

Rapport Animation Alexandre Meyer

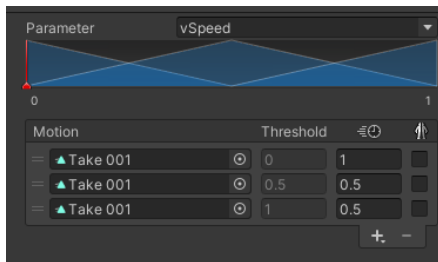
Barros Mikel-ange

I) Prise en main de Unity

I.a) machine à état

La mise en place d'une machine à état n'a pas posé de problèmes particuliers et à permis de poser un premier pieds dans l'environnement fourni par Unity. Cette approche simple m'as permis de vraiment me familiariser et de comprendre comment fonctionne les machines à état sous Unity.

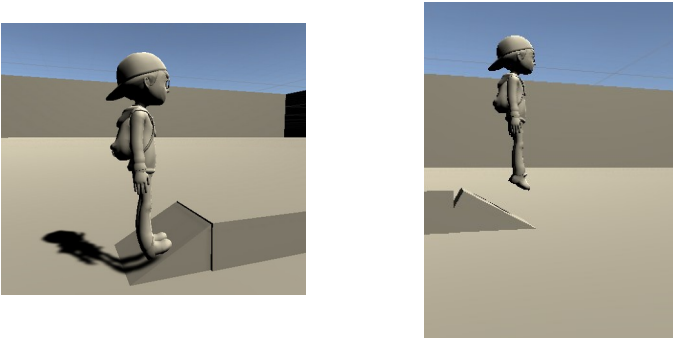
La mise en place d'un script, obligatoire pour gérer les transitions entre les animations, à permis de gérer facilement le déplacement et les poses du joueur.



I.b) IK Unity

Malgré le tutoriel et la bibliographie. La mise en place de IK pour les pieds du personnage est encore buguée chez moi. En effet, si j'arrive bien à monter sur un plan incliné et que mes pieds semblent bien orientés, la descente d'un plan inclinée ne se fait pas et mon personnage reste bloqué dans les airs.

L'implémentation me semble pourtant la même que celle du tutoriel, il semblerait donc que le problème vienne d'un paramètre que je n'ai pas compris ou d'une étape que j'ai manqué.



I.c) collision avec des corps rigides

La collision avec des corps rigides c'est avérée plus simple que prévu et à permis d'utiliser le cours de Mme Zara et de le lier a votre cours. Cela m'a permis de mieux comprendre le fonctionnement des rigides body. Cependant j'ai eu un problème avec la ragdoll et IK. En effet, en créant la ragdoll le mouvement des jambes de mon personnage est devenu totalement incohérent. Mais la collision avec des sphères générée automatiquement fonctionnait.

II) Fabrik

L'algorithme de Fabrik est un algorithme de cinématique inverse fonctionnant particulièrement bien et ce quelque soit le nombre de chaînes cinématiques. Cet algorithme était donc tout adapté pour comprendre et coder la cinématique inverse (IK) vue dans la partie précédente. Une approche simple de fabrik est de d'abord le faire fonctionner pour une chaine cinématique puis de le faire fonctionner pour des chaines multiples. C'est donc l'approche que j'ai choisie.

II.a) Fabrik pour une chaine

Fabrik pour une chaine cinématique consiste à comprendre le fonctionnement de fabrik et de le remettre en place. Je vais donc rapidement en détaillé le fonctionnement. Fabrik part du principe que l'on peut approximer un déplacement entre deux positions d'un squelette en en déplaçant les points petits à petits. Pour cela on doit définir la cible à atteindre et déplacer l'extrémité la plus proche de l'articulation en bout de chaîne sur la cible, puis de déplacer l'articulation précédente dans la direction du point en maintenant la distance de base entre les deux articulations. Et ainsi, de suite jusqu'à atteindre la dernière articulation. Cette phase est appelée forward.

Une fois la phase forward terminée, une nouvelle phase commence, la phase backward. Elle reprends le principe du forward vu précédemment mais cette fois en inversant le sens de déplacement.

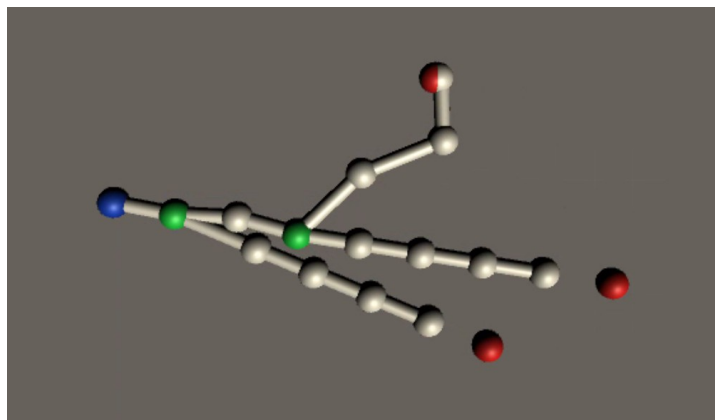
Ces deux phases sont appelées une dizaine de fois et permettent de facilement approximer des positions satisfaisantes.

Une fois cet algorithme compris le mettre en place ne pose aucune difficulté.

II.b) Fabrick pour N-chaîne

Cette méthode pose deux problèmes principaux :

- définir les bonnes chaînes cinématique : pour cela, on défini une chaîne cinématique comme étant un ensemble de trois éléments : un début, une fin et une cible et on remplit récursivement un tableau de chaîne selon trois cas : nœud à un seul fils, nœud à plusieurs fils, nœud sans fils.
- fusionner les différentes chaînes cinématiques qui partagent un nœud commun pour que les modifications d'une chaîne affecte les chaînes reliées au même nœud. Pour cela, il suffit de recréer un nœud commun à toutes les chaînes et qui sera donc modifié par chacune des chaînes.



III Conclusion

Ce Tp est une bonne manière d'aborder l'animation de personnage et la cinématique inverse, tout en permettant de prendre en main un des moteurs graphiques les plus utilisés au monde.