

Devoir 3

8TRD147 Animation et images par ordinateur

H2016

Prof: Yves Chiricota

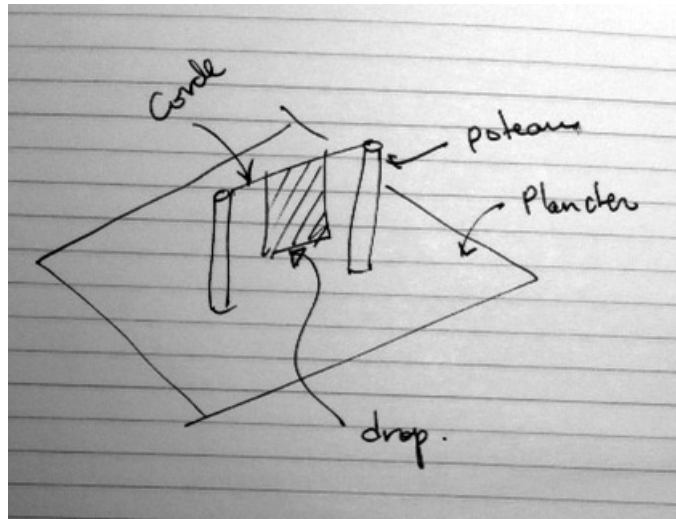
Date de remise: 28 avril 2016 12h00 (Midi)

Objectif

Le but de ce travail est de vous familiariser les systèmes masses-ressorts.

Description

Il s'agit d'implémenter sur le CPU (coté client) une simulation de type masses-ressorts d'un drap suspendu sur une corde. Vous devrez utiliser une partie du code que vous avez produit pour le deuxième devoir. Les données de la géométrie initiale sont les mêmes que pour le devoir 2.



Spécifications

- Les points sur la corde sont fixés.
- Le maillage est de même format que dans le devoir 2.
- Il faut faire la mise à jour du VBO avec les fonctions `glMapBuffer` et `glUnmapBuffer`. Pour ce faire, ajouter une fonction `UpdateVBO` dans la classe `CMesh` pour effectuer ce travail.
- Le calcul de chaque frame doit se faire à partir de la fonction `calc_frame`.
- Il faut simuler un coup de vent en direction perpendiculaire au drap. En chaque point, la force du vent doit varier en fonction de la position et du temps (utilisez une fonction trigonométrique).
- Il faut utiliser la force de gravité.

- Utiliser le modèle d'illumination de Phong (donc calculer les vecteurs normaux à chaque frame).
- L'état initial du système correspond à la figure précédente (avec vitesses nulles).

Quelques indications sur l'implémentation.

Dans une même fichier définir les classes suivantes:

```
class CSMR {
    CMesh* mesh; // Pour affichage. Mettre à jour à chaque itération.
    list<CParticule*> particules;
    list<CRessort*> ressorts;
    .
    .
    .
};

class CParticule {
    CVertex* vertex; // Sommet du maillage associé à la particule.
    CPoint3D pos[2];
    CVect3D vel[2];
    float masse;

    .
    .
    .
};

class CRessort {
    CParticule *P0, *P1;
    float longueur_repos;
    float k; // Constante de Hooke.

    CVect3D F() const; // Calcul la force du ressort.

};

// Classe qui effectue le calcul d'intégration avec la méthode d'Euler.
CIntegrateur {
    CSMR* smr;

    void step();
    .
    .
    .
};
```

```
};
```

Placer un ressort par coté de triangle dans le maillage en plus de ressort qui relie des sommets à distance 2 (horizontal et vertical).

La boucle d'animation comportera les étapes suivantes:

- Calculer un pas d'intégration ($\Sigma_k \rightarrow \Sigma_{k+1}$).
- Appliquer la correction géométrique sur les ressorts.
- Mettre à jour le maillage, les vecteurs normaux, le VBO.
- Dessiner le frame.

Suggestion: Ajouter l'étape de correction géométrique à la fin, lorsque la simulation fonctionnera correctement.

Valeurs de départ suggérées pour les constantes:

Variable	Valeur
h (pas d'intégration)	0.001
k (Hooke)	100
m (masse)	1000
γ (amortissement)	20
g (gravité)	$(0, -10, 0)$
Force du vent	Environ 5 à 10 variable.

La force du vent doit varier en fonction du temps de la position sur le drap. Le prototype de la fonction qui calcule le vent devrait être:

```
CVect3D f_vent(const CPoint3D& pos, const float& t);
```

Livrable

Vous devez remettre par courriel un fichier .zip; le nom du fichier sera d3-AAAA12345678, où AAAA12345678 est le code permanent d'un des deux membres de l'équipe. Le fichier contiendra le projet XCode (sauf le dossier DerivedData) et un rapport d'une page environ. Le rapport mentionnera les méthodes utilisées, difficultés rencontrées, rôle de chaque membre de l'équipe, etc.

Critères de correction

Item	%
Pertinence des commentaires	10
Clarté du code en tant que tel, limpidité. Le code est-il facile à comprendre? La solution est-elle simple?	30
Algorithmes. La méthode implémentée est-elle valide? Est-elle efficace?	30
Respect des spécifications et consignes	20
Exécution	10