TP Unity:
Cinématique inverse avec l'algorithme CCD

Steve Tonneau
IRISA

8 novembre 2013

Introduction

Ce TP propose de traiter un problème concret, récurent dans le domaine de l'animation, celui de la cinématique inverse. La partie 1 de ce document est consacrée à la description du problème. Aussi les lecteurs les plus éclairés pourront directement passer à la partie 2. Cette seconde partie a pour but de guider les étudiants dans l'implémentation de l'algorithme CCD (pour Cyclic Coordinate Descent), très utilisé, en particulier dans le domaine du jeu vidéo, pour résoudre le problème de cinématique inverse. On l'implémentera d'abord en 2, puis en 3 dimensions.

Public concerné

Ce tp s'adresse à des étudiants en première année d'informatique. Il requiert une connaissance minimale du moteur Unity 3d [1] (Menus, système de scripts). Les signatures des méthodes utilisées sont présentées en C#, mais il est facile de les transposer pour les langages Boo et javascript également acceptés sous Unity. Aussi il est également nécessaire d'avoir des connaissances dans un de ces langages cibles.

Pour comprendre et implémenter l'algorithme CCD sous Unity, des connaissances mathématiques de niveau lycée sont nécessaires.

Durée

TODO

Matériel

Un squelette de projet est disponible à l'adresse TODO. Il nécessite une version d'Unity (gratuite ou pro) supérieure ou égale à la version 4.1.

1. http://unity3d.com/unity.

Le projet comprend :

- Une scène CCD.scene. Elle comprend elle-même :
  - Une hiérarchie de sphères qui servira de chaîne cinématique pour l'exercice ;
  - Un cube Target, qui représentera la cible à atteindre par notre chaîne cinématique.
- Un squelette de script CCD.cs, qu'il va falloir modifier.

1 Problématique : La cinématique inverse

1.1 Exemple introductif : réglage de l'heure

Considérons la montre à gauche de la figure 1 :

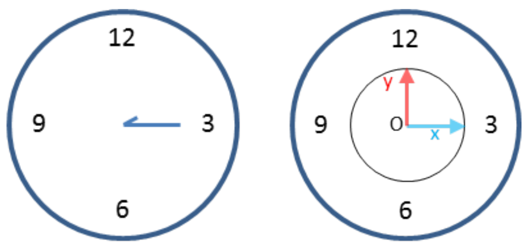

FIGURE 1 – *A gauche : Une montre indiquant 2h15. A droite : la même montre munie d'un repère cartésien (Ox, Oy).*

Cinématique directe

Nous munissons notre montre d'un repère cartésien *(Ox, Oy)* (figure 1, droite), dans lequel la longueur de la grande aiguille vaut 1. Nous appelons l'extrémité extérieure de l'aiguille **effecteur**. On observe en fait que l'effecteur se déplace le long du cercle trigonomètrique associé au repère *(Ox, Oy)*.
Ce déplacement est réalisé par une rotation d'un angle θ autour d'un troisiéme axe $Oz = Ox \wedge Oy$.

On définit donc la fonction *f*, telle que

$$f(\theta) = \begin{pmatrix} cos(\theta) \\ sin(\theta) \end{pmatrix}$$

avec $\theta \in [0, 2\pi[$.

Grâce à f on peut connaître la position de l'effecteur de la grande aiguille en fonction de l'angle θ. Par exemple, si $\theta = -\pi/2$, l'effecteur est à la position $\begin{pmatrix} 0 \\ -1 \end{pmatrix}$, comme montré sur la figure 2

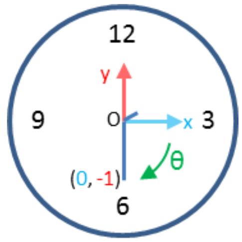

FIGURE 2 – *Si on choisit $\theta = -\pi/2$, la montre indique maintenant 2h30.*

Ceci constitue un exemple de **cinématique directe** : à partir d'un angle θ, nous calculons une position $\begin{pmatrix} x \\ y \end{pmatrix}$.

Cinématique inverse

On considère le problème inverse : On souhaite régler l'heure à 2h30, c'est à dire que l'extrémité de la grande aiguille ait pour coordonnées $\begin{pmatrix} 0 \\ -1 \end{pmatrix}$. Quelle valeur doit prendre l'angle θ pour atteindre cet objectif ?

On définit la fonction f^{-1}, telle que

$$f^{-1}(x, y) = acos(x) * sgn(y)$$

avec $sgn(y) = -1$ si $y < 0$, $sgn(y) = 1$ sinon.

f^{-1} permet de calculer l'angle θ nécessaire à l'atteinte de la position $\begin{pmatrix} x \\ y \end{pmatrix}$. Ceci est un exemple de **cinématique inverse**.

1.2 Cinématique en deux dimensions

Définition d'une chaîne cinématique

Pour ce tp, nous définissons une chaîne cinématique comme un ensemble n de barres rigides $c_i, i = 0, 1, ..., n-1$ maintenues deux à deux ensemble à leurs extrêmités par des articulations q_i. L'articulation $q_j, j = 1, ..., n-1$ connecte les barres $c_j - 1$ et c_j. q_0 connecte la première barre et l'environnement. Cette articulation est appelée racine. La dernière barre n'est connectèe qu'à une de ses extrémités ; on appelle l'extrémité libre **effecteur** e.

Par exemple la grande aiguille de la montre présentée (figure 3) est une chaîne composée d'une seule barre rigide, qui comprend donc la racine et l'effecteur.

Un angle θ_i qui décrit la rotation associée à une articulation q_i, comme le montre la figure 4.

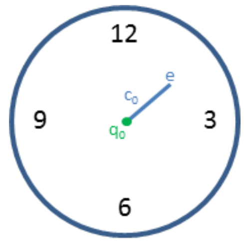

FIGURE 3 – *La grande aiguille de la montre est une chaîne cinématique composée d'une seule barre rigide.*

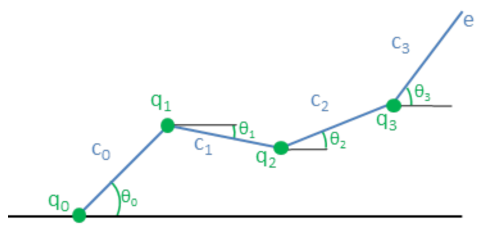

FIGURE 4 – *Une chaine cinématique composée de quatre corps rigides.*

Cinématique directe

Dans un problème de cinématique directe, on cherche à déterminer la position de l'effecteur e en fonction des valeurs d'angles θ_i de la chaîne cinématique.

On veut donc trouver la fonction f telle que :

$$f\begin{pmatrix}\theta_0\\ \dots\\ \theta_{n-1}\end{pmatrix} = \begin{pmatrix}x_e\\ y_e\end{pmatrix}$$

Cinématique inverse

Dans un problème de cinématique inverse, on cherche à déterminer une combinaison de valeurs d'angles θ_i qui permette d'atteindre la position $\begin{pmatrix}x_e\\ y_e\end{pmatrix}$.

On veut donc trouver la fonction f^{-1} telle que :

$$f^{-1}\begin{pmatrix}x_e\\ y_e\end{pmatrix} = \begin{pmatrix}\theta_0\\ \dots\\ \theta_{n-1}\end{pmatrix}$$

Le problème est que, dans les cas complexes, f n'est pas bijective :

- elle n'est pas surjective car f^{-1} n'est pas forcément définie partout (figure 5) ;

– elle n'est pas injective car il peut exister plusieurs solutions qui permettent d'atteindre la position $\begin{pmatrix} x_e \\ y_e \end{pmatrix}$(figure 6).

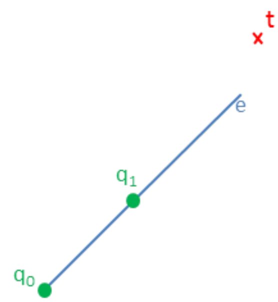

FIGURE 5 – *Le point t ne peut être atteint par l'effecteur e.*

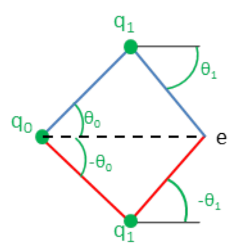

FIGURE 6 – *Différentes combinaisons d'angles peuvent aboutir à la même position pour l'effecteur e.*

Quand une chaîne cinématique est trop complexe, il est difficile, voire impossible de définir correctement f^{-1} ; c'est pour cela que des méthodes numériques ont été mises au point pour calculer les valeurs de f^{-1}, comme la méthode CCD.

1.3 Cinématique en 3 dimensions

Nous avons vu qu'en deux dimensions nous considérons des rotations autour de l'axe $Oz = Ox \wedge Oy$. En 3 dimensions, il nous faudra également considérer des rotations autour des axes Ox et Oy (figure 7).

2 Implémentation de l'algorithme CCD

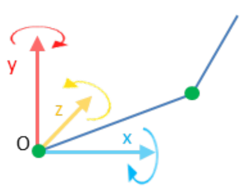

FIGURE 7 – *Articulation en 3 dimensions : 3 rotations sont possibles.*