

Projet Beyblade

Joseph El-Forzli
Arthur Renard

1 Réponses

1. On a fait tout ce qui était demandé ou suggéré dans le cours (sauf la caméra collée à la toupie). Plusieurs extensions de notre propre initiative ont également été rajoutées :
 - authenticité des sauvegardes avec le hashage,
 - possibilité de lancer des balles (touche w),
 - un mode WTF (touche t).Remarques : Toupie Générale marche seulement sur un court laps de temps et Énergétique fait une boucle infinie (une exception est lancée et catchée). De plus, certains invariants ne le sont pas tant que ça.
2. Oui, le programme est axée sur le graphisme. La version texte n'est utilisée que pour les tests. La bibliothèque graphique est Qt 5.14.1
3. Environ 7h.

2 Descriptif du projet

2.1 Introduction

Le projet Beyblade, dont le nom est issu d'un dessin animé relatant la vie de valeureux combattants ayant pour uniques armes des toupies, a pour objectif principal de permettre à l'utilisateur d'observer des toupies de formes différentes évoluer selon les valeurs initiales et les intégrateurs numériques de son choix parmi les choix suivants :

2.2 Toupie :

- Toupie Conique (CôneSimple)
- Toupie Chinoise
- Toupie Générale Conique (Une toupie conique avec des équations d'évolution plus générales)
- Toupie Énergétique Conique (Une toupie conique avec des équations d'évolution basées sur l'énergie)

2.3 Intégrateurs :

- Euler-Crommer (pour des équations du 1^{er} et 2^e degré)
- Newmark (uniquement 2^e degré)
- Runge-Kutta (pour le 1^{er} et 2^e degré)

Remarques : Toupie Générale et Énergétique Conique ont quelques soucis avec leurs équations d'évolution. L'intégrateur d'Euler-Crommer ne semble pas être adapté à Toupie Chinoise qui diverge rapidement.

2.4 Améliorations

Plusieurs améliorations, qui font la fierté de leurs créateurs, ont ensuite été ajoutées au projet telles que (liste non exhaustive) :

- Possibilité d'ajouter des toupies de hauteur, rayon et forme variables.
- Possibilité de choisir si les toupies se déplacent dans l'espace ou restent à un endroit fixe.
- Possibilité d'afficher la trace de leurs axes de symétrie.
- Possibilité de sauvegarder l'état de plusieurs toupies et de charger d'anciennes sauvegardes en étant certain de leur authenticité.
- Les couleurs des toupies varient en fonction de leur vitesse de rotation.
- La création de balles rebondissantes.
- Un mode WTF

Et bien plus...

3 Installation de beyblade

1. Ouvrir Beyblade/project/beyblade.pro, QtCreator devrait s'ouvrir, appuyez sur Ctrl+R ou sur la flèche d'exécution en bas à droite.
2. Une erreur "No rule to make target.." pourrait apparaître (nous n'avons pas réussi à la faire disparaître). Un dossier build-beyblade-"Kit sélectionne" devrait s'être créé dans le dossier Beyblade.
3. Déplacez les fichiers "music.wav" et "toupie.png" de Beyblade/projet/Qt_GL vers Beyblade/build-beyblade-.../QT_GL.
4. Retournez sur QtCreator et exécutez de nouveau, le programme devrait alors démarrer.

4 Utilisation de beyblade

4.1 Configuration des paramètres initiaux

Une fois le projet exécuté, deux fenêtres devraient apparaître, l'une avec la photo des auteurs et une description telle que "By Arthur et Jojo" et l'autre étant un panneau de configuration ayant pour titre "Toupie N°1". Dans ce dernier vous pouvez sélectionner le type de la toupie, l'intégrateur, les valeurs initiales des angles (en radians) et de leurs dérivées ainsi que la position initiale de la toupie. Appuyez ensuite sur *Valider*.

La toupie sélectionnée devrait alors apparaître sur la première fenêtre citée.

4.2 Lancement et interactions durant l'exécution

Pour que la toupie se mette en mouvement, appuyez sur la touche *Espace* de votre clavier. Appuyez à nouveau sur celle-ci pour mettre pause. Plusieurs autres touches de votre clavier permettent d'interagir avec le programme :

- u** Permet aux toupies de se déplacer dans l'espace (appuyez à nouveau pour annuler).
- y** Affiche la *trace* de la toupie (prolongement du vecteur (point de contact - centre de gravité)).
- w** Lance d'une balle de couleur aléatoire.
- t** Lance de l'option *WTF* qui met l'arène en plein écran et chamboule le programme en cours.
- o** Diminue le *dt* pour une meilleure précision.
- p** Augmente le *dt*.
- F2** Active le mode plein écran.
- ECHAP** Quitte le programme.

4.3 Déplacement de la caméra

La caméra évolue dans un repère sphérique centré à l'origine. Il n'est pas possible de changer ce repère.

- z** Zoom.
- s** Dézoom.
- q** Tourne la caméra dans le sens anti-horloger.
- d** Tourne la caméra dans le sens horloger.
- h** Réinitialise la caméra.
- Les touches *flèche directionnelle* permettent quant à elles d'évoluer dans le repère sphérique centré à l'origine.
- La souris (en maintenant le clic gauche) permet également de déplacer la caméra.

4.4 Paramètres avancés

Pour le bon déroulement de ces opérations, mettez sur pause le programme. Dans les options du programme plusieurs choix s'offrent à vous :

4.4.1 Ajouter une toupie

Rouvrez le panneau de configurations pour ajouter une toupie dans l'arène.

4.4.2 Supprimer une toupie

Supprime la toupie sélectionnée.

4.4.3 Sauvegarde et chargement d'anciennes sauvegardes

Sauvegarder le système ouvre une fenêtre vous permettant de sélectionner le nom du fichier où les toupies seront sauvegardées, ainsi que l'emplacement de ce dernier. (Si un fichier du même nom existe déjà, il sera remplacé). Le fichier de sauvegarde ainsi créé contient un code de sécurité généré par une fonction de hachage permettant de vérifier l'authenticité du document lors de sa réouverture.

Charger un système ouvre une fenêtre vous permettant de sélectionner le fichier de sauvegarde. Si le fichier de sauvegarde a été corrompu, un message d'erreur sera lancé et le programme s'arrêtera.

4.4.4 Informations

Une fenêtre apparaît. Sélectionnez la toupie de votre choix pour obtenir des informations.

4.5 Exécution des tests

Pour exécuter un test il suffit de :

- "décommenter" celui qui vous intéresse dans `main_text.cc`
- de cliquer sur l'icône au dessus de la flèche verte d'exécution
- de sélectionner `text` et de compiler.

5 Développé avec

Le projet Beyblade a été codé en *C++* depuis l'IDE *Qt Creator, version 5.14.1*, où plusieurs de ses bibliothèques furent utilisées.

6 Auteurs

- Joseph El-Forzli, section de Physique
- Arthur Renard, section de Mathématiques