

Partitionnement de graphe

optimisation et applications

XIX^{èmes} rencontres de la Société Française de Classification

Charles-Edmond Bichot

charles-edmond.bichot@ec-lyon.fr

<http://perso.ec-lyon.fr/charles-edmond.bichot/>

Maître de conférences à l'École Centrale de Lyon et au Laboratoire d'informatique en image et systèmes d'information (LIRIS)

Plan de la présentation

- 1 Le problème du partitionnement de graphe
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image
- 7 Conclusion

- 1 Le problème du partitionnement de graphe
 - Périmètre du partitionnement de graphe
 - Quand utiliser le partitionnement de graphe ?
 - Définition de la partition d'un ensemble
 - Domaines d'application du partitionnement de graphe
 - Deux catégories de problèmes de partitionnement
 - Partitionnement contraint
 - Partitionnement non contraint
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image
- 7 Conclusion

Périmètre du partitionnement de graphe

Quand utiliser le partitionnement de graphe ?

Soit S un ensemble à partitionner.

Connaît-on une distance entre chaque élément de S ?

- Oui \implies utiliser des méthodes de classification basées sur la distance
- Non \implies utiliser le partitionnement de graphe

Périmètre du partitionnement de graphe

Définition de la partition d'un ensemble

Définition (Partition d'un ensemble)

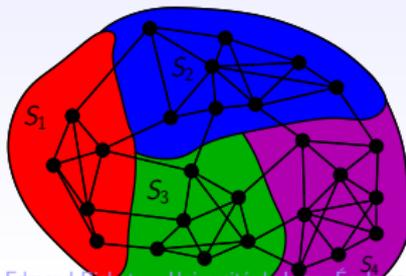
Soit S un ensemble quelconque. Un ensemble $P = \{S_1, \dots, S_k\}$ de sous-ensembles de S est appelé une **partition** de S si :

- 1 Aucun élément de P n'est vide
- 2 L'union des éléments de P est égale à S
- 3 Les éléments de P sont deux à deux disjoints

Les S_i sont appelés les **parties** de la partition P .

Partitionnement de graphe

Par partitionnement de graphe, on entend la partition de l'ensemble des sommets d'un graphe non orienté $G = (S, A)$.



Domaines d'application du partitionnement de graphe

Recherche et ingénierie.

De nombreuses applications

- répartition de charge dans les machines parallèles
- résolution de systèmes linéaires
- conception de circuits intégrés électroniques
- segmentation d'image
- exploration de données
- partitionnement de réseaux téléphoniques
- détection de communautés
- découpage géographique des régions
- trafic aérien
- *etc*

Domaines d'application du partitionnement de graphe

Le partitionnement de graphe possède de nombreuses applications !



Existe-t-il une formulation unique de ce problème ?

Domaines d'application du partitionnement de graphe

Le partitionnement de graphe possède de nombreuses applications !



Existe-t-il une formulation unique de ce problème ?

Non... mais nous allons proposer un regroupement en deux catégories de problèmes

Deux catégories de problèmes de partitionnement

Deux problèmes :

La littérature sur le partitionnement de graphe fait apparaître **deux catégories** de problèmes de partitionnement :

- le partitionnement **contraint**
- le partitionnement **non contraint**

Deux catégories de problèmes de partitionnement

Deux problèmes :

La littérature sur le partitionnement de graphe fait apparaître deux catégories de problèmes de partitionnement :

- le partitionnement contraint
- le partitionnement non contraint

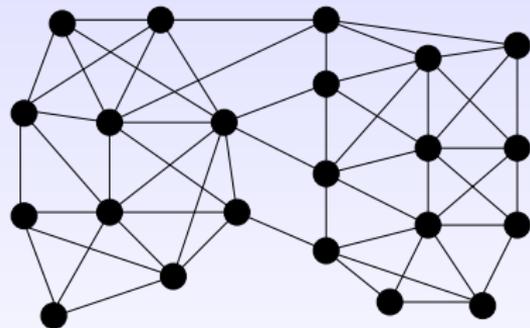
Difficulté :

Problèmes de la classe **NP-difficile**

Partitionnement contraint

Partitionnement contraint

Trouver une partition $P_k = \{S_1, \dots, S_k\}$ d'un graphe G pondéré sur ces arêtes tel que :



Partitionnement contraint

Partitionnement contraint

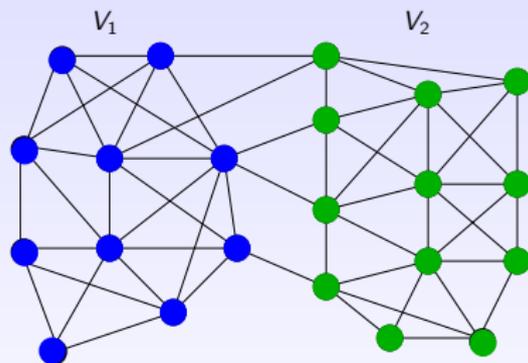
Trouver une partition $P_k = \{S_1, \dots, S_k\}$ d'un graphe G pondéré sur ces arêtes tel que :

- les parties sont de tailles **similaires** ($balance \leq 1,05$)

Balance de partitionnement

$$poids_{moy} = \left\lceil \frac{poids(S)}{k} \right\rceil$$

$$balance(P_k) = \frac{\max_i poids(S_i)}{poids_{moy}}$$



Le poids maximal d'une partie sera au plus égal à $balance(P_k)$ fois le poids moyen d'une partie

Partitionnement contraint

Partitionnement contraint

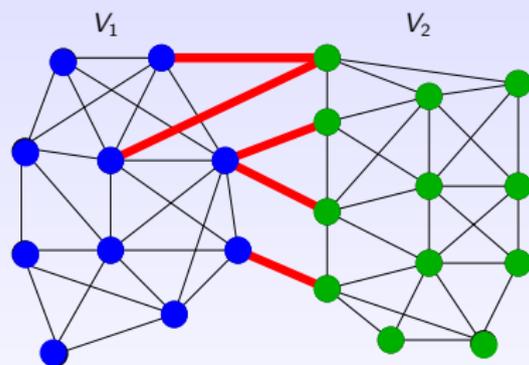
Trouver une partition $P_k = \{S_1, \dots, S_k\}$ d'un graphe G pondéré sur ces arêtes tel que :

- les parties sont de tailles similaires ($balance \leq 1,05$)
- le coût de coupe est **minimisé**

Coût de coupe d'une partition (cut)

$$cut(S_1, S_2) = \sum_{s_1 \in S_1, s_2 \in S_2} poids(s_1, s_2)$$

$$cut(P_k) = \sum_{i < j} cut(S_i, S_j)$$



Minimise le poids total des arêtes coupées entre les différentes parties

Partitionnement non contraint

Description du problème

Partitionnement non contraint

- Trouver une partition qui **minimise** une fonction objectif tenant compte des différences de tailles entre les parties
- La contrainte de balance de partitionnement est **facultative**

Partitionnement non contraint

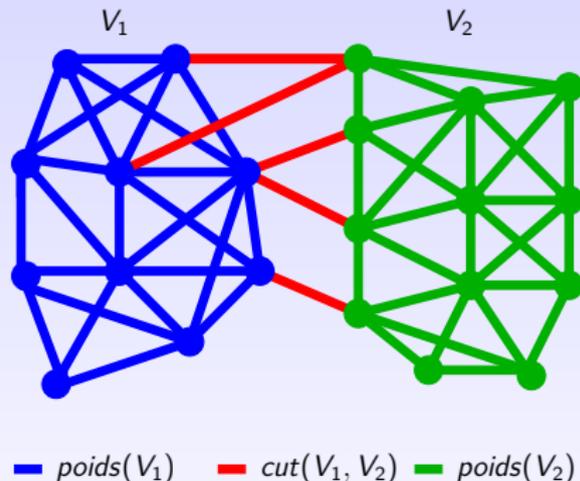
Fonctions objectifs pour le partitionnement non contraint

Ratio de coupe (*Ratio Cut*)

Soit une partition $P_k = \{S_1, \dots, S_k\}$ du graphe $G = (S, A)$

$$Ratio(P_k) = \sum_{i=1}^k \frac{cut(S_i, S - S_i)}{poids(S_i)}$$

C'est, pour chaque partie, le poids total de ses arêtes coupées sur son poids



Partitionnement non contraint

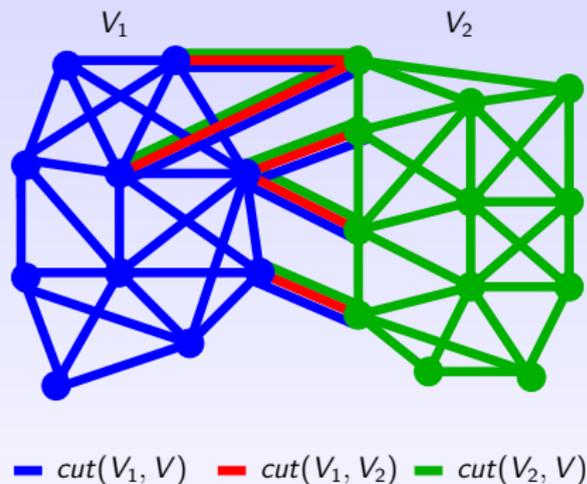
Fonctions objectifs pour le partitionnement non contraint

Coupe normalisée (*Normalized Cut*)

Soit une partition $P_k = \{S_1, \dots, S_k\}$ du graphe $G = (S, A)$

$$NCut(P_k) = \sum_{i=1}^k \frac{cut(S_i, S - S_i)}{cut(S_i, S)}$$

C'est, pour chaque partie, le poids total de ses arêtes coupées sur la somme de son poids et du poids total de ses arêtes coupées



- 1 Le problème du partitionnement de graphe
- 2 **Méthodes d'optimisation du partitionnement de graphes**
 - Listes de méthodes d'optimisation pour le partitionnement de graphe
 - Outils de partitionnement
- 3 Application au calcul numérique
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image
- 7 Conclusion

Listes de méthodes d'optimisation pour le partitionnement de graphe

Les méthodes classiques de partitionnement de graphe :

- Méthodes inertielles
- Expansion de région
- Méthode spectrale
- Méthode multi-niveaux
- Méthodes d'affinage d'une partition initiale

Listes de méthodes d'optimisation pour le partitionnement de graphe

Les méthodes classiques de partitionnement de graphe :

- Méthodes inertielles
- Expansion de région
- Méthode spectrale
- Méthode multi-niveaux
- Méthodes d'affinage d'une partition initiale

Autres méthodes utilisables :

- Analogie avec la mécanique des fluides : diffusion et percolation
- Métaheuristiques
- Méthodes hybrides
- Fusion-Fission

Listes de méthodes d'optimisation pour le partitionnement de graphe

Les méthodes classiques de partitionnement de graphe :

- Méthodes inertielles
- Expansion de région
- Méthode spectrale
- Méthode multi-niveaux
- Méthodes d'affinage d'une partition initiale

Autres méthodes utilisables :

- Analogie avec la mécanique des fluides : diffusion et percolation
- Métaheuristiques
- Méthodes hybrides
- Fusion-Fission

Outils de partitionnement

Pour le partitionnement contraint

- CHACO (*Sandia National Laboratories, USA*)
- METIS (*University of Minnesota, USA*)
- SCOTCH (INRIA/LaBRI, Bordeaux, France)
- JOSTLE (*University of Greenwich, UK*)
- PARTY (Université de Paderborn, Allemagne)

Pour le partitionnement non contraint

- GRACLUS (*University of Texas, USA*)

Outils de partitionnement parallèle

- ParMETIS (*University of Minnesota, USA*)
- PT-SCOTCH (INRIA/LaBRI, Bordeaux, France)
- JOSTLE parallel (*University of Greenwich, UK*)

- 1 Le problème du partitionnement de graphe
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique**
 - Placement statique
 - Résolution de systèmes linéaires creux
 - Méthodes de résolution
 - La méthode multi-niveaux
 - Algorithmes d'affinage
 - Défis
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image
- 7 Conclusion

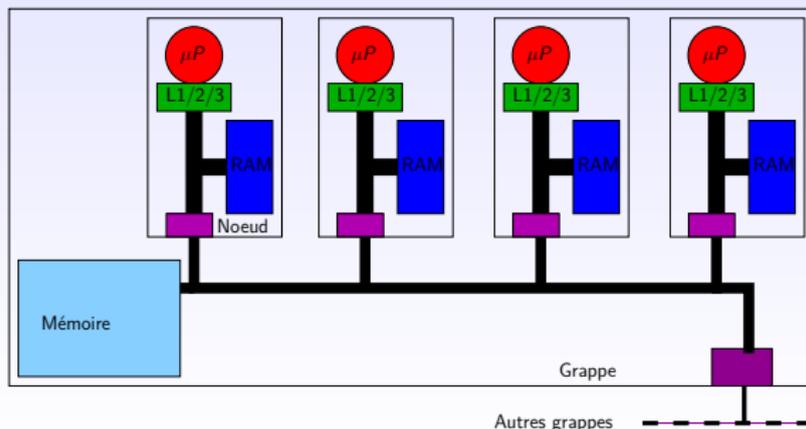
Placement statique

Contexte

« Machines » parallèles de grandes tailles

- Processeurs multi-coeurs
- Plusieurs processeurs par machine
- Différents niveaux de mémoire (cache et vive)
- Grappes de serveurs

⇒ on parle d'architecture NUMA (*Non Uniform Memory Architecture*)



Placement statique

Problème du placement statique

Placer n processus s'exécutant en parallèle sur p processeurs

Placement statique car calculé à l'avance

≠ ordonnancement car pas de dépendance temporelle entre processus

≠ placement dynamique où les processus peuvent être déplacés en cours d'exécution

Objectifs

Nombreux critères... en résumé :

- Minimiser le coût total de communication
- Équilibrer la charge de calcul entre processeurs

Placement statique

Problème du placement statique

Placer n processus s'exécutant en parallèle sur p processeurs

Placement statique car calculé à l'avance

≠ ordonnancement car pas de dépendance temporelle entre processus

≠ placement dynamique où les processus peuvent être déplacés en cours d'exécution

Objectifs

Nombreux critères... en résumé :

- Minimiser le coût total de communication
- Équilibrer la charge de calcul entre processeurs

Problème de partitionnement sous contrainte

- Minimiser la fonction de coupe *cut*
- Balance des parties stricte

Résolution de systèmes linéaires creux

Problématiques

- Factorisation/décomposition LU
- Parallélisation du calcul

Principe de la factorisation LU

Soit A une matrice inversible. Il existe une matrice de permutation P , une matrice triangulaire inférieur L et une matrice triangulaire supérieur U tel que $A = PLU$

Permet la résolution $Ax = b$ avec $LUx = b$ puis $Ly = b$ avec $Ux = y$

Résolution de systèmes linéaires creux

Problème de la factorisation LU

Minimiser le remplissage dans les facteurs L et U en permutant la matrice A .

Pourquoi ?

$$A_1 = \begin{pmatrix} x & x & x & x & x \\ x & x & & & \\ x & & x & & \\ x & & & x & \\ x & & & & x \end{pmatrix} \quad A_2 = \begin{pmatrix} x & & & & x \\ & x & & & x \\ & & x & & x \\ & & & x & x \\ x & x & x & x & x \end{pmatrix}$$

$A_1 \rightarrow$ facteurs L et U denses

$A_2 \rightarrow$ pas de remplissage dans K et U

Résolution de systèmes linéaires creux

Comment ?

Trouver un séparateur S_e (ensemble de sommets) qui partitionne le graphe de la matrice (les sommets sont les lignes et les colonnes) en deux parties disjointes, S_1 et S_2 .

La matrice permutée PAP^T est de la forme :

$$\begin{array}{ccc} S_1 & S_2 & S_e \\ \begin{pmatrix} A_{11} & & A_{13} \\ & A_{22} & A_{23} \\ A_{13}^T & A_{23}^T & A_{33} \end{pmatrix} \end{array}$$

Cette stratégie peut être utilisée récursivement

Problème de partitionnement contraint

- Minimiser la fonction de coupe *cut*
- Balance des parties

Méthodes de résolution

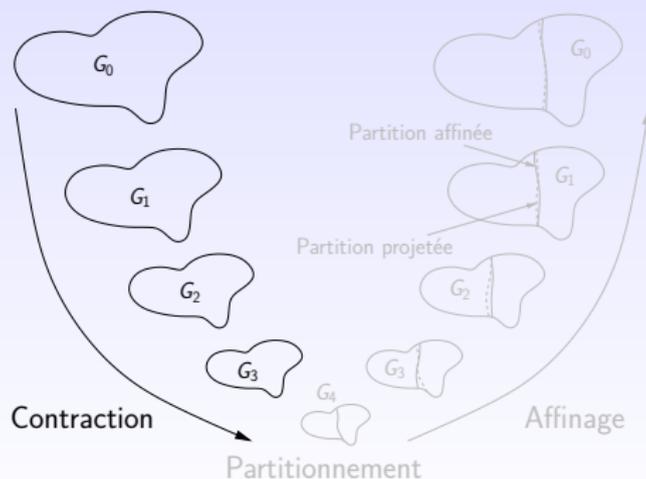
Méthodes de résolution

- Expansion de région (*Region Growing*)
 - Choix d'un sommet initial
 - Qualité du résultat très variable
 - Graphes de quelques centaines de sommets
- Affinage de la partition
 - Heuristique locale
 - Prévues pour le partitionnement contraint
- Stratégie multi-niveaux

La méthode multi-niveaux

Méthode en 3 temps

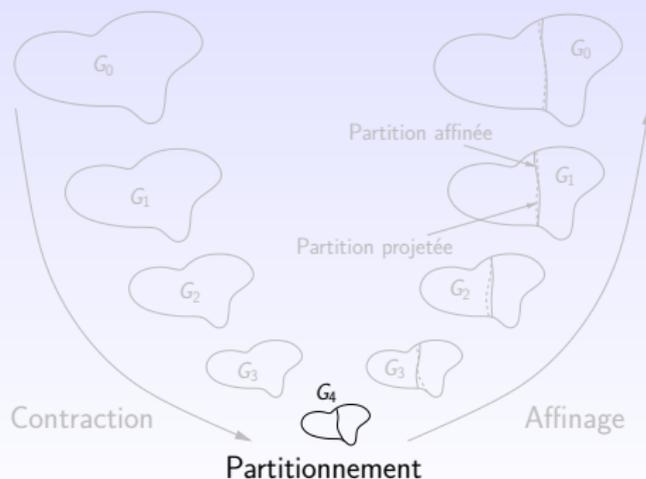
- 1 réduction de la taille du graphe par agrégation des sommets



La méthode multi-niveaux

Méthode en 3 temps

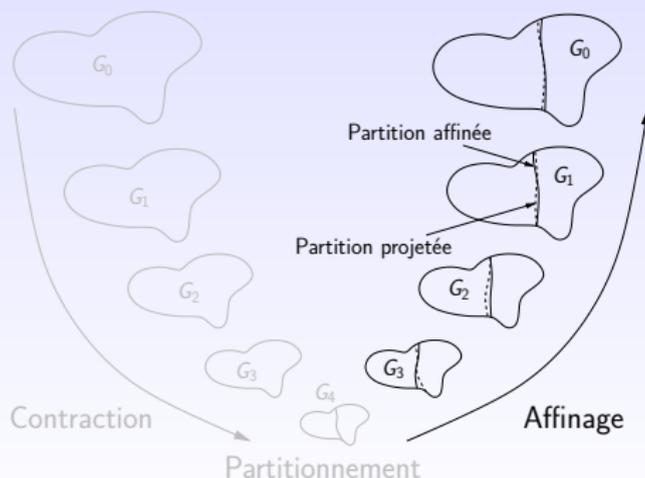
- 1 réduction de la taille du graphe par agrégation des sommets
- 2 partition du graphe réduit



La méthode multi-niveaux

Méthode en 3 temps

- 1 réduction de la taille du graphe par agrégation des sommets
- 2 partition du graphe réduit
- 3 retour progressif à la taille originelle avec améliorations locales de la partition



Algorithmes d'affinage

Principe de l'algorithme de Kernighan-Lin

But :

Affiner une partition, c.-à-d. minimiser localement la fonction objectif ; ici, le coût de coupe.

Idée :

Algorithme itératif par voisinage qui cherche à minimiser le coût de coupe en échangeant des ensembles de sommets entre 2 parties.

Article fondateur [1] :



B. W. Kernighan and S. Lin.

An efficient heuristic procedure for partitioning graphs.

Bell System Technical Journal, 49(2) :291–307, 1970.

Algorithmes d'affinage

La notion de gain introduite par Kernighan-Lin

Notion de gain :

Gain de coût de coupe résultant de l'échange du sommet $s \in S_1$ avec la partie S_2 .

$$gain(s) = \sum_{s' \notin S_1} poids(s, s') - \sum_{s' \in S_1} poids(s, s')$$

Si $gain(s) > 0$ alors le coût de coupe de la partition diminue.

Algorithmes d'affinage

Algorithme de Kernighan-Lin

Procédure $KL(G = (S, A), P_2)$

Initialisation : calculer le gain de chaque sommet

tant que il existe des sommets non marqués **faire**

 Sélectionner le couple de sommets dont l'échange constitue le plus grand gain

 Échanger et marquer ces deux sommets

 Mettre à jour les gains des voisins de 2 sommets

fin tant que

 Échanger réellement dans P_2 les 2 ensembles de sommets dont la somme des gains est maximale.

fin Procédure

Problème :

- N'est adapté qu'à la bissection de graphe parfaitement équilibrée !
- Complexité : $O(n_A^2 \log(n_A))$

Algorithmes d'affinage

Algorithme de Fiduccia-Mattheyses

Algorithme :

Procédure FM($G = (S, A)$)

Initialisation : calculer le gain de chaque sommet

tant que il existe des sommets non marqués dans la partie la plus grosse avec un gain ≥ 0 **faire**

 Sélectionner le sommet de plus grand gain

 Déplacer et marquer ce sommet

 Mettre à jour le gain des voisins de ce sommet

fin tant que

fin Procédure

Complexité : $O(n_A)$

Présenté dans [1] :



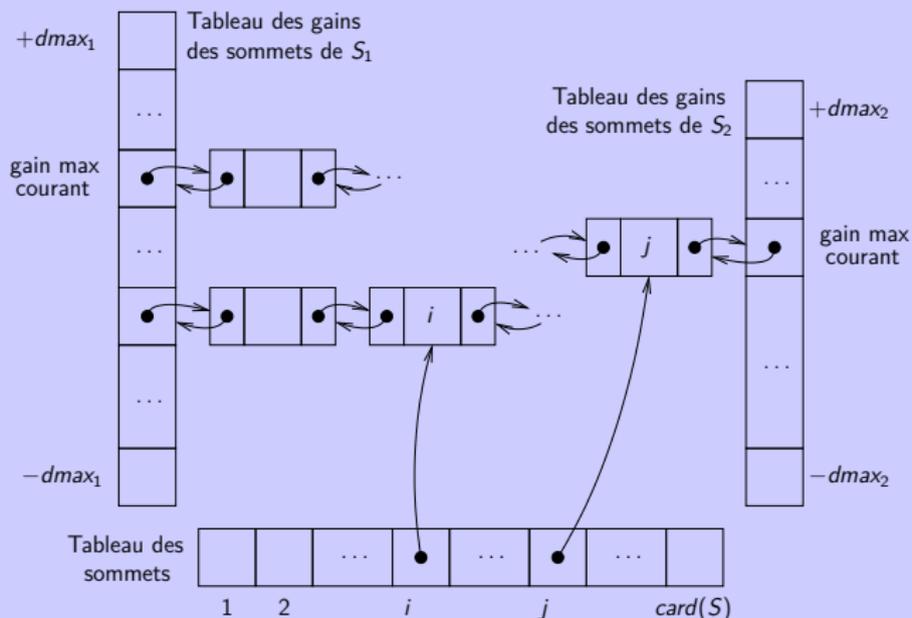
C. M. Fiduccia and R. M. Mattheyses.

A linear-time heuristic for improving network partitions.

In *Proceedings of 19th ACM/IEEE Design Automation Conference*, pages 175–181, 1982.

Algorithmes d'affinage

Structure de contrôle des gains de Fiduccia-Mattheyses



- Accès à un sommet en temps constant
- Déplacement des gains en temps constant

Défis

Défis

- Graphes de très grandes tailles (plusieurs milliards de sommets)
- Conjuguer performance et rapidité
- Robustesse des algorithmes (qualité des solutions)
- Paralléliser efficacement les algorithmes (CPU en bonne voie, GPU ?)
- Affinage avec d'autres fonctions de coûts et contraintes ?
- k -partitionnement direct plutôt que récursif ?

Travaux personnellement menés...

- Affinage
 - k -partitionnement direct → structure de donnée complexe, efficacité
 - intervalles de gains
 - tenir compte du poids des sommets (trier les listes de gains)
- Fusion-Fission

- 1 Le problème du partitionnement de graphe
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique
- 4 Application à la classification de documents**
 - Définition du problème
 - Graphe biparti
 - Fusion-Fission
 - Principe
 - Paradigme
 - Algorithme
 - Résultats
 - Quelques problèmes. . .
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image
- 7 Conclusion

Définition du problème

Problème de la classification de documents

Le problème de la classification de documents consiste à classer une collection de documents en différentes catégories partageant des sujets communs

Définition du problème

Problème de la classification de documents

Le problème de la classification de documents consiste à classer une collection de documents en différentes catégories partageant des sujets communs

Approche classique

- La classification de document est basée sur la distribution des mots (tokens) dans les documents
- Entre chaque couple de documents on calcule une distance (similarité entre les tokens utilisés)
- Les documents proches au sens de la distance sont classés ensemble

Définition du problème

Problème de la classification de documents

Le problème de la classification de documents consiste à classer une collection de documents en différentes catégories partageant des sujets communs

Approche classique

- La classification de document est basée sur la distribution des mots (tokens) dans les documents
- Entre chaque couple de documents on calcule une distance (similarité entre les tokens utilisés)
- Les documents proches au sens de la distance sont classés ensemble

Partitionnement d'un graphe biparti

- Classer documents et tokens en même temps
- Ne pas calculer de distance entre les documents
- Principe : le documents qui partagent les mêmes tokens sont classés ensemble

Graphe biparti

Graphe biparti

Classer en même temps documents et tokens

Processus

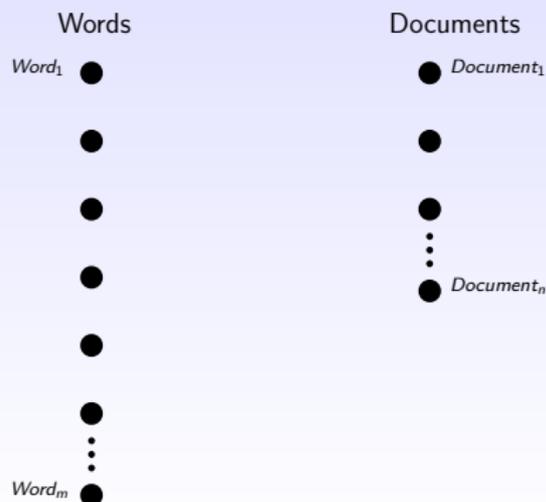
Graphe biparti

Graphe biparti

Classer en même temps documents et tokens

Processus

- 1 Extraire les tokens des documents
(Porter stemming algorithm)



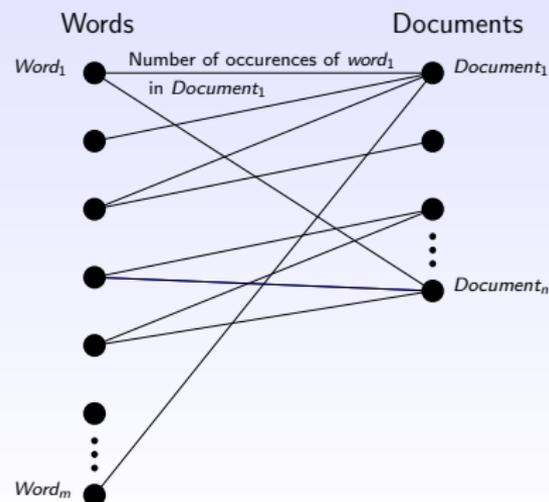
Graphe biparti

Graphe biparti

Classer en même temps documents et tokens

Processus

- 1 Extraire les tokens des documents (Porter stemming algorithm)
- 2 Créer un graphe biparti G tel que
 - tokens et documents sont les sommets du graphe
 - pondérer les arêtes par le nombre d'occurrence de chaque tokens dans le document



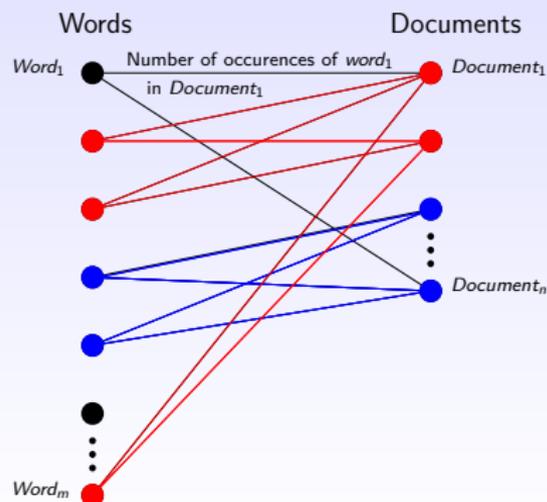
Graphe biparti

Graphe biparti

Classer en même temps documents et tokens

Processus

- 1 Extraire les tokens des documents (Porter stemming algorithm)
- 2 Créer un graphe biparti G tel que
 - tokens et documents sont les sommets du graphe
 - pondérer les arêtes par le nombre d'occurrence de chaque tokens dans le document
- 3 partitionner le graphe G



Graphe biparti

Les problème de la classification de documents devient un problème de partitionnement de graphe où

- documents et tokens sont classés ensemble
- la taille des classes n'est pas connue
- le nombre de classes doit être connu *a priori*

Problème → il faut définir une fonction objectif!

Graphe biparti

Les problème de la classification de documents devient un problème de partitionnement de graphe où

- documents et tokens sont classés ensemble
- la taille des classes n'est pas connue
- le nombre de classes doit être connu *a priori*

Problème → il faut définir une fonction objectif!

la coupe normalisée (NCut) a été utilisée dans



Inderjit S. Dhillon

Co-clustering documents and words using bipartite spectral graph partitioning
In *Proceedings of SIGKDD*, pages 269–274, 2001.

Fusion-Fission

Principe

Principe de la méthode de Fusion-Fission

la méthode de Fusion-Fission est basée sur une analogie avec les processus nucléaire de fusion et de fission

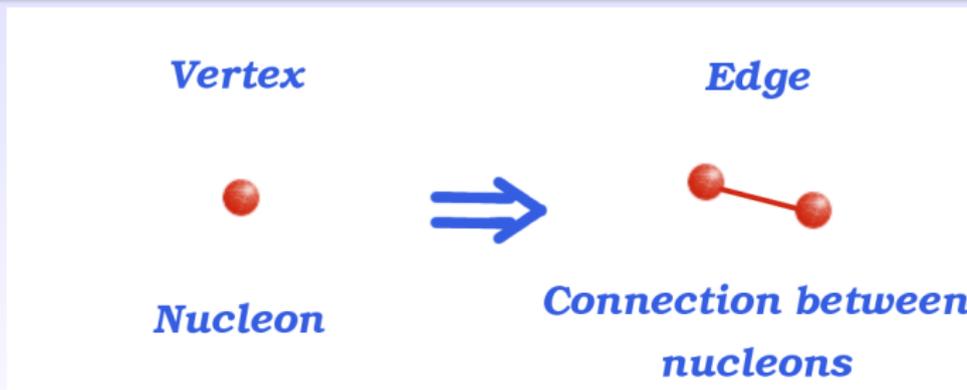
Fusion-Fission

Principe

Principe de la méthode de Fusion-Fission

la méthode de Fusion-Fission est basée sur une analogie avec les processus nucléaire de fusion et de fission

- nucléons = sommets



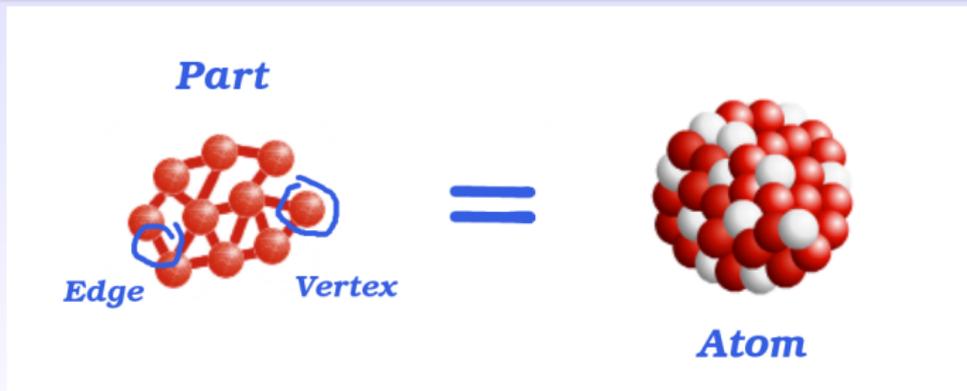
Fusion-Fission

Principe

Principe de la méthode de Fusion-Fission

la méthode de Fusion-Fission est basée sur une analogie avec les processus nucléaire de fusion et de fission

- nucléons = sommets
- atomes = parties



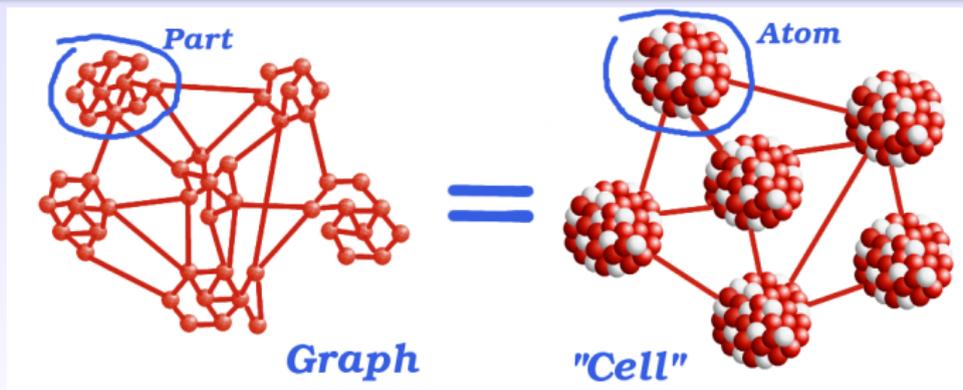
Fusion-Fission

Principe

Principe de la méthode de Fusion-Fission

la méthode de Fusion-Fission est basée sur une analogie avec les processus nucléaire de fusion et de fission

- nucléons = sommets
- atomes = parties



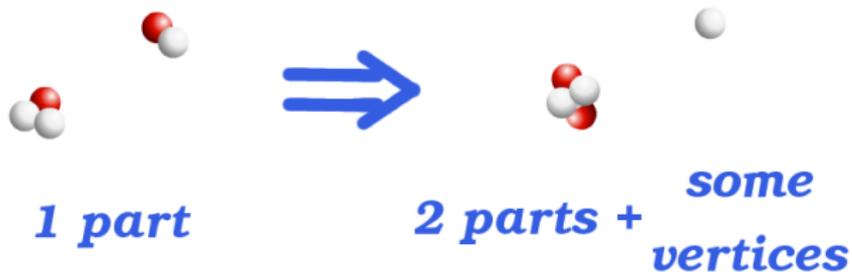
Fusion-Fission

Principe

Principe de la méthode de Fusion-Fission

la méthode de Fusion-Fission est basée sur une analogie avec les processus nucléaire de fusion et de fission

- nucléons = sommets
- atomes = parties
- étape de fusion



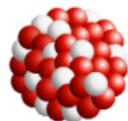
Fusion-Fission

Principe

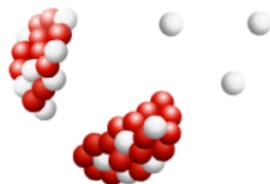
Principe de la méthode de Fusion-Fission

la méthode de Fusion-Fission est basée sur une analogie avec les processus nucléaire de fusion et de fission

- nucléons = sommets
- atomes = parties
- étape de fusion
- étape de fission



1 part

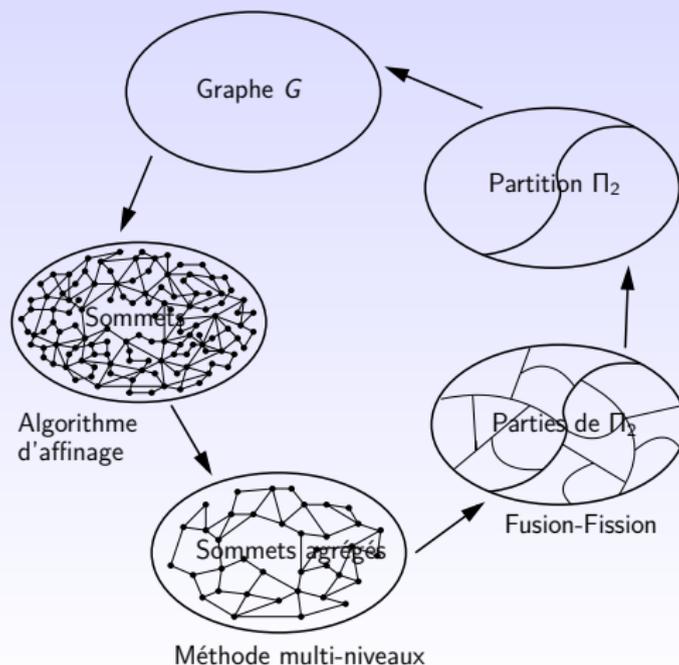


***2 parts +
some
vertices***

Fusion-Fission

Paradigme

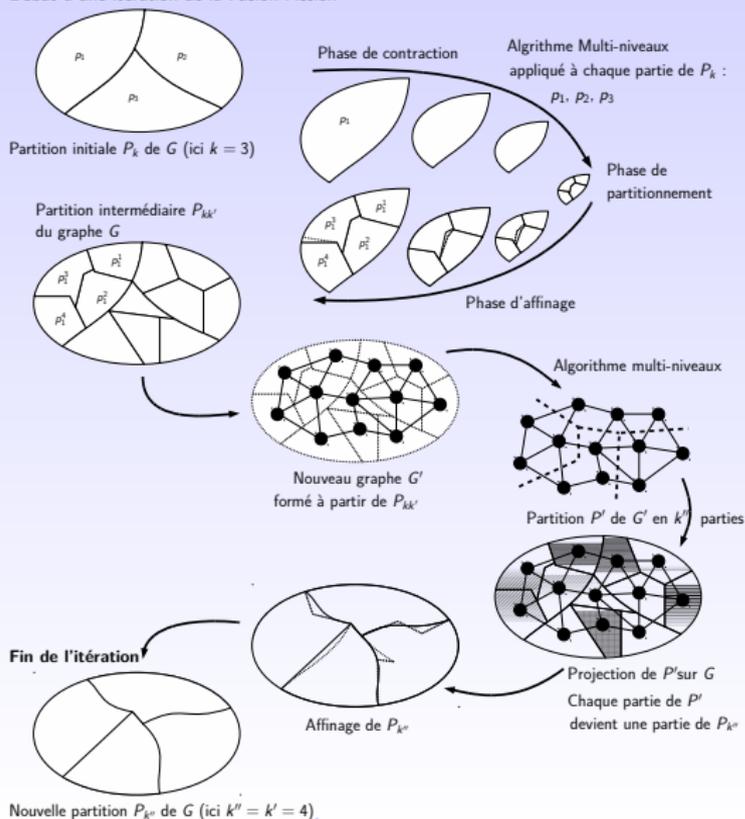
Paradigme du partitionnement de graphe



Fusion-Fission

Algorithme

Début d'une itération de la Fusion-Fission



Principe imagé de l'algorithme de fusion-fission

Fusion-Fission

Algorithme

Let *multi* be a black-box multilevel algorithm,
and *KL* a refinement algorithm.

```

Procédure FUSIONFISSION( $G = (V, E)$ ,  $k$ ,  $n$ )
   $P \leftarrow \text{multi}(G, k)$ 
   $P_{best} \leftarrow P = \{P_1, \dots, P_{best}\}$ 
  pour  $t = 1$  to  $n$  faire
    choose a new number of parts  $k'$ 
    split each part of  $P$  into  $k'$  parts % fission step
    make a graph  $G'$  based on these  $\text{card}(P) * k'$ 
     $P \leftarrow \text{multi}(G', k')$  % fusion step
     $P' \leftarrow \text{KL}(P)$  % ejection of nucleons
    If  $k' = k$  and  $\text{cut}(P) < \text{cut}(P_{best})$  then  $P_{best} \leftarrow P$ 
  fin pour
  return  $P_{best}$ 
fin Procédure

```

Résultats

Jeu de tests

Nom	# documents	sujet
Medline	1 033	médical
Cranfield	1 400	physique
Cisi	1 460	library

Résultats

Bi-partitioning results for Medline/Cranfield

Algorithm	Cluster	Medline	Cranfield	Normalized cut
Original clustering	D_0	1033	0	
	D_1	0	1400	
Fusion-Fission	D_0	1019	0	
	D_1	14	1400	
Graclus	D_0	765	0	
	D_1	268	1400	

W_0 : cell, patient, studi, increas, normal, growth
 W_1 : flow, pressur, number, boundari, layer, result

Bi-partitioning results for Medline/Cisi

Algorithm	Cluster	Medline	Cisi	Normalized cut
Original clustering	D_0	1033	0	
	D_1	0	1460	
Fusion-Fission	D_0	988	4	
	D_1	45	1456	
Graclus	D_0	825	40	
	D_1	208	1420	

W_0 : cell, patient, case, increas, effect, normal
 W_1 : librari, inform, system, index, data, research

Résultats

Bi-partitioning results for Medline/Cranfield

Algorithm	Cluster	Medline	Cranfield	Normalized cut
Original clustering	D_0	1033	0	0.3037
	D_1	0	1400	
Fusion-Fission	D_0	1019	0	0.3030
	D_1	14	1400	
Graclus	D_0	765	0	0.3706
	D_1	268	1400	

W_0 : cell, patient, studi, increas, normal, growth
 W_1 : flow, pressur, number, boundari, layer, result

Bi-partitioning results for Medline/Cisi

Algorithm	Cluster	Medline	Cisi	Normalized cut
Original clustering	D_0	1033	0	0.3402
	D_1	0	1460	
Fusion-Fission	D_0	988	4	0.3346
	D_1	45	1456	
Graclus	D_0	825	40	0.3474
	D_1	208	1420	

W_0 : cell, patient, case, increas, effect, normal
 W_1 : librari, inform, system, index, data, research

Résultats

3-way partitioning results for Medline/Cranfield/Cisi

Algorithm	Cluster	Medline	Cranfield	Cisi	Normalized cut
Original clustering	D_0	1033	0	0	
	D_1	0	1400	0	
	D_2	0	0	1460	
Fusion-Fission	D_0	903	0	4	
	D_1	7	1385	8	
	D_2	123	13	1448	
Graclus	D_0	874	0	0	
	D_1	18	1384	9	
	D_2	141	14	1451	

W_0 : cell, patient, increas, normal, growth, rat
 W_1 : flow, pressur, number, boundari, layer, result
 W_2 : librari, inform, system, index, data, research

Résultats

3-way partitioning results for Medline/Cranfield/Cisi

Algorithm	Cluster	Medline	Cranfield	Cisi	Normalized cut
Original clustering	D_0	1033	0	0	0.7586
	D_1	0	1400	0	
	D_2	0	0	1460	
Fusion-Fission	D_0	903	0	4	0.7499
	D_1	7	1385	8	
	D_2	123	13	1448	
Graclus	D_0	874	0	0	0.7602
	D_1	18	1384	9	
	D_2	141	14	1451	

W_0 : cell, patient, increas, normal, growth, rat
 W_1 : flow, pressur, number, boundari, layer, result
 W_2 : librari, inform, system, index, data, research

Quelques problèmes...

Problème d'adéquation entre le modèle et la vérité terrain !

Voix d'amélioration

- créer et/ou utiliser une autre fonction objectif
- poids exagéré des tokens les plus récurrents ?
- biais du à une approche non sémantique (pas de compréhension des documents) ?

- 1 Le problème du partitionnement de graphe
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien**
 - La structure de l'espace aérien
 - Objectifs et contraintes du découpage de l'espace aérien
 - Modélisation de l'espace aérien
 - Résultats
- 6 Application à la segmentation d'image
- 7 Conclusion

La structure de l'espace aérien

Les secteurs de contrôle

Le rôle des contrôleurs aériens :

Assurer la sécurité des aéronefs en faisant respecter les distances de séparation.

La structure de l'espace aérien

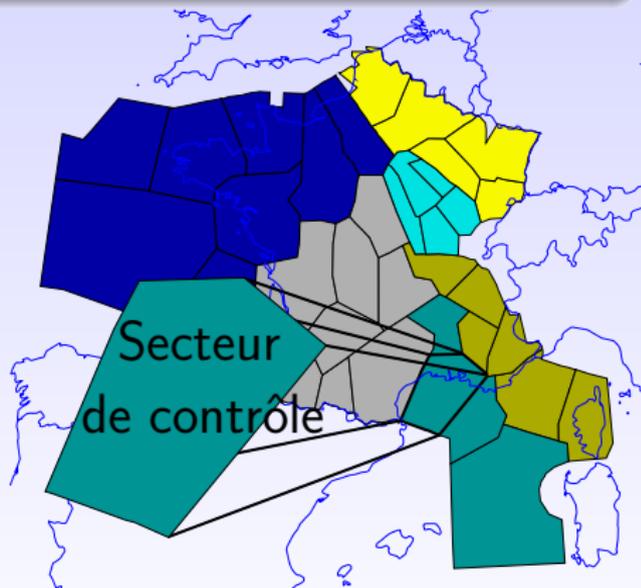
Les secteurs de contrôle

Le rôle des contrôleurs aériens :

Assurer la sécurité des aéronefs en faisant respecter les distances de séparation.

Secteur de contrôle :

Volume limité d'espace entre deux niveaux de vol, sous la responsabilité d'un contrôleur aérien.



Coupe horizontale au FL320 des secteurs aériens de contrôle français.

La structure de l'espace aérien

Les secteurs de contrôle

Le rôle des contrôleurs aériens :

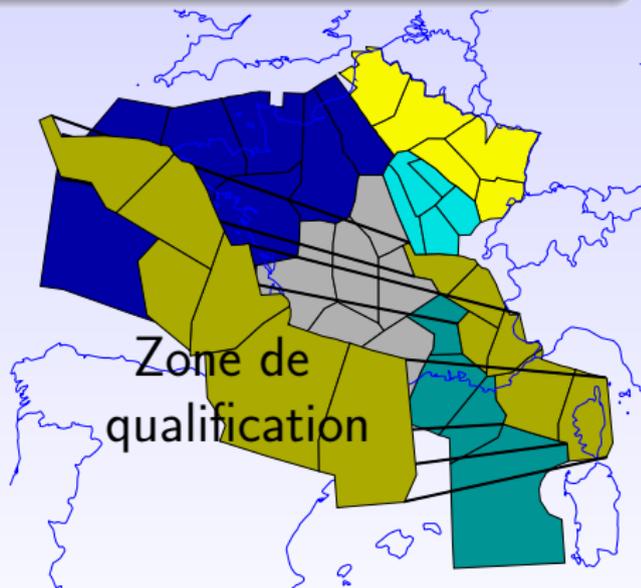
Assurer la sécurité des aéronefs en faisant respecter les distances de séparation.

Secteur de contrôle :

Volume limité d'espace entre deux niveaux de vol, sous la responsabilité d'un contrôleur aérien.

Zone de qualification :

Ensemble des secteurs sur lesquels un contrôleur aérien est habilité à contrôler.



Coupe horizontale au FL320 des zones de qualification françaises.

La structure de l'espace aérien

Les secteurs de contrôle

Le rôle des contrôleurs aériens :

Assurer la sécurité des aéronefs en faisant respecter les distances de séparation.

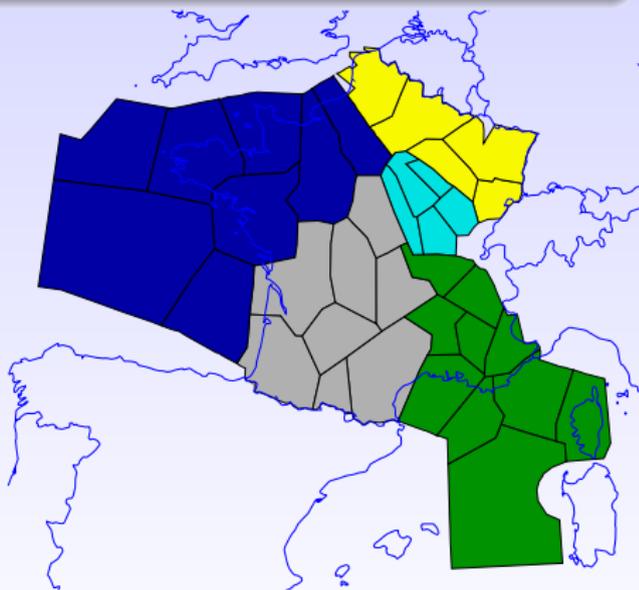
Secteur de contrôle :

Volume limité d'espace entre deux niveaux de vol, sous la responsabilité d'un contrôleur aérien.

Zone de qualification :

Ensemble des secteurs sur lesquels un contrôleur aérien est habilité à contrôler.

Centre de contrôle : lieu physique de contrôle rassemblant une ou plusieurs zones de qualifications.



Coupe horizontale au FL320 des centres de contrôle français.

La structure de l'espace aérien

Charge de contrôle

Travail du contrôleur dans un secteur :

- accueillir les aéronefs ;
- surveiller le trafic ;
- résoudre les conflits ou les anticiper ;
- passer les aéronefs aux autres secteurs.

Charge de contrôle :

- surveillance du trafic ;
- résolution des conflits ou conflits potentiels ;
- charge de coordination.

charge de coordination = accueil + passage aux autres secteurs.

La structure de l'espace aérien

La charge de coordination

Coordination :

La coordination est plus **simple** et **efficace** entre secteurs d'une **même** zone de qualification.

La structure de l'espace aérien

La charge de coordination

Coordination :

La coordination est plus simple et efficace entre secteurs d'une même zone de qualification.

Trois raisons :

- Le contrôleur connaît bien les secteurs de sa zone ;
- Les contrôleurs d'une même zones sont dans la même salle ;
- Pas de coordination à l'intérieur des secteurs *regroupés*.

Secteurs regroupés :

Quand le trafic est faible, des secteurs de contrôle d'une même zone peuvent être regroupés.

La structure de l'espace aérien

La charge de coordination

Coordination :

La coordination est plus simple et efficace entre secteurs d'une même zone de qualification.

Trois raisons :

- Le contrôleur connaît bien les secteurs de sa zone ;
- Les contrôleurs d'une même zones sont dans la même salle ;
- Pas de coordination à l'intérieur des secteurs *regroupés*.

Secteurs regroupés :

Quand le trafic est faible, des secteurs de contrôle d'une même zone peuvent être regroupés.

Objectif de notre étude :

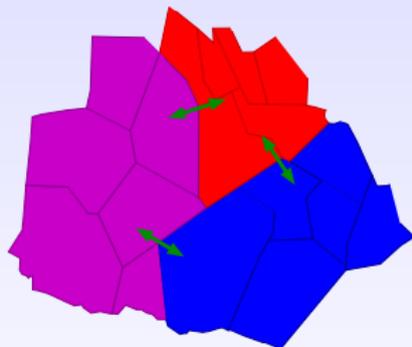
Minimiser la charge de coordination entre zones de qualification pour augmenter la fluidité et la capacité de l'espace aérien.

Objectifs et contraintes du découpage de l'espace aérien

Objectifs

Objectifs du découpage :

- **Minimiser** les flux d'avions **entre** les zones de qualification ;
- Concentrer les flux d'avions dans les zones de qualification.



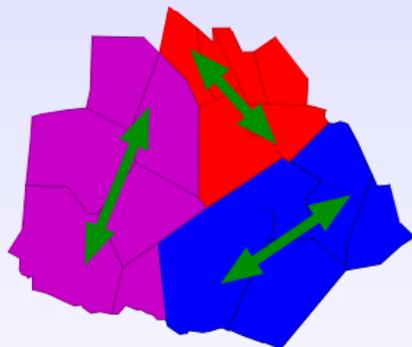
Flux entre secteurs = nombre de passage d'aéronefs entre ces secteurs

Objectifs et contraintes du découpage de l'espace aérien

Objectifs

Objectifs du découpage :

- Minimiser les flux d'avions entre les zones de qualification ;
- **Concentrer** les flux d'avions **dans** les zones de qualification.



Flux entre secteurs = nombre de passage d'aéronefs entre ces secteurs

Objectifs et contraintes du découpage de l'espace aérien

Contraintes

Contraintes sur le nombre de secteurs d'une zone de qualification :

- en moyenne 25 secteurs ;
- ne pas dépasser 50 secteurs ;

Permet de respecter les recommandations de l'agence Eurocontrol.

Objectifs et contraintes du découpage de l'espace aérien

Contraintes

Poids d'une zone de qualification :

Somme du nombre d'avions passant dans les secteurs de cette zone.

Objectifs et contraintes du découpage de l'espace aérien

Contraintes

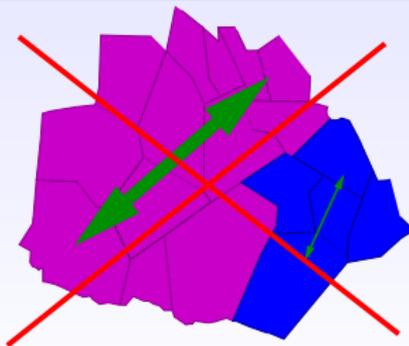
Poids d'une zone de qualification :

Somme du nombre d'avions passant dans les secteurs de cette zone.

Contraintes sur le poids d'une zone de qualification

Une zone de qualification ne doit pas être **2 fois plus grosse** que le poids moyen d'une zone.

Éviter de trop grandes disparités entre zones.



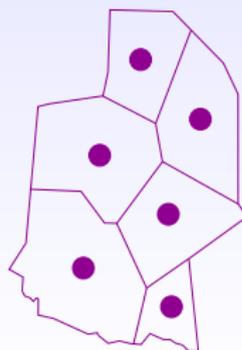
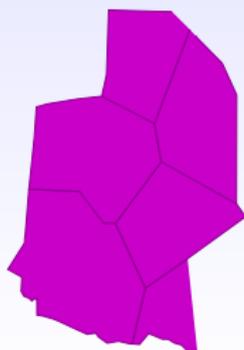
Modélisation de l'espace aérien

Graphe

Modélisation de la structure de l'espace aérien

L'espace aérien est modélisé par un **graphe pondéré** $G = (S, A)$,

- secteurs de contrôle \iff sommets du graphe ;



Modélisation de l'espace aérien

Graphe

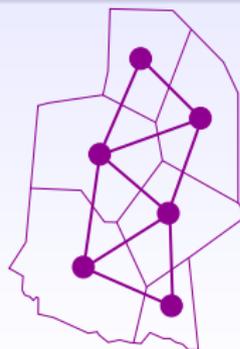
Modélisation de la structure de l'espace aérien

L'espace aérien est modélisé par un graphe pondéré $G = (S, A)$,

- secteurs de contrôle \iff sommets du graphe ;
- flux d'avions entre secteurs \iff arêtes pondéré entre sommets.

Poids et notations :

- d'une arête $(s_1, s_2) \in A$, $poids(s_1, s_2)$.



Modélisation de l'espace aérien

Graphe

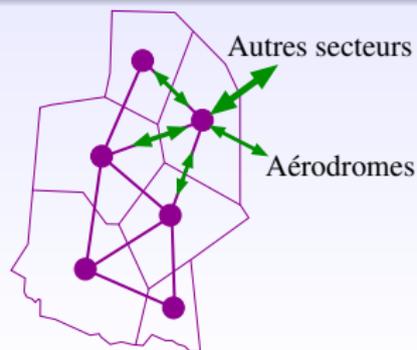
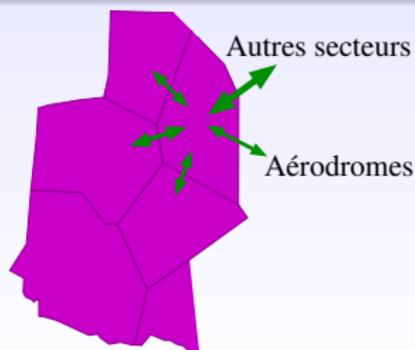
Modélisation de la structure de l'espace aérien

L'espace aérien est modélisé par un graphe pondéré $G = (S, A)$,

- secteurs de contrôle \iff sommets du graphe ;
- flux d'avions entre secteurs \iff arêtes pondérées entre sommets.

Poids et notations :

- d'une arête $(s_1, s_2) \in A$, $poids(s_1, s_2)$.
- d'un sommet $s \in S$, $poids(s)$;



Modélisation de l'espace aérien

Graphe

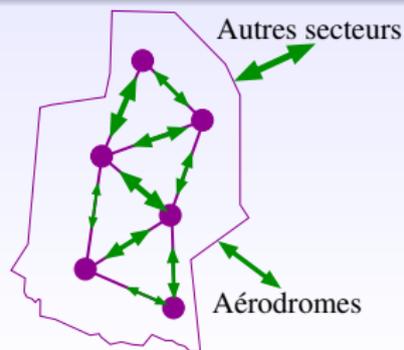
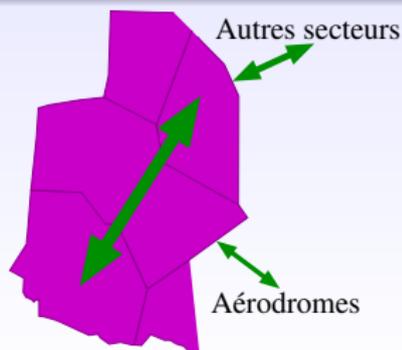
Modélisation de la structure de l'espace aérien

L'espace aérien est modélisé par un graphe pondéré $G = (S, A)$,

- secteurs de contrôle \iff sommets du graphe ;
- flux d'avions entre secteurs \iff arêtes pondérées entre sommets.

Poids et notations :

- d'une arête $(s_1, s_2) \in A$, $poids(s_1, s_2)$.
- d'un sommet $s \in S$, $poids(s)$;
- d'un ensemble de sommets S' , $poids(S') = \sum_{s \in S'} poids(s)$;



Modélisation de l'espace aérien

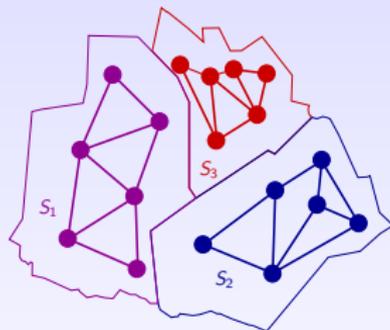
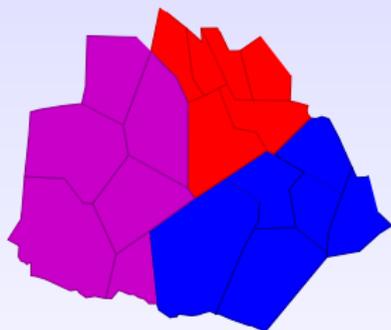
Découpage

Modélisation du découpage de l'espace aérien

Découper l'espace aérien en k zones de qualification



Partitionner le graphe de l'espace aérien en k parties.



Partition des sommets d'un graphe $G = (S, A)$:

Les sommets de S sont répartis dans k sous-ensembles S_1, \dots, S_k , pour former la partition $P_k = \{S_1, \dots, S_k\}$.

Résultats

Comparaison des performances des méthodes de partitionnement

Performances de différentes méthodes de partitionnement :

Moyenne sur 100 résultats :

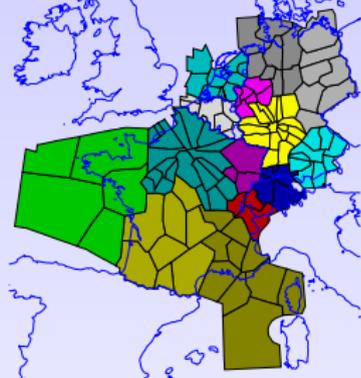
Algorithmes	FF	Recuit	Scotch	Graclus	Jostle	Metis
<i>ratio</i>	5,85	6,04	6,11	6,25	6,29	6,88
<i>balance</i>	1,87	1,92	1,72	1,75	1,37	1,13
Zone max	45,0	57,5	45,0	45,25	45,5	47,0

Analyse :

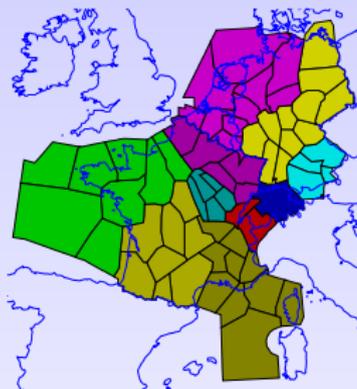
- Bon résultat de la fusion-fission ;
- Paramétrage facile de la balance pour : Scotch et fusion-fission ;
- Temps d'exécution long du recuit et de la fusion-fission.

Résultats

Découpage de 6 pays du cœur de l'Europe



Découpage réel au FL200



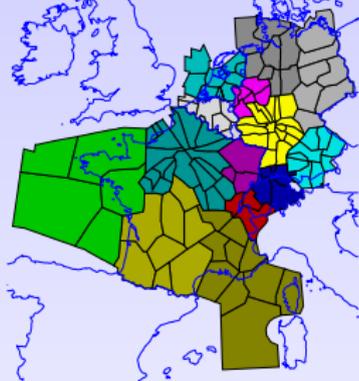
Découpage réel au FL360

Découpage en 16 parties :

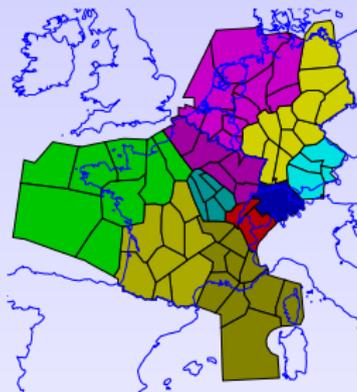
	<i>ratio</i>	<i>coupe</i>	<i>balance</i>	max, min secteurs	
Centres, le 17/6/2005	4,36	24 875	1,91	38	7
Fusion-fission	3,55	21 674	1,28	26	15
Scotch	3,82	23 094	1,20	29	14
Graclus	4,03	24 524	1,42	28	12

Résultats

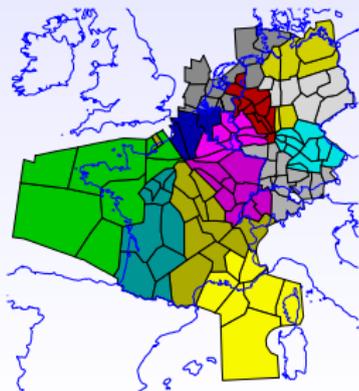
Découpage de 6 pays du cœur de l'Europe



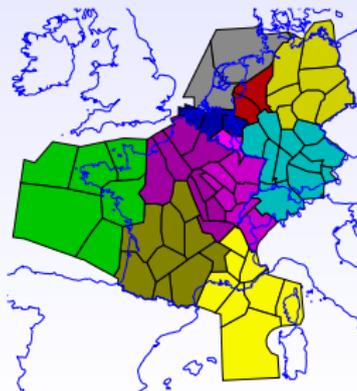
Découpage réel au FL200



Découpage réel au FL360



Découpage trouvé au FL200



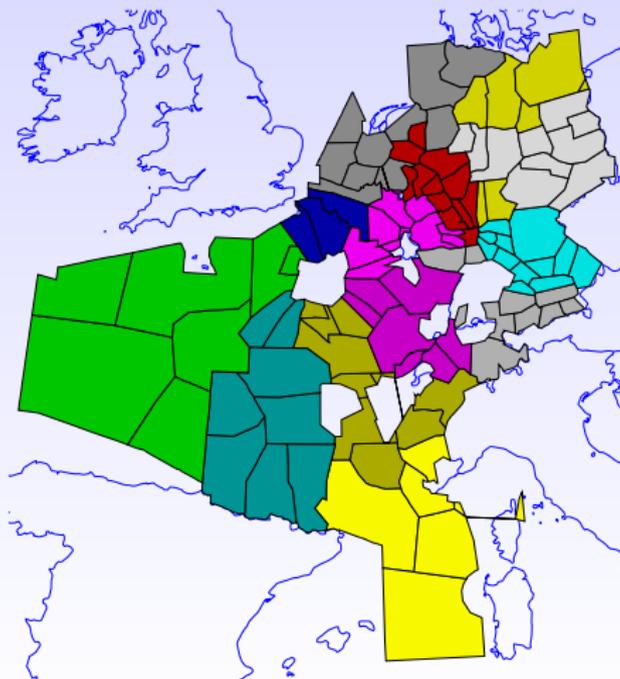
Découpage trouvé au FL360

Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL120

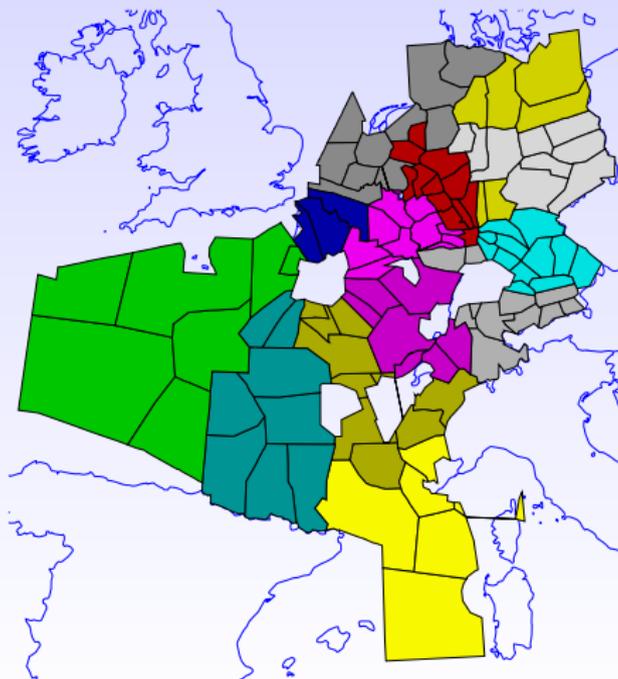


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL140

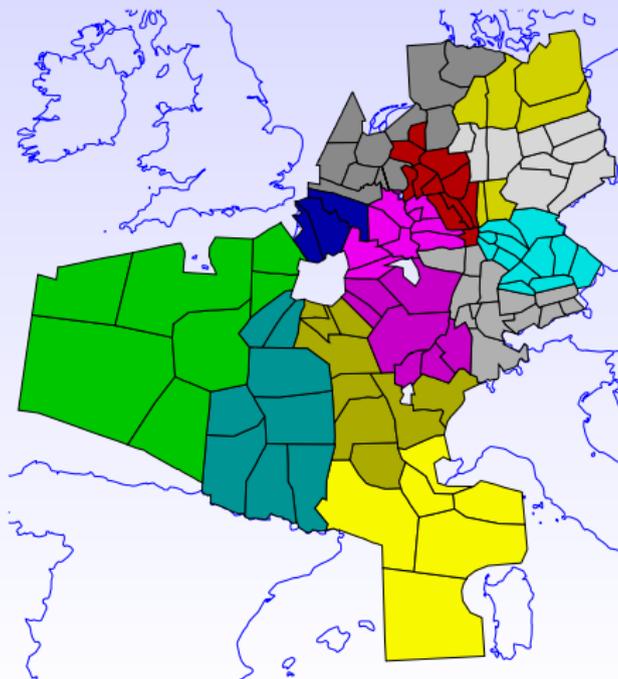


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL160

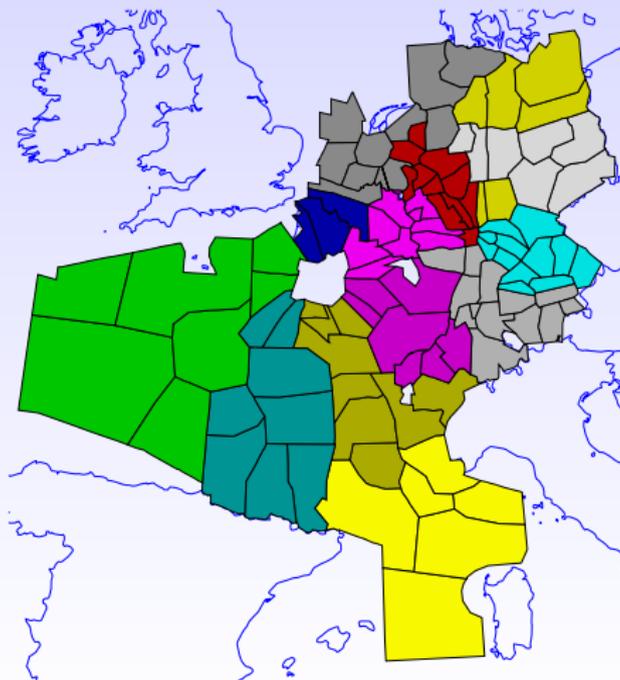


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL180

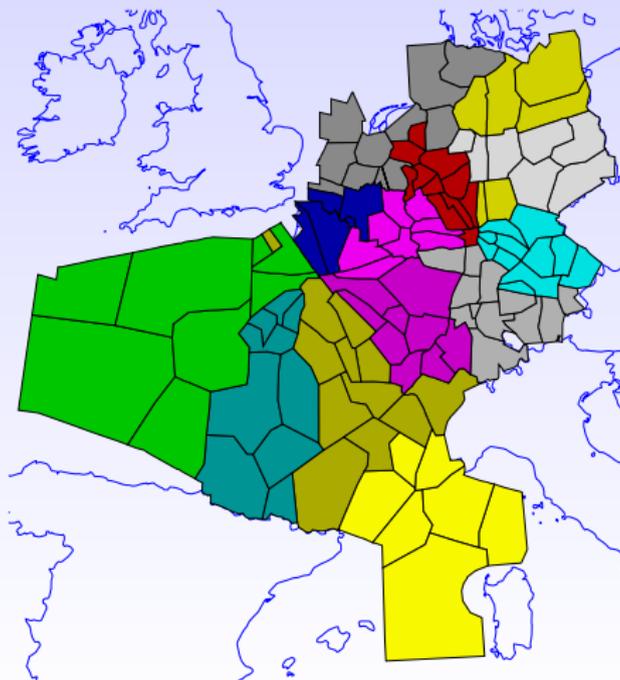


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL200

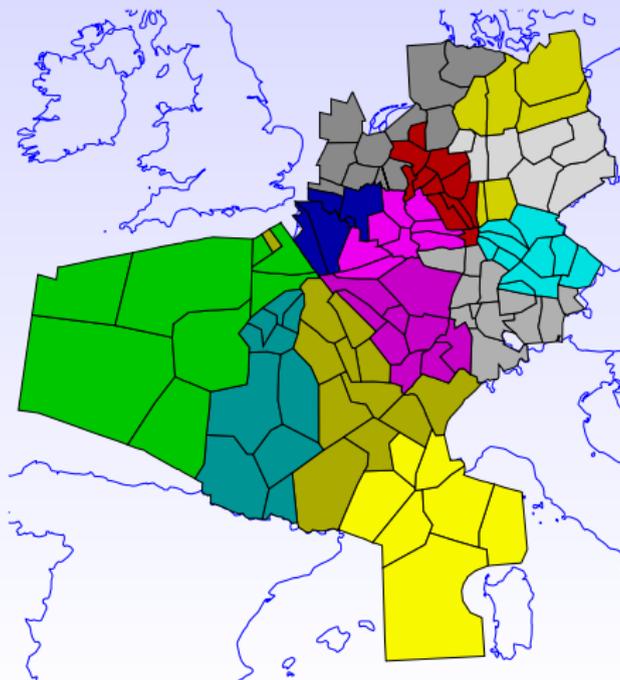


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL220

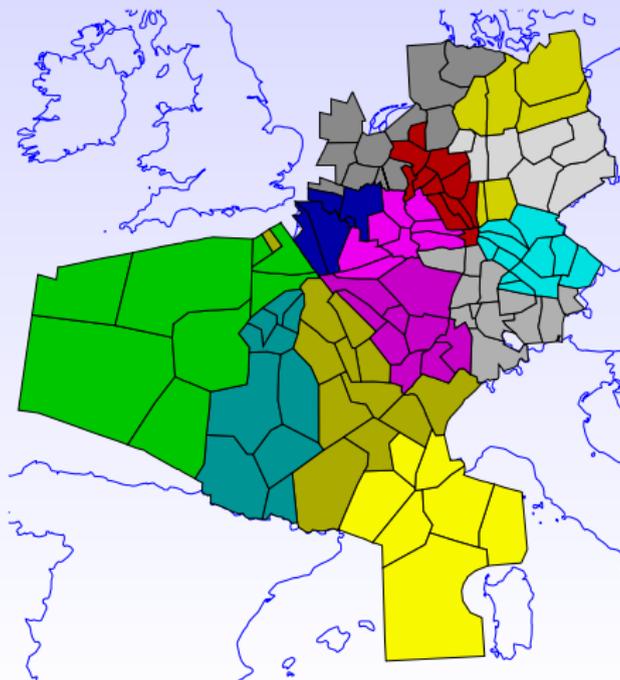


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL240

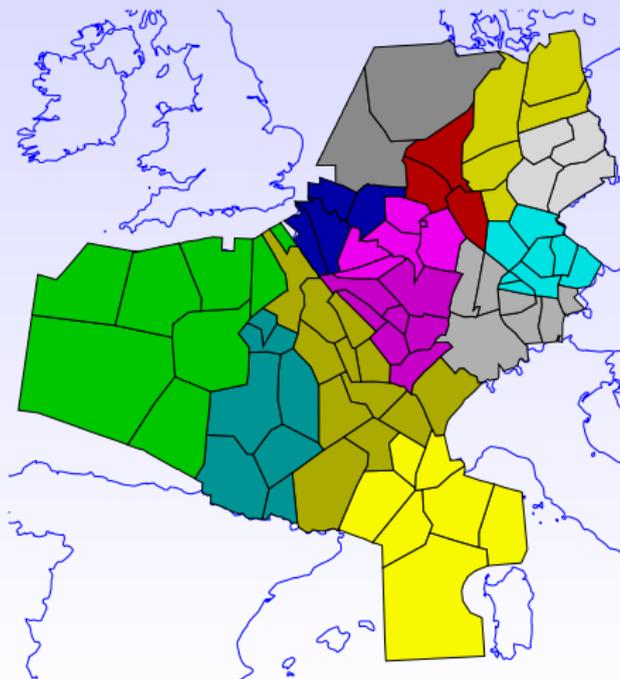


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL260

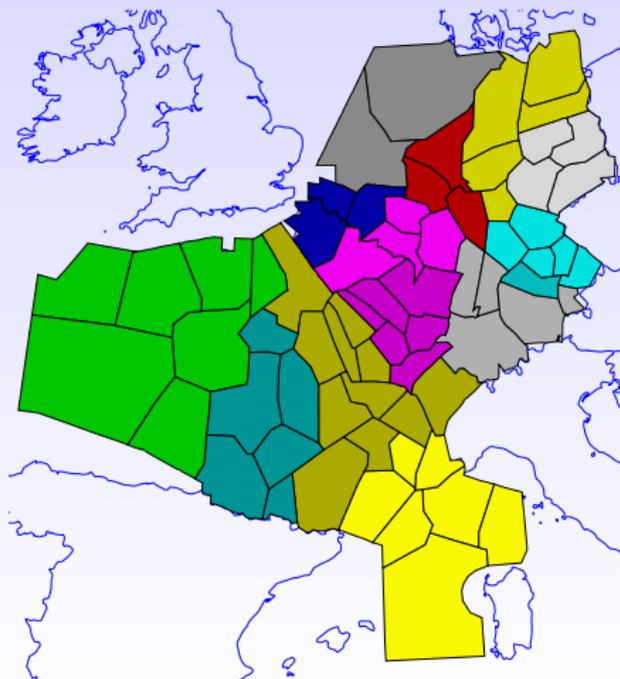


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL280

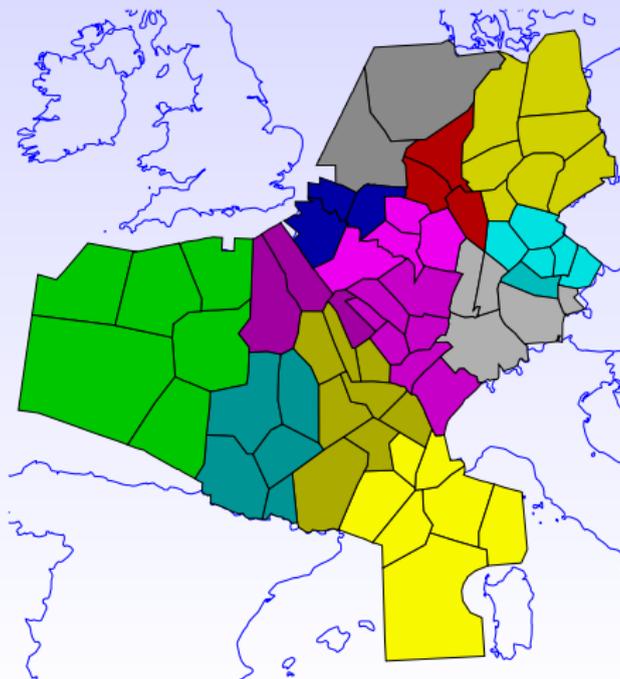


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL300

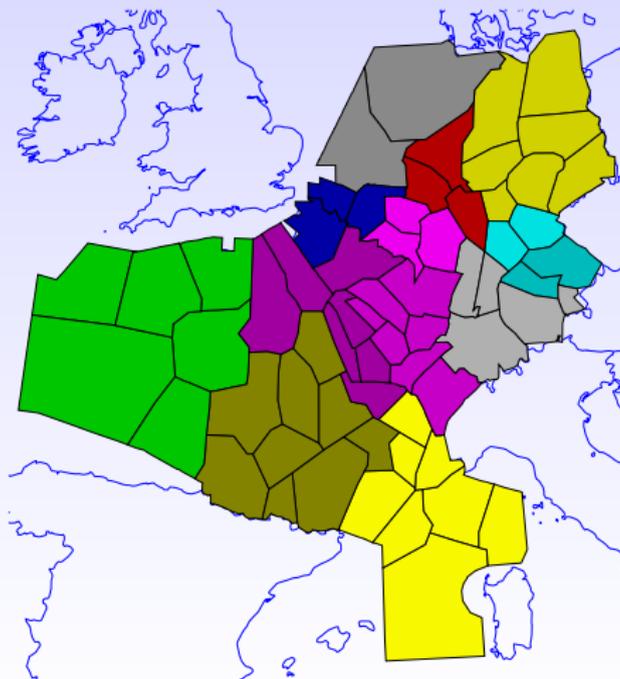


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL320

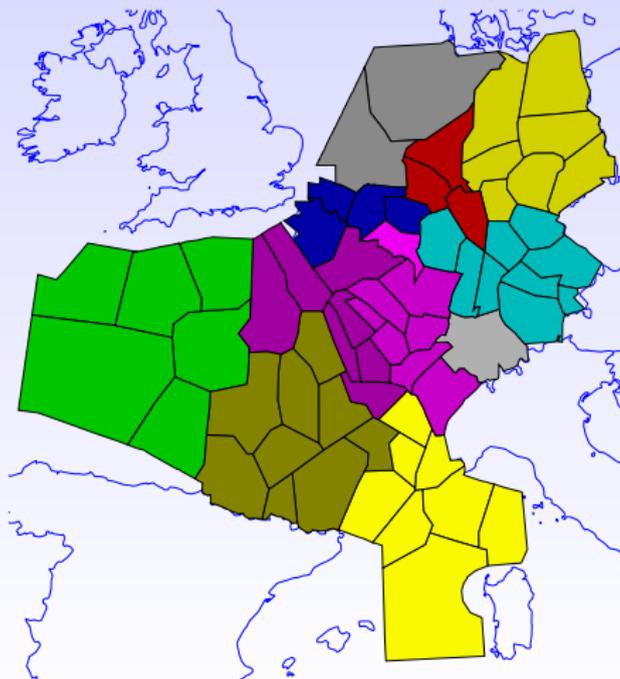


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL340

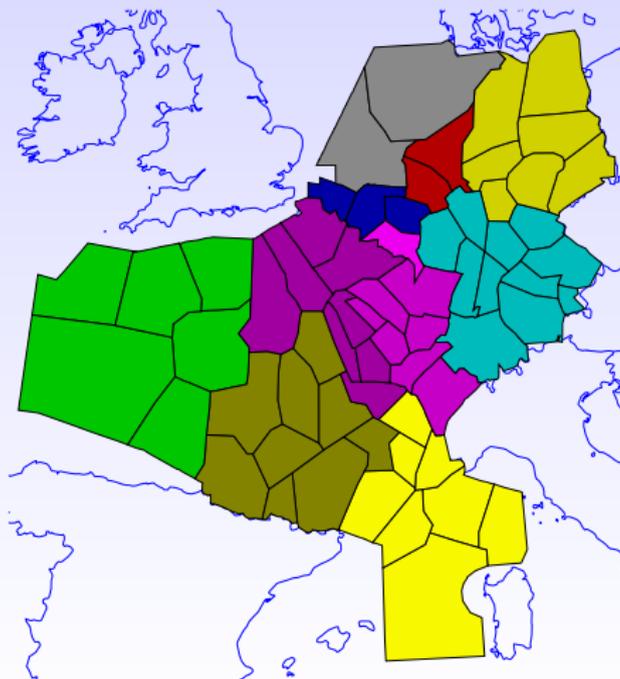


Résultats

Découpage de 6 pays du cœur de l'Europe

Représentation par niveaux de vol des zones de qualification trouvées par la fusion-fission

FL360



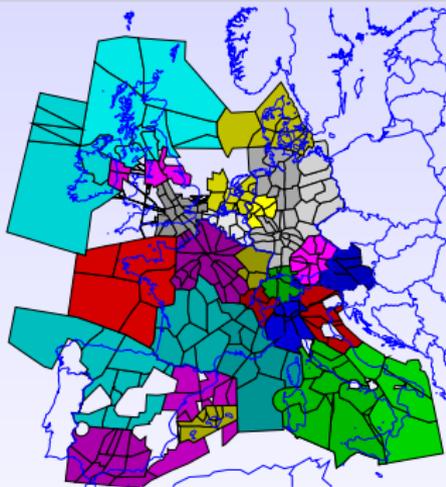
Résultats

Découpage de la *Core area*

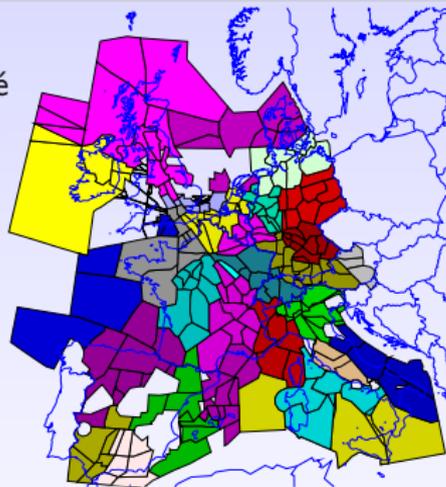
Core area :

Région du ciel aérien européen de plus forte densité de trafic

Réel
FL200



Trouvé
FL200



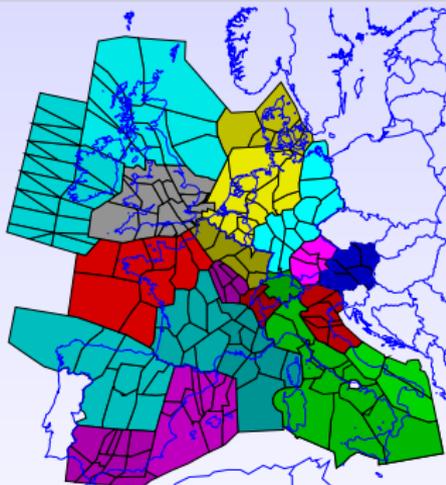
Résultats

Découpage de la *Core area*

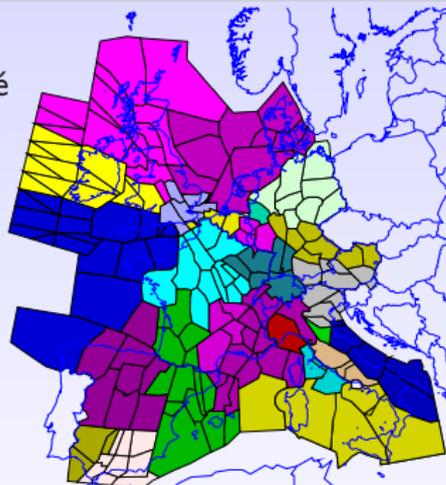
Core area :

Région du ciel aérien européen de plus forte densité de trafic

Réel
FL360



Trouvé
FL360



Grand espace aérien :

- Interprétation délicate ;
- Selon les pays, critères de découpage différents.

- 1 Le problème du partitionnement de graphe
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image**
 - La segmentation comme un problème de partitionnement de graphe
 - Résolution par méthode spectrale
 - Quelques résultats
- 7 Conclusion

Segmentation sémantique d'image

Le problème de la **segmentation sémantique d'image** est de découper une image en différentes régions telles que chaque région représente un concept sémantique (objet, groupe d'objets, etc)



FIGURE: Image originale

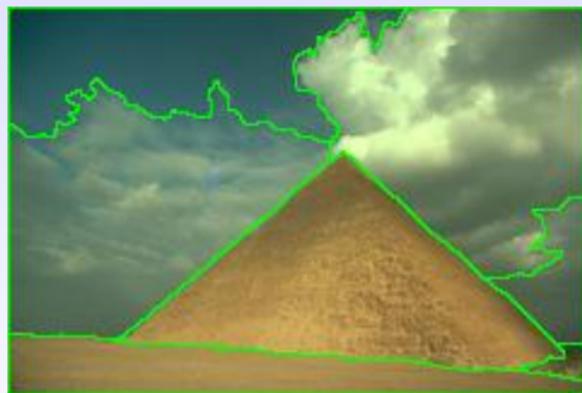


FIGURE: Image segmentée

Quelques questions

Cette définition est très subjective. . .

- concrètement, qu'est-ce qu'une région ? peut-on en donner une définition précises ?
- combien de région une image possède-t-elle ?

Le nombre de régions

Le nombre de région dépend de l'interprétation humaine !



FIGURE: 4 régions



FIGURE: 4 régions



FIGURE: 3 régions

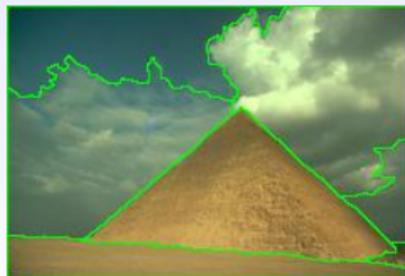
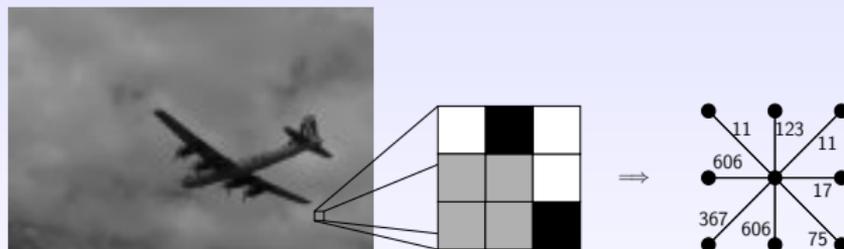


FIGURE: 7 régions

La segmentation comme un problème de partitionnement de graphe

Principe

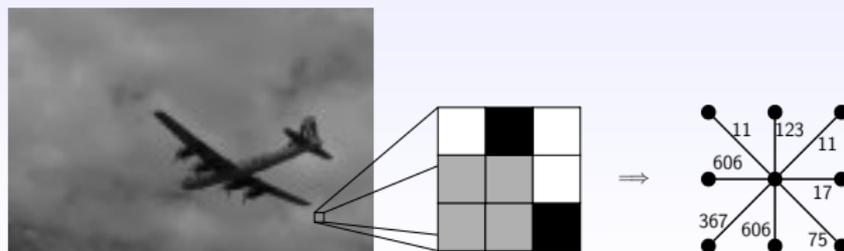
- Modéliser une image par un graphe
- Partitionner le graphe
- Chaque partie représente un segment de l'image



La segmentation comme un problème de partitionnement de graphe

Construction du graphe

- chaque pixel devient un sommet du graphe
- arêtes pondérées
 - arête basée sur le voisinage (comment relier deux sommets ?)
 - pondération basée sur les caractéristiques de l'image (contours, couleur, texture, etc)



La segmentation comme un problème de partitionnement de graphe

Construction du graphe

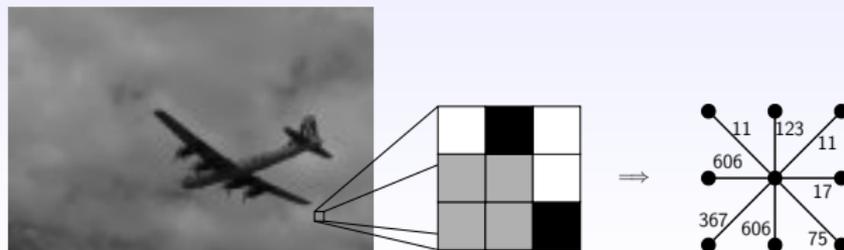
- chaque pixel devient un sommet du graphe
- arêtes pondérées
 - arête basée sur le voisinage (comment relier deux sommets ?)
 - pondération basée sur les caractéristiques de l'image (contours, couleur, texture, etc)

Poids d'une arête (i, j) :

$$w_{ij} = \exp \frac{-\|X(i) - X(j)\|^2}{\sigma_X} * \exp \frac{-\|I(i) - I(j)\|^2}{\sigma_I}$$

$\|X(i) - X(j)\|$: distance entre les sommets i et j

$\|I(i) - I(j)\|$: différence d'intensité lumineuse entre i et j



Résolution par méthode spectrale

Principe des méthodes spectrales

Trouver les premiers vecteurs propres (vecteurs de Fiedler) de la matrice Laplacienne du graphe

Relation avec la segmentation d'image

- chaque vecteur de Fiedler permet d'obtenir une partition du graphe en deux parties
- plusieurs fonctions de coûts peuvent être modélisées
→ coupe normalisée (NCut)



Jianbo Shi and Jitendra Malik.

Normalized cuts and image segmentation.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8) :888–905, 2000.

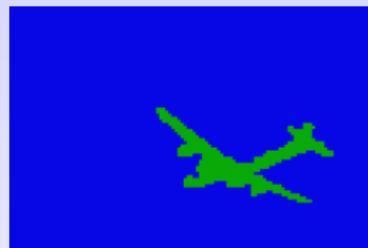
Quelques résultats

Segmentation d'une image d'avion

Segmentation de l'image d'un avion sur fond de ciel nuageux.



Image d'origine



fusion-fission



Expansion



Spectrale

Bilan :

Pour une image simple, la segmentation est correcte.

Quelques résultats

Segmentation de l'image d'un palmier

Segmentation de l'image d'un palmier au bord de l'eau sur fond de lagune.



Image d'origine



fusion-fission



Expansion

Bilan :

Pour une image plus compliquée, la segmentation n'est pas au point.

Quelques résultats

Quelques résultats

Comparaison méthode spectrale (Ncut) – multiniveaux (Graclus)

(a) Graclus $k = 2$ (b) Graclus $k = 3$ (c) Graclus $k = 4$ (d) Graclus $k = 5$ (e) Ncut $k = 2$ (f) Ncut $k = 3$ (g) Ncut $k = 4$ (h) Ncut $k = 5$

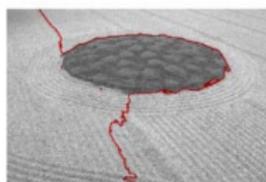
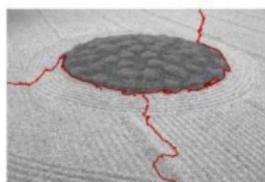
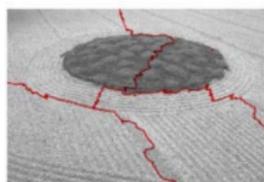
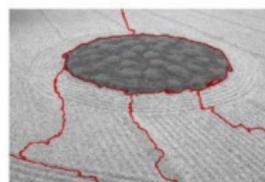
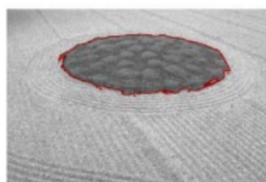
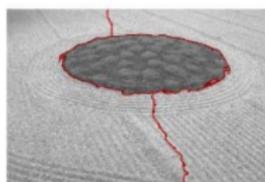
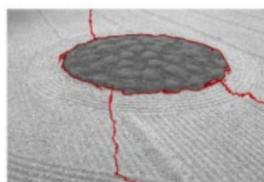
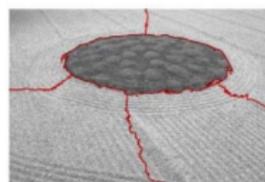
Quelques résultats

Comparaison méthode spectrale (Ncut) – multiniveaux (Graclus)

(a) Graclus $k = 2$ (b) Graclus $k = 3$ (c) Graclus $k = 4$ (d) Graclus $k = 5$ (e) Ncut $k = 2$ (f) Ncut $k = 3$ (g) Ncut $k = 4$ (h) Ncut $k = 5$

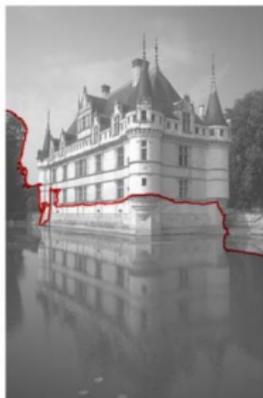
Quelques résultats

Comparaison méthode spectrale (Ncut) – multiniveaux (Graclus)

(a) Graclus $k = 2$ (b) Graclus $k = 3$ (c) Graclus $k = 4$ (d) Graclus $k = 5$ (e) Ncut $k = 2$ (f) Ncut $k = 3$ (g) Ncut $k = 4$ (h) Ncut $k = 5$

Quelques résultats

Comparaison méthode spectrale (Ncut) – multiniveaux (Graclus)

(a) Graclus $k = 2$ (b) Graclus $k = 4$ (c) Ncut $k = 2$ (d) Ncut $k = 4$

- 1 Le problème du partitionnement de graphe
- 2 Méthodes d'optimisation du partitionnement de graphes
- 3 Application au calcul numérique
- 4 Application à la classification de documents
- 5 Application au découpage de l'espace aérien
- 6 Application à la segmentation d'image
- 7 Conclusion**
 - Directions actuelles et perspectives
 - Défis
 - Livres

Directions actuelles et perspectives

Directions actuelles et perspectives

- Algorithme de partitionnement de graphe
 - Calcul sur GPU
 - Publier les sources de la Fusion-Fission
- Affinage des partitions
 - Modifier la structure de données pour permettre une plage de valeurs
 - Changer de fonction objectif
 - Prendre en compte le poids des sommets
- Segmentation d'image
 - Modéliser plus finement l'image : travail sur la texture

Défis

Défis à relever

- Affinage des partitions
- Parallélisation du partitionnement (GPU)
- Construction du graphe
- Nouvelle modélisation de la classification de documents basée sur le partitionnement de graphe
- Segmentation réellement sémantique (associer détection d'objet et classification d'image)
- Autres applications !!!

Livres

- « Partitionnement de graphe » chez Hermes-Lavoisier (14 chapitres)
- « Graph Partitioning » chez ISTE-Wiley (13 chapitres)

Édités par Charles-Edmond Bichot et Patrick Siarry
Liste des 19 auteurs :

Jean-Baptiste Angelelli (IML) ; Thomas Aynaud (LIP6) ; Charles-Edmond Bichot (LIRIS) ;
Vincent D. Blondel (Université Catholique de Louvain) ; Alexandre Caminada (UTBM) ; Cédric
Chevalier (Sandia National Laboratories) ; Nicoals Durand (IRIT) ; Laura Grigori (INRIA
Saclay-Ile de France) ; Alain Guénoche (IML) ; Jean-Loup Guillaume (LIP6) ; Renaud Lambiotte
(Imperial College London) ; Sid Lamrous (UTBM) ; Amir Nakib (LiSSi) ; Laurent Najman
(LIGM) ; Mustapha Oughdi (UTBM) ; François Pellegrini (LaBRI) ; Laurence Reboul (IML) ;
Patrick Siarry (LiSSi) ; Hugues Talbot (LIGM)

Sites des livres :

- http://cebichot.netne.net/graph_partitioning_book/
- http://cebichot.netne.net/livre_partitionnement/

