

MODELISATION DECLARATIVE ET TRANSFORMATION D'UN OBJET GEOMETRIQUE SOUS CONTRAINTES

Mireille MOINET, Philippe SERRE, Alain RIVIERE

LISMMA Supméca Paris
3 rue Fernand Hainaut
93407 saint Ouen cedex

[mireille.moinet, philippe.serre, alain.riviere]@supmeca.fr,

André CLEMENT

DASSAULT SYSTEMES
9 quai Marcel Dassault
92150 Suresnes
andre.clement@3ds.com

RESUME : Afin de spécifier et résoudre les problèmes géométriques 2D et 3D définis par des contraintes géométriques, nous avons élaboré un modèle déclaratif où les objets et les contraintes sont représentés par un ensemble de vecteurs. Ces vecteurs ne sont pas représentés par leurs coordonnées cartésiennes habituelles mais par leur produit scalaire respectif. C'est alors un tenseur métrique qui définit l'objet géométrique. Pour résoudre le problème, nous avons mis au point un modèle faisant appel à la transformation de l'objet de son état initial à son état final.

MOTS-CLES : contraintes géométriques, géométrie non-cartésienne, modèle tensoriel, perturbation, modélisation déclarative.

1. INTRODUCTION

Un logiciel de CAO se décompose de la manière suivante :

- un modeleur géométrique,
- un outil de visualisation,
- un ensemble d'applications.

Nous allons nous intéresser aux modeleurs géométriques. Il en existe différents types : les modeleurs de type paramétrique regroupent les modeleurs de type B-REP (Boundary Representation), CSG (Constructive Solid Geometry) et hybrides et les modeleurs de type variationnel, d'un niveau d'abstraction plus élevé.

Ainsi, le principe du modèle B-REP est de modéliser la frontière entre l'intérieur et l'extérieur de l'objet volumique souhaité. Cette frontière est décrite par un ensemble de faces limitées par des arêtes elles-mêmes limitées par des sommets. L'objet volumique est parfaitement défini lorsque les supports géométriques de ces objets topologiques sont connus.

Le modèle CSG est de type procédural (et volumique). Dans ce type d'approche, un arbre de construction rend compte de la manière dont l'objet est créé. On y retrouve la liste ordonnée des constructions élémentaires (union, soustraction, etc.) qui s'appliquent sur des primitives volumiques élémentaires (extrusion, révolution, etc.) pour obtenir l'objet volumique souhaité.

Le modèle hybride CSG-BREP, modèle le plus souvent utilisé actuellement, pour lequel, les primitives de base sont, soit celles du modèle CSG (cône, pyramide, cube,...), soit des solides définis par leurs frontières. Lors de chaque opération sur l'arbre de construction,

les éléments frontières (représentant la « peau de l'objet ») sont calculés.

Dans ces trois types d'approche, chaque objet élémentaire est défini par un ensemble de paramètres que le concepteur peut facilement modifier. Ainsi, l'utilisateur peut changer facilement l'objet généré. Il lui suffit de modifier la valeur d'un paramètre définissant la forme d'une primitive et le système se chargera d'exécuter la procédure pour obtenir le nouvel objet. En revanche, si l'utilisateur veut modifier un paramètre qui n'apparaît pas dans l'arbre de construction, alors ce dernier devra revoir entièrement la construction de l'objet. Cette remarque peut sembler anodine, elle est cependant fondamentale. En effet, durant son cycle de vie, une maquette numérique est amenée à être régulièrement modifiée pour une multitude de raisons différentes. Et la plupart du temps, ces modifications sont extrêmement difficiles à anticiper. Par conséquent, l'arbre de construction de la maquette numérique le mieux adapté à supporter ces modifications est impossible à déterminer a priori.

Les modeleurs variationnels apportent une réponse à l'inconvénient cité ci-dessus. En effet, avec ces modèles, on ne décrit pas la procédure permettant de créer les objets mais les objets élémentaires et les contraintes qui les lient. On retrouve ce genre d'approche dans des modules de certains logiciels actuels pour réaliser des esquisses 2D (sketcher (Lesage, 2002)) et/ou pour définir des assemblages de pièces (assembly).

Dans cet article, nous nous appuyons sur un modèle variationnel déjà exposé par notre équipe (Serré, 2000). En 2D les entités géométriques qui décrivent l'objet sont : le point et la droite et en 3D ce sont : le point, la droite, le plan, le cylindre et la sphère. A ces objets, sont ajoutées des contraintes géométriques qui peuvent être du type : coïncidence, orthogonalité, parallélisme, distance et angle. Nous les décrirons dans la section suivante. Ce modèle a la particularité d'être basé sur une représentation vectorielle des entités géométriques et des contraintes. Nous décrirons dans la troisième section l'approche retenue à savoir, une modélisation tensorielle qui ne fait pas intervenir de repère. Dans la section 4, nous détaillerons la mise en équation du problème. Chaque type de spécifications imposées par le concepteur se traduira par une équation algébrique déterminée à partir du tenseur métrique.

Par rapport aux travaux déjà réalisés par l'équipe, ce document présente une nouvelle méthode d'expression des équations. Elle est basée sur la perturbation du tenseur métrique. Le but de la résolution est alors de trouver la transformation qui permet de passer d'une géométrie initiale à une géométrie finale qui respecte les contraintes spécifiées. Cette méthode est particulièrement bien adaptée à la CAO car l'utilisateur commence toujours par « dessiner » une forme initiale qu'il modifie petit à petit pour obtenir l'objet souhaité.

Pour finir, nous avons mis en œuvre une méthode itérative permettant de résoudre pas à pas le système d'équations. Le principe de cette méthode est similaire à celle de Newton-Raphson, elle sera décrite dans la section 5 où nous illustrerons également notre proposition sur un exemple.

2. LES OBJETS ET LES CONTRAINTES

La déclaration des contraintes géométriques est assurée par le modèle SATT « Surfaces Associées Topologiquement et Technologiquement ». Ce modèle fut d'abord utilisé pour décrire le tolérancement de pièces mécaniques ((Rivière, 1993) (Clement *et al*, 1994)). Il est basé sur la notion de classe d'invariance des surfaces. On dénombre sept classes de surfaces invariantes : quelconque, prismatique, de révolution, hélicoïdale, cylindrique, plane et sphérique. Chaque classe est définie par son degré d'invariance et les déplacements en rotation et en translation correspondant. A chaque classe d'invariance de SATT, nous définissons des Eléments Géométriques de Référence Minimum (EGRM) formant eux-mêmes un SATT de la même classe d'invariance mais composés seulement de trois éléments parmi le point, la droite ou le plan.

On exprime ainsi treize contraintes de positionnement relatif entre deux EGRM. Celles-ci permettent

d'exprimer le positionnement relatif de toutes les combinaisons de deux classes entre elles (Voir tableau 1)

Les contraintes que nous allons traiter sont de trois types : les contraintes de type distance (distance entre deux points, ...), d'angle (angle entre deux droite, ...) et de coïncidence (coïncidence de deux points, ...). Nous verrons dans la section 3 que ces familles de contraintes génèrent trois types d'équations.







	EGRM droite D_1 	EGRM Plan P_1 	EGRM point O_1 
EGRM droite D_2 	C11 : $D_1 = D_2$ coïncidence Classe cylindrique C12 : $D_1 // D_2$ $D_1 \neq D_2$ distance Classe prismatique C13 : sinon angle et distance Classe quelconque	C8 : $D_2 \perp P_1$ perpendicularité Classe révolution C9 : $D_2 // P_1$ distance Classe prismatique C10 : sinon angle Classe quelconque	C4 : $O_1 \in D_2$ coïncidence Classe révolution C5 : sinon distance Classe quelconque
EGRM Plan P_2 	sym	C6 : $P_1 // P_2$ coïncidence Classe plane C7 : sinon distance Classe prismatique	C3 : distance Classe révolution
EGRM point O_2 	sym	sym	C1 : $O_1 = O_2$ coïncidence Classe sphérique C2 : sinon distance Classe révolution

Tableau 1. Les 13 contraintes SATT et leur classe d'invariance

3. MODELISATION TENSORIELLE

Dans la plupart des modèles géométriques définis par contraintes, les points caractéristiques de l'objet sont représentés dans une base cartésienne et les coordonnées de ces points sont les variables du système d'équations algébriques à résoudre. Il est connu qu'un mauvais choix de repère d'expression peut impliquer des problèmes de configurations du système d'équations et engendre des problèmes de résolution (Durand, 1998) (MacDonald, *et al*, 2001) (Lamure et Michelucci, 1995).

Le modèle proposé s'appuie sur une modélisation géométrique non-cartésienne. Cette approche fait intervenir un ensemble de vecteurs décrivant les objets et les contraintes. A partir de cet ensemble de vecteurs, on construit un tenseur métrique. Ce tenseur métrique définit complètement la métrique de l'objet et ne dépend pas d'une base d'expression particulière.

Un avantage supplémentaire de cette approche est la possibilité d'assurer la cohérence des spécifications en vérifiant les propriétés mathématiques du tenseur métrique (symétrique, semi défini positif, ...). Par exemple, à partir du calcul de certains déterminants on peut savoir si le problème aura une solution...

3.1 Représentation de la géométrie et des contraintes

Les entités géométriques que l'on manipule sont le point, la droite et le plan, ce que nous appelons les EGRM. A ce modèle de données on applique un modèle vectoriel associé. Ainsi :

- le point est représenté par un point ;
- la droite par un vecteur et un point ;
- le plan par un vecteur et un point.

Prenons un exemple, soient \vec{e}_1 et \vec{e}_2 deux éléments d'un espace vectoriel E. Nous les choisissons comme vecteurs directeurs des droites passant par les points M et N.

Nous ajoutons un nouveau vecteur unitaire à cet espace : \vec{e}_3 .

Appelons L_3 la longueur entre les deux points M et N, nous avons ainsi : $\overline{MN} = L_3 \vec{e}_3$

L'angle entre les droites passant par \vec{e}_1 et \vec{e}_2 sera noté α_{12} et pourra être obtenu en effectuant le produit scalaire entre ces deux vecteurs unitaires.

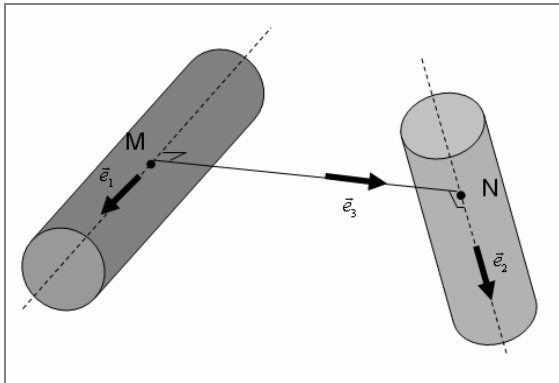


Figure 1. Vectorisation

3.3. Le modèle géométrique

3.3.1 Quelques rappels

Tout vecteur \vec{x} dans un espace vectoriel à n dimensions dans lequel on a défini une base $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_n$ se met sous la forme :

$$\vec{x} = \sum_{i=1}^n \vec{e}_i x^i$$

Où x^1, x^2, \dots, x^n sont les composantes **contravariantes** du vecteur. Dans une autre base $\vec{E}_1, \vec{E}_2, \dots, \vec{E}_n$ il aura les coordonnées X^1, X^2, \dots, X^n .

Si les bases sont liées par les relations

$$\vec{E}_j = \sum_{i=1}^n a_j^i \vec{e}_i$$

$$\vec{e}_i = \sum_{j=1}^n b_i^j \vec{E}_j$$

Avec les matrices a_j^i et b_i^j qui sont deux matrices inverses :

$$\sum_{i=1}^n a_j^i b_i^k = \sum_{i=1}^n b_k^i a_i^j = \begin{cases} 1, & \text{si } k=j \\ 0, & \text{si } k \neq j \end{cases}$$

Alors les coordonnées contravariantes se transforment comme

$$x^j = \sum_{i=1}^n a_i^j X^i$$

$$X^i = \sum_{j=1}^n b_j^i x^j$$

Par la suite on utilisera la convention d'Einstein pour simplifier l'écriture : si dans une expression le même indice se répète en haut et en bas, on sous-entend la sommation sur cet indice.

Dans un espace euclidien, le produit scalaire de deux vecteurs \vec{x} et \vec{y} s'écrit :

$$\langle \vec{x}, \vec{y} \rangle = \sum_{i=1}^n \sum_{j=1}^n \langle \vec{e}_i, \vec{e}_j \rangle x^i y^j$$

On définit le tenseur métrique comme

$$g_{ij} = \langle \vec{e}_i, \vec{e}_j \rangle = g_{ji}$$

A l'aide du tenseur métrique on définit les coordonnées **covariantes** d'un vecteur

$$x_i = \langle \vec{e}_i, \vec{x} \rangle = \langle \vec{e}_i, \vec{e}_j x^j \rangle = g_{ij} x^j$$

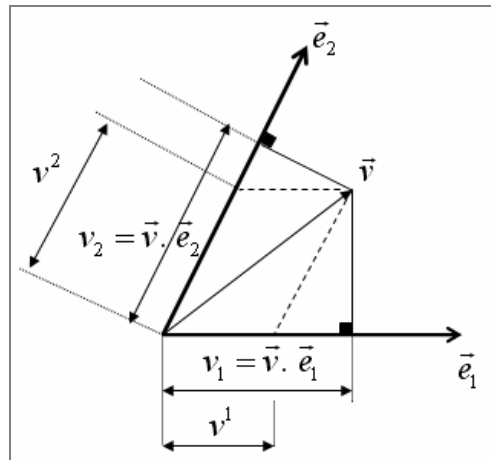


Figure 2. Représentation des composantes covariantes et contravariantes en 2D

Les composantes contravariantes s'obtiennent en effectuant une projection oblique du vecteur \vec{v} sur les vecteurs de la base $\{\vec{e}_1, \vec{e}_2\}$.

$$\vec{v} = v^1 \vec{e}_1 + v^2 \vec{e}_2$$

Les composantes covariantes s'obtiennent en projetant le vecteur \vec{v} sur les vecteurs de la base $\{\vec{e}_1, \vec{e}_2\}$.

$$v_1 = \langle \vec{v}, \vec{e}_1 \rangle$$

$$v_2 = \langle \vec{v}, \vec{e}_2 \rangle$$

3.3.2 Modèle proposé

Un objet géométrique est représenté par une liste de vecteurs. Soit V un ensemble de n vecteurs non normés.

On construit le tenseur métrique $\overline{\overline{G}}(V)$:

$$\overline{\overline{G}}(V) = \begin{pmatrix} \vec{v}_1 \\ \vec{v}_2 \\ \vdots \\ \vec{v}_n \end{pmatrix} \otimes (\vec{v}_1 \quad \vec{v}_2 \quad \dots \quad \vec{v}_n)$$

3.4. Perturbation du tenseur métrique et des longueurs

Si V est un ensemble de vecteurs de rang r et T une transformation vectorielle telle que $V' = T(V)$. Alors $\overline{\overline{G}}(V')$, tel que :

$$\overline{\overline{G}}(V') = T \cdot \overline{\overline{G}}(V) \cdot {}^t T \quad (1)$$

est un tenseur métrique de rang r .

Notons $\Omega(V)$ la perturbation d'un ensemble de vecteurs V : $\Delta V = \Omega \cdot V$, V_{init} l'ensemble des vecteurs de la géométrie initiale et V_{final} ceux de la géométrie finale.

Sachant que :

$$\begin{aligned} V_{final} &= V_{init} + \Delta V_{init} \text{ et } \overline{\overline{G}}(V_{final}) = V_{final} \otimes {}^t V_{final} \\ \overline{\overline{G}}(V_{final}) &= (V_{init} + \Delta V_{init}) \otimes {}^t (V_{init} + \Delta V_{init}) \\ \overline{\overline{G}}(V_{final}) &= (V_{init} + \Omega \cdot V_{init}) \otimes {}^t (V_{init} + \Omega \cdot V_{init}) \\ \overline{\overline{G}}(V_{final}) &= (I + \Omega) \overline{\overline{G}}(V_{init}) (I + \Omega) \end{aligned} \quad (2)$$

La formule (2) ci-dessus représente les variations de l'objet géométrique.

Les éléments de la matrice de perturbation Ω sont les inconnues de notre problème.

4. MISE EN EQUATION DU PROBLEME

Les contraintes géométriques spécifiées par l'utilisateur induisent un ensemble d'équations. Nous avons classé ces équations en trois types suivant qu'elles proviennent de spécifications de longueurs, d'angles ou de fermeture vectorielle.

Dans la suite du document, la case ij du tenseur $\overline{\overline{G}}(V_{init})$ sera notée $G_{init_{ij}}$, (ie. pour $\overline{\overline{G}}(V_{final})$)

4.1. Equations de longueurs

Par définition, un élément de la diagonale du tenseur métrique $G_{init_{ii}}$ (resp. $G_{final_{ii}}$) représente la longueur au carré d'un vecteur initial (resp. final). On appelle S_{L_i} la spécification de longueur du vecteur \vec{v}_i imposée par le concepteur. Ainsi,

$$\sqrt{G_{final_{ii}}} = S_{L_i} \quad (3)$$

Sachant que chaque case du tenseur final s'exprime, en fonction des éléments de la matrice de perturbation Ω , de la manière suivante:

$$G_{final_{ij}} = \sum_{p=1}^n \sum_{q=1}^n (\delta_{ip} + \omega_{ip}) G_{init_{pq}} (\delta_{qj} + \omega_{jq}) \quad (4)$$

Où δ_{ip} représente le symbole de kronecker qui vaut 1 si $i=p$ et 0 sinon.

En développant cette dernière expression au 1^{er} ordre, nous obtenons :

$$G_{final_{ij}} \cong G_{init_{ij}} + \sum_{p=1}^n (\omega_{ip} G_{init_{pj}} + G_{init_{ip}} \omega_{jp}) \quad (5)$$

En particulier, pour les éléments de la diagonale :

$$G_{final_{ii}} \cong G_{init_{ii}} + 2 \sum_{p=1}^n (\omega_{ip} G_{init_{pi}})$$

Et en développant l'équation (3), nous obtenons :

$$\frac{1}{\sqrt{G_{init_{ii}}}} \sum_{p=1}^n (\omega_{ip} G_{init_{pi}}) \cong S_{L_i} - \sqrt{G_{init_{ii}}} \quad (6)$$

Les termes S_{L_i} et $G_{init_{ii}}$ sont connus. Les inconnues de cette équation sont les ω_{ip} .

4.2. Equations d'angles

Par définition, un élément G_{ij} d'un tenseur métrique représente le produit scalaire entre les vecteurs \vec{v}_i et \vec{v}_j .

Le concepteur souhaite imposer que le cosinus de l'angle entre \vec{v}_i et \vec{v}_j respecte la spécification $S\alpha_{ij}$, ceci se traduit par l'équation suivante :

$$G_{final_{ij}} = \sqrt{G_{final_{ii}}} \sqrt{G_{final_{jj}}} S\alpha_{ij} \quad (7)$$

Nous réécrivons l'équation (7) de façon à passer tous les termes de G_{final} à gauche de signe égal :

$$G_{final_{ij}} \cdot (G_{final_{ii}})^{-1/2} (G_{final_{jj}})^{-1/2} = S\alpha_{ij}$$

En développant $G_{final_{ij}}$ provenant de l'équation (4) pour faire apparaître les termes de la matrice de perturbation et en linéarisant au 1^{er} ordre, on obtient l'équation suivante :

$$\sum_{p=1}^n \left(G_{init_{ip}} - \frac{G_{init_{ij}} \cdot G_{init_{pj}}}{G_{init_{jj}}} \right) \omega_{jp} + \left(G_{init_{pj}} - \frac{G_{init_{ij}} \cdot G_{init_{pi}}}{G_{init_{ii}}} \right) \omega_{ip} \quad (8)$$

$$\cong \sqrt{G_{init_{ii}}} \sqrt{G_{init_{jj}}} S\alpha_{ij} - G_{init_{ij}}$$

$G_{init_{ij}}$ et $S\alpha_{ij}$ sont des termes connus. Les inconnues sont ω_{jp} et ω_{ip} .

4.3. Equations de fermeture vectorielle

Ces équations représentent le fait que certains vecteurs forment des boucles lorsque le concepteur définit certaines contraintes de coïncidence. Bien évidemment ces dernières doivent être respectées dans la configuration finale. Il convient de trouver une base de boucles indépendantes, de façon à écrire les équations de boucles correspondantes.

Pour chaque boucle, nous écrivons que la somme des vecteurs qui la composent est nulle.

$$\sum_{p \in Boucle} \varepsilon_p \vec{v}_p = \vec{0} \quad (9)$$

Avec p l'indice des vecteurs composant la boucle étudiée et ε_p le signe du vecteur v_p dans cette boucle.

Pour écrire les relations algébriques correspondantes nous avons étudié deux méthodes : la méthode des coupes métriques et la méthode des projections.

Méthode des coupes métriques

Cette méthode consiste à écrire que les longueurs de certains vecteurs sommes se correspondent.

Soient V un ensemble de n vecteurs d'un espace euclidien et V_1 et V_2 deux ensemble complémentaires de V ($V_1 \cap V_2 = \emptyset$ et $V_1 \cup V_2 = V$). Nous pouvons associer à chaque sous-ensemble V_i une quantité que nous appellerons *coupe métrique*, correspondant à la somme de tous les termes de la sous-matrice de Gram associée.

Lorsque l'ensemble V forme une somme vectorielle nulle, les coupes métriques de toutes les paires de sous-ensembles complémentaires sont égales. En effet, si $\vec{v}_1 + \dots + \vec{v}_n = \vec{0}$ alors :

$$\forall k \in [1, n], (\vec{v}_1 + \dots + \vec{v}_k) = -(\vec{v}_{k+1} + \dots + \vec{v}_n)$$

$$(\vec{v}_1 + \dots + \vec{v}_k)^2 = (\vec{v}_{k+1} + \dots + \vec{v}_n)^2$$

La coupe métrique d'un ensemble de vecteurs V_i peut donc être interprétée comme la norme du vecteur \vec{S} construit à partir de ses vecteurs.

Pour un contour fermé composé de n vecteurs nous aurons,

$$\forall n \in [1, n],$$

$$G(\vec{v}_1, \dots, \vec{v}_n) = \begin{pmatrix} \langle \vec{v}_1, \vec{v}_1 \rangle & \dots & \langle \vec{v}_1, \vec{v}_k \rangle & \dots & \langle \vec{v}_1, \vec{v}_{k+1} \rangle & \dots & \langle \vec{v}_1, \vec{v}_n \rangle \\ \vdots & & \vdots & & \vdots & & \vdots \\ \langle \vec{v}_k, \vec{v}_1 \rangle & \dots & \langle \vec{v}_k, \vec{v}_k \rangle & \dots & \langle \vec{v}_k, \vec{v}_{k+1} \rangle & \dots & \langle \vec{v}_k, \vec{v}_n \rangle \\ \langle \vec{v}_{k+1}, \vec{v}_1 \rangle & \dots & \langle \vec{v}_{k+1}, \vec{v}_k \rangle & \dots & \langle \vec{v}_{k+1}, \vec{v}_{k+1} \rangle & \dots & \langle \vec{v}_{k+1}, \vec{v}_n \rangle \\ \vdots & & \vdots & & \vdots & & \vdots \\ \langle \vec{v}_n, \vec{v}_1 \rangle & \dots & \langle \vec{v}_n, \vec{v}_k \rangle & \dots & \langle \vec{v}_n, \vec{v}_{k+1} \rangle & \dots & \langle \vec{v}_n, \vec{v}_n \rangle \end{pmatrix}$$

$$= \begin{pmatrix} A_k & | & B_k \\ \hline C_k & | & D_k \end{pmatrix}$$

La notion de coupe métrique permet d'établir la relation suivante :

$$\sum (A_k)_{ij} = \sum (D_k)_{mp}$$

La symétrie des matrices de Gram induit aussi que

$$\sum (B_k)_{ij} = \sum (C_k)_{mp}$$

L'inconvénient de ce mode de génération des équations est le nombre de paramètres qu'elle fait intervenir lorsque la boucle vectorielle compte beaucoup de vecteurs.

Méthode des projections

Cette méthode consiste à écrire que les projections du vecteur somme sur d vecteurs indépendants sont nulles (où d est la dimension de l'espace).

On projette l'équation (9) sur des vecteurs de projection, marqués par l'indice i :

$$\sum_{p \in Boucle} \varepsilon_p \langle \vec{v}_i, \vec{v}_p \rangle = \sum_{p \in Boucle} \varepsilon_p G_{final_{ip}} = 0 \quad (10)$$

Il est à noter que la base des vecteurs indépendants $\{\vec{v}_i\}$, avec $i=1,2$ en 2D et $i=1,2,3$ en 3D, est extraite du champ de vecteurs V .

En développant $G_{final_{ij}}$ provenant de l'équation (4) pour faire apparaître les termes de la matrice de perturbation et en linéarisant au 1^{er} ordre l'équation (10), on obtient l'équation suivante :

$$\sum_{p \in \text{Boucle}} \varepsilon_p \sum_{q=1}^n G_{init_{qp}} \omega_{iq} + G_{init_{iq}} \omega_{pq} \quad (11)$$

$$\cong - \sum_{p \in \text{Boucle}} \varepsilon_p G_{init_{ip}}$$

Les termes connus sont les $G_{init_{..}}$ alors que les ω_{iq} et ω_{pq} sont les inconnues du système.

On constate que le choix des vecteurs de projection est très important puisque le nombre d'inconnues intervenant dans le système en est directement lié. En effet, si le vecteur choisi participe à k angles spécifiés, alors le nombre de paramètres inconnus est égal à $(n-1) \cdot k$. Par conséquent, nous sélectionnons comme vecteurs de projections ceux sur lesquels le maximum d'angles sont spécifiés.

Dans la suite de nos investigations, nous avons développé un prototype basé sur la méthode des projections.

5. RESOLUTION

Le système à résoudre est composé des relations non-linéaires (3),(7) et (10). Les termes connus sont les éléments du tenseur initial G , les spécifications d'angles $S\alpha_{ij}$ et les spécifications de longueur S_{L_i} .

Résoudre le problème revient à trouver les éléments de la matrice de perturbation ω_{ij} qui définissent la transformation de l'état initial à l'état final de l'ensemble des vecteurs.

Après une analyse qualitative du système à résoudre :

- par le dénombrement du nombre d'équations,
- par le dénombrement du nombre d'inconnues,
- par l'analyse qualitative des spécifications imposées par l'utilisateur,
- et aussi par une analyse qualitative de certaines spécifications angulaires [SER01],

nous pouvons lancer la tâche de résolution.

La méthode la plus utilisée pour résoudre ce genre de système ($F(X)=0$) est celle de Newton-Raphson. Cependant, nous avons mis en œuvre une variante de cette méthode que nous présenterons au paragraphe suivant. Finalement, nous illustrerons cette méthode sur l'exemple d'un tétraèdre.

5.1 Description de la méthode

La méthode de résolution implémentée est une méthode itérative résolvant, pas à pas, les équations linéarisées (6), (8) et (11).

A chaque pas, le système linéaire à résoudre est généralement sous-contraint : seules interviennent les équations correspondant aux termes spécifiés du tenseur final.

Notre système s'écrit sous la forme générique $AX=B$. On peut chercher une solution approchée : on appelle solution des moindres carrés du système linéaire $AX=B$ la solution de norme minimale du système si elle existe et sinon la solution X minimisant la norme quadratique de $AX-B$: $\text{Min}_A \|AX - B\|_2$

La solution des moindres carrés d'un système linéaire existe toujours et est unique. On montre qu'elle est donnée par la formule :

$$X = A^\dagger B \quad (12)$$

Où A^\dagger est la pseudo-inverse de A .

Dans le prototype que nous avons implémenté sous Matlab®, on obtient la pseudo-inverse de A par la fonction $\text{pinv}(A)$.

Après résolution du système linéaire, la matrice de perturbation Ω est reconstruite. Nous calculons alors, en appliquant l'équation (2), un tenseur $G_{courant}$ qui servira de paramètre d'entrée pour l'itération suivante.

Les itérations sont poursuivies jusqu'à ce que les spécifications soient vérifiées.

On peut représenter l'algorithme sous la forme du diagramme suivant :

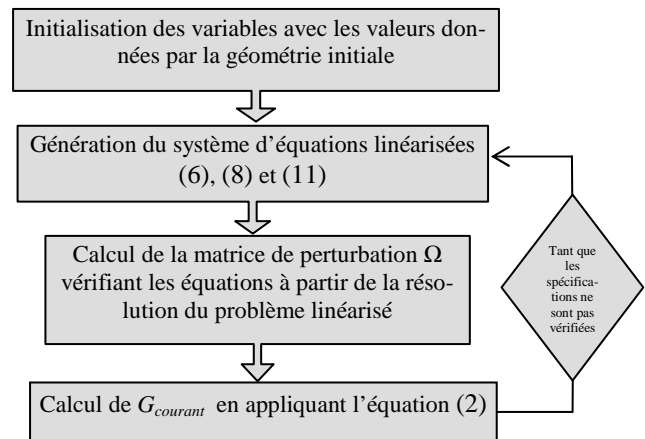


Figure 3. Algorithme implémenté

5.2 Exemple iso-contraint en 3D

Nous avons choisi d'illustrer cette méthode par un exemple en 3D, le tétraèdre.

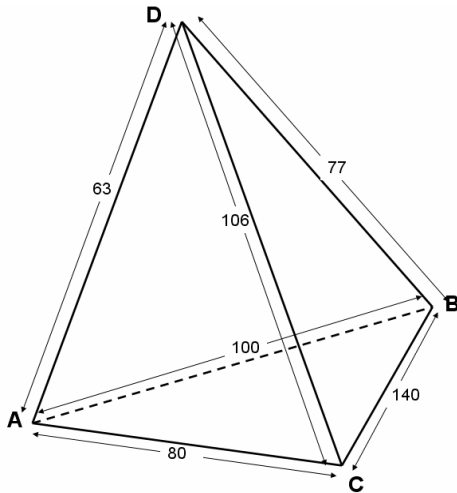


Figure 4. Objet initial

Le tétraèdre initial de l'exemple traité est défini par le tenseur G_{init} .

$G_i =$					
1.0e+004 *					
1.0000	-0.1600	0.4020	-1.1600	-0.5980	0.5620
-0.1600	0.6400	-0.0433	0.8000	0.1166	-0.6833
0.4020	-0.0433	0.3969	-0.4454	-0.0051	0.4402
-1.1600	0.8000	-0.4454	1.9600	0.7146	-1.2454
-0.5980	0.1166	-0.0051	0.7146	0.5929	-0.1218
0.5620	-0.6833	0.4402	-1.2454	-0.1218	1.1236

Sur cette image, G_i équivaut au G_{init} précédemment défini et l'ordre des vecteurs est le suivant : AB, AC, AD, BC, BD et CD

Le scénario que nous avons choisi d'étudier est le suivant : l'utilisateur souhaite modifier cette géométrie initiale en imposant de nouvelles dimensions aux barres et de nouveaux angles. Les spécifications de longueurs sont les suivantes :

- AB reste égal à 100,
- $AC = 85$ au lieu de 80,
- $AD = 65$ au lieu de 63,
- $BD = 75$ au lieu de 77,
- $CD = 110$ au lieu de 106.

La spécification angulaire est :

$$\text{angle}(\overrightarrow{AB}, \overrightarrow{AC}) = 90^\circ \text{ au lieu de } 101^\circ$$

Nous avons développé notre prototype sous Matlab® en suivant le principe de résolution développé dans la section précédente. Les performances obtenues sont les suivantes :

- le système converge en 5 itérations en 0.0520s,
- les spécifications de longueurs et angulaires sont respectées à 10^{-11} près.

L'objet modifié est défini par le tenseur G_{final} . Les spécifications données par l'utilisateur (valeurs encadrées) sont respectées. Le rang du tenseur métrique vaut 3 et correspond bien à la dimension de l'objet recherché.

$G_n =$					
1.0e+004 *					
1.0000	-0.0000	0.4300	-1.0000	-0.5700	0.4300
-0.0000	0.7225	-0.0325	0.7225	-0.0325	-0.7550
0.4300	-0.0325	0.4225	-0.4625	-0.0075	0.4550
-1.0000	0.7225	-0.4625	1.7225	0.5375	-1.1850
-0.5700	-0.0325	-0.0075	0.5375	0.5625	0.0250
0.4300	-0.7550	0.4550	-1.1850	0.0250	1.2100

Sur cette image, G_n équivaut au G_{final} précédemment défini et l'ordre des vecteurs reste le suivant : AB, AC, AD, BC, BD et CD

Nous pouvons illustrer la géométrie obtenue :

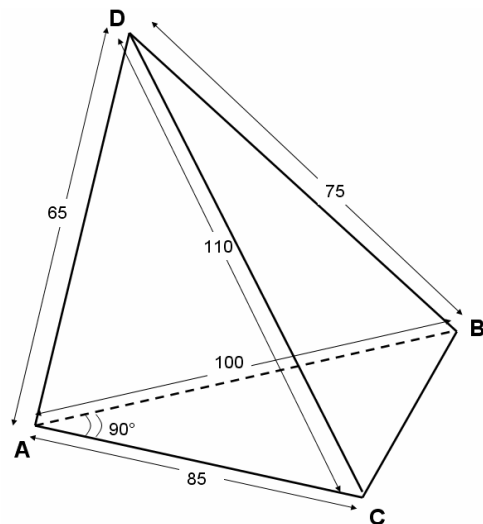


Figure 5. Objet final

6. CONCLUSION

Nous avons présenté dans cet article une modélisation, une méthode de génération des équations et une méthode de résolution originales. Celles-ci ont été validées par une expérimentation numérique sur le même prototype pour les cas 2D et 3D.

La méthode employée permet de transformer une géométrie 2D ou 3D d'une configuration initiale à une configuration finale. Les modifications apportées à la géométrie sont de type dimensionnel mais il est envisageable d'appliquer cette méthode sur des objets définis par d'autres types de contraintes (par exemple des contraintes de chiralité).

A l'avenir, nous pensons modéliser et résoudre des problèmes plus complexes avec des objets de type face ou volume liés par des contraintes de type angle entre deux plans, distance entre deux plans, aire de face, etc...

REFERENCES

Clement A., Rivière A., Termmerman M., 1994. «Cotation tridimensionnelle des systèmes mécaniques-Théorie et pratique », PYC éd.

Cubeles-Valade C., 1998. « Contribution à la synthèse des langages déclaratifs de spécifications géométriques ». thèse de doctorat de l'Ecole Centrale Paris.

Durand C., 2000. A systematic framework for solving geometric constraints analytically. *Journal of Symbolic Computation* Volume 30, Issue 5 pp 493 - 519.

Lesage D., 2002. Un modèle dynamique de spécifications d'ingénierie basé sur une approche de géométrie variationnelle, thèse de mécanique, Institut National Polytechnique de Grenoble.

Lamure H., Michelucci D., 1995. Solving geometric constraints by homotopy, *Proceedings of the third Symposium on Solid Modeling and Applications*, éditions ACM Press, pp.263-269.

MacDonald I., Pach J., and Theobald T., 2001. Common tangents to four unit balls. *Discr. Comp. Geometry*, 26:1-17.

Rivière A., 1993. « La géométrie du groupe des déplacements appliquée à la modélisation du tolérancement », thèse de doctorat de l'Ecole Centrale Paris.

Serré P., 2000. «cohérence de la spécification d'un objet de l'espace euclidien à n dimensions», thèse de doctorat de l'Ecole Centrale Paris.

Serré P., Rivière A., Duong A., Ortuzar, 2001. Analysis of a geometric specification, DETC2001, CD Rom paper n°DAC-21123, 6 pages, Pittsburgh (Pennsylvania, USA), September 9-12, 2001, pp 1-6.