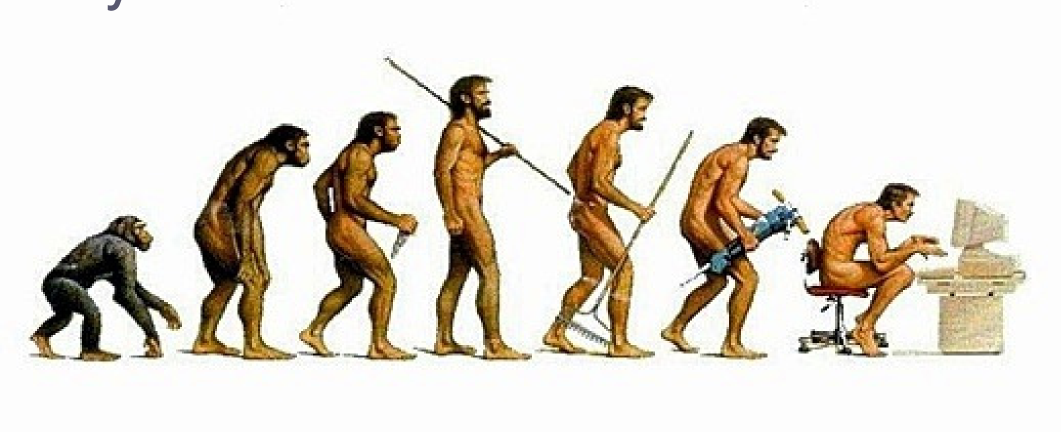
QUATERNION POUR L'ANIMATION DE PERSONNAGE

Alexandre Meyer

Equipe SAARA, Laboratoire LIRIS

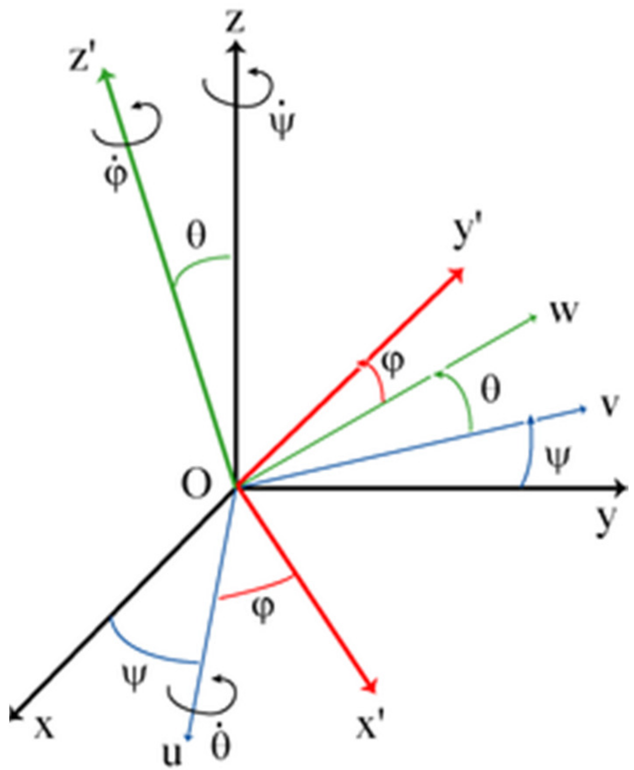
Université Lyon 1



Angles d'Euler

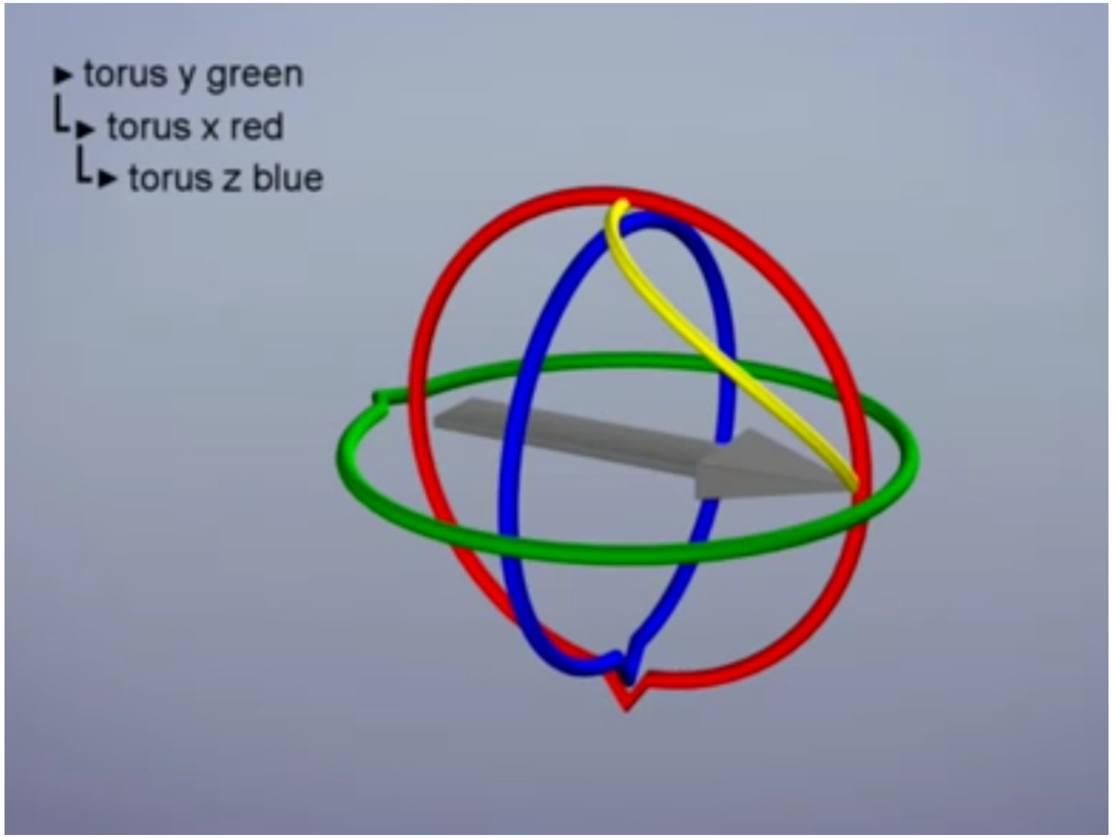
- 3 rotations pour faire tourner un repère

- La *précession* ψ, autour de l'axe O**z**, fait passer de (O,*x*,*y*,**z***)* au référentiel (O,*u*,*v*,**z***)* (en bleu).
- La *nutation* θ, autour de l'axe O*u* (ligne des nœuds), fait passer de (O,*u*,*v*,**z***)* à (O,*u*,*w*,**z'***)* (en vert).
- La *rotation propre* φ, ou giration, autour de l'axe O*z'*, fait passer de (O,*u*,*w*,**z'***)* au référentiel lié au solide (O,*x'*,*y'*,**z'***)* (en rouge).



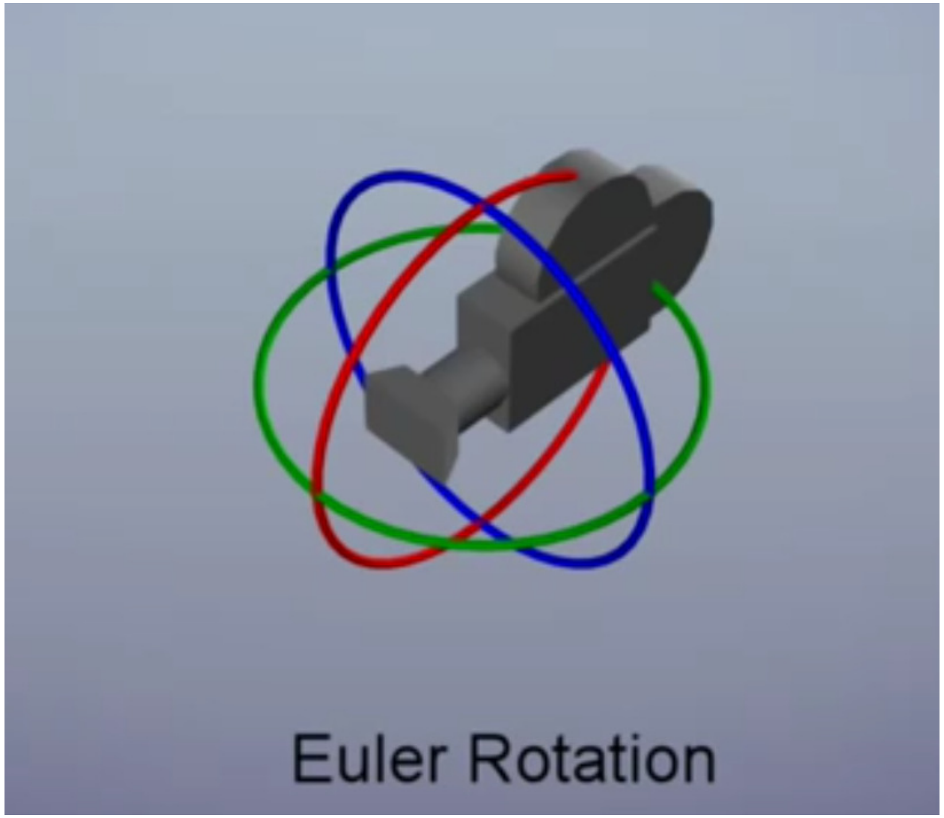
Angle d’Euler : interpolation

- Interpolation des 3 angles donne la trajectoire jaune
  - Ce n’est pas tout à fait la géodésique (trajectoire la plus courte)



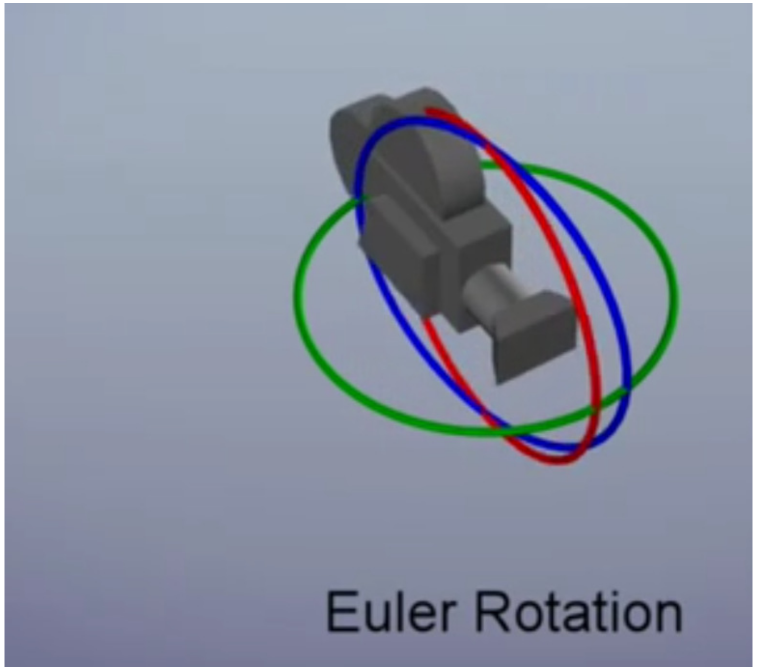
Angle d'Euler

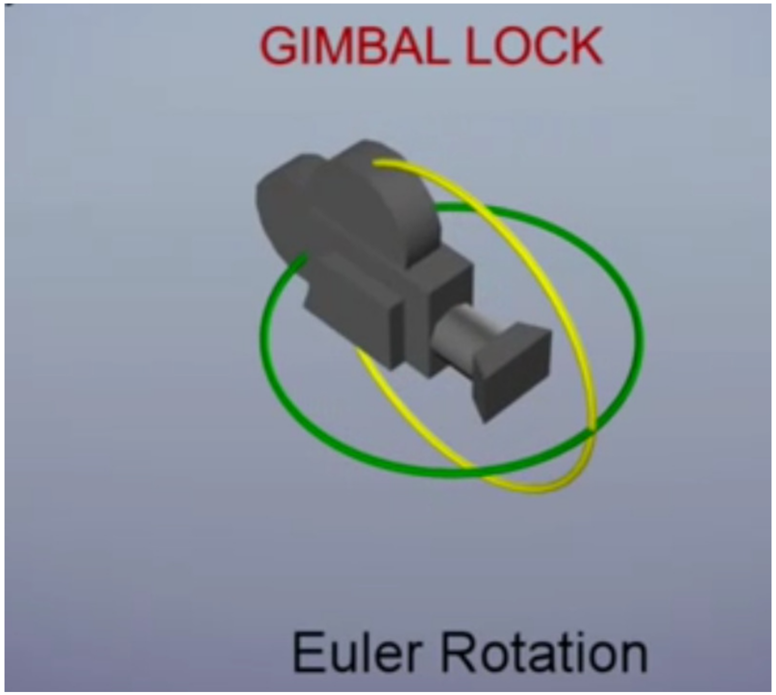
- Rotation autour de bleu, vert, rouge
  - ok



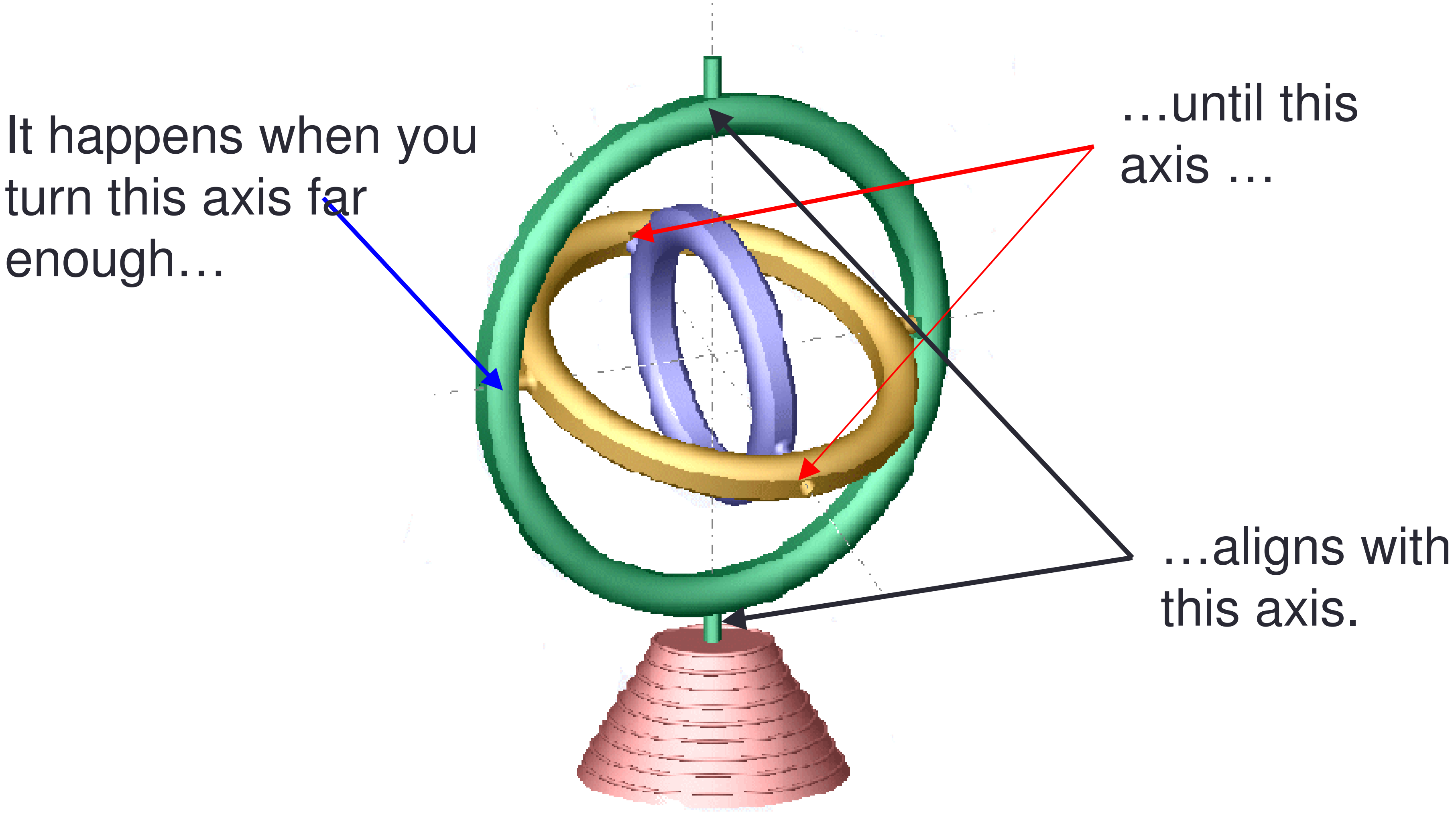
Angle d'Euler
Gimbal lock ou blocage de cardan

- Rotation : rotation autour de rouge et bleu
  - Blocage
  - Perte d'un degré de liberté





Angle d'Euler
Gimbal lock ou blocage de cardan



Et donc ?

Solution au blocage de cardan

- Matrice de rotation 3x3

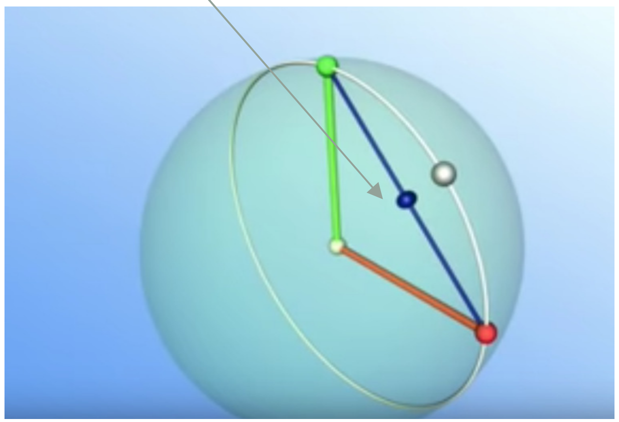
Oui mais pour l'interpolation ?

- Interpoler les 9 paramètres de la matrice ?

  → bof bof elle ne sera plus orthonormée

  Et dans certain cas, perte de degré de liberté aussi

➔Quaternion



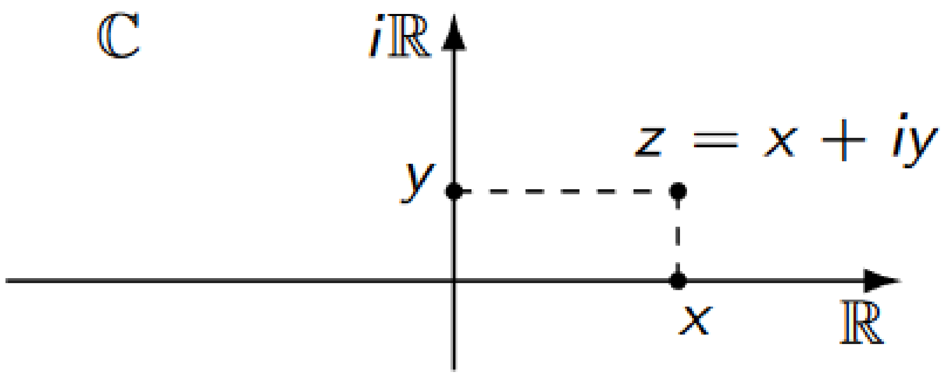
Nombres complexes

Un nombre complexe est un nombre de la forme

$$\boxed{z = x + i.y}$$

où x et y sont des nombres réels et i est un symbole ayant la propriété de $\boxed{i^2 = -1}$

Le plan complexe :



Nombres complexes

- Pour un nombre complexe

$$\boxed{z = x + i.y}$$

- **Partie réelle**, le nombre réel Re(z) = x
- **Partie imaginaire**, le nombre réel Im(z) = y

- Le conjugué complexe est le nombre complexe

$$\boxed{\overline{z} = x - i.y}$$

Nombres complexes : opérations

- Soit les deux nombres complexes

  $\boxed{\mathbf{z = x + i.y}}$ et $\boxed{\mathbf{w = u + i.v}}$

- **Egalité** : si et seulement si $x=u$ et $y=v$

- **Addition :** $z+w = (x+u) + i.(y+v)$
- **Soustraction :** $z-w = (x-u) + i.(y-v)$

- **Produit par scalaire :** $\lambda.z = \lambda.x + i.(\lambda.y)$

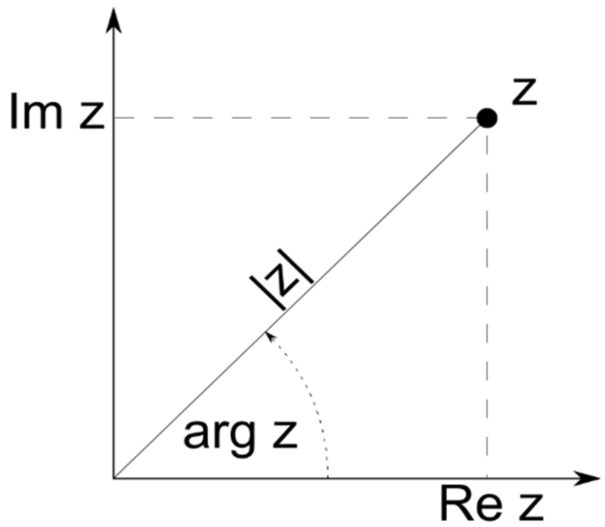
- **Multiplication :** $z.w = xu + iyu + ixv + i^2yv$
- $= (x.u-y.v) + i.(x.v+y.u)$

Nombres complexes : forme exponentielle

Tout nombre complexe *z* peut s'écrire sous une forme trigonométrique

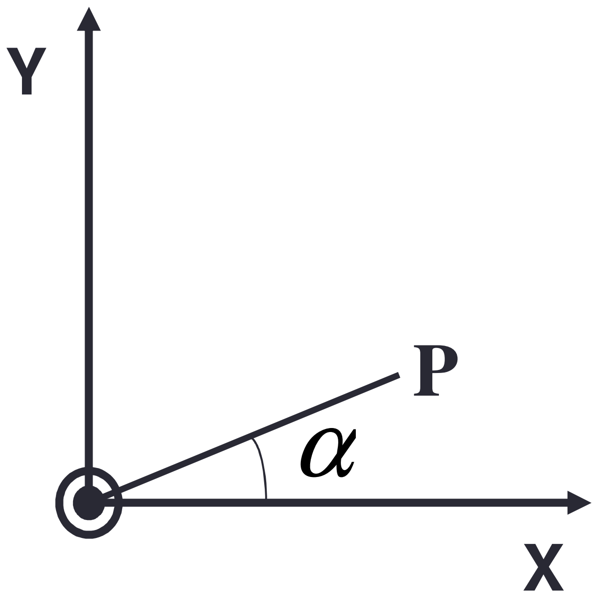
$\boxed{z = r.(\cos(\theta) + i.\sin(\theta))}$ avec $r \geq 0$, forme polaire

$\boxed{z = r.e^{i\theta}}$ forme exponentielle



Rotation d’un angle θ autour de l’origine O

- Soit le point P avec pour image le complexe z

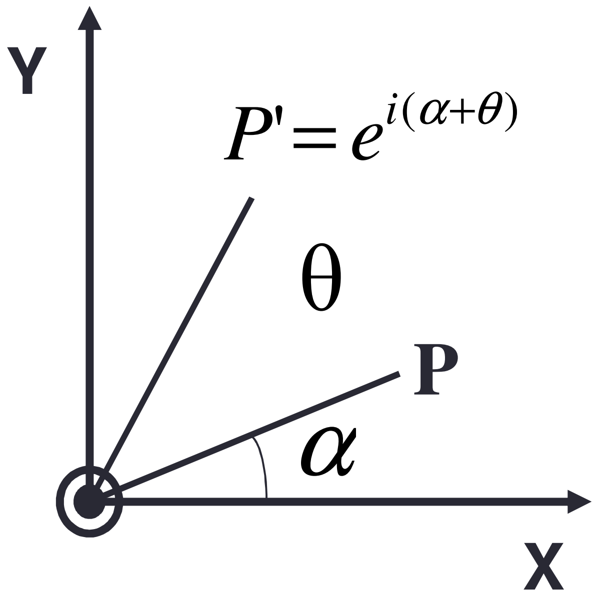


$$P = \begin{pmatrix} x & y \end{pmatrix}$$
$$= \begin{pmatrix} r.\cos(\alpha) & r.\sin(\alpha) \end{pmatrix}$$

$$z = r.(\cos(\alpha) + i\sin(\alpha))$$
$$= r.e^{i\alpha}$$

Rotation d’un angle θ autour de l’origine O

- Soit le point P avec pour image le complexe z
- Soit le complexe R=$e^{i\theta}$
- P’ le point résultat de la rotation de P autour de l’origine O d’un angle θ

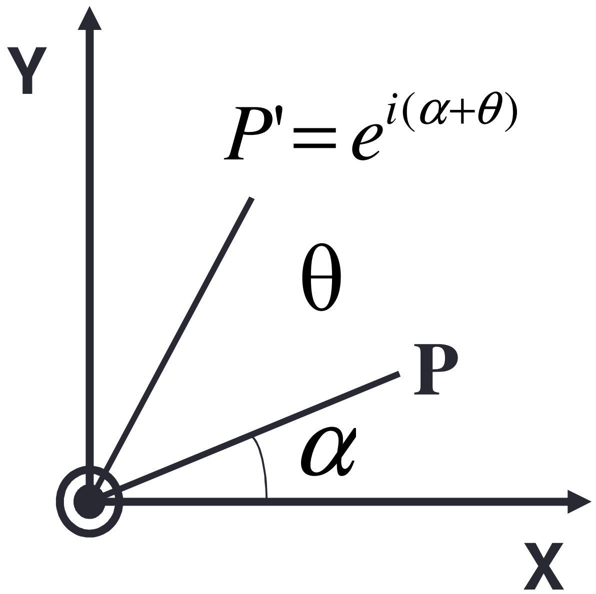


$$P' = r.e^{i.(\alpha+\theta)}$$
$$= r.e^{i\alpha}.e^{i\theta}$$
$$= P.R$$

Interprétations géométriques

Multiplier un complexe z (image du point P) par le complexe $e^{i\theta}$ de module 1 équivaut à faire tourner P(z) d'un angle θ autour de l'origine O

➔ **Propriété très sympa !**



$$P' = r.e^{i.(\alpha+\theta)}$$

$$= r.e^{i\alpha}.e^{i\theta}$$

$$= P.R$$

Quaternions

- Comment construire un type de nombre permettant d'obtenir des résultats similaires en 3D?

Quaternions

- Comment construire un type de nombre permettant d'obtenir des résultats similaires en 3D?
- Un triplet de nombres réels ne suffit pas, il faut un quadruplet!
  - Un scalaire
  - Et un vecteur imaginaire pur
- Forme d'un quaternion

  $a+b\mathbf{i}+c\mathbf{j}+d\mathbf{k}$
- Propriétés

  $\mathbf{i}^2=\mathbf{j}^2=\mathbf{k}^2=\mathbf{ijk}=-1$

Quaternions

- Attention le produit entre quaternion n'est pas toujours commutatif!

.	**1**	**i**	**j**	**k**
1	1	i	j	k
i	i	-1	k	-j
j	j	-k	-1	i
k	k	j	-i	-1

- Cohérent avec le fait que les rotations ne sont pas commutatives!

Mulitplication de 2 quaternions

$Q=a+b\mathbf{i}+c\mathbf{j}+d\mathbf{k}$

$Q'=a'+b'\mathbf{i}+c'\mathbf{j}+d'\mathbf{k}$

$$Q.Q'=(aa'-(bb'+cc'+dd'))$$
$$+(ab'+ba')\mathbf{i}+(ac'+ca')\mathbf{j}+(ad'+da')\mathbf{k}$$
$$+(cd'-c'd)\mathbf{i}+(db'-d'b)\mathbf{j}+(bc'-cb')\mathbf{k}$$

Si on note $Q=a+\mathbf{V}$ et $Q'=a'+\mathbf{V'}$

On obtient

$$Q.Q'=aa'-V\cdot V'$$
$$+(ab'+ba')\mathbf{i}+(ac'+ca')\mathbf{j}+(ad'+da')\mathbf{k}+$$
$$+\mathbf{V}\wedge\mathbf{V'}$$

où on retrouve les produits scalaire et vectoriels des 2 parties vectorielles!

Conjugué d'un quaternion

$Q = a+b\mathbf{i}+c\mathbf{j}+d\mathbf{k} = a+\mathbf{V}$

Conjugué $Q^*=a-b\mathbf{i}-c\mathbf{j}-d\mathbf{k}=a-\mathbf{V}$

- Permet d'obtenir la norme d'un quaternion par $||Q||^2=Q.Q^*=a^2+\mathbf{V}^2$
- Les quaternions de norme 1 sont dits unitaires

Inverse d'un quaternion

- Inverse d'un quaternion
  $Q^{-1}=Q^{*} / ||Q||^{2}$

- Propriétés
  $Q^{**}=Q$
  $(Q+Q')^{*}=Q^{*}+Q'^{*}$
  $(Q.Q')^{*}=Q'^{*}.Q^{*}$
  $(Q.Q')^{-1}=Q'^{-1}.\ Q^{-1}$

Rotation

- C'est bien beau toutes ces formules mathématiques, mais quel rapport avec les rotations 3D et plus généralement avec les similitudes?

- Etant donné un quaternion Q=a+**V** de norme q

- On considère l'angle Φ tel que a/q=cos Φ et ||**V**||/q=sin Φ

  Donc si le quaternion est normalisé : a=cos Φ et ||V||=sin Φ

- Alors Q=q(cosΦ+sinΦ**N**) où **N** est le vecteur unitaire **V**/q

Rotation

- La rotation d'un vecteur **U** d'angle 2Φ autour de l'axe porté par **N** s'exprime par le produit de quaternions suivants :

  U' =(cosΦ+sinΦ**N**).(0+**U**). (cosΦ-sinΦ**N**)

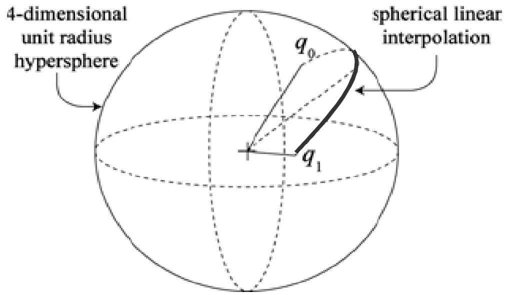
  =(cosΦ+sinΦ**N**).(0+**U**). (cosΦ+sinΦ**N**)*
- On dit que le quaternion cos Φ+sin Φ**N** représente la rotation R[**N**,2Φ] axe N et angle 2Φ
- Pratique pour combiner les rotations
  - Si on applique la rotation R1 de quaternion Q1 puis R2 de quaternion Q2 on obtient une rotation représentée par Q2.Q1

- Donc le quaternion
  - (a,b,c,d)=(cosΦ, N_x.sinΦ, N_y.sinΦ, N_z.sinΦ)

  =rotation d'axe N et d'angle 2Φ

Spherical Linear Interpolation (SLERP)

- On voudrait que l'interpolation produise un arc au lieu d'une ligne



- Rappel de l'interpolation :parameter t entre 0 et 1
- La formule d'interpolation entre le quaternion q et r et le param t

  $s(t)= (rq^{-1})^t q$

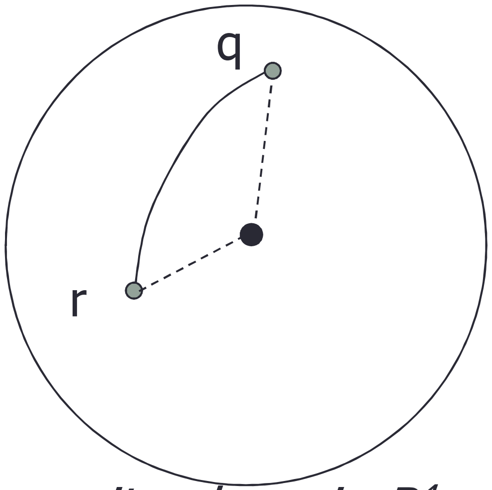
avec t=0, on a rotation=s(0)=q
avec t=1, on a rotation=s(1)=r

- Le problème de la puissance

$$\text{if } q = [\cos\theta, \boldsymbol{u}\sin\theta] \text{ then } q^t = [\cos t\theta, \boldsymbol{u}\sin t\theta]$$

Spherical Linear intERPolation (SLERP)

$$s(t) = \frac{\sin \varphi\,(1-t)}{\sin \varphi}\,q + \frac{\sin \varphi\, t}{\sin \varphi}\,r$$



unit sphere in R^4

Avec t allant de 0 à 1, le quaternion rotation s(t) tourne autour d'un axe fixe à vitesse constante

References

http://www.sjbrown.co.uk/quaternions.html
http://www.theory.org/software/qfa/writeup/node12.html

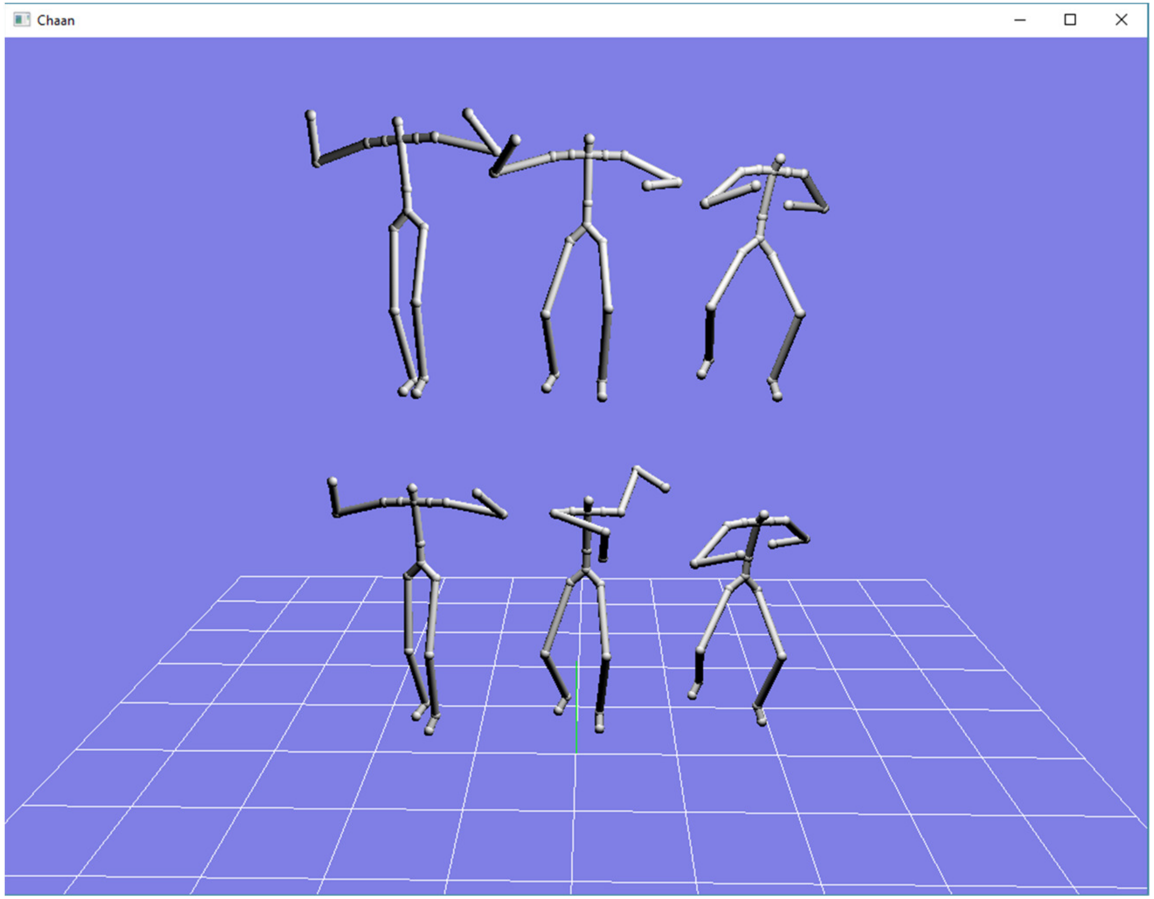
Spherical Linear Interpolation

$$slerp(t; q_0, q_1) = \frac{\sin((1-t)\theta) q_0 + \sin(t\theta) q_1}{\sin\theta}$$

- q_0 et q_1 sont unitaires
- θ = angle entre q_0 et q_1
- t : coefficient d'interpolation entre [0, 1]
- Si q_0 *et* q_1 *sont identiques, alors* $\theta = 0$*, mais dans ce cas,* $q(t) = q_0$ *pour toutes valeurs de* t

Spherical Linear Interpolation

- Voir la démo du TP
  - interpolation entre animation de gauche et de droite
  - En bas interpolation des angles : problème avec le bras droit
  - En haut interpolation avec les quaternion



Coût mémoire

- Espace mémoire nécessaire
  - Matrice de rotation 9
  - Quaternion 4
  - Axe et angle 4*

  * Note : la représentation sous forme d'angle et d'axe peut être stockée dans 3 emplacements seulement en multipliant l'axe de rotation par l'angle de rotation ; néanmoins, avant de l'utiliser, il faut récupérer le vecteur unitaire et l'angle en renormalisant, ce qui coûte des opérations mathématiques supplémentaires.

source http://fr.wikipedia.org/wiki/Quaternions_et_rotation_dans_l'espace

Performance

- Comparaison de performances de la composition de rotations

Méthode	Multiplications	Additions et soustractions	Nombre total d'opérations
Matrices de rotation	27	18	45
Quaternions	16	12	28

Comparaison de performances de la rotation de vecteurs

Méthode	Multiplications	Additions et soustractions	sin et cos	Nombre total d'opérations
Matrice de rotation	9	6	0	15
Quaternions	21	18	0	39
Axe et angle	23	16	2	41

Pourquoi utiliser les quaternions?

- Les opérations sont numériquement
  - plus stables que sur les matrices
  - Plus rapide que avec les matrices

- Evite Gimbam Lock

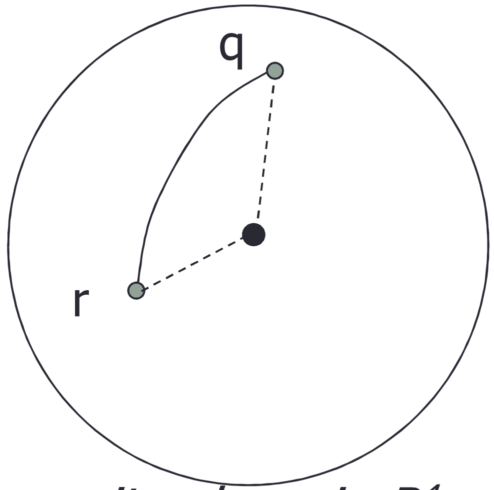
- Utiles pour faire de l'interpolation entre 2 rotations ☺

Spherical Linear intERPolation (SLERP)

$$s(t) = \frac{\sin\phi(1-t)}{\sin\phi} q + \frac{\sin\phi t}{\sin\phi} r$$

$$\cos\phi \equiv q_0 r_0 + q_1 r_1 + q_2 r_2 + q_3 r_3$$

quaternion



unit sphere in R^4

Avec t allant de 0 à 1, le quaternion rotation s(t) tourne autour d’un axe fixe à vitesse constante

References

http://www.sjbrown.co.uk/quaternions.html
http://www.theory.org/software/qfa/writeup/node12.html