Modélisation surfacique et volumique

Christian Nguyen

Département d'informatique
Université de Toulon

Plan

Quel modèle ?

Le modèle conditionne structure de données, calculs, rendu et interactivité (et réciproquement).

Chaque modèle est caractérisé par :

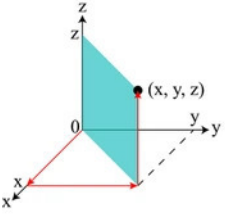
- la structure de données (langage de description, données d'acquisition, formes géométriques),
- le coût (pipeline graphique, calculs),
- l'apparence finale (degré de réalisme, vitesse de génération),
- le degré d'interaction (modifications, analyse, déformations).

Plan

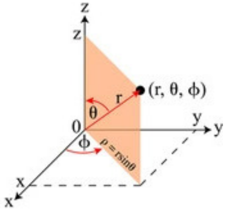
Systèmes de coordonnées

Deux systèmes de coordonnées :

- cartésien : un point P est défini par ses coordonnées (x, y, z) dans un repère orthonormé direct (ou indirect),
- polaire : un point P est défini par une distance r et deux angles θ et ϕ relatifs aux axes x et z.



Cartesian



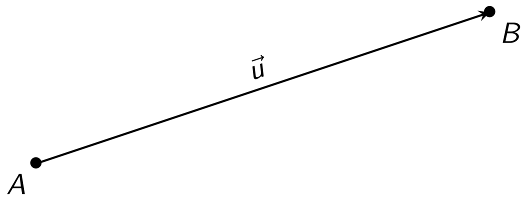
Spherical

Espace vectoriel, espace affine

La définition formelle d'un espace affine présuppose la donnée d'un espace vectoriel, appelé l'espace directeur.

Dans un espace *affine* E associé à l'espace *vectoriel* V, les éléments de E sont appelés les *points* et les éléments de V sont appelés les *vecteurs*.

À chaque couple de points (ou bipoints) (A, B), on associe un vecteur $\phi(A, B) = \vec{u}$.



Application linéaire, calcul matriciel

Soient E et F deux K-espaces vectoriels de dimensions finies respectives $p > 0$ et $n > 0$ et de bases respectives e et f, et soit φ une application linéaire de E dans F.

Toute application linéaire préserve les combinaisons linéaires et la composée de deux applications linéaires est linéaire.

Lien entre calcul matriciel et application linéaire φ : on appelle *matrice représentative* de φ dans les bases e et f la matrice (α_{ij}) déterminée par

$$\forall j \in \{1, \ldots, p\}, \quad \varphi(e_i) = \sum_{i=1}^{n} \alpha_{ij} f_i.$$

Espaces projectifs, coordonnées homogènes

Justification : traiter points et vecteurs de façon indifférenciée, en particulier pour la translation.

L'équation cartésienne d'une droite affine dans le plan est :

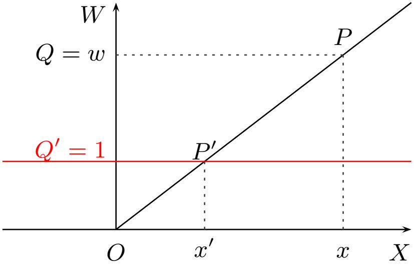
$$ax + by + c = 0$$

que l'on peut écrire :

$$[a, b, c][x, y, 1]^T = [a, b, c][wx, wy, w]^T = 0$$

Dans le sous-espace de codimension 1 du plan, un même point a plusieurs représentations distinctes, correspondant à une relation de proportionnalité : $[wx, wy, w]$ avec $w \neq 0$.

Espaces projectifs, coordonnées homogènes



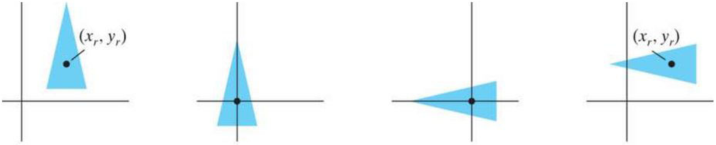
$$\boxed{x = w\,x'}$$

- tous les points vérifiant cette relation sont sur la droite OP,
- ils ont tous la même projection P' sur la droite $Q' = 1$,
- tout couple de coordonnées (wx', w) peut être utilisé pour définir P', et en particulier le couple $(x', 1)$.

Calcul matriciel, compositions de transformation

Exemple, rotation autour d'un point quelconque :

$$v' = T^{-1} \cdot R \cdot T \cdot v$$



Avantage principal : gain en efficacité, cette transformation affine nécessite 9 multiplications et 6 additions, sa structure fixe permet de simplifier le calcul en 4 multiplications et 4 additions.

Espace vectoriel, produit scalaire

La structure d'espace vectoriel est le support de base pour faire de la géométrie.

Deux notions supplémentaires pour manipuler les objets du plan : distance et angle, toutes deux obtenues grâce au *produit scalaire*.

On appelle *produit scalaire* l'application $\phi : K^n \times K^n \to K$ définie dans une base orthonormée par

$$\phi(\vec{u}, \vec{v}) := u_1 v_1 + u_2 v_2 + \cdots + u_n v_n \qquad (1)$$

avec $\vec{u} = (u_1, \ldots, u_n)$ et $\vec{v} = (v_1, \ldots, v_n)$.

On note souvent $(\vec{u} \,|\, \vec{v})$ le nombre réel $\phi(\vec{u}, \vec{v})$.

Espace vectoriel, produit scalaire

Un produit scalaire permet de définir une *norme* puis une *distance* dites *induites*. Si $\vec{u}$ et $\vec{v}$ désignent deux vecteurs du plan,

$$\|\vec{u}\| := \sqrt{(\vec{u} \mid \vec{u})} \quad \text{et} \quad d(\vec{u}, \vec{v}) := \|\vec{v} - \vec{u}\|. \tag{2}$$

la norme et la distance ainsi définies sont la norme et la distance *euclidiennes*.

Un vecteur est dit *normé* ou unitaire si $\|\vec{u}\| = 1$.

Deux vecteurs $\vec{u}$ et $\vec{v}$ sont orthogonaux si leur produit scalaire est nul, autrement dit si $(\vec{u} \mid \vec{v}) = \|\vec{u}\| \|\vec{v}\| \cos(\vec{u}, \vec{v}) = 0$.

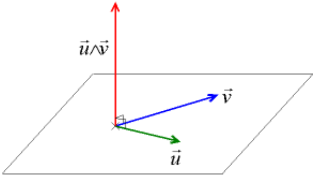
Ils sont orthonormés s'ils sont orthogonaux et si chacun des vecteurs est normé.

Espace vectoriel, produit vectoriel

Opération effectuée dans les espaces euclidiens orientés de dimension trois.

Le produit vectoriel $\vec{u} \wedge \vec{v}$ de deux vecteurs $\vec{u}$ et $\vec{v}$ de E non colinéaires se définit comme l'unique vecteur $\vec{w}$ tel que :

- le vecteur $\vec{w}$ est orthogonal aux deux vecteurs donnés,
- la base $(\vec{u}, \vec{v}, \vec{w})$ est de sens direct,
- $\|\vec{w}\| = \|\vec{u}\| \|\vec{v}\| \, |\sin(\vec{u}, \vec{v})|$.



Produit vectoriel, intersection de deux droites

Soient $\vec{u}(a_1, b_1, c_1)$, $\vec{v}(a_2, b_2, c_2)$ les vecteurs directeurs des droites δ_1 et δ_2.

Le point d'intersection de ces deux droites peut être déterminé grâce au produit vectoriel par un calcul en composantes :

$$\vec{u} \wedge \vec{v} = \begin{pmatrix} b_1 c_2 - b_2 c_1 \\ a_2 c_1 - c_2 a_1 \\ a_1 b_2 - b_1 a_2 \end{pmatrix}$$

puis en divisant les deux premières composantes du vecteur résultat par la troisième composante.

Déterminant

Propriétés géométriques

Soient les vecteurs $\vec{u}$, $\vec{v}$ dans le plan euclidien, l'expression analytique de leur déterminant est :

$$\det(u, v) = \begin{vmatrix} x & x' \\ y & y' \end{vmatrix} = xy' - yx'$$

L'expression géométrique équivalente est :

$$\det(\vec{u}, \vec{v}) = \|\vec{u}\| \|\vec{v}\| \sin(\widehat{u, v})$$

Propriétés :

- la valeur absolue du déterminant est égale à l'aire du parallélogramme défini par u et v,
- le déterminant est nul ssi les deux vecteurs sont colinéaires,
- son signe est strictement positif ssi la mesure de l'angle $(\widehat{u, v})$ est comprise dans l'intervalle $]0, \pi[$.

Puissance d'un point

En géométrie euclidienne du plan, la puissance d'un point M par rapport à un cercle Γ de centre O et de rayon R est un nombre qui indique la position de M par rapport à ce cercle :

$$P_\Gamma(M) = OM^2 - R^2$$

Dans le cas d'une droite orientée $\overrightarrow{AB}$, celle-ci définit 3 régions :

- un demi-plan contenant tous les points de puissance positive,
- un demi-plan contenant tous les points de puissance négative,
- la droite elle-même dont tous les points ont une puissance nulle.

La puissance d'un point M par rapport à une droite AB se calcule par le déterminant (en *coordonnées homogènes*) :

$$P_\Delta(M) = \det(M, A, B)$$

Droite

Une droite d est caractérisée par l'un de ses points $A(a_1, a_2)$ et un vecteur $\vec{v}(v_1, v_2)$ définissant son orientation. Ce vecteur $\vec{v}$ est appelé vecteur directeur de d.

L'ensemble de tous les points $M(x, y)$ du plan qui appartiennent à la droite d est caractérisé par l'équation vectorielle paramétrique de la droite suivante :

$$\overrightarrow{OM} = \overrightarrow{OA} + \overrightarrow{AM} = \overrightarrow{OA} + k.\vec{v}$$

Que l'on peut représenter sous forme d'un système d'équations paramétriques :

$$\begin{cases} x = a_1 + k.v_1 \\ y = a_2 + k.v_2 \end{cases} \quad \text{avec } k \in \mathbb{R}$$

Droite

Soit un plan affine muni d'un repère $(O, \vec{i}, \vec{j})$; x désigne l'abscisse d'un point, et y l'ordonnée de ce point.

Une droite peut s'exprimer sous la forme d'une fonction :

$$f : x \mapsto mx + h$$

avec m coefficient directeur, pente de la droite, tel que :

$$m = \frac{y_B - y_A}{x_B - x_A} = \frac{\Delta y}{\Delta x}$$

et h ordonnée à l'origine.

Une telle droite peut également s'écrire sous la forme d'une équation appelée équation cartésienne d'une droite affine :

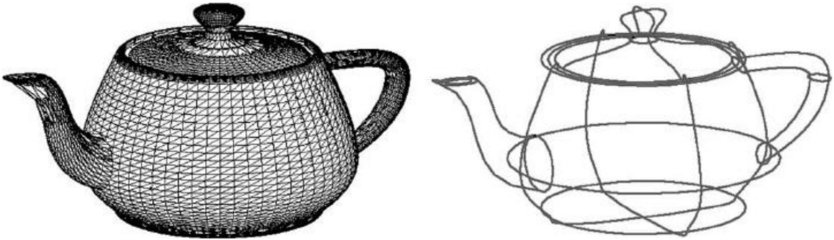
$$ax + by + c = 0$$

Plan

1. Rappels mathématiques
2. Structures de données

La plupart des modèles sont construits par association de primitives, dont le forme la plus simple est le triangle.

Mais des modèles plus complexes associent des carreaux (*patch*), comme les carreaux de Bézier ou les NURBS (*non-uniform rational B-spline*).



décomposition polygonale vs carreaux de Bézier

Coordonnées

Quel que soit le système retenu (cartésien, polaire ...), le type des données qui semble aller de soi est le type flottant.

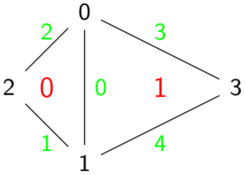
En modélisation, les cas particuliers abondent (exemple : intersections demi-droite/polygone) et ne conduisent pas aux mêmes résultats en fonction de la précision offerte ($|x - y| < \epsilon$).

L'adoption du type entier pour les coordonnées a conduit à l'émergence de la géométrie discrète, un ensemble d'outils mathématiques pour la géométrie adapté aux points discrets.

Sommets et polygones

La description minimale d'un modèle géométrique nécessite une liste de frontières et une liste de sommets.

Chaque entrée dans la liste des frontières identifie la liste des sommets (ou des arêtes) qui les constituent [1].



Polygones

0	0 1 2
1	3 1 0
⋮	⋮

Sommets

0	(x_0, y_0)
1	(x_1, y_1)
2	(x_2, y_2)
3	(x_3, y_3)
⋮	⋮

Arêtes

0	0 1
1	1 2
2	2 0
⋮	⋮

1. ainsi que certaines propriétés surfaciques, hors sujet ici.

Algorithmes de mise à jour

Le modèle de données à privilégier est la liste doublement chaînée.

Dans les cas où son implantation n'est pas possible, il faut disposer de fonctions d'insertion et de suppression efficaces.

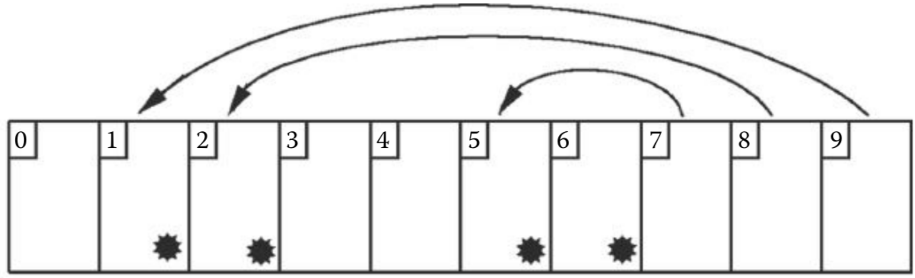
Si la relation d'ordre est inutile, l'insertion en fin est à privilégier, mais la suppression peut être très coûteuse (réallocation, interdépendance des structures de données).

Une solution est de déplacer l'élément de fin à la place de l'élément à supprimer (et de réduire la taille en conséquence) et de conserver son indice pour la mise à jour de la structure des polygones.

Algorithmes de mise à jour maj_sommets($\mathcal{S}$)

Entrées : $\mathcal{S}$ une liste de sommets
Output : $\mathcal{S}$ $(\#\mathcal{S}^+ < \#\mathcal{S}^-)$
$i \leftarrow j \leftarrow 0$; $n_1 \leftarrow n_2 \leftarrow \#\mathcal{S}$
tant que $i < n_1$ **faire**
  si $\mathcal{S}[j]$ *est marqué pour suppression* **alors**
    ▷ dernier sommet : cas particulier
    si $i < n_1 - 1$ **alors**
      $\mathcal{S}[j] \leftarrow \mathcal{S}[n_2 - 1]$ ▷ utilisation de l'emplacement supprimé
      $\mathcal{S}[n_2 - 1].idx \leftarrow j$ ▷ mémorisation de l'indice
    fin
    $n_2 \leftarrow n_2 - 1$
  sinon
    $j \leftarrow j + 1$ ▷ indice des sommets à supprimer
  fin
  $i \leftarrow i + 1$ ▷ indice de la liste des sommets
fin
$n_1 \leftarrow n_2$

Algorithmes de mise à jour



Algorithmes de mise à jour $\texttt{maj\_polygones}(\mathcal{P})$

Entrées : $\mathcal{P}$ une liste de polygones
Output : indices des sommets de $\mathcal{P}$ mis à jour

$i \leftarrow 0\,;\ n_p \leftarrow \#\mathcal{P}$
tant que $i < n_p$ **faire**
  $j \leftarrow 0\,;\ n_s \leftarrow \#\mathcal{P}[i]$
  tant que $j < n_s$ **faire**
    $i_s \leftarrow \mathcal{P}[i][j]$ ▷ indice du sommet courant
    si $i_s \geq \#\mathcal{S}$ **alors**
      $\mathcal{P}[i][j] \leftarrow \mathcal{S}[i_s].idx$
    fin
    $j \leftarrow j + 1$ ▷ indice des sommets
  fin
  $i \leftarrow i + 1$ ▷ indice des polygones
fin

Liste d'arêtes depuis une liste de polygones

Pour convertir un modèle d'une représentation à une autre, il est parfois nécessaire de construire une liste des arêtes du modèle.

C'est une opération coûteuse car la plupart des polygones du modèle partagent des arêtes (pour n polygones, il faut approximativement $3n^2$ tests).

Un algorithme plus efficace que de comparer tous les polygones entre eux consiste à construire une liste d'adjacence des sommets reliés par une arête pour chaque sommet du modèle :

- pour toutes les arêtes des polygones du modèle, on considère les deux sommets associés,
- si aucun des deux sommets n'a l'autre comme sommet adjacent, on l'ajoute à la liste d'adjacence d'un sommet.

Recherche de polygones adjacents

De nombreux algorithmes (triangulation de Delaunay, subdivisions de surfaces, interpolation des normales, ...) nécessitent que chaque polygone (dans un maillage) connaisse ses polygones voisins.

Un algorithme qui évite encore une fois une comparaison exhaustive des polygones gère une information supplémentaire : l'identité des facettes adjacentes (3 entiers pour des facettes triangulaires, -1 en l'absence de voisin) :

- pour tous les polygones du modèle, on ajoute le polygone à la liste d'adjacence de chaque sommet,
- un deuxième parcours des polygones permet de vérifier si les listes d'ajacence de deux sommets d'une arête contiennent l'identité d'un polygone commun (sauf le polygone courant) : il est ajouté à la liste d'adjacence du polygone.

Carte combinatoire

Structures de données efficaces en modélisation géométrique, appelé dans ce cas B-Rep (*Boundary Representation*).

Version la plus simple : la carte planaire, structure combinatoire pour la représentation de graphes planaires dans le plan.

Étant donné un graphe G, une carte planaire est un plongement de G dans $\mathbb{R}^2$, qui satisfait les conditions suivantes :

1. Les sommets du graphe sont représentés par des points.
2. Les arêtes sont représentées par des arcs de courbes ne se coupant qu'aux sommets.
3. Le complément $S \backslash G$ est une union disjointe de régions appelées faces (ne contenant ni sommets ni arêtes) qui ont toutes sauf une [2] la topologie d'un disque ouvert.

2. face externe (infinie) ayant la topologie d'un disque ouvert avec un trou

Structure de demi-arêtes

Informations topologiques : quels segments limitent une région données, quelles régions sont adjacentes, ...

Chaque arête est décomposée en deux demi-arêtes, d'orientations opposées et incidentes à un sommet source et à un sommet destination.

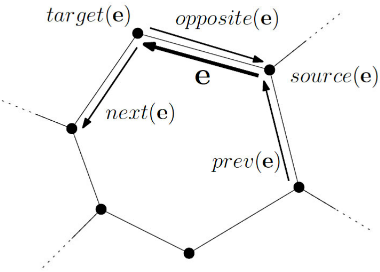
Représentation d'une demi-arête :

- une référence à la face incidente, ainsi qu'aux sommets source et destination,
- une référence à la demi-arête opposée,
- une référence vers la demi-arête qui suit et celle qui précède, dans la même face incidente.

Chaque face et sommet est représenté en stockant une référence vers l'*une* des demi-arêtes incidentes.

Introduction

structure de demi-arêtes



demi-arêtes

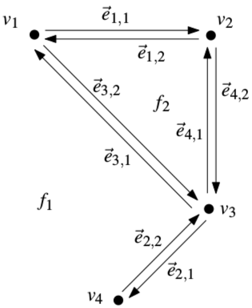
```
class Halfedge{
  Halfedge  prev, next, opposite;
  Vertex source, target;
  Face f;
}
 class Vertex{
  Halfedge  e;
  Point p;
}
 class Face{
  Halfedge  e;
}
```

information combinatoire

```
class Point{
  double x;
  double y;
}
```

information géométrique

Structure de demi-arêtes



Vertex	**Coordinates**	**IncidentEdge**
v_1	$(0,4)$	$\vec{e}_{1,1}$
v_2	$(2,4)$	$\vec{e}_{4,2}$
v_3	$(2,2)$	$\vec{e}_{2,1}$
v_4	$(1,1)$	$\vec{e}_{2,2}$

Face	**OuterComponent**	**InnerComponents**
f_1	**nil**	$\vec{e}_{1,1}$
f_2	$\vec{e}_{4,1}$	**nil**

Half-edge	**Origin**	**Twin**	**IncidentFace**	**Next**	**Prev**
$\vec{e}_{1,1}$	v_1	$\vec{e}_{1,2}$	f_1	$\vec{e}_{4,2}$	$\vec{e}_{3,1}$
$\vec{e}_{1,2}$	v_2	$\vec{e}_{1,1}$	f_2	$\vec{e}_{3,2}$	$\vec{e}_{4,1}$
$\vec{e}_{2,1}$	v_3	$\vec{e}_{2,2}$	f_1	$\vec{e}_{2,2}$	$\vec{e}_{4,2}$
$\vec{e}_{2,2}$	v_4	$\vec{e}_{2,1}$	f_1	$\vec{e}_{3,1}$	$\vec{e}_{2,1}$
$\vec{e}_{3,1}$	v_3	$\vec{e}_{3,2}$	f_1	$\vec{e}_{1,1}$	$\vec{e}_{2,2}$
$\vec{e}_{3,2}$	v_1	$\vec{e}_{3,1}$	f_2	$\vec{e}_{4,1}$	$\vec{e}_{1,2}$
$\vec{e}_{4,1}$	v_3	$\vec{e}_{4,2}$	f_2	$\vec{e}_{1,2}$	$\vec{e}_{3,2}$
$\vec{e}_{4,2}$	v_2	$\vec{e}_{4,1}$	f_1	$\vec{e}_{2,1}$	$\vec{e}_{1,1}$