

Outils Mathématiques: Position, Orientation et Mouvement

DEA IVR

François Faure¹ , UJF-Grenoble

30 octobre 2002

¹<http://www-imagis.imag.fr/Membres/Francois.Faure/faure.html>

Ce cours est une introduction aux techniques permettant de définir les mouvement de points et solides dans l'espace. Il se compose de quatre séances d'1h30 :

1. rappels mathématiques (algèbre linéaire, géométrie euclidienne, matrices)
2. positionnement (repères, rotations, chaînes cinématiques)
3. cinématique (vitesses et accélérations, lois de composition)
4. cinématique inverse (contrôle de mouvement)

Table des matières

1	Rappels d'algèbre linéaire	4
1.1	Structures	4
1.1.1	Groupes	4
1.1.2	Espaces vectoriels	4
1.1.3	Sous-espaces vectoriels	4
1.1.4	Systèmes de vecteurs, bases	5
1.1.5	Espaces affines	5
1.2	Espaces euclidiens	6
1.2.1	Produit scalaire	6
1.2.2	Normes	6
1.2.3	Orthogonalité et orthonormalité	6
1.2.4	Produit vectoriel	7
1.2.5	Produit mixte	7
1.3	Applications linéaires et affines	7
1.3.1	Application linéaire	7
1.3.2	Expression matricielle d'une application linéaire	8
1.3.3	Changement de base d'un vecteur	9
1.3.4	Changement de base d'une application linéaire	9
1.3.5	Application affine	10
1.3.6	Changement de repère d'un point	10
1.3.7	Changement de repère d'une application affine	10
1.4	Matrices	11
1.4.1	Généralités	11
1.4.2	Formes particulières	11
1.4.3	Produit de matrices	11
1.4.4	Transposée	11
1.4.5	Trace	12
1.4.6	Matrice inverse	12
1.4.7	Matrice orthogonale	12
1.4.8	Déterminant	12
1.4.9	Valeurs propres et vecteurs propres	13
1.4.10	Résolution de systèmes d'équations linéaires	13
2	Positionnement	15
2.1	Rotations	15
2.1.1	Propriétés	15
2.1.2	Représentation matricielle	15

2.1.3	Décomposition de Jordan	15
2.1.4	Conversion entre matrice et (angle,axe). Formule de Rodrigues.	16
2.1.5	Composition des rotations. Angles d'Euler.	16
2.1.6	Quaternions	17
2.1.7	Interpolation linéaire des rotations	17
2.1.8	Interpolation lisse des rotations	18
2.2	Transformations affines	18
3	Cinématique	19
3.1	Rappels	19
3.1.1	Dérivée dans \mathcal{R}_0 d'un vecteur fixe dans \mathcal{R}_1 . Vecteur rotation.	19
3.1.2	Vitesse dans \mathcal{R}_0 d'un point fixe dans \mathcal{R}_1 . Champ des vitesses.	19
3.1.3	Accélération dans \mathcal{R}_0 d'un point fixe dans \mathcal{R}_1 . Champ des accélérations.	20
3.1.4	Dérivée dans \mathcal{R}_0 d'un vecteur non fixe dans \mathcal{R}_1	20
3.1.5	Vitesse dans \mathcal{R}_0 d'un point mobile dans \mathcal{R}_1	20
3.1.6	Accélération dans \mathcal{R}_0 d'un point mobile dans \mathcal{R}_1 . Accélération de Coriolis.	20
3.2	Cinématique des chaînes articulées	21
3.2.1	Composition des vitesses	21
3.2.2	Liaisons cinématiques. Degrés de liberté.	21
3.2.3	Paramétrage des liaisons. Espace articulaire.	23
3.2.4	Algorithmes de cinématique directe	24
4	Cinématique inverse	26
4.1	Équations linéaires	26
4.2	Équation non linéaires	28
4.2.1	Linéarisation. Matrice Jacobienne.	29
4.2.2	Contraintes d'orientation	30
4.2.3	Orientation d'un vecteur	30
4.3	Résolution des équations linéaires	30
4.3.1	Matrice carrée	31
4.3.2	Plus d'inconnues que d'équations	31
4.3.3	Plus d'équations que d'inconnues	31
4.3.4	Singularités	31
4.3.5	Décomposition en valeurs singulières (SVD)	32
4.4	Résolution itérative des équations non linéaires	32
4.5	Utilisation de l'espace libre	33
4.5.1	Projection sur le noyau de \mathbf{J}	33
4.5.2	Optimisation de poses	33
A	Liaisons cinématiques	34
B	Glossaire français-anglais	38

Chapitre 1

Rappels d'algèbre linéaire

1.1 Structures

1.1.1 Groupes

Définition	Exemple
Ensemble	\mathbb{Z}
muni d'une loi de composition interne	$+$
associative	$a + (b + c) = (a + b) + c$
munie d'un élément neutre	$a + 0 = a$
tout élément a un symétrique	$a + (-a) = 0$
Si l'opération est commutative ($a + b = b + a$) alors le groupe est dit commutatif.	

1.1.2 Espaces vectoriels

Définition	Exemple
\vec{E} est un espace vectoriel (EV) sur K si	\mathbb{R}^3 EV sur \mathbb{R}
\vec{E} a une opération interne qui lui confère une structure de groupe commutatif	$\mathbb{R}^3, +$
\vec{E} possède une opération externe tq :	multiplication scalaire par vecteur
$\lambda \cdot (\mu \cdot x) = (\lambda \mu) \cdot x$	
$(\lambda + \mu) \cdot x = \lambda \cdot x + \mu \cdot x$	
$\lambda \cdot (x + y) = \lambda \cdot x + \lambda \cdot y$	
$1 \cdot x = x$ (élément neutre)	

1.1.3 Sous-espaces vectoriels

combinaison linéaire: Une combinaison linéaire d'une partie A de \vec{E} est : $\sum_i \lambda_i a_i$, avec $a_i \in A$

sous-espace vectoriel (SEV): Un sous-espace vectoriel de \vec{E} est une sous-ensemble de \vec{E} stable par combinaison linéaire.

Exemple : une droite vectorielle est un SEV de \mathbb{R}^3 .

SEV engendré par une partie A de \vec{E} : L'ensemble des combinaisons linéaires des vecteurs de A.

Exemple : plan vectoriel engendré par deux vecteurs non colinéaires.

somme de SEV \vec{E}_1 et \vec{E}_2 de \vec{E} : $\vec{E}_1 + \vec{E}_2 = \{x + y; x \in \vec{E}_1, y \in \vec{E}_2\}$. La somme de deux SEV est un SEV. L'intersection aussi. Si l'intersection de deux SEV est nulle, la somme est dite directe et notée $\vec{E}_1 \oplus \vec{E}_2$

Exemple dans \mathbb{R}^3 : la somme de deux droites vectorielles distinctes est directe et c'est un plan vectoriel.

1.1.4 Systèmes de vecteurs, bases

système de vecteurs: ensemble $\{x_i, i \in I\}$ où I est un ensemble fini de cardinal p , appelé *ordre du système*

système linéairement indépendant: système de vecteurs dans lequel $\sum_i^p \lambda_i x_i = 0 \Rightarrow \forall i \lambda_i = 0$

système lié: système non linéairement indépendant

base d'un EV: système libre $\{e_i, i \in I\}$ de vecteurs de \vec{E} tel que pour tout élément x de \vec{E} il existe une décomposition unique $x = \sum_{i \in I} \lambda_i e_i$

base orthonormée: Base dont tous les vecteurs sont orthogonaux deux à deux (orthogonalité définie en section 1.2.3).

base orthonormée directe: En dimension 3, base orthonormée dont les vecteurs sont orientés comme le pouce, l'index et le majeur de la main droite. Par extension, base construite (ou constructible) par produits vectoriels successifs. (produit vectoriel défini en section 1.2.4).

rang d'un système: Le rang d'un système vaut r si :

- il existe un système libre extrait de r vecteurs
- tout système extrait de $r + 1$ vecteurs est lié

EV de dimension finie: EV qui admet une base de rang fini. Toutes les bases d'un même EV sont de même rang, ce rang est appelé dimension de l'espace.

composantes: Si n est la dimension (finie) d'un EV et $B = \{e_i, i \in [1..n]\}$ une base, on peut définir un vecteur $x = \sum_{i=1}^n x_i e_i$ par ses composantes dans la base : ${}^B x = (x_1, \dots, x_n)^T$

codimension: Si \vec{E}_1 et \vec{E}_2 sont deux SEV de \vec{E} tels que $\vec{E}_1 \oplus \vec{E}_2 = \vec{E}$ alors la dimension de \vec{E}_2 est appelée codimension de \vec{E}_1 , et réciproquement.

Exemple : dans \mathbb{R}^3 , la codimension d'une droite est 2, la codimension d'un plan est 1.

hyperplan: SEV de codimension 1

1.1.5 Espaces affines

Soit \vec{E} un espace vectoriel. Un ensemble E est un espace affine (EA) associé à \vec{E} s'il existe une application φ telle que

- $\forall (M, N, P) \in E^3 \varphi(M, N) + \varphi(N, P) = \varphi(M, P)$ (Chasles)
- $\forall M \in E \forall v \in \vec{E} \exists N \in E : \varphi(M, N) = v$
- $\forall (M, N) \in E^2 \varphi(M, N) = 0 \Rightarrow M = N$

On note $\varphi(M, N) \overrightarrow{MN}$.

sous-espace affine: F est un sous-espace affine de E si F est un espace affine associé à un SEV \vec{F} de \vec{E} . On dit que \vec{F} est la direction de F .

SEA parallèles: SEA ayant la même direction.

Propriétés :

- l'intersection de deux SEA est un SEA
- par un point donné il passe un unique SEA de direction donnée.

Repère: Couple (O, B) où O est un point de E et B une base de \vec{E} . Pour tout point M de E , $\vec{OM} \in \vec{E}$.

1.2 Espaces euclidiens

Un espace est dit euclidien s'il est muni d'un produit scalaire.

1.2.1 Produit scalaire

Forme bilinéaire, symétrique, définie, positive :

- $\langle x, y \rangle \in \mathbb{R}$
- $\langle \alpha x_1 + \beta x_2, y \rangle = \alpha \langle x_1, y \rangle + \beta \langle x_2, y \rangle$
- $\langle x, \alpha y_1 + \beta y_2 \rangle = \alpha \langle x, y_1 \rangle + \beta \langle x, y_2 \rangle$
- $\langle x, y \rangle = \langle y, x \rangle$
- $\langle x, x \rangle = 0 \Rightarrow x = \mathbf{0}$
- $\langle x, x \rangle \geq 0$

En dimension finie on peut le définir par une matrice A tq : $\langle x, y \rangle = x^T A y$.

1.2.2 Normes

norme euclidienne (ou norme indice 2): $\|x\| = \sqrt{\langle x, x \rangle}$

Propriétés :

- $|\langle x, y \rangle| \leq \|x\| \|y\|$
- $\|x + y\| \leq \|x\| + \|y\|$ (Minkovski)

Autres normes :

- norme indice 1 : $\|x\|_1 = \sum_i |x_i|$ ("distance de Manhattan")
- norme infinie : $\|x\|_{\text{inf}} = \max_i |x_i|$

1.2.3 Orthogonalité et orthonormalité

orthogonalité: $x \perp y \Leftrightarrow \langle x, y \rangle = 0$

système orthogonal: vecteurs orthogonaux 2 à 2.

vecteur unitaire: vecteur de norme 1.

système orthonormal: système orthogonal de vecteurs unitaires.

base orthonormale: base dont les vecteurs forment un système orthonormal. Dans une base orthonormale, la matrice du produit scalaire est l'identité.

base canonique: dans un espace euclidien, base constituée des vecteurs $(1, 0, 0 \dots)$, $(0, 1, 0, 0 \dots)$ etc.

SEV orthogonaux: \vec{E}_1 et \vec{E}_2 sont des SEV orthogonaux si tout élément de l'un est orthogonal à tout élément de l'autre.

SEA orthogonaux: SEA dont les SEV associés sont orthogonaux.

1.2.4 Produit vectoriel

Soient \mathbf{u}, \mathbf{v} deux vecteurs en dimension 3. Le produit vectoriel $\mathbf{w} = \mathbf{u} \wedge \mathbf{v}$ se calcule par :

$$w_i = u_{i+1 \bmod 3} v_{i+2 \bmod 3} - u_{i+2 \bmod 3} v_{i+1 \bmod 3}$$

Propriétés :

- si les vecteurs sont liés le produit vectoriel est nul
- $\mathbf{u} \wedge \mathbf{v} = -\mathbf{v} \wedge \mathbf{u}$
- $\mathbf{u} \wedge \mathbf{v} \perp \mathbf{u}$
- $\mathbf{u} \wedge \mathbf{v} \perp \mathbf{v}$
- $\|\mathbf{u}\| = 1 \quad \|\mathbf{v}\| = 1 \quad \mathbf{u} \perp \mathbf{v} \Rightarrow (\mathbf{u}, \mathbf{v}, \mathbf{u} \wedge \mathbf{v})$ base orthonormée directe
- $\langle \mathbf{u}, \mathbf{v} \rangle^2 + \|\mathbf{u} \wedge \mathbf{v}\|^2 = \|\mathbf{u}\|^2 \|\mathbf{v}\|^2$
- $\mathbf{u} \wedge (\mathbf{v} \wedge \mathbf{w}) = (\mathbf{u}\mathbf{w})\mathbf{v} - (\mathbf{u}\mathbf{v})\mathbf{w}$

Angles entre vecteurs :

$$\begin{aligned} \langle \mathbf{u}, \mathbf{v} \rangle &= \|\mathbf{u}\| \|\mathbf{v}\| \cos(\theta) \\ \|\mathbf{u} \wedge \mathbf{v}\| &= \|\mathbf{u}\| \|\mathbf{v}\| \sin(\theta) \end{aligned}$$

Forme matricielle : Pour tout vecteur $\mathbf{n} = (n_x, n_y, n_z)^T$ et tout vecteur \mathbf{u} on a :

$$\mathbf{n} \wedge \mathbf{u} = (\mathbf{n} \wedge) \mathbf{u} \text{ avec } (\mathbf{n} \wedge) = \begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix} \quad (1.1)$$

1.2.5 Produit mixte

En dimension 3 le produit mixte de trois vecteurs $\mathbf{u}, \mathbf{v}, \mathbf{w}$ est défini par :

$$(u, v, w) = (u \wedge v)w$$

Propriétés :

- le produit mixte est nul si les vecteurs sont liés
- le produit mixte est invariant par permutation circulaire : $(u \wedge v)w = (w \wedge u)v$

1.3 Applications linéaires et affines

1.3.1 Application linéaire

Application linéaire (AL): Soient \vec{E} et \vec{F} deux espaces vectoriels. Une application $f : \vec{E} \rightarrow \vec{F}$ est dite linéaire si elle vérifie :

- $f(x + y) = f(x) + f(y)$
- $f(\lambda x) = \lambda f(x)$

endomorphisme: AL de \vec{E} dans \vec{E}

composition: la composée $g \circ f$ de deux applications linéaires est l'application qui à x associe $g(f(x))$. La composée de deux AL est une AL.

image d'un EV par une AL: L'image $Im_f = f(\vec{E})$ d'un EV \vec{E} est un sous-espace vectoriel formé des transformées des éléments de \vec{E} .

Exemple : dans \mathbb{R}^3 , une projection sur un plan a pour image ce plan.

image réciproque: L'image réciproque d'un SEV de \vec{F} par une AL de \vec{E} dans \vec{F} est l'ensemble des éléments de \vec{E} se transformant en éléments de \vec{F} . C'est un SEV de \vec{E} .

noyau d'une AL: Le noyau d'une application f de \vec{E} dans \vec{F} , noté Ker_f , est l'image réciproque de l'élément nul de \vec{F} Exemple : dans \mathbb{R}^3 , le noyau d'une projection sur un plan est la droite dans la direction de projection.

En dimension finie, $dim(Im_f) + dim(Ker_f) = dim(\vec{E})$

Exemple : Dans un EV de dimension n , projection f sur une droite. $dim(Im_f) = 1$, $dim(Ker_f) = n - 1$

isométrie vectorielle: application linéaire qui conserve le produit scalaire. Transforme une base orthonormale en une base orthonormale.

1.3.2 Expression matricielle d'une application linéaire

Notons :

$$g : \vec{E}_m \longrightarrow \vec{E}_n$$

$$\mathbf{x} \longmapsto \mathbf{x}'$$

une AL g d'un espace \vec{E}_m de dimension m vers un espace \vec{E}_n de dimension n . Soient les bases :

- $B = \{\mathbf{e}_j, j \in [1..m]\}$ une base de \vec{E}_m telle que $\mathbf{x} = \sum_{j=1}^m x_j \mathbf{e}_j$, et

- $B' = \{\mathbf{f}_i, i \in [1..n]\}$ une base de \vec{E}_n telle que $\mathbf{x}' = \sum_{i=1}^n x'_i \mathbf{f}_i$

La transformée d'un vecteur de B peut s'écrire : $g(\mathbf{e}_j) = \sum_{i=1}^n a_{ij} \mathbf{f}_i$. On peut donc formuler la transformation de \mathbf{x} par :

$$g(\mathbf{x}) = \sum_{j=1}^m x_j g(\mathbf{e}_j)$$

$$= \sum_{j=1}^m x_j \sum_{i=1}^n a_{ij} \mathbf{f}_i$$

$$= \sum_{i=1}^n \sum_{j=1}^m a_{ij} x_j \mathbf{f}_i$$

Donc pour chaque composante de \mathbf{x}' on a $x'_i = \sum_{j=1}^m a_{ij} x_j$. Nous pouvons ainsi décrire l'application g sous forme d'une matrice \mathbf{G} telle que $\mathbf{x}' = \mathbf{G}\mathbf{x}$:

$$\begin{pmatrix} a_{11} & \dots & a_{1m} \\ \vdots & & \\ a_{i1} & \dots & a_{ij} & \dots & a_{im} \\ \vdots & & & & \\ a_{n1} & \dots & a_{nm} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \\ x'_1 \\ \vdots \\ x'_i \\ \vdots \\ x'_n \end{pmatrix}$$

Les colonnes de la matrice \mathbf{G} sont les transformés des vecteurs de base de l'ensemble de départ par l'application.

1.3.3 Changement de base d'un vecteur

Un changement de base est une application linéaire qui aux composantes d'un vecteur dans une certaine base associe les composantes de ce vecteur dans une autre base du même espace. Soient les bases :

- $B = \{e_j, j \in [1..n]\}$ une base de \vec{E} telle que ${}^B \mathbf{x} = \sum_{j=1}^n x_j e_j$, et

- $B' = \{f_i, i \in [1..n]\}$ une base de \vec{E} telle que ${}^{B'} \mathbf{x} = \sum_{i=1}^n x'_i f_i$

Connaissant ${}^B \mathbf{x}$ on cherche à déterminer ${}^{B'} \mathbf{x}$. Sachant exprimer les vecteurs de B dans B' : $e_j = \sum_{i=1}^n p_{ij} f_i$, nous pouvons écrire :

$$\begin{aligned} \mathbf{x} &= \sum_{j=1}^n x_j e_j \\ &= \sum_{j=1}^n x_j \sum_{i=1}^n p_{ij} f_i \\ &= \sum_{i=1}^n \sum_{j=1}^n p_{ij} x_j f_i \end{aligned}$$

d'où $x'_i = \sum_{j=1}^n p_{ij} x_j$ Nous pouvons ainsi décrire le passage de B à B' sous forme d'une matrice ${}^B \mathbf{R}$ telle que ${}^{B'} \mathbf{x} = {}^B \mathbf{R} {}^B \mathbf{x}$:

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \\ a_{i1} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & & & & \\ a_{n1} & \dots & & & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \\ x'_1 \\ \vdots \\ x'_n \end{pmatrix}$$

Les colonnes de la matrice ${}^B \mathbf{R}$ sont les vecteurs de la base B exprimés dans la base B'.

1.3.4 Changement de base d'une application linéaire

Notons :

$$\begin{aligned} g : \vec{E}_m &\longrightarrow \vec{E}_n \\ \mathbf{x} &\longmapsto \mathbf{y} \end{aligned}$$

une AL g d'un espace \vec{E}_m de dimension m vers un espace \vec{E}_n de dimension n . Soient

- B_m et B'_m deux bases de \vec{E}_m

- B_n et B'_n deux bases de \vec{E}_n

- ${}^{B_n} \mathbf{G}$ la matrice de g transformant un vecteur exprimé dans B_m en un vecteur exprimé dans B_n , c'est-à-dire ${}^{B_n} \mathbf{y} = {}^{B_n} \mathbf{G} {}^{B_m} \mathbf{x}$

Nous cherchons la **matrice équivalente** ${}_{B'_m}^{B'_n} \mathbf{G}'$ transformant un vecteur ${}^{B'_m} \mathbf{x}'$ exprimé dans B'_m en un vecteur ${}^{B'_n} \mathbf{y}'$ exprimé dans B'_n . Le principe est de ramener la donnée dans B_m , appliquer g , puis amener le résultat dans la base souhaitée. On peut ainsi écrire :

$${}^{B'_n} \mathbf{y}' = \underbrace{{}_{B'_n} \mathbf{R} \underbrace{{}_{B'_m} \mathbf{G} \underbrace{{}^{B'_m} \mathbf{R} \mathbf{x}'}}_{B_m \mathbf{x}}}}_{B_n \mathbf{y}}$$

et donc ${}_{B'_m}^{B'_n} \mathbf{G}' = {}_{B'_n} \mathbf{R} {}_{B'_m}^{B_n} \mathbf{G} {}_{B'_m}^{B_m} \mathbf{R}$

1.3.5 Application affine

Soit E un EA associé à un EV \vec{E} . Une application f de E dans E est une application affine (AA) s'il existe un endomorphisme φ de \vec{E} tel que pour tout couple (M, N) de E on ait $f(M)f(N) = \varphi(MN)$

Propriétés :

- une AA est déterminée par la donnée d'un point A , de son transformé A' , et de l'endomorphisme associé. L'image M' d'un point M est alors donnée par $\overrightarrow{AM'} = \overrightarrow{AA'} + \varphi(\overrightarrow{AM})$
- f étant donné, φ est unique
- l'image d'un SEA par une AA est un SEA
- la **composée de deux AA est une AA**. Son endomorphisme associé est la composée des endomorphismes associés.
- l'ensemble des points invariants par une AA est un SEA
- toute AA conserve le barycentre
- **deux applications affines sont égales** \Leftrightarrow leurs endomorphismes associés sont égaux et il existe un point qui a le même transformé par les deux applications

1.3.6 Changement de repère d'un point

Soit un EA E de dimension n et les repères :

- (O, B) avec B une base de \vec{E}
- (O', B') avec B' une autre base de \vec{E}

Étant donné (x_1, \dots, x_n) les composantes d'un point M dans le repère (O, B) , nous obtenons les composantes (x'_1, \dots, x'_n) de M dans (O', B') :

$$\begin{aligned} \mathbf{O}'M &= \mathbf{O}'\mathbf{O} + \mathbf{O}M \\ &= \mathbf{O}M - \mathbf{O}\mathbf{O}' \\ {}^{B'}\mathbf{O}'M &= {}^{B'}\mathbf{O}'\mathbf{O} + {}^{B'}\mathbf{O}M \\ &= \frac{{}^{B'}}{B} \mathbf{R}^B \mathbf{O}'\mathbf{O} + \frac{{}^{B'}}{B} \mathbf{R}^B \mathbf{O}M \\ &= \frac{{}^{B'}}{B} \mathbf{R} ({}^B \mathbf{O}M - {}^B \mathbf{O}\mathbf{O}') \end{aligned}$$

1.3.7 Changement de repère d'une application affine

Soit une application affine définie pour un repère (O', B') :

$$f : {}^{B'}\mathbf{O}'M \mapsto {}^{B'}\mathbf{O}'P = {}^{B'}\mathbf{O}'\mathbf{O}'_f + \frac{{}^{B'}}{B'} \mathbf{F}^{B'} \mathbf{O}'M$$

Nous désirons calculer la transformée P d'un point M , tous deux définis par leurs coordonnées dans un repère (O, B) . Pour cela nous passons M dans (O', B') , appliquons la transformation puis ramenons le résultat dans (O, B) :

$${}^BOP = {}^B_B R ({}^{B'}O'O'_f + {}^{B'}_B F_B {}^B R ({}^B OM - {}^B OO') - {}^{B'}O'O)$$

1.4 Matrices

1.4.1 Généralités

Une matrice $n \times m$ est un tableau de n lignes et m colonnes. Une telle matrice peut servir notamment à représenter :

- une application linéaire de \vec{E}_m dans \vec{E}_n
- les coefficients d'un système de n équations linéaires à m inconnues

On notera a_{ij} la valeur d'une matrice \mathbf{A} en ligne i , colonne j .

1.4.2 Formes particulières

matrice carrée: $\mathbf{A}_{n \times m}$ est dite carrée si $m = n$. Dans une matrice carrée, les éléments de (1,1) à (n,n) forment la *diagonale principale*.

matrice identité: matrice carrée dont les éléments de la diagonale principale valent tous 1, et tous les autres éléments, 0. En dimension n , on la note \mathbf{I}_n Pour toute matrice $\mathbf{A}_{n \times n}$, $\mathbf{A}\mathbf{I}_n = \mathbf{I}_n\mathbf{A} = \mathbf{A}$

matrice triangulaire: Une matrice est dite triangulaire supérieure (resp. inférieure) si tous les éléments en-dessous (resp. au-dessus) de sa diagonale principale sont nuls.

matrice symétrique: Matrice égale à sa transposée

matrice symétrique définie positive: Une matrice symétrique \mathbf{A} telle que pour tout vecteur \mathbf{x} non nul, $\mathbf{x}^T \mathbf{A} \mathbf{x} > 0$

matrice symétrique semi-définie positive: Une matrice symétrique \mathbf{A} telle que pour tout vecteur \mathbf{x} non nul, $\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0$

matrice singulière: Une matrice dont les lignes ou les colonnes forment un système lié.

1.4.3 Produit de matrices

$$\mathbf{A}_{n \times m} \mathbf{B}_{m \times p} = \mathbf{C}_{n \times p}$$

avec $c_{ij} = \sum_{k=1}^m a_{ik} b_{kj}$.

Si \mathbf{F} est la matrice associée à une application linéaire f et \mathbf{G} la matrice associée à une application linéaire g , alors $\mathbf{F}\mathbf{G}$ est la matrice associée à l'application linéaire $f \circ g$.

Les vecteurs en dimension n sont considérés comme des matrices à n lignes et une colonne.

1.4.4 Transposée

La transposée $\mathbf{A}'_{m \times n}$ d'une matrice $\mathbf{A}_{n \times m}$ est une matrice notée \mathbf{A}^T telle que $a'_{ij} = a_{ji}$. Transposer une matrice revient à permuter les lignes avec les colonnes.

Propriétés :

- $(\mathbf{A} + \mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T$
- $(\mathbf{A}\mathbf{B})^T = \mathbf{B}^T \mathbf{A}^T$

1.4.5 Trace

La trace d'une matrice carrée est la somme des éléments de sa diagonale.

Propriétés :

- $tr(\mathbf{A} + \mathbf{B}) = tr(\mathbf{A}) + tr(\mathbf{B})$
- $tr(\lambda\mathbf{A}) = \lambda tr(\mathbf{A})$
- $tr(\mathbf{AB}) = tr(\mathbf{BA})$

1.4.6 Matrice inverse

L'inverse d'une matrice carrée $\mathbf{A}_{n \times n}$ est une matrice carrée $\mathbf{A}_{n \times n}^{-1}$ telle que : $\mathbf{AA}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}_n$. Une matrice est réversible ssi ses vecteurs colonnes forment un système libre.

pseudo-inverse: Une matrice $\mathbf{A}_{m \times n}$ peut admettre une pseudo-inverse $\mathbf{A}_{n \times m}^+$ telle que $\mathbf{AA}^+ = \mathbf{I}_m$ (plus de détails en section 4.3)

Propriétés :

- $(\mathbf{A}^{-1})^T = (\mathbf{A}^T)^{-1}$
- si \mathbf{A} et \mathbf{B} sont carrées, $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$

1.4.7 Matrice orthogonale

Matrice carrée dont toutes les lignes ont pour norme 1, sont orthogonales 2 à 2, de même pour les colonnes. Propriétés :

- si \mathbf{A} est orthogonale, alors $\mathbf{A}^{-1} = \mathbf{A}^T$.
- Les matrices de changement de base orthonormale sont orthogonales.

1.4.8 Déterminant

Soit \mathbf{A} une matrice carrée d'ordre n . Le déterminant de \mathbf{A} est un scalaire noté $|\mathbf{A}|$, nul si les vecteurs-colonnes sont liés. Propriétés :

- ne change pas si on ajoute à un vecteur-colonne une combinaison linéaire des vecteurs-colonnes
- $|\mathbf{A}| = |\mathbf{A}^T|$
- $|\mathbf{AB}| = |\mathbf{A}||\mathbf{B}|$
- \mathbf{A} est inversible ssi son déterminant est non nul

Soit a_{ij} un élément de \mathbf{A} . Notons \mathbf{a}_{ij}^- , la matrice de taille $(n-1) \times (n-1)$ obtenue en retirant la ligne i et la colonne j de \mathbf{A} . Le déterminant de \mathbf{a}_{ij}^- est appelé cofacteur de a_{ij} .

Calcul par développement de la première ligne :

$$|\mathbf{A}| = a_{11}|\mathbf{a}_{11}^-| - a_{12}|\mathbf{a}_{12}^-| + a_{13}|\mathbf{a}_{13}^-| \dots a_{1n}|\mathbf{a}_{1n}^-|$$

On peut aussi développer par rapport à la première colonne. Les déterminants d'ordre inférieur peuvent se calculer par la même méthode.

Notons $\tilde{\mathbf{A}}$ la matrice des cofacteurs. L'inverse de \mathbf{A} vaut : $\mathbf{A}^{-1} = \frac{1}{|\mathbf{A}|}\tilde{\mathbf{A}}^T$ (méthode de calcul extrêmement inefficace)

Le déterminant d'une matrice triangulaire est le produit des éléments de sa diagonale principale.

1.4.9 Valeurs propres et vecteurs propres

Définition

Soit une matrice carrée \mathbf{A}_n , un vecteur \mathbf{x} non nul et un scalaire λ tq : $\mathbf{Ax} = \lambda\mathbf{x}$. On dit que λ est une valeur propre de \mathbf{A} et que \mathbf{x} est un vecteur propre de \mathbf{A} associé à λ . Notons que pour $k \in \mathbb{R}$, $k\mathbf{x}$ est aussi vecteur propre.

Détermination des valeurs propres

De la dérivation suivante :

$$\begin{aligned}\mathbf{Ax} &= \lambda\mathbf{x} \\ \mathbf{Ax} - \lambda\mathbf{x} &= \mathbf{0} \\ (\mathbf{A} - \lambda\mathbf{I}_n)\mathbf{x} &= \mathbf{0}\end{aligned}$$

et puisque \mathbf{x} est non nul, nous pouvons déduire que les vecteurs-colonnes de $\mathbf{A} - \lambda\mathbf{I}_n$ sont liés. Le déterminant de cette matrice est donc nul. Ce déterminant s'exprime comme un polynôme d'ordre n en λ , et les racines de ce polynôme sont les valeurs propres de \mathbf{A} .

La recherche des vecteurs propres associés à une valeur propre λ s'effectue en résolvant le système d'équations : $(\mathbf{A} - \lambda\mathbf{I}_n)\mathbf{x} = \mathbf{0}$. On obtient une famille de solutions qui constitue le **sous-espace propre** associé à λ .

Propriétés :

- les sous-espaces propres sont des SEV.
- les sous-espaces propres sont orthogonaux entre eux.
- il ne peut y avoir plus de n valeurs propres pour une matrice d'ordre n .

Exemples (dans \mathbb{R}^3) :

- l'identité a une valeur propre : 1, et son SEP associé est de dimension 3
- une projection orthogonale sur un plan a deux valeurs propres 1 (SEP associé de dim 2) et 0 (SEP associé de dim 1)
- une rotation a une valeur propre : 1 (SEP associé de dim 1)

Diagonalisation

Si la somme des SEP a une dimension n égale à celle de la matrice, alors on peut trouver une matrice diagonale équivalente par changement de base. Soit B la base dans laquelle sont exprimés les vecteurs-colonnes de la matrice \mathbf{A} , et B' la base constituée par ses vecteurs propres (unitaires). Nous pouvons écrire :

$$\mathbf{A} = \overset{B}{\mathbf{R}} \mathbf{diag}(\lambda) \overset{B'}{\mathbf{R}} = \overset{B}{\mathbf{R}} \mathbf{diag}(\lambda) \overset{B'}{\mathbf{R}}^T$$

où $\mathbf{diag}(\lambda)$ est une matrice diagonale composée des valeurs propres de la matrice associées aux vecteurs-colonnes de la matrice $\overset{B}{\mathbf{R}}$, qui sont les vecteurs propres de \mathbf{A} . Nous pouvons réciproquement écrire :

$$\mathbf{diag}(\lambda) = \overset{B'}{\mathbf{R}}^T \mathbf{A} \overset{B}{\mathbf{R}}$$

1.4.10 Résolution de systèmes d'équations linéaires

Un système d'équations linéaires peut s'écrire sous forme matricielle $\mathbf{Ax} = \mathbf{b}$. On est tenté de le résoudre par $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. C'est généralement une mauvaise idée pour deux raisons :

- la matrice peut ne pas être inversible (matrice non carrée ou singulière) ;
- même quand c'est possible, inverser une matrice est très coûteux.

Voici quelques points de repère très généraux :

- matrice carrée, non singulière : décomposition LU
- matrice symétrique définie positive : décomposition de Cholesky
- matrice non carrée ou singulière : décomposition en valeurs singulières (SVD)
- matrice creuse : gradient biconjugué
- matrice creuse, symétrique définie positive : gradient conjugué

Les décompositions LU, Cholesky et SVD permettent en outre de calculer l'inverse ou le déterminant d'une matrice à un coût raisonnable. À certaines matrices particulières peuvent s'appliquer des méthodes spéciales non listées ici. Pour plus de détails, se reporter par exemple à [1].

Chapitre 2

Positionnement

Dans ce chapitre on considèrera l'espace en dimension 3, sachant que la dimension 2 peut trivialement s'y ramener.

2.1 Rotations

Une rotation est une isométrie dont une direction unique de l'espace (son *axe*) reste invariante. L'angle maximum entre un vecteur et sa transformée apparaît pour les vecteurs du plan orthogonal à l'axe. C'est par définition l'angle de la rotation. Nous noterons $\mathbf{R}_{\theta,u}$ une rotation d'angle θ autour de l'axe \mathbf{u} .

2.1.1 Propriétés

- la composée de deux rotations est une rotation.
- $\mathbf{R}_{\alpha,u}\mathbf{R}_{\beta,v} \neq \mathbf{R}_{\beta,v}\mathbf{R}_{\alpha,u}$ sauf si $\alpha\mathbf{u}$ est colinéaire à $\beta\mathbf{v}$
- $\mathbf{R}_{\alpha,u}\mathbf{R}_{\beta,u} = \mathbf{R}_{(\alpha+\beta),u}$
- $\mathbf{R}_{\alpha,u}^{-1} = \mathbf{R}_{-\alpha,u} = \mathbf{R}_{\alpha,-u}$
- les matrices de rotation sont orthogonales
- l'espace des rotations est de dimension 3 (9 coefficients de la matrice -3 contraintes de vecteurs-colonnes unitaires -3 contraintes d'orthogonalité)

2.1.2 Représentation matricielle

Les matrices ci-dessous représentent respectivement les rotations $\mathbf{R}_{\alpha,x}$, $\mathbf{R}_{\beta,y}$ et $\mathbf{R}_{\gamma,z}$.

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} \sin \beta & 0 & \cos \beta \\ 0 & 1 & 0 \\ \cos \beta & 0 & -\sin \beta \end{pmatrix} \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2.1)$$

2.1.3 Décomposition de Jordan

Quand la rotation ne s'effectue pas autour d'un des axes principaux de la base, on peut obtenir son expression matricielle par la **décomposition de Jordan** : $\mathbf{R}_{\theta,u} = \mathbf{R}_a\mathbf{R}_{\theta,x}\mathbf{R}_a^{-1} = \mathbf{R}_a\mathbf{R}_{\theta,x}\mathbf{R}_a^T$ où \mathbf{R}_a est une rotation qui aligne l'axe \mathbf{x} de la base sur l'axe \mathbf{u} de rotation. Un autre vecteur que \mathbf{x} peut être choisi. Cette décomposition correspond à un changement de base d'application linéaire comme vu en section 1.3.4.

Cette représentation n'est pas unique car l'alignement est défini à une rotation près autour de l'axe \mathbf{u} . On peut en effet remplacer \mathbf{R}_a par $\mathbf{R}_a \mathbf{R}_{\phi, \mathbf{u}}$ pour tout ϕ .

question : si \mathbf{u} est unitaire :

1. que vaut la première colonne de \mathbf{R}_a ?
2. comment en déduire des valeurs pour les colonnes 2 et 3 ?

2.1.4 Conversion entre matrice et (angle,axe). Formule de Rodrigues.

La formule de Rodrigues fournit un passage direct entre (angle,axe) et matrice de rotation :

$$\mathbf{R}_{\theta, \mathbf{u}} = \mathbf{I} + \sin \theta (\mathbf{n} \wedge) + (1 - \cos \theta) (\mathbf{n} \wedge)^2 \quad (2.2)$$

avec \mathbf{I} l'identité et $(\mathbf{n} \wedge)$ la matrice qui du produit vectoriel par \mathbf{n} (voir équation 1.1).

Le problème inverse est de trouver l'axe \mathbf{n} et l'angle θ d'après la matrice \mathbf{R} . Il n'y a pas de solution unique (angle défini à 2π près, axe ou son opposé). On peut en trouver une en remarquant que $\text{tr}(\mathbf{R}) = 1 + 2 \cos \theta$ et que $\mathbf{R} - \mathbf{R}^T = 2 \sin \theta (\mathbf{n} \wedge)$

2.1.5 Composition des rotations. Angles d'Euler.

Soit une base \mathcal{B}_1 tournée de $\mathbf{R}_{\theta, \mathbf{u}}$ par rapport à une base de référence \mathcal{B}_0 . Nous voulons lui appliquer une rotation supplémentaire, par exemple $\mathbf{R}_{\alpha, \mathbf{x}}$ d'angle α suivant l'axe \mathbf{x} . Pour calculer sa nouvelle matrice d'orientation il est important de savoir de *quel* axe \mathbf{x} on parle :

- s'il s'agit de l'axe \mathbf{x} de la base \mathcal{B}_0 , alors on multiplie à *gauche* et la nouvelle matrice de rotation vaut $\mathbf{R}_{\alpha, \mathbf{x}} \mathbf{R}_{\theta, \mathbf{u}}$
- s'il s'agit de l'axe \mathbf{x} de la base \mathcal{B}_1 , alors on multiplie à *droite* et la nouvelle matrice de rotation vaut $\mathbf{R}_{\theta, \mathbf{u}} \mathbf{R}_{\alpha, \mathbf{x}}$

Les angles d'Euler consistent à appliquer trois rotations successives selon des axes principaux. Suivant le contexte, on les interprète comme des rotations suivant les axes d'une base de référence, ou suivant les axes des bases intermédiaires successives. Par exemple, la rotation $\mathbf{R}_{\alpha, \mathbf{x}} \mathbf{R}_{\beta, \mathbf{y}} \mathbf{R}_{\gamma, \mathbf{z}}$ (matrices définies en éq. 2.1) peut s'interpréter de deux manières mathématiquement équivalentes mais intuitivement différentes :

- rotation d'angle α selon \mathbf{x} (de \mathcal{B}_0) puis rotation d'angle β autour du *nouveau* \mathbf{y} (ayant subi une rotation) puis rotation d'angle γ autour du *nouveau* \mathbf{z} (ayant subi deux rotations)
- ou rotation d'angle γ autour de \mathbf{z} de \mathcal{B}_0 puis rotation d'angle β autour de \mathbf{y} de \mathcal{B}_0 puis rotation d'angle α selon \mathbf{x} de \mathcal{B}_0 .

Les angles d'Euler présentent toutefois certains inconvénients :

- non-unicité : par exemple en xyz, la rotation $(\pi, \pi, 0)$ est équivalente à $(0, 0, \pi)$
- perte d'un degré de liberté : par exemple pour $(\alpha, \pi/2, \gamma)$ en xyz, α et γ ont le même effet car ils font pivoter autour d'un même axe. Aucune rotation n'est alors possible autour du troisième axe. Ceci représente le cas extrême, mais toute configuration proche de celle-ci posera des problèmes numériques
- les interpolations peuvent prendre des chemins compliqués ou inattendus.

2.1.6 Quaternions

Extension des nombres complexes : $\mathbf{q} = w + xi + yj + zk = (w, \mathbf{v})$.
 w est la partie réelle, \mathbf{v} la partie imaginaire. Propriétés de i, j, k :

$$\begin{aligned} i^2 &= j^2 = k^2 = -1 \\ ij &= k, \quad ji = -k \\ jk &= i, \quad kj = -i \\ ki &= j, \quad ik = -j \end{aligned}$$

Vecteur 3D :

$$\mathbf{p} = (0, x, y, z)$$

Produit de quaternions (non commutatif) :

$$\mathbf{q}_1 \mathbf{q}_2 = (w_1 w_2 - \mathbf{v}_1 \cdot \mathbf{v}_2, w_1 \mathbf{v}_2 + w_2 \mathbf{v}_1 + \mathbf{v}_1 \wedge \mathbf{v}_2)$$

Quaternion conjugué :

$$\begin{aligned} \bar{\mathbf{q}} &= w - xi - yj - zk \\ \mathbf{q} \bar{\mathbf{q}} &= w^2 + x^2 + y^2 + z^2 \end{aligned}$$

Quaternions unitaires, utilisés pour représenter les rotations :

$$\mathbf{q} \bar{\mathbf{q}} = 1$$

Rotation $(\theta, \mathbf{u}) : (\mathbf{u}^2 = 1)$

$$\mathbf{q}_{(\theta, \mathbf{u})} = \left(\cos \frac{\theta}{2}, u_x \sin \frac{\theta}{2}, u_y \sin \frac{\theta}{2}, u_z \sin \frac{\theta}{2} \right)$$

Rotation d'un vecteur $\mathbf{p} : \mathbf{q} \mathbf{p} \bar{\mathbf{q}}$

Matrice de rotation associée à un quaternion unitaire :

$$\begin{pmatrix} 1 - 2y^2 - 2z^2 & 2xy - 2wz & 2xz + 2wy \\ 2xy + 2wz & 1 - 2x^2 - 2z^2 & 2yz - 2wx \\ 2xz - 2wy & 2yz + 2wx & 1 - 2x^2 - 2y^2 \end{pmatrix}$$

Composition des rotations : $\mathbf{R}_{\alpha, \mathbf{u}} \mathbf{R}_{\beta, \mathbf{v}} \longrightarrow q_{(\alpha, \mathbf{u})} q_{(\beta, \mathbf{v})}$

Rotation inverse : $q_{(\theta, \mathbf{u})}^{-1} = q_{(-\theta, \mathbf{u})} = q_{(\theta, -\mathbf{u})} = (-w, \mathbf{v}) = (w, -\mathbf{v})$

Conversion $(w, \mathbf{v}) \longrightarrow (\theta, \mathbf{u})$:

$$\begin{aligned} \cos(\theta/2) &= w \\ \sin(\theta/2) &= \|\mathbf{v}\| \\ \mathbf{u} &= \mathbf{v}/\|\mathbf{v}\| \end{aligned}$$

2.1.7 Interpolation linéaire des rotations

Soit une base initialement définie par ${}^0_1 \mathbf{R}$ à $t = 0$, que nous voulons faire évoluer continument vers ${}^0_2 \mathbf{R}$ à $t = 1$.

- On ne peut interpoler simplement les coefficients d'une matrice de rotation car on viole alors les contraintes sur les vecteurs-colonnes (unitaires, orthogonaux).
- On peut interpoler les angles d'Euler mais le chemin emprunté sera rarement le plus direct, et la vitesse rarement constante.

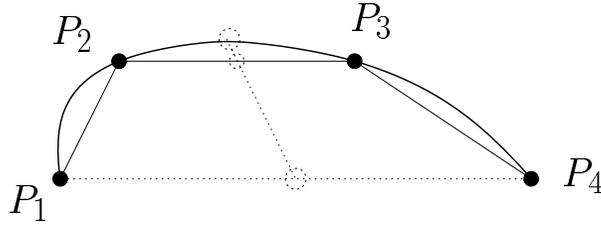


FIG. 2.1 – Approximation itérative d’une spline. Un nouveau point est calculé par somme pondérée des points P_1, P_2, P_3, P_4 ayant respectivement pour coefficients $\frac{-c}{8}, \frac{-c}{8} + \frac{1}{2}, \frac{-c}{8} + \frac{1}{2}$, et $\frac{-c}{8}$. Le coefficient c exprime la continuité de la spline et peut être fixé à 1.

Il convient de :

1. calculer la matrice de passage de la configuration initiale à la configuration finale :

$$\begin{aligned} {}^1_2\mathbf{R} &= {}^0_2\mathbf{R}_2^0\mathbf{R} \\ &= ({}^0_1\mathbf{R}^{-1})_2^0\mathbf{R} \end{aligned}$$

2. calculer angle et axe θ , tels que : $\mathbf{R}_{\theta,u} = {}^1_2\mathbf{R}$
3. interpoler en appliquant $\mathbf{R}(t) = {}^0_1\mathbf{R}\mathbf{R}_{t\theta,u}$

2.1.8 Interpolation lisse des rotations

Nous voulons passer de manière lisse par différentes orientations au cours du temps. Nous nous basons sur la méthode d’approximation itérative des splines présentée en figure 2.1 et construisons une analogie entre vecteurs de \mathbb{R}^3 et quaternions unitaires :

	vecteur \mathbf{v}	quaternion $q(\theta, \mathbf{u}) = (w, \mathbf{v})$
composition	$\mathbf{v}_1 + \mathbf{v}_2 (= \mathbf{v}_2 + \mathbf{v}_1)$	$q_1 q_2 (\neq q_2 q_1 !)$
inverse	$\mathbf{v}_1^{-1} = -\mathbf{v}$	$q_1^{-1} = (-w, \mathbf{v})$
produit avec un scalaire	$k \mathbf{v}$	$q^k = q(k\theta, \mathbf{u})$
distance	$\Delta \mathbf{v} = \mathbf{v}_2 - \mathbf{v}_1 = -\mathbf{v}_1 + \mathbf{v}_2$	$\Delta q = q_1^{-1} q_2$
interpolation	$\mathbf{v} = \text{lerp}(\mathbf{v}_1, \mathbf{v}_2, \alpha) = \mathbf{v}_1 + \alpha(-\mathbf{v}_1 + \mathbf{v}_2)$	$q = \text{slerp}(q_1, q_2, \alpha) = q_1 (q_1^{-1} q_2)^\alpha$
moyenne	$\text{moy}(\mathbf{v}_1, \mathbf{v}_2) = \text{lerp}(\mathbf{v}_1, \mathbf{v}_2, 0.5)$	$\text{smoy}(q_1, q_2) = \text{slerp}(q_1, q_2, 0.5)$

Nous pouvons maintenant à partir de quatre quaternions en générer un supplémentaire, et à partir de $m + 1$ quaternions en obtenir $2m + 1$:

$$\begin{aligned} \{q_i : i = 0, 1, \dots, m\} &\longrightarrow \{q_j : j = 0, 1, \dots, 2m\} \\ j = 2i + 1 : q_j &= \text{slerp}(\text{smid}(q_i, q_{i+3}), \text{smid}(q_{i+1}, q_{i+2}), 1 + c/4) \\ j = 2i : q_j &= q_i \end{aligned}$$

2.2 Transformations affines

A COMPLETER

Chapitre 3

Cinématique

3.1 Rappels

3.1.1 Dérivée dans \mathcal{R}_0 d'un vecteur fixe dans \mathcal{R}_1 . Vecteur rotation.

Considérons un vecteur $\mathbf{u}_{\in 1}$, fixe dans un repère \mathcal{R}_1 . Le repère \mathcal{R}_1 est en rotation par rapport à un repère \mathcal{R}_0 . Nous nous intéressons à la projection ${}^0\mathbf{u}_{\in 1}$ de ce vecteur dans \mathcal{R}_0 , que nous appellerons parfois \mathbf{u} pour simplifier, et à sa dérivée dans \mathcal{R}_0 que nous noterons $\dot{\mathbf{u}}_{\in 1}^{(0)}$.

Soit $\mathbf{R}(dt)$ la rotation subie par \mathcal{R}_1 entre l'instant t et l'instant $t + dt$. Nous pouvons écrire :

$$\mathbf{u}(t + dt) = \mathbf{R}(dt)\mathbf{u}(t) \quad (3.1)$$

$$\mathbf{u}(t + dt) - \mathbf{u}(t) = (\mathbf{R}(dt) - \mathbf{I})\mathbf{u}(t) \quad (3.2)$$

Soit $\dot{\theta}$ la vitesse angulaire autour de l'axe de rotation, que nous choisissons égal à \mathbf{z} pour simplifier. Nous obtenons par un développement limité à l'ordre 1 :

$$\mathbf{R}(dt) - \mathbf{I} = \begin{pmatrix} \cos(\dot{\theta}dt) - 1 & -\sin(\dot{\theta}dt) & 0 \\ \sin(\dot{\theta}dt) & \cos(\dot{\theta}dt) - 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \longrightarrow \begin{pmatrix} 0 & -\dot{\theta}dt & 0 \\ \dot{\theta}dt & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \dot{\theta}dt(\mathbf{z}\wedge)$$

où $(\mathbf{z}\wedge)$ est l'opérateur matriciel de produit vectoriel par \mathbf{z} . Cette formule peut s'étendre par changement de base à n'importe quel axe de rotation \mathbf{n} (unitaire). Posons $\boldsymbol{\Omega}_{1/0} = \dot{\theta}\mathbf{n}$. En divisant l'expression 3.2 par dt et en faisant tendre dt vers 0 nous obtenons $\dot{\mathbf{R}} = (\boldsymbol{\Omega}_{1/0}\wedge)$.

Nous pouvons écrire la dérivée temporelle dans \mathcal{R}_0 : $\dot{\mathbf{u}}_{\in 1}^{(0)} = \boldsymbol{\Omega}_{1/0} \wedge \mathbf{u}_{\in 1}$, ou en simplifiant la notation :

$$\begin{aligned} \dot{\mathbf{u}} &= \dot{\mathbf{R}}\mathbf{u} \\ &= (\boldsymbol{\Omega}_{1/0}\wedge)\mathbf{u} \\ &= \boldsymbol{\Omega}_{1/0} \wedge \mathbf{u} \end{aligned} \quad (3.3)$$

3.1.2 Vitesse dans \mathcal{R}_0 d'un point fixe dans \mathcal{R}_1 . Champ des vitesses.

Considérons la vitesse dans \mathcal{R}_0 d'un point A fixe dans \mathcal{R}_1 en mouvement par rapport à \mathcal{R}_0 . Soient O_0 l'origine de \mathcal{R}_0 et O_1 l'origine de \mathcal{R}_1 . Nous dérivons en coordonnées homogènes la

projection de A dans \mathcal{R}_0 :

$$\begin{aligned} \mathbf{O}_0\mathbf{A} &= \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \mathbf{O}_1\mathbf{A} \\ \mathbf{O}_0\dot{\mathbf{A}} &= \begin{pmatrix} (\boldsymbol{\Omega}\wedge) & \dot{\mathbf{t}} \\ \mathbf{0} & 0 \end{pmatrix} \begin{pmatrix} \overrightarrow{O_1A} \\ 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Omega} \wedge \overrightarrow{O_1A} + \dot{\mathbf{t}} \\ 0 \end{pmatrix} = \begin{pmatrix} \vec{V}_A^{1/0} \\ 0 \end{pmatrix} \end{aligned}$$

Nous appelons $\vec{V}_A^{1/0}$ vitesse de A fixe dans \mathcal{R}_1 par rapport à \mathcal{R}_0 . On peut noter que $\vec{V}_{O_1}^{1/0} = \dot{\mathbf{t}}$. Nous en déduisons les relations du **champ des vitesses d'un solide** :

$$\vec{V}_A^{1/0} = \vec{V}_{O_1}^{1/0} + \boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A} \quad (3.4)$$

$$\vec{V}_A^{1/0} = \vec{V}_B^{1/0} + \boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{BA} \text{ pour tous points } A \text{ et } B \quad (3.5)$$

la dernière relation se déduisant trivialement par soustraction $\vec{V}_A^{1/0} - \vec{V}_B^{1/0}$.

3.1.3 Accélération dans \mathcal{R}_0 d'un point fixe dans \mathcal{R}_1 . Champ des accélérations.

En dérivant la relation 3.4, et sachant que $\overrightarrow{O_1A}$ est fixe dans \mathcal{R}_1 , nous obtenons la relation du **champ des accélérations d'un solide** :

$$\vec{\Gamma}_A^{1/0} = \vec{\Gamma}_{O_1}^{1/0} + \dot{\boldsymbol{\Omega}}_{1/0} \wedge \overrightarrow{O_1A} + \boldsymbol{\Omega}_{1/0} \wedge (\boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A}) \quad (3.6)$$

3.1.4 Dérivée dans \mathcal{R}_0 d'un vecteur non fixe dans \mathcal{R}_1

Soit $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ une base de \mathcal{R}_1 . Nous pouvons écrire :

$$\begin{aligned} {}^1u &= \sum_i x_i \mathbf{e}_i \\ \dot{\mathbf{u}} &= \sum_i \dot{x}_i \mathbf{e}_i + \sum_i x_i \dot{\mathbf{e}}_i \end{aligned}$$

d'où :

$$\dot{\mathbf{u}}^{(0)} = \dot{\mathbf{u}}^{(1)} + \boldsymbol{\Omega}_{1/0} \wedge \mathbf{u} \quad (3.7)$$

3.1.5 Vitesse dans \mathcal{R}_0 d'un point mobile dans \mathcal{R}_1 .

Soit $\vec{V}_A^{1/1}$ la vitesse d'un point A dans le repère \mathcal{R}_1 . Celle-ci vient s'ajouter à la vitesse qu'aurait le point s'il était fixe dans \mathcal{R}_1 . Nous avons donc :

$$\vec{V}_A^{1/0} = \vec{V}_A^{1/1} + \vec{V}_{O_1}^{1/0} + \boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A} \quad (3.8)$$

Notons que le point O_1 étant l'origine du repère \mathcal{R}_1 , on a $\vec{V}_{O_1}^{1/0} = \vec{V}_{O_1}^{1/1}$.

3.1.6 Accélération dans \mathcal{R}_0 d'un point mobile dans \mathcal{R}_1 . Accélération de Coriolis.

En dérivant la relation 3.8 nous obtenons :

$$\vec{\Gamma}_A^{1/0} = \underbrace{\vec{\Gamma}_A^{1/1} + \boldsymbol{\Omega}_{1/0} \wedge \vec{V}_A^{1/1}}_{\vec{\Gamma}_A^{1/1}} + \vec{\Gamma}_{O_1}^{1/0} + \underbrace{\dot{\boldsymbol{\Omega}}_{1/0} \wedge \overrightarrow{O_1A} + \boldsymbol{\Omega}_{1/0} \wedge \vec{V}_A^{1/1} + \boldsymbol{\Omega}_{1/0} \wedge (\boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A})}_{\boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A}}$$

ou encore :

$$\vec{\Gamma}_A^{/0} = \vec{\Gamma}_A^{/1} + \vec{\Gamma}_{O_1}^{/0} + \boldsymbol{\Omega}_{1/0} \wedge (\boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A}) + 2\boldsymbol{\Omega}_{1/0} \wedge \vec{V}_A^{/1} \quad (3.9)$$

avec :

- $\vec{\Gamma}_A^{/1} = \sum_i \ddot{x}_i \mathbf{e}_i$ accélération relative
- $\vec{\Gamma}_{O_1}^{/0}$ accélération d'entraînement
- $\boldsymbol{\Omega}_{1/0} \wedge (\boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A})$ accélération centripète
- $2\boldsymbol{\Omega}_{1/0} \wedge \vec{V}_A^{/1}$ accélération de Coriolis

3.2 Cinématique des chaînes articulées

3.2.1 Composition des vitesses

Si nous appliquons la relation 3.8 à un point fixe dans un repère \mathcal{R}_2 , nous pouvons directement écrire

$$\begin{aligned} \vec{V}_A^{2/0} &= \vec{V}_A^{2/1} + \vec{V}_{O_1}^{1/0} + \boldsymbol{\Omega}_{1/0} \wedge \overrightarrow{O_1A} \\ &= \vec{V}_A^{2/1} + \vec{V}_A^{1/0} \end{aligned}$$

et plus généralement :

$$\vec{V}_A^{n/0} = \sum_{i=1}^n \vec{V}_A^{i/i-1} \quad (3.10)$$

La figure 3.1 illustre une application de cette formule. La vitesse $\vec{V}_A^{3/0}$ est la somme des contributions de chaque mouvement relatif. On note que la vitesse due à une rotation est perpendiculaire à la droite joignant le point au centre de rotation. La vitesse de translation est reportée tel quel.

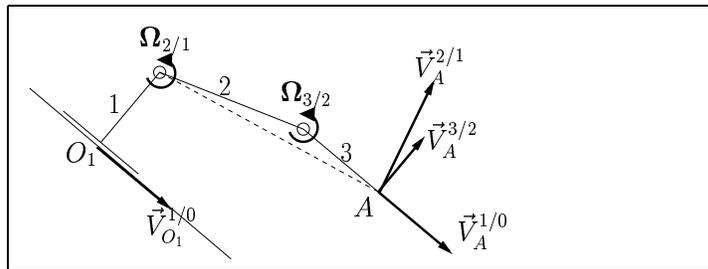


FIG. 3.1 – Composition des vitesses.

3.2.2 Liaisons cinématiques. Degrés de liberté.

Les liaisons cinématiques définissent les mouvements possibles entre deux solides parmi les trois translations et trois rotations possibles. Les liaisons se caractérisent par leurs **degrés de liberté** (ddl). La figure 3.2 montre l'exemple de la liaison pivot glissant, qui autorise deux degrés de liberté : un en translation et un en rotation. La plus contrainte est la liaison *encastrement*, avec zéro ddl, la plus libre est la liaison *libre* avec six ddl. Les principales liaisons sont présentées en annexe A.

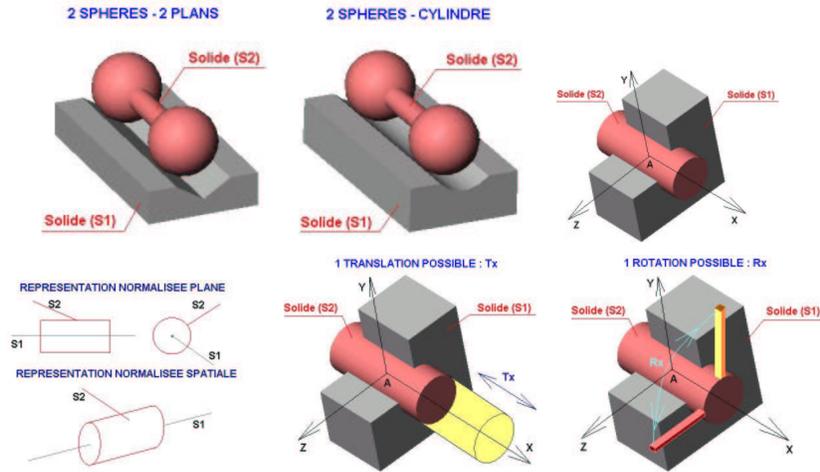


FIG. 3.2 – Liaison pivot glissant (images de Roger Bouisset). De haut en bas et de gauche à droite : des surfaces pouvant créer ce type de liaison, les axes principaux de la liaison, la représentation normalisée de la liaison, les mouvements relatifs possibles.

Cas général

Une liaison permet le mouvement relatif de deux repères qui dans le cas général ne sont pas directement les repères principaux des solides. La transformation ${}^i_{i-1}\mathbf{C}$ de \mathcal{R}_i vers \mathcal{R}_{i-1} due à une liaison peut se décomposer en trois transformations :

$${}^i_{i-1}\mathbf{C} = ({}^i_{i-1}\mathbf{C}_p)({}^i_{i-1}\mathbf{C}_l)({}^i_{i-1}\mathbf{C}_c) \quad (3.11)$$

comme illustré sur la figure 3.3.

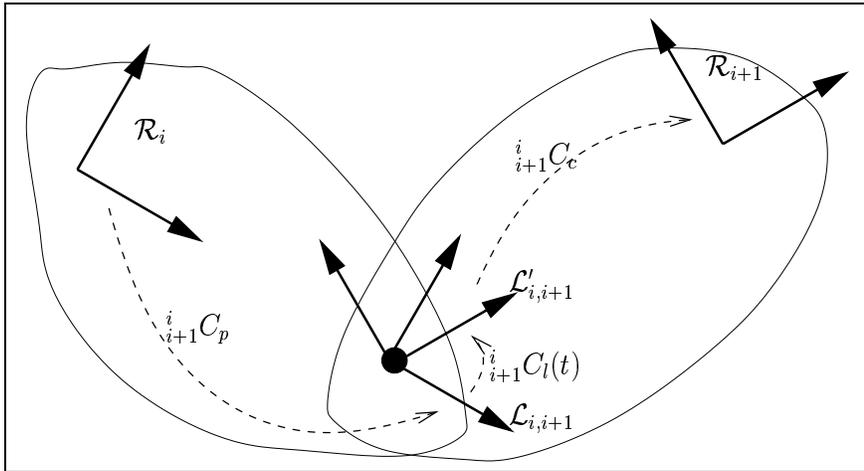


FIG. 3.3 – Une liaison entre les repères i et $i+1$, avec ses repères intermédiaires.

La matrice ${}^i_{i-1}\mathbf{C}_p$ traduit la position de la liaison par rapport à \mathcal{R}_{i-1} . La matrice ${}^i_{i-1}\mathbf{C}_l$ traduit le déplacement dans la liaison, par rapport à une position de référence. Elle peut varier avec le temps et conformément aux ddl. La matrice ${}^i_{i-1}\mathbf{C}_c$ (plus précisément, son inverse) traduit la position de la liaison par rapport à \mathcal{R}_i .

La figure 3.4 présente un exemple de chaîne articulée. Un socle mobile de repère \mathcal{R}_1 peut se déplacer sur le sol de repère \mathcal{R}_0 . Un bras de repère \mathcal{R}_2 est articulé par rapport au socle mobile. À droite de la figure, le **graphe cinématique** représente l'accrochage des solides (noeuds) par les liaisons (arcs). Ce graphe est orienté en partant du repère absolu, qui est toujours à la racine.

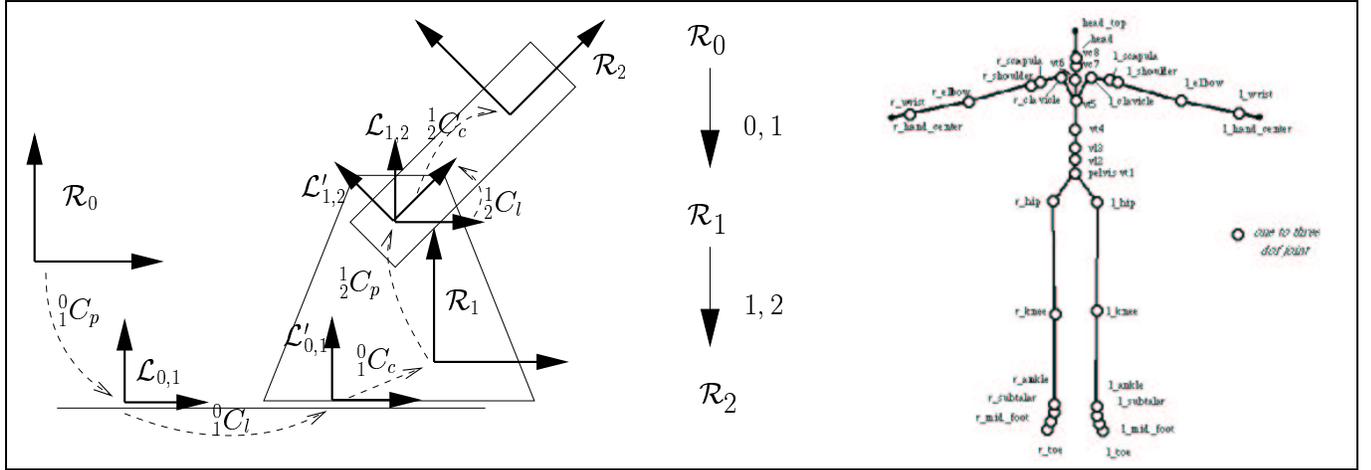


FIG. 3.4 – Corps articulés et graphes cinématiques. À gauche, un mécanisme avec ses solides, liaisons et repères intermédiaires. Au centre, le graphe cinématique correspondant. À droite, un graphe cinématique plus complexe.

Modèle de Denavit-Hartenberg

- La notation de Denavit-Hartenberg, très utilisée en robotique, correspond à un cas simplifié :
- le repère du solide enfant est centré sur le repère mobile de l'articulation : ${}^i_{i-1}\mathbf{C}_c = \mathbf{I}_4$
 - les liaisons n'autorisent qu'un degré de liberté en translation et en rotation, le long du même axe (z par convention).

De plus, l'axe x_i d'un repère \mathcal{R}_i est par définition confondu avec la perpendiculaire commune aux axes (O_i, z_i) et (O_{i+1}, z_{i+1}) . La figure 3.5 illustre la mise en position d'un repère par rapport à son père. Pour aller du repère père \mathcal{R}_i au repère fils \mathcal{R}_{i+1} , on translate selon x (paramètre a_i), puis on tourne selon x (paramètre α_i), puis on translate selon z (paramètre d_{i+1}) et finalement on pivote selon z (paramètre θ_{i+1}). Les paramètres a_i et α_i sont constants, les deux autres correspondant à la rotation et la translation selon l'axe. La transformation géométrique vaut :

$$\begin{aligned} {}^i_{i+1}\mathbf{C} &= \mathbf{T}_{x_i, a_i} \mathbf{R}_{x_i, \alpha_i} \mathbf{T}_{z_{i+1}, d_{i+1}}(t) \mathbf{R}_{z_{i+1}, \theta_{i+1}}(t) \\ &= ({}^i_{i+1}\mathbf{C}_p) ({}^i_{i+1}\mathbf{C}_l(t)) \end{aligned}$$

Notez que nous numérotions les repères en partant du repère absolu \mathcal{R}_0 . Il arrive de rencontrer l'ordre inverse, particulièrement dans la littérature robotique.

3.2.3 Paramétrage des liaisons. Espace articulaire.

À chaque ddl i est associé un axe \mathbf{n}_i (vecteur unitaire), une valeur de position q_i et une valeur de vitesse \dot{q}_i . Les valeurs aux ddl définissent la position d'une liaison. En translation, ces valeurs sont reportées dans le vecteur translation de la matrice de passage. Le cas des rotations est plus

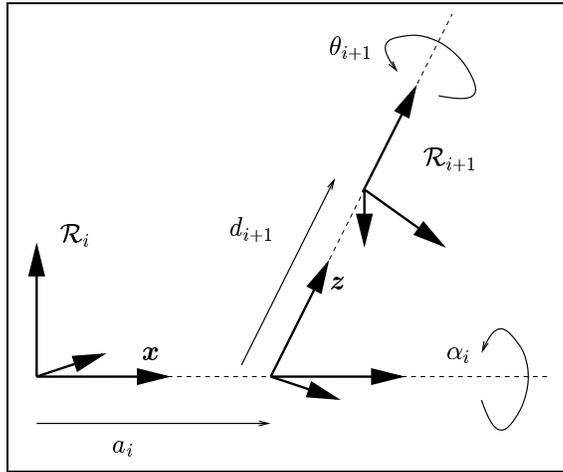


FIG. 3.5 – Liaison selon Denavit-Hartenberg. Quatre paramètres décrivent la position du fils (i+1) par rapport au père (i).

complexe. Pour les liaisons à un ou deux ddl, on utilise les angles d'Euler et on calcule la matrice correspondante. Pour les liaisons à trois ddl et mouvements de grande amplitude, on ne peut employer les angles d'Euler à cause des limitations analytiques de ceux-ci, notamment la perte d'un ddl dans certaines configurations. On a alors recours aux quaternions, ou directement aux matrices. Le paramétrage par angles d'Euler reste possible pour exprimer des petites variations autour de la position courante.

On peut rassembler l'ensemble des valeurs associées aux ddl dans un vecteur \mathbf{q} qui définit l'état du système. L'ensemble des états possibles est appelé **espace articulaire** du système. De même on peut regrouper les vitesses articulaires dans un vecteur $\dot{\mathbf{q}}$.

3.2.4 Algorithmes de cinématique directe

La cinématique dite directe consiste à calculer des positions et vitesses dans l'espace cartésien à partir de l'état articulaire du système. Dans cette partie nous considérerons un mécanisme modélisé par la notation de Denavit-Hartenberg (cf. section 3.2.2), constitué d'une chaîne cinématique dont les solides sont numérotés de 0 (repère absolu) à n. Le repère fils est translaté et tourné selon l'axe \mathbf{x} du père, puis translaté et tourné selon le nouvel axe \mathbf{z} .

La figure 3.6 présente un algorithme de calcul des matrices de passage de chaque solide de la chaîne.

$$\begin{array}{l}
 {}^0_0\mathbf{C} \leftarrow \mathbf{I}_4 \\
 i \text{ parcourant } 1..n \\
 {}^{i-1}_i\mathbf{C} \leftarrow \mathbf{T}_{\mathbf{x},a_{i-1}} \mathbf{R}_{\mathbf{x},\alpha_{i-1}} \mathbf{T}_{\mathbf{z},d_i} \mathbf{R}_{\mathbf{z},\theta_i} \\
 {}^0_i\mathbf{C} \leftarrow {}^0_{i-1}\mathbf{C} {}^{i-1}_i\mathbf{C}
 \end{array}$$

FIG. 3.6 – Calcul des matrices de positions dans le formalisme de Denavit-Hartenberg.

La figure 3.7 présente un algorithme de calcul de vitesses. Les données sont :

- la position nO_nA d'un point A dans le repère \mathcal{R}_n

- les positions \mathbf{q} et vitesses articulaires $\dot{\mathbf{q}}$

Les résultats sont :

- la position 0O_0A de A dans le repère absolu \mathcal{R}_0
- la vitesse de rotation $\boldsymbol{\Omega}_{n/0}$
- la vitesse $\vec{V}_A^{n/0}$ de A fixe dans \mathcal{R}_n par rapport à \mathcal{R}_0 , exprimée dans \mathcal{R}_0

Au passage, toutes les matrices ${}^i{}_{i-1}\mathbf{C}$ sont calculées.

$$\begin{array}{l}
 \vec{O\dot{A}} \leftarrow {}^n O_n \vec{A} \\
 \vec{V} \leftarrow \vec{0} \\
 \vec{\Omega} \leftarrow \vec{0} \\
 \text{i décroissant de n à 1} \\
 {}^i{}_{i-1}\mathbf{C} \leftarrow \mathbf{T}_{x,\alpha_{i-1}} \mathbf{R}_{x,\alpha_{i-1}} \mathbf{T}_{z,d_i} \mathbf{R}_{z,\theta_i} \\
 \vec{\Omega} \leftarrow {}^i{}_{i-1}\mathbf{R}(\vec{\Omega} + \dot{\theta}_i \mathbf{z}) \\
 \vec{V} \leftarrow {}^i{}_{i-1}\mathbf{R}(\vec{V} + \dot{d}_i \mathbf{z} + \dot{\theta}_i \mathbf{z} \wedge \vec{O\dot{A}}) \\
 \vec{O\dot{A}} \leftarrow {}^i{}_{i-1}\mathbf{C} \vec{O\dot{A}}
 \end{array}$$

FIG. 3.7 – Calcul de vitesse dans le formalisme de Denavit-Hartenberg.

Chapitre 4

Cinématique inverse

Problématique : trouver des configurations articulaires pour satisfaire des contraintes définies dans l'espace cartésien.

4.1 Équations linéaires

Considérons un mécanisme comportant deux liaisons glissière alignées sur les axes du repère absolu, comme illustré sur la figure 4.1. Chaque degré de liberté est caractérisé par son axe \mathbf{a} et sa coordonnée articulaire q . L'axe portant \mathbf{a}_1 est fixe tandis que l'axe portant \mathbf{a}_2 se translate. Nous désirons amener le point P jusqu'à la position P' et pour cela nous devons satisfaire une

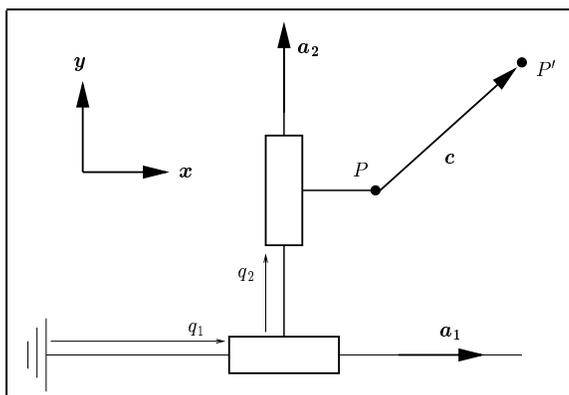


FIG. 4.1 – Un mécanisme à axes orthogonaux.

contrainte de déplacement $\Delta \mathbf{P} = \overrightarrow{OP'} - \overrightarrow{OP} = \mathbf{c} = (c_x, c_y)^T$ au moyen d'un déplacement articulaire $\Delta \mathbf{q} = (\Delta q_1, \Delta q_2)^T$. Dans ce cas simple la réponse est évidemment $\Delta \mathbf{q} = (c_x, c_y)^T$.

Considérons maintenant un mécanisme similaire, mais à axes quelconques, comme illustré sur la figure 4.2. La résolution d'une contrainte de déplacement nécessite de poser et résoudre un système d'équations linéaires :

$$\begin{pmatrix} a_{1x} & a_{2x} \\ a_{1y} & a_{2y} \end{pmatrix} \begin{pmatrix} \Delta q_1 \\ \Delta q_2 \end{pmatrix} = \begin{pmatrix} c_x \\ c_y \end{pmatrix} \quad (4.1)$$

Considérons maintenant la contrainte suivante : le point P doit se positionner sur la droite définie par un point P' et un vecteur normal $\mathbf{n} = (n_x, n_y)^T$, comme illustré sur la figure 4.3.

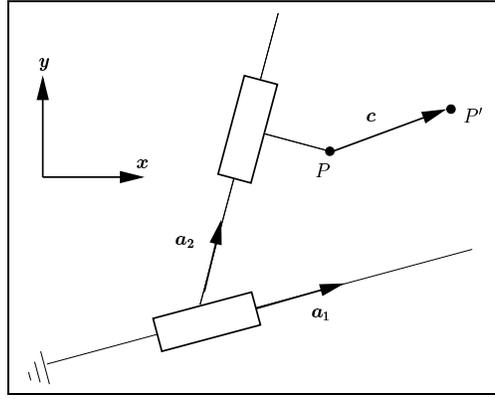


FIG. 4.2 – Un mécanisme à axes non orthogonaux.

Nous sommes en présence d'une contrainte scalaire, puisqu'on peut l'exprimer par une équation unique $\Delta \mathbf{P} \cdot \mathbf{n} = \overrightarrow{PP'} \cdot \mathbf{n}$. Posons, développons et regroupons l'équation sur le déplacement :

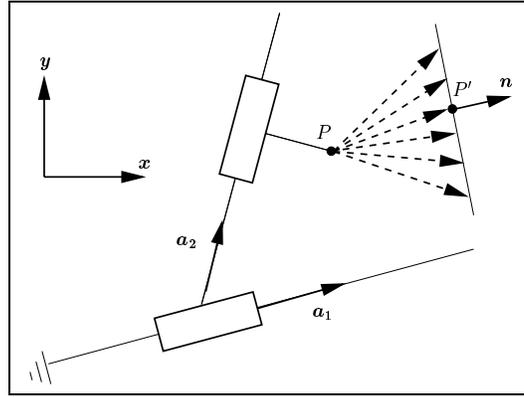


FIG. 4.3 – Une contrainte scalaire appliquée au mécanisme 2D : rejoindre une droite.

$$\begin{aligned} \Delta \mathbf{P} \cdot \mathbf{n} &= \overrightarrow{PP'} \cdot \mathbf{n} \\ (\mathbf{a}_1 \quad \mathbf{a}_2) \Delta \mathbf{q} &= \overrightarrow{PP'} \cdot \mathbf{n} \\ \begin{pmatrix} a_{1x} & a_{2x} \\ a_{1y} & a_{2y} \end{pmatrix} \begin{pmatrix} \Delta q_1 \\ \Delta q_2 \end{pmatrix} \cdot \mathbf{n} &= \overrightarrow{PP'} \cdot \mathbf{n} \\ (a_{1x} \Delta q_1 + a_{2x} \Delta q_2) n_x + (a_{1y} \Delta q_1 + a_{2y} \Delta q_2) n_y &= \overrightarrow{PP'} \cdot \mathbf{n} \\ (a_{1x} n_x + a_{1y} n_y) \Delta q_1 + (a_{2x} n_x + a_{2y} n_y) \Delta q_2 &= \overrightarrow{PP'} \cdot \mathbf{n} \\ (\mathbf{a}_1 \cdot \mathbf{n} \quad \mathbf{a}_2 \cdot \mathbf{n}) \begin{pmatrix} \Delta q_1 \\ \Delta q_2 \end{pmatrix} &= \overrightarrow{PP'} \cdot \mathbf{n} \end{aligned}$$

Chaque contrainte scalaire correspond à une ligne d'un système d'équation. Une contrainte au sens général peut être vue comme un ensemble de contraintes scalaires. Par exemple, nous pouvons écrire l'équation 4.1 comme :

$$\begin{pmatrix} \mathbf{a}_1 \cdot \mathbf{x} & \mathbf{a}_2 \cdot \mathbf{x} \\ \mathbf{a}_1 \cdot \mathbf{y} & \mathbf{a}_2 \cdot \mathbf{y} \end{pmatrix} \begin{pmatrix} \Delta q_1 \\ \Delta q_2 \end{pmatrix} = \begin{pmatrix} \mathbf{c} \cdot \mathbf{x} \\ \mathbf{c} \cdot \mathbf{y} \end{pmatrix}$$

Nous avons une équation à deux inconnues, ce qui dans le cas général fournit un espace de solutions de dimension 1. Plus généralement, pour un système de n équations à m inconnues, on peut s'attendre à un espace de solutions de dimension $m - n$. Toutefois, les systèmes peuvent présenter des "déficiences de rang" quand une colonne de la matrice est combinaison linéaire des autres. Le mécanisme sur la figure 4.4 a beau posséder trois degrés de liberté, ils sont tous dans le plan (x, y) et il ne pourra jamais satisfaire une contrainte en z . Par contre, la dimension de l'espace des solutions dans le plan augmente de 1.

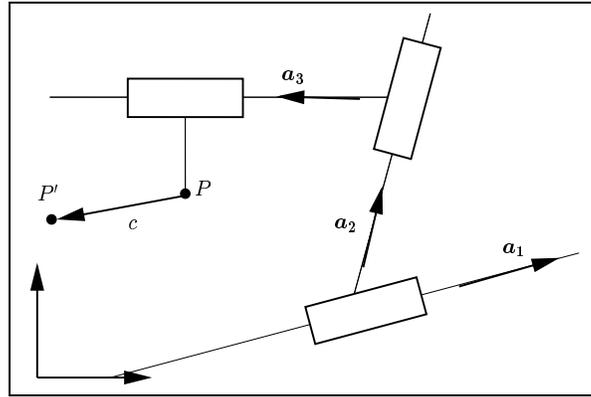


FIG. 4.4 – Mécanisme redondant. Les trois degrés de liberté dans le plan augmentent la dimension de l'espace des solutions. Ce mécanisme ne peut cependant pas satisfaire de contraintes dans la direction z .

4.2 Équation non linéaires

Les degrés de liberté en rotation induisent des fonctions trigonométriques dans les équations géométriques, rendant celles-ci non linéaires. On en voit quelques conséquences sur la figure 4.5. Il y a deux manières de mener le point P au point P' , mais aucune d'atteindre P'' qui est trop loin.

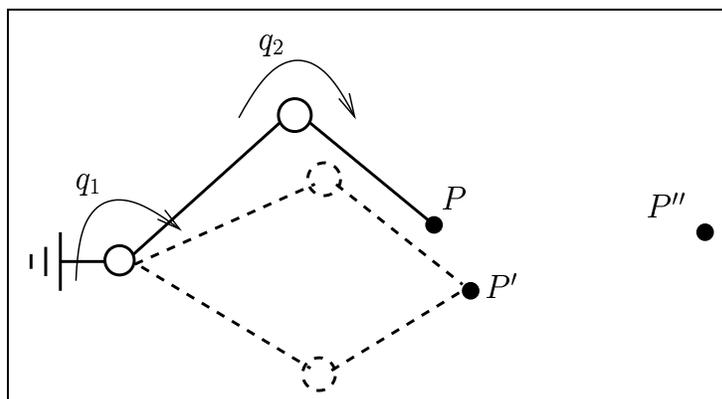


FIG. 4.5 – Un mécanisme avec rotations. Un même type de contrainte peut avoir une, plusieurs ou aucune solution.

4.2.1 Linéarisation. Matrice Jacobienne.

L'équation de cinématique du solide (3.4) nous donne, en remarquant que $\frac{dP}{dt} = \frac{dP}{dq} \frac{dq}{dt}$:

$$\frac{\delta P}{\delta q_i} = \mathbf{a}_i \quad (\text{ddl en translation}) \quad (4.2)$$

$$\frac{\delta P}{\delta q_i} = \mathbf{a}_i \wedge \overrightarrow{O_i P} \quad (\text{ddl en rotation}) \quad (4.3)$$

Autrement dit, les variations de coordonnées articulaires en translation se répercutent tel quel sur tout point ; le cas des rotations est illustré sur la figure 4.6, où la variation est orthogonale à la droite menant à l'axe de rotation pour des (infinitement) petites variations d'angle.

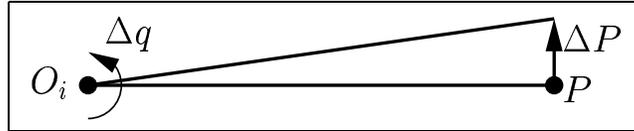


FIG. 4.6 – Effet d'une petite rotation.

Nous considérons la position d'un point P influencée par des ddl numérotés de 1 à n . Nous pouvons regrouper toutes les dérivées partielles en colonnes dans une matrice appelée matrice jacobienne de vitesse (ou de petit déplacement) au point P :

$$\mathbf{J}_p = \frac{dP}{dq} = \left(\begin{array}{ccc} \frac{\delta P}{\delta q_1} & \cdots & \frac{\delta P}{\delta q_n} \end{array} \right)$$

de dimensions $3 \times n$ et qui permet d'approximer :

$$\Delta P \simeq \mathbf{J}_p \Delta q \quad (4.4)$$

Cette formule est exacte si tous les ddl sont des translations. La figure 4.7 illustre les vecteurs-colonnes de la matrice \mathbf{J}_p sur un exemple de mécanisme.

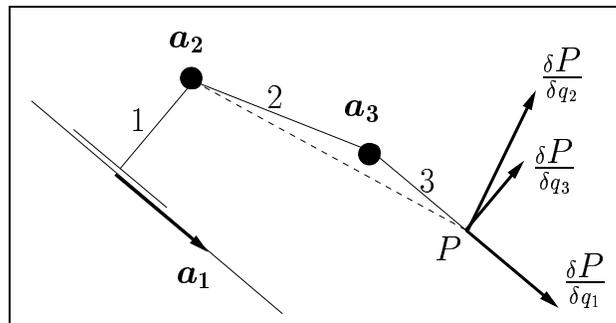


FIG. 4.7 – Les vecteurs colonnes de la jacobienne des vitesses du point P . Les vecteurs orthogonaux au plan de la figure sont illustrés par un disque plein.

Les contraintes scalaires du type $\Delta P \cdot \mathbf{n} = b$ se traduisent par une équation :

$$\left(\begin{array}{ccc} \frac{\delta P}{\delta q_1} \cdot \mathbf{n} & \cdots & \frac{\delta P}{\delta q_n} \cdot \mathbf{n} \end{array} \right) \Delta q = b$$

4.2.2 Contraintes d'orientation

Similairement à la position d'un point, la variation d'orientation d'un repère peut se représenter sous forme linéaire :

$$\mathbf{J}_r \Delta q = \Delta r$$

La quantité Δr est colinéaire à l'axe de rotation et a pour norme l'angle de rotation : $\Delta r = \theta \mathbf{n}$ correspond à une rotation d'axe \mathbf{n} et d'angle θ . Les ddl en rotation influent directement sur l'orientation, tandis que les ddl en translation n'ont pas d'influence. Nous avons donc :

$$\frac{\delta r}{\delta q_i} = \mathbf{0} \quad (\text{ddl en translation}) \quad (4.5)$$

$$\frac{\delta r}{\delta q_i} = \mathbf{a}_i \quad (\text{ddl en rotation}) \quad (4.6)$$

Nous considérons l'orientation d'un repère influencée par des ddl numérotés de 1 à n . La jacobienne d'orientation est :

$$\mathbf{J}_r = \frac{dr}{dq} = \left(\begin{array}{ccc} \frac{\delta r}{\delta q_1} & \dots & \frac{\delta r}{\delta q_n} \end{array} \right)$$

Orientation d'un repère

Soit ${}^0_n \mathbf{R}$ la matrice d'orientation du repère considéré. Nous voulons l'amener à une orientation ${}^0_n \mathbf{R}'$ au moyen d'une rotation $\mathbf{R}_{\mathbf{n},\theta}$ telle que : ${}^0_n \mathbf{R}' = \mathbf{R}_{\mathbf{n},\theta} {}^0_n \mathbf{R}$. Pour appliquer cette rotation au moyen des ddl disponibles nous résolvons donc :

$$\mathbf{J}_r \Delta q = \theta \mathbf{n}$$

Notez que cette solution n'est juste que pour les rotations infiniment petites, car elle ne tient pas compte de la non-commutativité des rotations.

Exemple d'application : orientation d'une caméra.

4.2.3 Orientation d'un vecteur

Nous désirons rendre un vecteur \mathbf{u} colinéaire à un vecteur \mathbf{u}' comme illustré sur la figure 4.8. La rotation la plus directe transformant \mathbf{u} en \mathbf{u}' a son axe orthogonal à \mathbf{u} et \mathbf{u}' , et pour angle l'angle θ entre ces deux vecteurs. L'alignement entre les deux vecteurs nous laissant un degré de liberté, nous pourrions appliquer ensuite une rotation arbitraire autour de \mathbf{u}' . La rotation autour de \mathbf{v} , orthogonal à \mathbf{u} et \mathbf{n} , doit quant à elle être nulle. L'alignement de \mathbf{u} sur \mathbf{u}' nécessite donc de poser les équations :

$$\left(\begin{array}{ccc} \frac{\delta r}{\delta q_1} \cdot \mathbf{n} & \dots & \frac{\delta r}{\delta q_n} \cdot \mathbf{n} \\ \frac{\delta r}{\delta q_1} \cdot \mathbf{v} & \dots & \frac{\delta r}{\delta q_n} \cdot \mathbf{v} \end{array} \right) = \left(\begin{array}{c} \theta \\ 0 \end{array} \right)$$

Exemple d'application : orientation d'un axe de perceuse.

4.3 Résolution des équations linéaires

Soit $\mathbf{J}_{(n \times m)} \delta \mathbf{q}_{(m)} = \mathbf{c}_{(n)}$ le système de n équations et m inconnues à résoudre. Nous considérons une matrice sans déficience de rang, sachant qu'une déficience entraîne une diminution du nombre d'inconnues indépendantes.

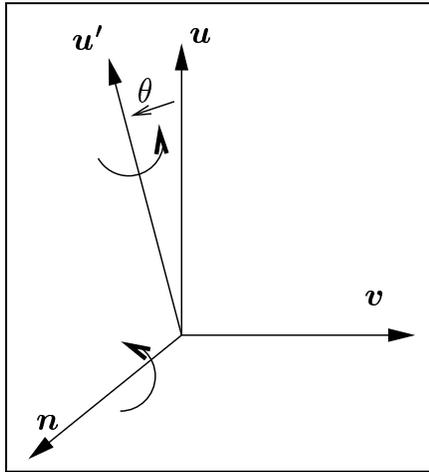


FIG. 4.8 – Alignement d’un vecteur sur un autre.

4.3.1 Matrice carrée

Si $n = m$ et sous les hypothèses établies, le système a une solution unique. Il peut se résoudre par la factorisation LU de \mathbf{J} .

4.3.2 Plus d’inconnues que d’équations

Dans ce cas le système a une infinité de solutions. La pseudo-inverse de \mathbf{J} , notée \mathbf{J}^+ , nous permet d’en calculer une :

$$\delta \mathbf{q} = \mathbf{J}^+ \mathbf{c} \quad (4.7)$$

$$\text{avec } \mathbf{J}^+ = \mathbf{J}^T (\mathbf{J} \mathbf{J}^T)^{-1} \quad (4.8)$$

Cette solution minimise $\delta \mathbf{q}^2$ parmi toutes les solutions.

4.3.3 Plus d’équations que d’inconnues

Dans ce cas on doit trouver un compromis. Une autre formule de pseudo-inverse de \mathbf{J} nous permet d’en calculer un :

$$\delta \mathbf{q} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{c} \quad (4.9)$$

$$(4.10)$$

Cette solution minimise $(\mathbf{J} \delta \mathbf{q} - \mathbf{c})^2$.

4.3.4 Singularités

Certains mécanismes peuvent présenter des configurations spéciales appelées singularités, dans lesquelles des colonnes de la matrice Jacobienne se retrouvent linéairement dépendantes. La méthode de résolution employée jusqu’alors peut échouer. Ces cas se traitent, soit en appliquant une légère perturbation au système pour l’éloigner de la singularité, soit en appliquant une méthode de résolution plus perfectionnée comme celle présentée en section 4.3.5.

4.3.5 Décomposition en valeurs singulières (SVD)

Les méthodes précédentes peuvent échouer, quand par exemple une matrice $n \times n$ est de rang $n-1$, auquel cas aucune inverse ni pseudo-inverse ne peut être calculée. Ces cas se produisent lors de singularités, ou pour certains mécanismes redondants. On peut alors utiliser la décomposition SVD :

$$\mathbf{J} = \mathbf{U}\mathbf{W}\mathbf{V}^T$$

La matrice \mathbf{W} est diagonale et contient des valeurs positives appelées valeurs singulières. Les matrices \mathbf{U} et \mathbf{V} sont orthogonales. Les vecteurs-colonnes de \mathbf{U} associés à une valeur singulière non nulle constituent une base de l'image de \mathbf{J} . Les vecteurs-colonnes de \mathbf{V} associés à une valeur singulière nulle constituent une base du noyau de \mathbf{J} . La solution du système est :

$$\delta\mathbf{q} = \mathbf{V}(\mathbf{W}^{-1})^*\mathbf{U}^T\mathbf{c}$$

où $(\mathbf{W}^{-1})^*$ est une matrice diagonale composée des inverses des valeurs singulières non nulles, et de 0 pour les valeurs singulières nulles.

Pour plus d'inconnues que d'équations la solution minimise $\delta\mathbf{q}^2$, dans le cas contraire elle minimise $(\mathbf{J}\delta\mathbf{q} - \mathbf{c})^2$.

La décomposition SVD est hélas un algorithme très coûteux.

4.4 Résolution itérative des équations non linéaires

Dans le cas d'équations non linéaires, la résolution des équations linéaires à l'aide du jacobien ne nous donne qu'une solution approchée. Il convient donc de réitérer le processus. La figure 4.9 présente l'algorithme itératif de base. Aux alentours des singularités cet algorithme peut

calculer \mathbf{c} tant que $\ \mathbf{c}\ > \epsilon$ calculer \mathbf{J} résoudre $\mathbf{J} \delta\mathbf{q} = \mathbf{c}$ $\mathbf{q} \leftarrow \mathbf{q} + \delta\mathbf{q}$ calculer \mathbf{c}

FIG. 4.9 – Algorithme itératif de base pour la résolution d'équations géométriques.

présenter un comportement d'oscillation autour de la solution. On peut résoudre ce problème en remplaçant l'instruction $\mathbf{q} \leftarrow \mathbf{q} + \delta\mathbf{q}$ par un algorithme de minimisation de $\|\mathbf{c}\|$ sur la droite passant par \mathbf{q} et de direction $\delta\mathbf{q}$. Cette idée est exploitée dans les algorithmes les plus performants ("quasi-Newton", ou à base de gradient conjugué).

Les mécanismes sont souvent pourvus de butées qui empêchent les coordonnées articulaires de dépasser certaines valeurs. On peut adapter l'algorithme précédent pour qu'en cas de dépassement, la valeur soit fixée à la valeur limite. Dans ce cas le ddl correspondant n'est plus libre et la résolution doit se poursuivre aux moyens des autres ddl. L'algorithme présenté en figure 4.10 applique cette idée. Cet algorithme n'est pas garanti de converger vers la solution dans les cas complexes. Il convient alors de le redémarrer en partant de la solution courante et en libérant tous les ddl.

calculer \mathbf{c}
tant que $\ \mathbf{c}\ > \epsilon$
calculer \mathbf{J}
résoudre $\mathbf{J} \delta \mathbf{q} = \mathbf{c}$
$\mathbf{q} \leftarrow \mathbf{q} + \delta \mathbf{q}$
pour chaque ddl i
si $q_i > q_{imax}$ alors
$q_i \leftarrow q_{imax}$
retirer i des ddl disponibles
calculer \mathbf{c}

FIG. 4.10 – Algorithme itératif avec prise en compte de butées articulaires.

4.5 Utilisation de l'espace libre

4.5.1 Projection sur le noyau de \mathbf{J}

Nous nous plaçons dans le cas où les ddl fournissent un espace de solutions appelé espace libre. On peut remarquer que

$$\begin{aligned}
 \forall \mathbf{z} \quad \mathbf{J}(\mathbf{J}^+ \mathbf{J} - \mathbf{I})\mathbf{z} &= \mathbf{J}(\mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J} - \mathbf{I})\mathbf{z} \\
 &= (\mathbf{J}\mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J} - \mathbf{J})\mathbf{z} \\
 &= (\mathbf{J} - \mathbf{J})\mathbf{z} \\
 &= \mathbf{0}
 \end{aligned}$$

L'opérateur $\mathbf{J}^+ \mathbf{J} - \mathbf{I}$ projette les vecteurs de coordonnées articulaires sur le noyau de \mathbf{J} . Il en résulte que

$$\forall \mathbf{z} \quad \mathbf{J}\delta \mathbf{q} = \mathbf{c} \Rightarrow \mathbf{J}(\delta \mathbf{q} + (\mathbf{J}^+ \mathbf{J} - \mathbf{I})\mathbf{z}) = \mathbf{c}$$

4.5.2 Optimisation de poses

Quand les ddl fournissent un espace de solutions, il peut être souhaitable d'y optimiser certains critères. Pour un personnage articulé, on peut par exemple souhaiter que les coordonnées articulaires soient le plus proche possible de valeurs de référence représentant une position "naturelle". On formule alors un coût e à minimiser : $e = \sum_i \alpha_i (q_i - q_{ic})^2$. Le problème se formule mathématiquement par : minimiser e sous la contrainte \mathbf{c} . Il existe des algorithmes perfectionnés pour résoudre ce genre de problèmes. Toutefois, il existe une approche rudimentaire, présentée en figure 4.11, qui peut donner de bons résultats. La difficulté est de définir ce qu'on entend par "petit" pas. Une autre application de l'optimisation de pose consiste à contrôler le centre

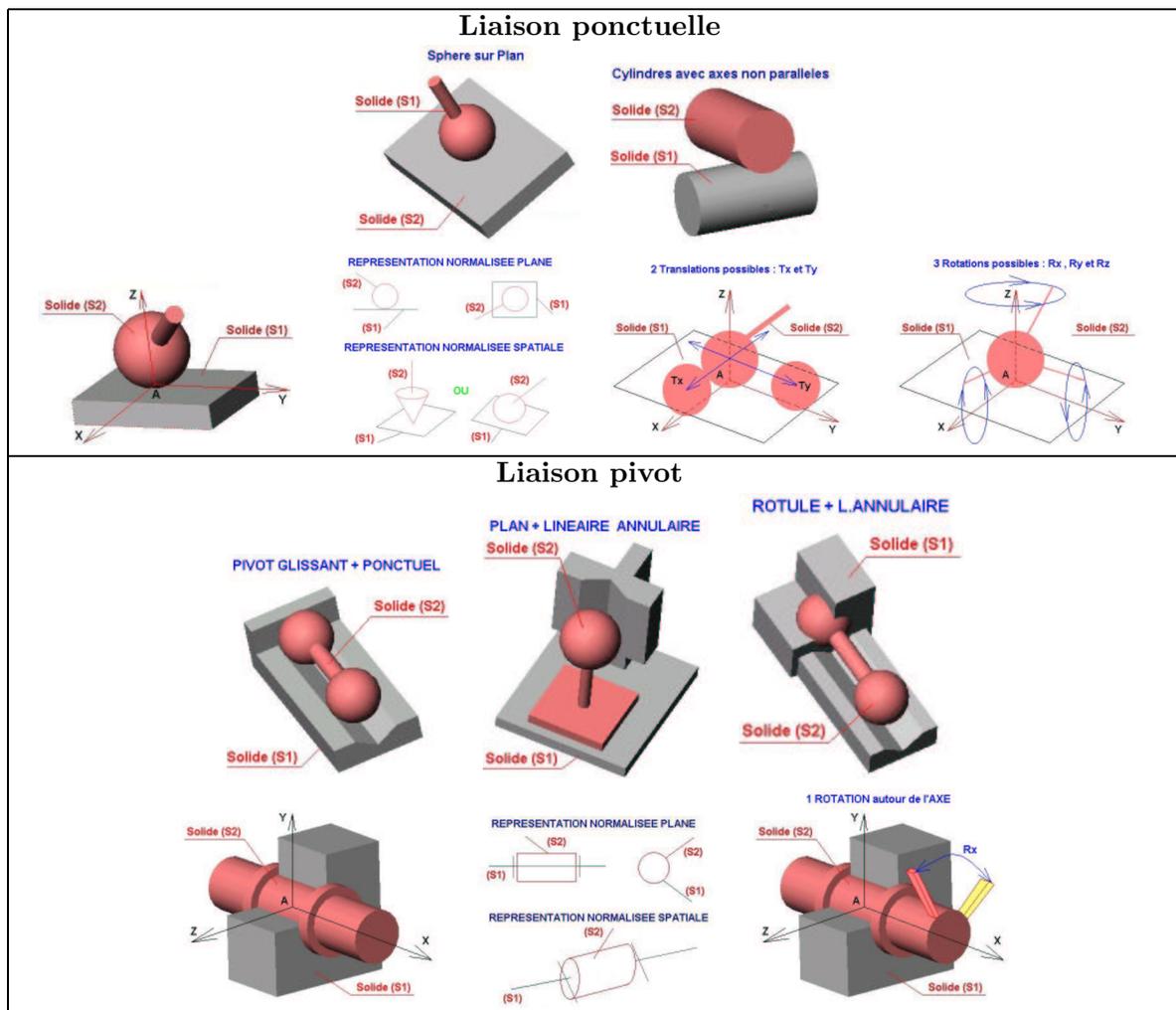
itérer:
résoudre la contrainte
faire un "petit" pas dans la direction $-(\mathbf{J}^+ \mathbf{J} - \mathbf{I}) \overrightarrow{\text{grad}} e$

FIG. 4.11 – Une approche simple pour minimiser un critère e tout en respectant des contraintes.

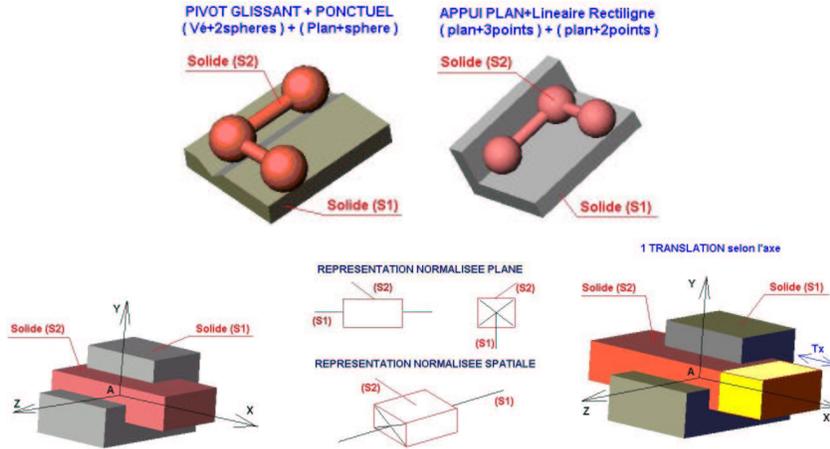
de gravité d'un personnage afin que sa projection au sol soit située entre les pieds.

Annexe A

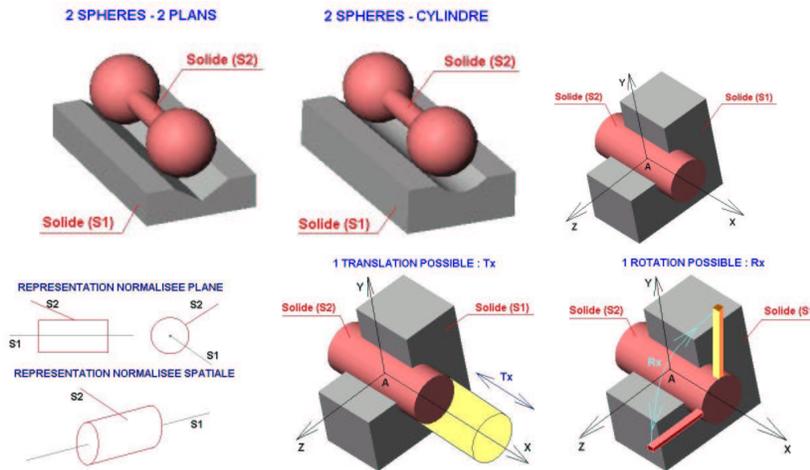
Liaisons cinématiques



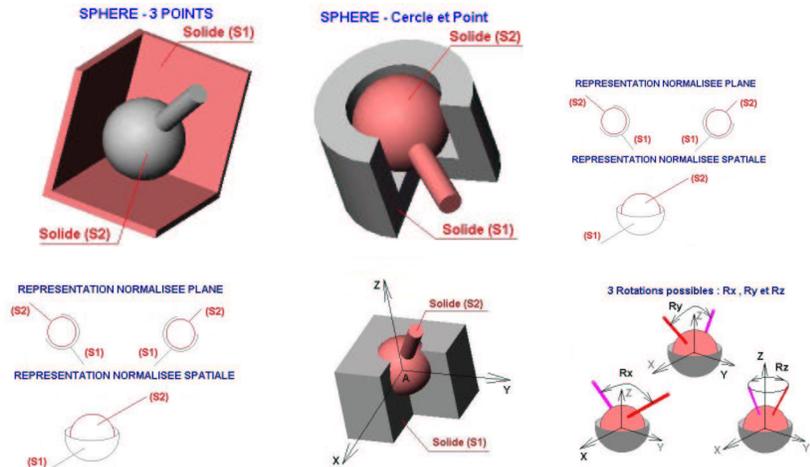
Liaison glissière



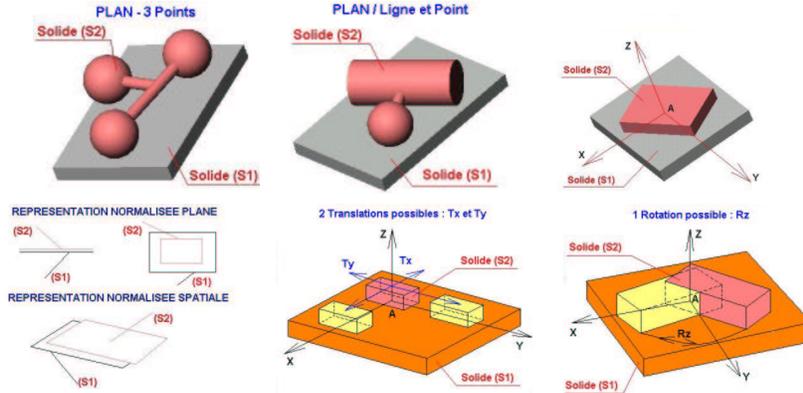
Liaison pivot glissant



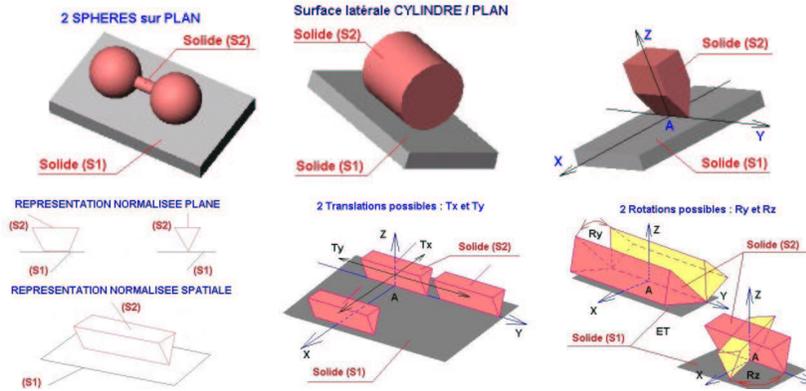
Liaison sphérique



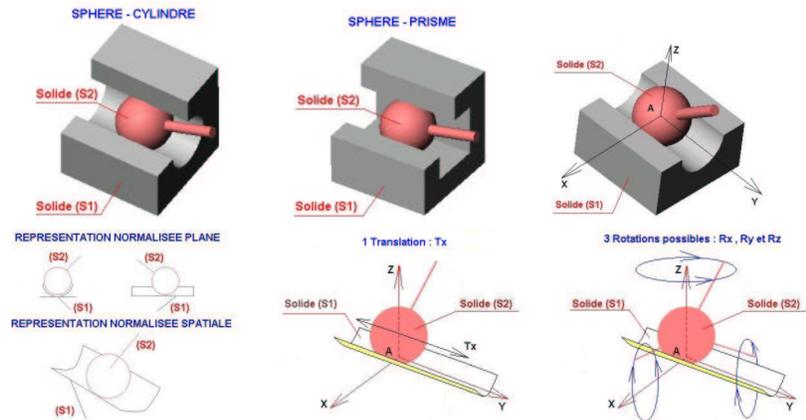
Appui plan



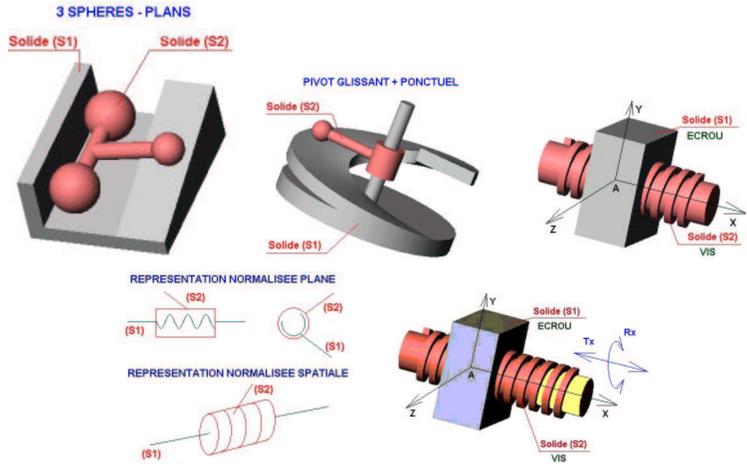
Linéaire rectiligne



Linéaire annulaire



Hélicoidale



Annexe B

Glossaire français-anglais

application de A vers B	mapping from A to B
base	basis
degré de liberté	degree of freedom
espace vectoriel	vector space
liaison cinématique	kinematic joint
liaison appui plan	flat joint
liaison glissière	prismatic joint
liaison pivot	revolute joint (pin joint)
liaison pivot glissant	cylindrical joint
liaison rotule	spherical joint
produit scalaire	dot product
produit vectoriel $u \wedge v$	cross product $u \times v$
repère	frame, coordinate system
valeur propre	eigenvalue
vecteur propre	eigenvector
vecteur unitaire	unit vector

Bibliographie

- [1] Press, Teukolski, Vetterling, and Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992.