4], 0.2	\0, \0, [0, 3], 0.49	
>0, [0, 6], 0.7	>0, \0, [0, 5], 0.59	v0, /0, [/0, 5], 0.4	\0, \0, [0, 4], 0.45	
>0, [v0, 6], 0.62	>0, \0, [\0, 5], 0.81	~0, v0, [0, 4], 0.29	\0, \0, [0, 6], 0.4	
>0, [\0, 5], 0.37	>0, \0, [/0, 4], 0.58	~0, v0, [0, 5], 0.35	\0, /0, [0, 5], 0.42	

Annexe A : résultats détaillés des expérimentations

Cycle de décision 6 : >O

v0, [4], 0.91	\0, [>0, 6], 0.7	>x, v0, [>0, 4], 0.58	v0, \0, [-0, 4], 0.58	\0, \0, [>0, 3], 0.28
[v0, 5], 0.85	\0, [\0, 3], 0.28	>x, v0, [/0, 4], 0.62	v0, \0, [-0, 6], 0.34	\0, \0, [>0, 4], 0.22
[\0, 3], 0.53	\0, [\0, 4], 0.52	>x, \0, [>0, 3], 0.57	v0, /0, [>0, 4], 0.5	\0, \0, [>0, 5], 0.35
[\0, 4], 0.81	\0, [/0, 3], 0.5	>x, \0, [>0, 4], 0.76	v0, /0, [>0, 5], 0.62	\0, \0, [/0, 3], 0.21
[>0, 6], 0.7	>0, v0, [>0, 2], 0.54	>x, \0, [>0, 5], 0.72	v0, /0, [>0, 6], 0.27	\0, \0, [/0, 4], 0.27
[v0, 6], 0.85	>0, v0, [>0, 3], 0.44	>x, \0, [\0, 4], 0.2	v0, /0, [/0, 4], 0.24	\0, /0, [v0, 5], 0.52
[\0, 5], 0.85	>0, v0, [>0, 4], 0.31	>F, \0, [>0, 4], 0.2	~0, v0, [v0, 4], 0.63	\0, /0, [v0, 6], 0.21
[\0, 6], 0.85	>0, v0, [>0, 5], 0.29	>F, \0, [>0, 5], 0.33	~0, v0, [v0, 5], 0.24	\0, /0, [\0, 5], 0.56
>0, [v0, 6], 0.7	>0, v0, [/0, 2], 0.36	>F, \0, [/0, 3], 0.27	~0, v0, [v0, 6], 0.44	\0, /0, [\0, 6], 0.62
>0, [\0, 5], 0.73	>0, v0, [/0, 4], 0.22	>F, \0, [/0, 4], 0.24	~0, v0, [\0, 4], 0.26	/0, v0, [>0, 3], 0.38
v0, [>0, 3], 0.85	>0, \0, [>0, 3], 0.47	v0, >0, [>0, 3], 0.65	~0, v0, [\0, 5], 0.27	/0, v0, [>0, 4], 0.42
v0, [>0, 4], 0.85	>0, \0, [>0, 4], 0.42	v0, ~0, [>0, 5], 0.57	~0, v0, [\0, 6], 0.69	/0, v0, [>0, 5], 0.56
v0, [>0, 5], 0.85	>0, \0, [>0, 5], 0.34	v0, ~0, [>0, 6], 0.25	\0, >0, [>0, 3], 0.36	/0, v0, [\0, 5], 0.3
\0, [>0, 3], 0.7	>0, \0, [>0, 6], 0.55	v0, ~0, [>0, 7], 0.4	\0, >0, [>0, 4], 0.39	/0, v0, [/0, 4], 0.23
\0, [>0, 4], 0.85	>0, \0, [v0, 5], 0.21	v0, ~0, [\0, 5], 0.68	\0, >0, [v0, 3], 0.24	/0, \0, [v0, 5], 0.45
\0, [>0, 5], 0.85	>0, \0, [0, 4], 0.46	v0, ~0, [\0, 6], 0.4	\0, >0, [\0, 2], 0.28	/0, \0, [v0, 6], 0.42
\0, [\0, 5], 0.85	>0, \0, [\0, 5], 0.4	v0, \0, [\0, 5], 0.67	\0, >0, [\0, 3], 0.22	/0, \0, [v0, 5], 0.21
\0, [/0, 4], 0.85	>0, \0, [/0, 4], 0.48	v0, \0, [>0, 5], 0.53	\0, >0, [\0, 4], 0.24	/0, \0, [\0, 6], 0.22
>0, [\0, 6], 0.85	>x, v0, [>0, 3], 0.49	v0, \0, [>0, 6], 0.78	\0, v0, [>0, 5], 0.2	

Cycle de décision 7 : >O

[v0, 6], 0.7	>0, v0, [>0, 4], 0.52	>x, \0, [\0, 4], 0.35	~0, v0, [\0, 6], 0.67	/0, >F, [v0, 2], 0.28
[\0, 5], 0.73	>0, v0, [>0, 5], 0.56	>x, \0, [\0, 5], 0.52	\0, >0, [>0, 3], 0.97	/0, >F, [v0, 3], 0.28
[\0, 6], 0.85	>0, v0, [/0, 2], 0.62	>F, \0, [/0, 3], 0.56	\0, >0, [>0, 4], 0.97	/0, >F, [\0, 1], 0.3
v0, [>0, 2], 0.56	>0, \0, [>0, 4], 0.38	v0, >0, [>0, 2], 0.62	\0, >0, [v0, 4], 0.85	/0, >F, [\0, 2], 0.27
v0, [>0, 3], 0.52	>0, \0, [>0, 5], 0.35	v0, >0, [>0, 3], 0.38	\0, >0, [\0, 3], 0.38	/0, >F, [\0, 3], 0.41
v0, [>0, 4], 0.35	>0, \0, [>0, 6], 0.52	v0, ~0, [v0, 6], 0.35	\0, >0, [\0, 4], 0.71	/0, v0, [>0, 3], 0.69
v0, [>0, 5], 0.38	>0, \0, [v0, 5], 0.56	v0, ~0, [\0, 5], 0.52	\0, >0, [\0, 5], 0.65	/0, v0, [>0, 4], 0.73
v0, [/0, 2], 0.36	>0, \0, [v0, 6], 0.62	v0, ~0, [\0, 6], 0.56	\0, v0, [>0, 5], 0.85	/0, v0, [>0, 5], 0.49
v0, [/0, 4], 0.22	>0, \0, [\0, 4], 0.38	v0, ~0, [\0, 7], 0.62	\0, \0, [>0, 3], 0.22	/0, v0, [>0, 6], 0.5
\0, [>0, 3], 0.62	>0, \0, [\0, 5], 0.35	v0, \0, [>0, 4], 0.38	\0, \0, [>0, 4], 0.25	/0, v0, [>0, 4], 0.28
\0, [>0, 4], 0.56	>0, \0, [\0, 6], 0.52	v0, \0, [>0, 5], 0.4	\0, \0, [>0, 5], 0.48	/0, v0, [\0, 5], 0.23
\0, [>0, 5], 0.52	>x, v0, [>0, 3], 0.56	v0, \0, [>0, 6], 0.66	\0, \0, [/0, 3], 0.5	/0, v0, [/0, 4], 0.46
\0, [>0, 6], 0.55	>x, v0, [>0, 4], 0.62	v0, \0, [-0, 4], 0.82	\0, \0, [/0, 4], 0.5	/0, \0, [v0, 5], 0.75
\0, [v0, 5], 0.52	>x, v0, [>0, 5], 0.38	v0, \0, [-0, 6], 0.81	\0, /0, [v0, 6], 0.36	/0, \0, [v0, 6], 0.54
\0, [\0, 4], 0.46	>x, v0, [/0, 3], 0.35	v0, \0, [/0, 5], 0.82	\0, /0, [\0, 5], 0.85	/0, \0, [\0, 5], 0.24
\0, [\0, 5], 0.4	>x, v0, [/0, 4], 0.52	v0, /0, [>0, 5], 0.23	\0, /0, [\0, 6], 0.71	/0, \0, [\0, 6], 0.39
\0, [/0, 4], 0.48	>x, \0, [>0, 3], 0.56	v0, /0, [>0, 6], 0.82	/0, >F, [>0, 2], 0.59	
>0, v0, [>0, 2], 0.33	>x, \0, [>0, 4], 0.62	~0, v0, [v0, 6], 0.43	/0, >F, [>0, 3], 0.43	
>0, v0, [>0, 3], 0.35	>x, \0, [>0, 5], 0.38	~0, v0, [\0, 5], 0.24	/0, >F, [v0, 1], 0.56	

Cycle de décision 8 :

v0, [>0, 2], 0.33	v0, [\0, 5], 0.57	>0, \0, [\0, 5], 0.66	v0, >0, [/0, 3], 0.56	\0, \0, [>0, 3], 0.85
v0, [>0, 3], 0.58	v0, [/0, 4], 0.63	>0, \0, [\0, 6], 0.22	v0, \0, [>0, 4], 0.67	\0, \0, [>0, 4], 0.71
v0, [>0, 4], 0.8	\0, [>0, 3], 0.83	>x, \0, [>0, 3], 0.37	v0, \0, [>0, 5], 0.57	\0, \0, [>0, 5], 0.59
v0, [>0, 5], 0.89	>0, v0, [>0, 2], 0.38	>x, v0, [>0, 4], 0.66	v0, \0, [>0, 6], 0.74	/0, v0, [>0, 4], 0.43
v0, [/0, 2], 0.62	>0, v0, [>0, 3], 0.85	>x, v0, [>0, 5], 0.23	v0, /0, [>0, 5], 0.27	/0, v0, [>0, 5], 0.6
\0, [>0, 4], 0.58	>0, v0, [>0, 4], 0.84	>x, v0, [/0, 4], 0.3	v0, /0, [v0, 6], 0.91	/0, v0, [>0, 6], 0.59
\0, [>0, 5], 0.58	>0, v0, [>0, 5], 0.48	>x, v0, [>0, 3], 0.42	v0, /0, [\0, 5], 0.76	/0, v0, [v0, 5], 0.46
\0, [>0, 6], 0.58	>0, v0, [/0, 2], 0.52	>x, \0, [>0, 4], 0.85	v0, /0, [\0, 6], 0.66	/0, v0, [\0, 4], 0.3
\0, [v0, 5], 0.58	>0, \0, [>0, 4], 0.89	>x, \0, [>0, 5], 0.53	\0, >0, [v0, 3], 0.82	/0, v0, [\0, 5], 0.58
\0, [v0, 6], 0.62	>0, \0, [>0, 5], 0.8	>x, \0, [\0, 4], 0.2	\0, >0, [v0, 4], 0.81	/0, \0, [v0, 6], 0.72
\0, [\0, 4], 0.84	>0, \0, [>0, 6], 0.23	>x, \0, [\0, 5], 0.45	\0, >0, [v0, 5], 0.82	
\0, [\0, 5], 0.58	>0, \0, [v0, 5], 0.34	v0, >0, [>0, 2], 0.72	\0, >0, [\0, 3], 0.48	
\0, [\0, 6], 0.58	>0, \0, [v0, 6], 0.68	v0, >0, [>0, 3], 0.21	\0, >0, [\0, 4], 0.5	
v0, [>0, 6], 0.58	>0, \0, [\0, 4], 0.73	v0, >0, [\0, 2], 0.53	\0, v0, [>0, 5], 0.5	

Cycle de décision 9 : >O

v0, [>0, 2], 0.38	\0, [\0, 5], 0.76	>0, \0, [>0, 4], 0.46	>x, v0, [/0, 4], 0.69	v0, \0, [>0, 5], 0.26
v0, [>0, 3], 0.85	\0, [\0, 6], 0.76	>0, \0, [>0, 5], 0.77	>x, \0, [>0, 3], 0.87	v0, \0, [>0, 6], 0.25
v0, [>0, 4], 0.84	v0, [>0, 6], 0.76	>0, \0, [>0, 6], 0.78	>x, \0, [>0, 4], 0.31	v0, /0, [>0, 5], 0.2
v0, [>0, 5], 0.76	v0, [\0, 5], 0.87	>0, \0, [v0, 5], 0.74	>x, \0, [>0, 5], 0.32	v0, /0, [v0, 6], 0.41
v0, [/0, 2], 0.52	v0, [/0, 4], 0.91	>0, \0, [v0, 6], 0.52	>x, \0, [\0, 4], 0.43	v0, /0, [\0, 5], 0.33
\0, [>0, 4], 0.89	\0, [>0, 3], 0.54	>0, \0, [\0, 4], 0.3	>x, \0, [\0, 5], 0.21	v0, /0, [\0, 6], 0.47
\0, [>0, 5], 0.8	>0, v0, [>0, 2], 0.3	>0, \0, [\0, 5], 0.33	v0, >0, [>0, 2], 0.58	\0, >0, [v0, 3], 0.27
\0, [>0, 6], 0.23	>0, v0, [>0, 3], 0.68	>0, \0, [\0, 6], 0.91	v0, >0, [>0, 3], 0.78	\0, >0, [v0, 4], 0.41
\0, [v0, 5], 0.76	>0, v0, [>0, 4], 0.71	>x, v0, [>0, 3], 0.87	v0, >0, [\0, 2], 0.21	\0, >0, [v0, 5], 0.36
\0, [v0, 6], 0.76	>0, v0, [>0, 5], 0.67	>x, v0, [>0, 4], 0.34	v0, >0, [/0, 3], 0.37	\0, >0, [\0, 3], 0.23
\0, [\0, 4], 0.73	>0, v0, [/0, 2], 0.8	>x, v0, [>0, 5], 0.31	v0, \0, [>0, 4], 0.21	\0, >0, [\0, 4], 0.3

\0, v0, [>0, 5], 0.2	\0, \0, [>0, 5], 0.56	/0, v0, [>0, 6], 0.36	/0, v0, [\0, 5], 0.51
\0, \0, [>0, 3], 0.25	/0, v0, [>0, 4], 0.78	/0, v0, [v0, 5], 0.39	/0, \0, [v0, 6], 0.54
\0, \0, [>0, 4], 0.22	/0, v0, [>0, 5], 0.37	/0, v0, [\0, 4], 0.64	

Cycle de décision 10 : vO

[>0, 2], 0.38	\0, [>0, 4], 0.58	>0, \0, [>0, 4], 0.48	~0, v0, [>0, 5], 0.7	\0, /0, [>0, 6], 0.46
[>0, 3], 0.85	\0, [>0, 5], 0.58	>0, \0, [/0, 3], 0.73	~0, v0, [\0, 5], 0.53	/0, v0, [>0, 5], 0.46
[>0, 4], 0.84	\0, [>0, 6], 0.58	>0, \0, [/0, 4], 0.66	\0, >0, [>0, 3], 0.56	/0, v0, [/0, 5], 0.77
[>0, 5], 0.8	/0, [>0, 5], 0.76	>F, \0, [>0, 4], 0.42	\0, >0, [>0, 4], 0.79	/0, \0, [>0, 3], 0.78
[/0, 2], 0.52	/0, [v0, 6], 0.41	>F, \0, [>0, 5], 0.72	\0, >0, [/0, 3], 0.33	/0, \0, [>0, 4], 0.74
[>0, 6], 0.76	/0, [\0, 5], 0.48	>F, \0, [/0, 4], 0.86	\0, >0, [/0, 4], 0.39	/0, \0, [>0, 5], 0.52
[\0, 5], 0.87	/0, [\0, 6], 0.47	v0, ~0, [>0, 5], 0.68	\0, ~0, [\0, 5], 0.65	/0, \0, [>0, 6], 0.91
[/0, 4], 0.91	>0, [>0, 4], 0.73	v0, ~0, [>0, 6], 0.52	\0, ~0, [\0, 6], 0.8	/0, \0, [/0, 4], 0.87
>0, [>0, 2], 0.58	\0, [/0, 4], 0.58	v0, ~0, [>0, 7], 0.61	\0, \0, [~0, 6], 0.89	
>0, [>0, 3], 0.8	\0, [/0, 5], 0.8	v0, /0, [>0, 5], 0.6	\0, \0, [/0, 5], 0.82	
>0, [\0, 2], 0.21	/0, [>0, 6], 0.76	v0, /0, [>0, 6], 0.54	\0, /0, [>0, 4], 0.6	
>0, [/0, 3], 0.73	>0, \0, [>0, 3], 0.48	~0, v0, [>0, 4], 0.84	\0, /0, [>0, 5], 0.59	

Cycle de décision 11 : >F

\0, [>0, 4], 0.69	>0, [\0, 1], 0.59	>0, >0, [/0, 1], 0.25	>0, \0, [>0, 3], 0.74	~0, v0, [\0, 4], 0.59
\0, [>0, 5], 0.72	>0, [/0, 1], 0.71	>0, v0, [>0, 3], 0.9	>0, \0, [>0, 4], 0.48	~0, v0, [\0, 5], 0.51
\0, [/0, 4], 0.87	v0, [>0, 4], 0.75	>0, v0, [>0, 4], 0.62	>0, \0, [/0, 3], 0.81	/0, v0, [>0, 4], 0.94
\0, [/0, 3], 0.87	v0, [>0, 5], 0.41	>0, v0, [/0, 2], 0.31	>0, \0, [/0, 4], 0.89	/0, v0, [>0, 5], 0.54
>0, [>0, 1], 0.87	v0, [/0, 4], 0.8	>0, v0, [/0, 3], 0.6	>x, \0, [>0, 4], 0.47	/0, v0, [/0, 4], 0.62
>0, [>0, 2], 0.24	/0, [>0, 5], 0.71	>0, v0, [/0, 4], 0.78	v0, /0, [>0, 5], 0.76	/0, \0, [\0, 5], 0.69

Annexe B : Simplification algorithmique de la proximité globale

Je propose une simplification algorithmique concernant le calcul de la proximité globale, dans le cadre de l'expérimentation décrite au chapitre 10 :

nous avons défini la *proximité globale* d'un objet qui afforde une interaction i comme :

$$\nu_{i,t}(M) = \int_{\tau \in T} c(i, E_{\tau,t}) \times f(d(\tau)) \quad (1)$$

Avec $E_{\tau,t}$ l'image du contexte E_t par la transformation τ , $c(i, E)$ la certitude qu'une interaction i puisse être énoncée avec succès dans le contexte E , et f une fonction qui caractérise l'influence d'une instance d'objet en fonction de sa distance. Dans notre implémentation, l'ensemble T des transformations est discret, et la fonction de certitude est définie à l'aide d'un neurone artificiel. Nous simplifierons également la fonction de nos neurones formels en remplaçons la fonction d'activation sigmoïde par une fonction linéaire. La fonction de certitude peut s'exprimer par :

$$c(i, E_t) = \sum_{k \in [1;n+1]} \epsilon_{t,k} \times w_{i,k} \quad (2)$$

Avec $n = \text{Card}(A)$. On note $e_{t,\tau(k)}$ l'entrée du contexte interactionnel E_t associée à l'acte localisé à la position p' obtenue en appliquant la transformation τ à p_k . La proximité globale peut donc s'écrire sous la forme :

$$\nu_{i,t}(M) = \sum_{\tau \in T} \sum_{k \in [1;n+1]} \epsilon_{t,\tau(k)} \times w_{i,k} \times f(d(\tau)) \quad (3)$$

Nous définissons le *poids image* de $w_{i,k}$ par la transformation $\tau^{-1}(k)$. Cette image vaut $w_{i,k'}$ si il existe une entrée ϵ'_k et une transformation τ telle que τ lie $\epsilon_{k'}$ à ϵ_k , et 0 sinon. La proximité globale peut alors s'écrire :

$$\nu_{i,t}(M) = \sum_{\tau \in T} \sum_{k \in [1;n+1]} \epsilon_{t,k} \times w_{i,\tau^{-1}(k)} \times f(d(\tau^{-1})) \quad (4)$$

Nous pouvons considérer que les poids des signatures évoluent lentement. Dès lors, nous pouvons réduire la fréquence de mise à jour du terme :

$$\sum_{\tau \in T} w_{i,\tau^{-1}(k)} \times f(d(\tau^{-1})) \quad (5)$$

Je propose de définir la *signature floue* B_i d'une interaction i comme un vecteur $[b_1, \dots, b_m]$, avec $m = \text{Card}(P')$ le nombre de positions définies par la mémoire spatiale, avec $b_{i,k} = \sum_{\tau \in T} w_{i,\tau^{-1}(k)} \times f(d(\tau^{-1}))$. Notons que la signature floue ne prend ainsi en compte que les poids des interactions qui peuvent être localisées dans la mémoire spatiale. La proximité globale peut alors s'écrire :

$$\nu_{i,t}(M) = M \times B_i^T \quad (6)$$

Les signatures floues sont mises à jour tous les 50 pas de simulation. Cette fréquence de mise à jour est jugée suffisante : en effet, les signatures floues ne sont pas utilisées par le mécanisme d'apprentissage des signatures, mais seulement par le mécanisme d'exploitation de la mémoire spatiale. Or, ce mécanisme n'est utilisé de façon régulière par l'agent que lorsque les signatures d'interaction deviennent pertinentes et stables.

Annexe C : Algorithmes

1 algorithme général du modèle PSRI

1. Select an act $i_t \in A$ and try to enact it.
 2. Trying to enact i_t will result in an enacted set E_t and τ_t the geometric transformation performed by the agent.
 3. Apply the geometric transformation τ_t to M.
 4. Increase ages o_a of all acts in M. Remove old acts (o_a greater than a threshold).
 5. Add enacted acts from E_t to M with recentness $o_a=1$.
 6. Goto 1.
-

2 Mise à jour de la mémoire spatio-séquentielle

Soit e_t la dernière interaction effectuée par l'agent,

Soit φ une interaction composite de longueur n , soient les fonctions

First : $\varphi \mapsto \varphi[1]$

Next : $\varphi \mapsto \varphi[2..n]$

algorithme de la mise à jour :

1. pour chaque interaction $\varphi \in \mathcal{T}$
retirer φ de \mathcal{T}
si $\text{First}(\varphi) = e_t$ alors ajouter $\text{Next}(\varphi)$ à \mathcal{T}
 2. pour chaque interaction $\varphi \in \mathcal{F}$
retirer φ de \mathcal{F}
si $\text{First}(\varphi) = e_t$ alors ajouter $\text{Next}(\varphi)$ à \mathcal{F}
 3. pour chaque interaction $\varphi \in \Phi$
si $c(\varphi, E_t) > \tau \wedge \varphi \notin \mathcal{T}$ alors ajouter φ à \mathcal{T}
si $c(\varphi, E_t) < -\tau \wedge \varphi \notin \mathcal{F}$ alors ajouter φ à \mathcal{F}
-

3 reconstruction du path d'une interaction composite dans le cas d'un système implémentant l'hypothèse vestibulaire

Soit une interaction composite φ de longueur n , avec un path de longueur k . On note $\varphi[i]$ le i^{me} élément de φ

Soient \mathcal{T} et \mathcal{F} les listes d'interactions composites stockées en mémoire.

Initialisation :

1. construire une séquence d'interactions primitives A de longueur n initialement vide.
 2. Pour chaque $i \in [k + 1, n]$, $A[i] \leftarrow \varphi[i]$
 3. Pour chaque $i \in [1; k]$ faire
 4. Si il existe $j \in \mathcal{T}$ de longueur k telle que j est inscrite dans φ alors :
 5. soit m la longueur du path de j
 6. pour chaque $k \in [m + 1 ; k]$, faire :
 7. si $S[k]$ est vide \vee si $s_{j[k]} > s_{A[k]}$ alors $A[k] \leftarrow j[k]$
 8. fin pour
 9. fin si
 10. fin pour
 11. Si il n'existe pas de i tel que $A[i]$ soit vide, alors retourner A
 12. sinon retourner *null*
-

Résumé

Cette thèse s'inscrit dans le cadre du projet IDEAL (Implementing DEvelopmentAl Learning) financé par L'agence Nationale de la Recherche (ANR). La capacité à percevoir, mémoriser et interpréter l'environnement qui nous entoure est une capacité vitale que l'on retrouve chez de nombreux êtres vivants. Cette capacité leur permet de générer des comportements adaptés à leur contexte, ou d'échapper à un prédateur sorti de leur champ de vision. L'objectif de cette thèse consiste à doter un agent artificiel de cette capacité. Nous proposons un modèle théorique permettant à un agent artificiel de générer des connaissances exploitables des éléments constituant son environnement et une structure reflétant l'espace. Ce modèle est basé sur la théorie de la contingence sensorimotrice, et implémente une forme de motivation intrinsèque. En effet, ce modèle débute avec un ensemble de structure indivisibles, appelées interactions, caractérisant les échanges entre l'agent et son environnement. L'apprentissage des connaissances est développemental et émerge de l'interaction entre l'agent et son environnement, sans qu'aucune intervention extérieure (récompense), ne soit nécessaire. Notre modèle propose des mécanismes permettant d'organiser et exploiter ces connaissances émergentes dans le but de générer des comportements. Nous proposons des implémentations de ce modèle pour démontrer l'émergence d'une connaissance à partir de l'interaction entre l'agent et son environnement, et les comportements qui émergent de cette connaissance.

Abstract

This thesis is a part of the IDEAL project (Implementing DEvelopmentAl Learning) funded by th Agence Nationale de la Recherche (ANR). The ability of perceiving, memorizing and interpreting the surrounding environment is a vital ability found in numerous living beings. This ability allows them to generate context adapted behaviors, or escaping from a predator that escape from their sensory system. The objective of this thesis consists in implementing such a capacity in artificial agents. We propose a theoretical model that allows an artificial agent to generate a usable knowledge of elements that compose its environment and a structure able to characterize the structure of surrounding space. This model is based on the sensorimotor contingency theory, and implements a form of intrinsic motivation. Indeed, this model begin with a set of indivisible structures, called interactions, that characterize the interaction possibilities between the agent and its environment . The learning is developmental and emerges from the interaction that occurs between the agent and the environment, without the need of any external intervention (like a reward). Our model propose a set of mechanisms that allow to organize and exploit emerging knowledge in order to generate behaviors. We propose implementations of our model to demonstrate the emerging knowledge based on agent-environment interaction, and behaviors that can emerge from this knowledge.

