

Rapport du TP1 du module Animation et Simulation

Skinning

Antoine BRALET et Guillaume DURET

22 Novembre 2020

1 Introduction

Ce rapport a pour but de présenter la résolution d'un problème simple de *skinning* sur un cylindre puis par extension sur une forme plus complexe, un monstre. Ce travail a été réalisé dans le cadre d'un TP du module d'Animation et de Simulation.

Le *skinning* est une technique graphique qui consiste à modéliser un objet par un squelette (plus ou moins complexe) et dont tous les os sont reliés par des joints assimilables à une articulation dans l'anatomie humaine par exemple. Ce lien d'un joint à l'autre est tel que tout joint du squelette aura toujours pour parent (même très éloigné) le premier joint créé. Ainsi, l'idée est d'utiliser ce squelette beaucoup plus facile à manipuler afin de modéliser les comportements de la forme lors de ses mouvements puis de se servir de ce modèle squelettique pour reproduire le comportement sur les sommets de l'objet, en fonction de sa position par rapport à chacun des joints. Si le concept reste encore très théorique ici et quelque peu obscur, il va être expliqué et détaillé tout le long du rapport.

Nous allons commencer par travailler avec une forme simple : un cylindre que nous allons créer en Section 2 puis animer en Section 3 à l'aide de la technique de *skinning* puis nous chercherons à extrapoler cette méthode vers un maillage beaucoup plus complexe dans la Section 4 avant de conclure quant à notre étude.

2 Initialisation

2.1 Surface géométrique

La première étape avant de pouvoir parler d'animation et de *skinning* est d'initialiser les objets et de notamment créer le mesh du cylindre.

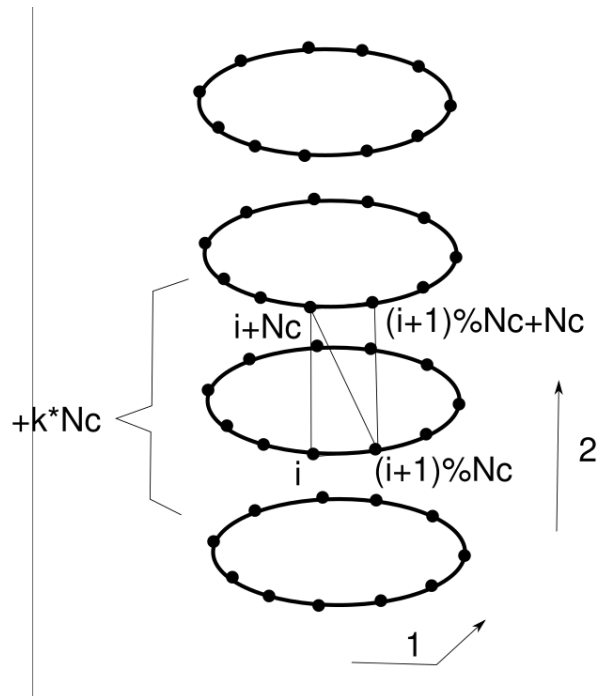


Figure 1: Schema de l'indexation du maillage d'un cylindre

Comme l'illustre la Figure 1 la méthode la plus simple pour réaliser un cylindre en utilisant des triangles se fait en 2 étapes: la première est de créer les points du cylindre en considérant le cylindre comme une

superposition de cercles. Il est ainsi possible à l'aide du rayon r , du nombre de points par cercle Nc et de la hauteur h du cercle d'obtenir les coordonnées des points p du cercle en 3D à l'aide de la formule 1 et en itérant sur la hauteur il est ainsi possible d'obtenir les points du cylindre total.

$$p = (r * \cos(\theta), r * \sin(\theta), h) \quad (1)$$

La deuxième étape consiste à créer les triangles à partir de ces points, pour cela il faut alors choisir de créer les triangles avec le cercle qui se trouve au dessus. Deux points du cercle et deux points situés à la même position mais sur le cercle du dessus forment 2 triangles. Il faut ainsi pour un indice i dans le tableau des points obtenir le point du cercle supérieur ou inférieur en faisant $+ou - Nc$ avec Nc le nombre de points par cercle. On a accès au point voisin en faisant $+1$ cependant il faut bien faire attention à ne pas faire plus d'un tour et de refermer le cercle en sachant que le voisin du dernier point du cercle est son premier point.

Une fois le maillage créé il ne reste plus qu'à l'afficher et de vérifier de le maillage est bien construit (Figure 2 et 3).

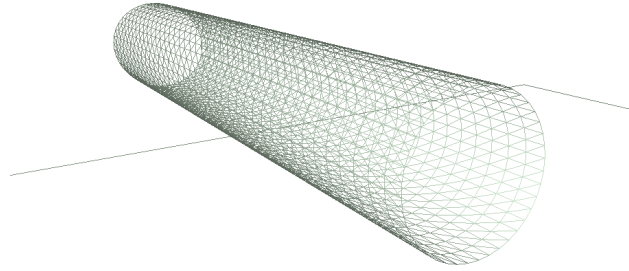


Figure 2: Affichage du maillage du cylindre en position de repos

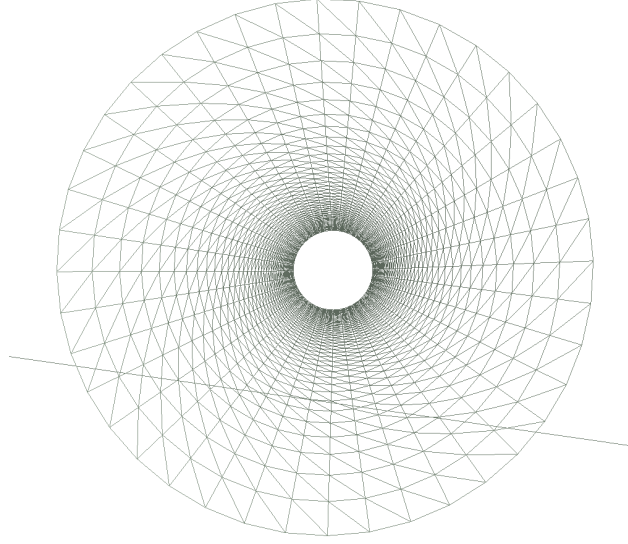


Figure 3: Affichage du maillage du cylindre en position de repos en mettant en valeur les triangles créés.
Le maillage ne semble pas contenir de defaults

2.2 Position de repos du squelette

Le but maintenant est de créer le squelette associé au cylindre précédemment créé. Un squelette est décrit par un ensemble de joints j_i . Comme leur nom l'indique, les joints représentent la jointure entre deux os, cette jointure peut être assimilée à une articulation possédant une position et décrit une rotation pour l'os suivant. En bref, chaque joint possède un repère propre composé d'une translation et d'une orientation.

Le squelette possède en réalité une structure récursive car chaque joint j_i est lié à un joint parent auquel il est attaché, ainsi ce joint a une position exprimée en fonction du repère du joint parent, suivant cette structure seul le premier joint j_0 peut être exprimé dans le repère global de la scène. Les autres joints sont exprimés dans un repère dit local.

Le squelette de notre cylindre est décrit par trois joints tels que le joint j_0 est le parent du joint j_1 et j_1 est le joint parent du joint j_2 comme décrit dans la Figure 4.

De plus sachant la longueur L du cylindre, la position du premier joint s'exprime dans le repère global soit $(0, 0, 0)$, le joint j_1 s'exprime par rapport au repère j_0 soit $(0, 0, L/2)$ et le joint j_2 s'exprime par rapport au repère j_1 soit $(0, 0, L/2)$ alors que dans le repère global se serait $(0, 0, L)$

De plus les joints caractérisent une rotation, informativement il se trouve être plus simple de décrire la rotation d'un joint à l'aide d'un quaternion. Les quaternions sont une généralisation des nombres complexes à l'espace 3D, tel que le quaternion q caractérisant une rotation d'angle nul s'obtient ainsi : $q = (0, 0, 0, 1)$

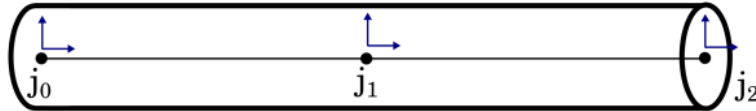


Figure 4: Schema des trois points du squelette d'un cylindre avec leur repère respectif

Le squelette du cylindre au repos est ainsi créé cependant il n'est pas possible de le visualiser du fait de l'expression local des joints. Pour palier ce problème il est possible de créer un deuxième squelette associé

au premier mais cette fois ci exprimé avec des positions globales.

En effet en partant du joint j_0 il est possible de retrouver les positions globales des joints du squelette tel que:

$$G_j = G_{p(j)} L_j \quad (2)$$

avec G_j la matrice correspondante aux coordonnées du repère j dans le repère global, L_j la matrice correspondante aux coordonnées locales du joint j par rapport au joint parent, $p(j)$ est l'indice du parent du joint j . Notons également qu'un repère R formé d'un quaternion q et d'une translation t s'exprime matriciellement comme présenté dans l'équation 3 si bien au niveau global que local.

$$R = \left(\begin{array}{ccc|c} & & & \\ & q & & t \\ - & - & - & - \\ 0 & 0 & & 1 \end{array} \right) \quad (3)$$

En réalisant le produit de l'équation il résulte la matrice:

$$R = \left(\begin{array}{ccc|c} & & & \\ & q_g q_l & & q_g t_l + t_g \\ - & - & - & - \\ 0 & 0 & & 1 \end{array} \right) \quad (4)$$

Il faut cependant savoir que le quaternion $q = (q_0, q_1, q_2, q_3)$ représente une matrice de rotation 3x3 tel que:

$$R = \begin{pmatrix} 1 - 2(q_1^2 + q_2^2) & 2(q_0 q_1 - q_3 q_2) & 2(q_0 q_2 + q_3 q_1) \\ 2(q_0 q_1 + q_3 q_2) & 1 - 2(q_0^2 + q_2^2) & 2(q_1 q_2 - q_3 q_0) \\ 2(q_0 q_2 - q_3 q_1) & 2(q_1 q_2 + q_3 q_0) & 1 - 2(q_0^2 + q_1^2) \end{pmatrix} \quad (5)$$

Ainsi le produit de quaternion défini dans l'équation 6 entre $q_0 = (v_0, w_0)$ et $q_1 = (v_1, w_1)$ avec v_0 et v_1 de dimension 3 est défini de telle sorte que le quaternion résultant est le quaternion décrivant la matrice résultante du produit des deux matrices. La multiplication de quaternion est ainsi beaucoup moins coûteuse que le sera la multiplication des deux matrices de rotation.

$$q_0 q_1 = (w_0 v_1 + w_1 v_0 - v_0 x v_1, w_0 w_1 - v_0 x v_1) \quad (6)$$

Dans la même logique d'équivalence d'un quaternion q avec une matrice de rotation, le résultat p' de la multiplication de la matrice de rotation avec un vecteur p peut s'exprimer seulement avec les 3 premières composantes du quaternion $q = (q', w)$ ce qui est une fois encore moins coûteux (équation 7).

$$p' = q' p \overline{q'} \quad (7)$$

Ayant ces propriétés il est alors possible de déterminer les positions et les rotations dans le repère global en commençant les calculs par le joint j_0 qui est déjà dans le repère monde.

En obtenant le squelette dans le repère global il suffit alors de récupérer les segments joignant les joints qui représentent les os du squelette et finalement les afficher en tant que segment pour pouvoir dessiner le squelette (figure 5).

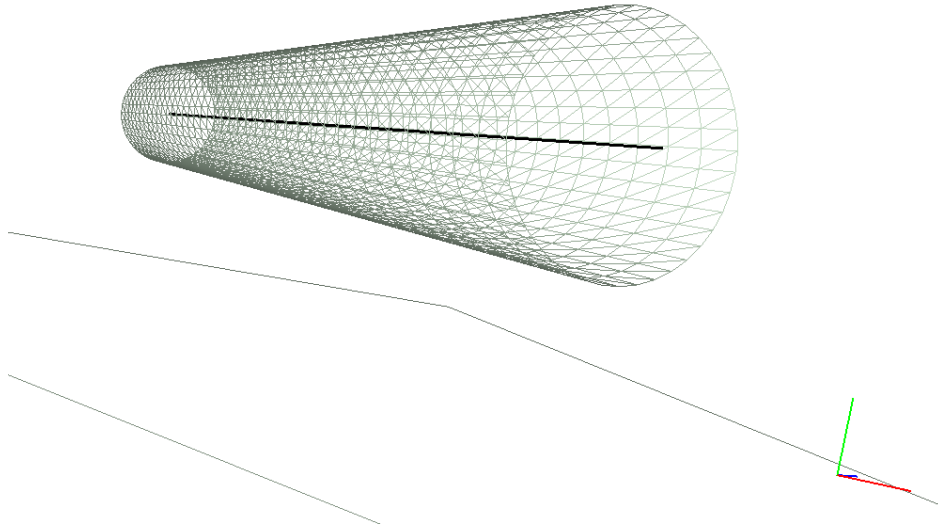


Figure 5: Affichage du squelette du cylindre en position de repos

3 Animation Cylindre

3.1 Animation du squelette

Une fois la position de repos obtenue, il faut maintenant réaliser d'autres squelettes qui correspondent à des *frames* différentes, ainsi dans le cas du cylindre il a été créé 4 positions différentes: 0° , 30° , 60° et enfin 90° . Ces angles correspondent ainsi au pliage du cylindre en 4 étapes (figure 6).

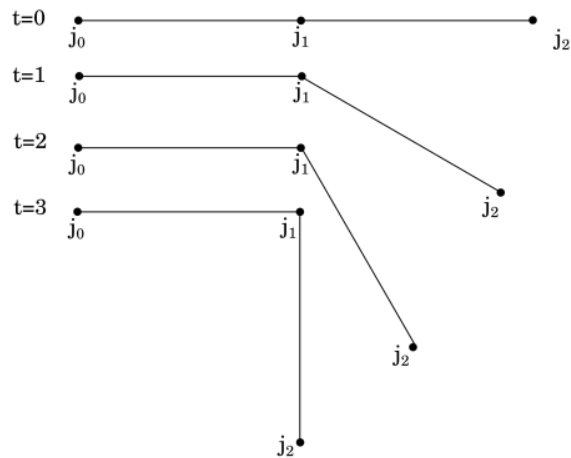


Figure 6: Schema des 4 *frames* du squelette du cylindre pour les angles : 0° , 30° , 60° et 90°

La création de ces squelettes se fait de la même manière que précédemment cependant il faut aussi modifier

le quaternion du joint j_1 qui détermine la rotations de l'os entre j_1 et j_2 . Les quaternions caractérisent une rotation d'angle α selon l'axe normalisé $n = (nx, ny, nz)$ suivant la formule 8.

$$Q = (\sin(\alpha/2) * nx, \sin(\alpha/2) * ny, \sin(\alpha/2) * nz, \cos(\alpha/2)) \quad (8)$$

Dans notre problème, l'axe du cylindre est $n = (1, 0, 0)$ ainsi les quaternions des 4 angles 0, 30, 60 et enfin 90 sont respectivement $(0, 0, 0, 1)$, $(nx * \sin(\Pi/12), 0, 0, \cos(\Pi/12))$, $(ny * \sin(\Pi/6), 0, 0, \cos(\Pi/6))$, $(nz * \sin(\Pi/4), 0, 0, \cos(\Pi/4))$

Ces squelettes sont ainsi créés et stockés dans un ensemble de squelettes pour pouvoir contrôler l'affichage des squelettes successivement et ainsi créer une animation du squelette initial (figure 7).

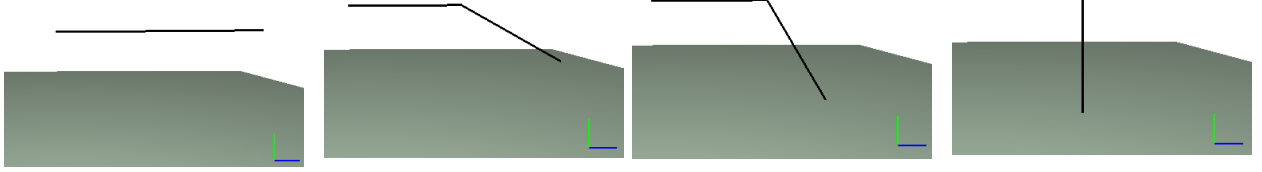


Figure 7: Affichage de l'animation du squelette pour les angles de 0°, 30°, 60° et 90° en allant de la gauche vers la droite

La gestion de l'animation se fait en mesurant le temps écoulé et le changement de *frame* se fait toutes les secondes, il y est ajouté aussi une gestion du sens de l'animation pour revenir en arrière une fois que l'on atteint la dernière frame.

3.2 Déformation de la surface

La déformation de la surface de notre objet à partir de l'animation du squelette se fait en deux étapes que nous allons détailler tour à tour dans cette section : le calcul des poids de *skinning* pour chaque couple sommet-joint puis le déplacement des sommets de l'objet en fonction de la position des joints du squelette. Nous évoquerons également à la fin de cette section une amélioration de l'initialisation des poids de *skinning* permettant de rendre le résultat d'autant plus réaliste.

L'idée majeure de cette partie est de pouvoir déterminer la position des sommets du cylindre en fonction des joints du squelette. Pour cela, il nous faut déterminer "à quel point un sommet donné est dépendant d'un joint donné". Toute cette phrase se résume en un concept : la création de poids de *skinning* pour chaque couple sommet-joint. En pratique, nous ne renseignerons que les poids non nuls, considérant que les couples non mentionnés ont un poids nul.

Ainsi, dès l'initialisation du cylindre, évoqué en Section 2, il nous faut mentionner pour chaque sommet ces poids. Dans un premier temps, afin de faciliter la compréhension de la méthode nous allons considérer que tous les sommets du cylindre dont la position suivant l'axe "z" est inférieure à $L/2$ avec L la longueur du cylindre, auront un poids de 1 avec le joint 0 et 0 avec tous les autres, et que tous les autres auront un poids de 1 vis à vis du joint 1 et 0 pour tous les autres joints. Attention, il est important de noter ici que la somme de tous les poids de *skinning* doit être unitaire afin de conserver au maximum le schéma initial du cylindre. Maintenant que les poids sont déterminés, il est possible de modifier les sommets de l'objet de sorte qu'ils suivent le comportement de leur joint associé.

Considérons p_0 la position d'un sommet du cylindre au repos. Alors calculer sa position à l'instant t notée $p(t)$ en tenant compte des changements de repères opérés par ses joints associés se calcule via l'équation 9.

$$p(t) = \sum_{j \in J_{p_0}} \omega_j S_j(t) p_0 \quad (9)$$

avec J_{p_0} l'ensemble des joints associés à p_0 , ω_j le poids de *skinning* du couple p_0 - j et $S_j(t)$ la matrice de transformation définie dans l'équation 10.

$$S_j(t) = T_j(t)B_j^{-1} \quad (10)$$

avec $T_j(t)$ la transformation du joint j dépendant de l'animation courante, dans le repère global, et B_j^{-1} l'inverse du repère global du joint j de la *bind-pose*.

Ainsi pour calculer la position du point p_0 à l'instant t en fonction de l'animation courante, il est nécessaire de calculer $S_j(t)$ qui dépend d'une inversion de repère de joint et d'une multiplication de repères. Nous allons donc détailler dans les deux prochains paragraphes comment effectuer de telles opérations sur nos repères composés de translations et quaternions.

Inversion de repères Pour calculer le quaternion et la translation de l'inverse d'un repère R , il nous faut repasser par la définition même de l'inversion de repère, à savoir : $R^{-1}.R = Id$ Ainsi en passant sous forme matricielle d'un repère comme rappelé en équation 11, il est possible d'obtenir l'équation 12 qui permet de récupérer la translation t_{inv} et le quaternion q_{inv} du repère inverse de R .

$$R = \left(\begin{array}{ccc|c} & & & \\ & q & & t \\ - & - & - & - \\ & 0 & & 1 \end{array} \right) \quad (11)$$

avec q le quaternion représentant l'orientation de R et donc une matrice 3x3 ici et t la translation sous forme de vecteur 3x1.

$$\begin{cases} q_{inv} = q^{-1} = \bar{q} \\ t_{inv} = -(q_{inv} * t) \end{cases} \quad (12)$$

Multiplication de repères Cette opération a déjà été opérée précédemment dans la Section 2.2 à l'aide de l'équation 4 qui renseigne le passage du repère local au repère global d'un joint. Ainsi donc en adaptant les notations, il est possible de multiplier un repère R_1 par un repère R_2 et d'obtenir R_{12} en calculant son quaternion q_{12} et sa translation t_{12} à l'aide des équations 13.

$$\begin{aligned} q_{12} &= q_1 * q_2 \\ t_{12} &= q_1 * t_2 + t_1 \end{aligned} \quad (13)$$

Ainsi, une fois ces opérations connues, il nous est possible de déterminer pour tout sommet p de l'objet, ici le cylindre, quelle est sa position dans l'espace en fonction de l'animation courante. Aussi ceci nous permet donc d'obtenir les résultats visibles dans la Figure 8, en conservant les mêmes *keyframes* que celles évoquées dans la Section 3.1. On peut ainsi y constater que le cylindre se voit déformé suivant les différentes animations du squelette, à savoir un changement d'orientation du joint central de 0, 30, 60 et 90°. Ainsi donc nous pouvons bien conclure quant au bon fonctionnement de notre algorithme.

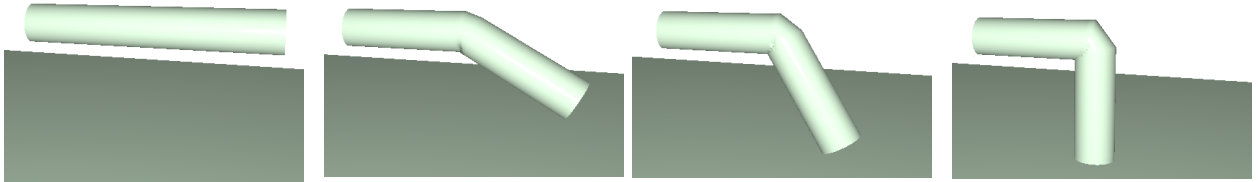


Figure 8: Résultats obtenus en appliquant le *skinning* sur le cylindre avec les animations du squelette décrites en Section 3.1. On peut constater que l'ensemble des sommets du cylindre sont bien placés en fonction de leur joint de référence.

Malgré tout, il semble qu'un problème persiste. En effet si l'on se concentre sur la zone de "cassure", c'est à dire la partie du cylindre où les sommets sont soudainement attribués au joint 1 contrairement aux

précédents qui étaient attribués au joint 0, on peut observer les résultats de la Figure 9. On peut alors voir sur la figure de gauche un problème de cassure au niveau du coude formé par le cylindre à l'endroit où le repère a été modifié lors de l'animation. Il est possible de vérifier d'où vient le problème à l'aide de la figure de droite qui affiche le même cylindre dans la même position mais ne remplit pas les triangles. Ainsi, deux choses peuvent être constatées : premièrement notre algorithme marche parfaitement bien : sur la gauche, tous les points ont l'orientation dictée par le joint 0, et vers le bas, tous les points ont l'orientation dictée par le joint 1. Deuxièmement, l'algorithme fonctionne si bien qu'aucune interpolation n'est faite au niveau du coude. En effet, alors qu'à cet endroit le cylindre devrait s'arrondir, tous les points formant le coude sont soit attribués au joint 1, soit attribués au joint 0 et par conséquent la seule manière de relier les sommets qui sont en bordure est de tracer un triangle qui rend le coude très géométrique.

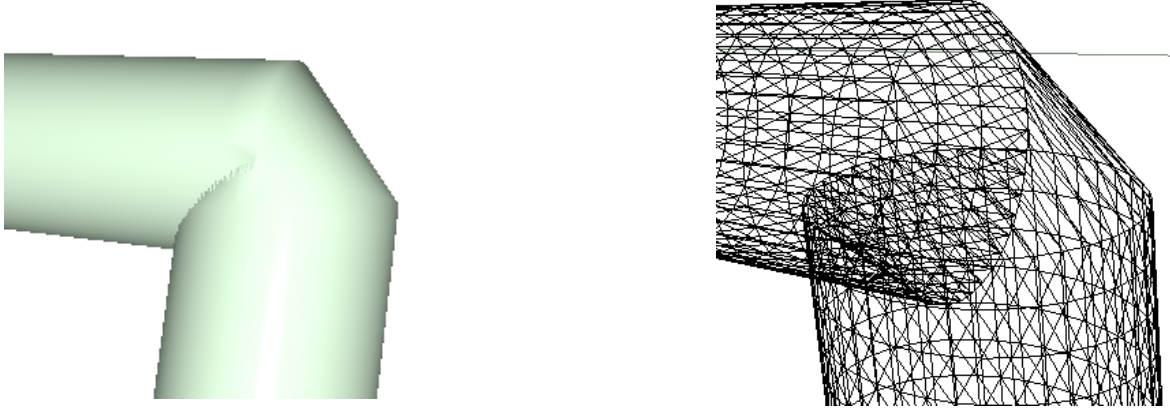


Figure 9: Visualisation du problème engendré par un mauvais choix de poids de *skinning*. À gauche le problème visualisé en triangles remplis, à droite le problème visualisé avec les triangles vides où l'on distingue le raccordement beaucoup trop brusques de sommets formant le coude de l'articulation.

Ces défauts apparaissent parce que nos poids de *skinning* sont très mal choisis. Aussi en déterminant les poids différemment, il est possible d'obtenir une transition continue lors du passage de repère. En effet nous considérons que chaque sommet ne dépendait que d'un seul joint. En réalité, et c'est particulièrement vrai dans le coude, ces sommets dépendent au moins de deux joints. Ainsi en répartissant linéairement les poids de *skinning* entre le joint 0 et le joint 1, en fonction de la composante z du sommet comme expliqué par l'équation 14, il est possible d'obtenir des résultats bien meilleurs illustrés en Figure 10.

$$W_0(p) = 1 - \frac{p_z}{L} \quad \text{et} \quad W_1(p) = \frac{p_z}{L} \quad (14)$$

avec $W_j(p)$ le poids associé au joint j pour le sommet p , p_z la composante du sommet p suivant l'axe z et L la longueur totale du cylindre.

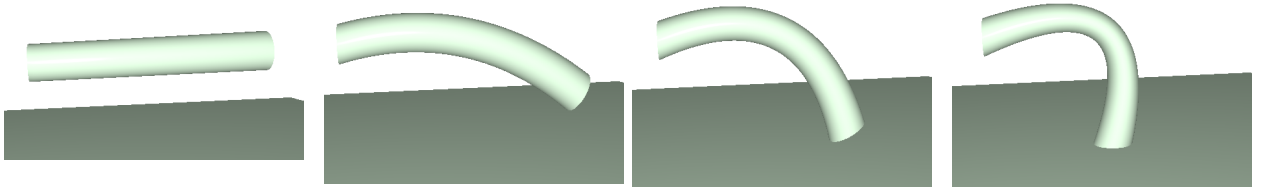


Figure 10: Visualisation de l'effet de la modification des poids de *skinning* sur le cylindre lors des animations décrites en Section 3.1. On peut alors constater que le mouvement autour du coude est beaucoup plus naturel malgré un problème de conservation des volumes.

Ainsi donc on peut en déduire que le choix des poids de *skinning* est particulièrement important afin de pouvoir avoir un rendu naturel. Dans le cas actuel, il est notable que nous avons très largement amélioré le lissage du cylindre au niveau du coude mais malgré tout, un nouveau problème survient, à savoir la conservation du volume n'est pas respectée. En effet, on peut constater en particulier sur la figure de droite de la Figure 10 qu'au niveau du coude formé, apparaît également un resserrement des parois du cylindre ce qui ne devrait pas arriver dans la réalité. Mais ceci étant connu, on peut tout de même voir que pour une assez grande gamme d'angles ($\leq 60^\circ$) le phénomène reste assez moindre et le rendu semble assez naturel. C'est pourquoi il semble pertinent d'utiliser cette méthode dans le sens où elle est rapide, efficace et peu coûteuse en mémoire, en sachant qu'un travail supplémentaire devra être effectué dans l'hypothèse où le changement de repère effectué est trop brusque et risquerait d'engendrer des modifications non réalistes de la structure de l'objet.

3.3 Interpolation de repères

Nous avons pu constater une nette amélioration de la continuité de la forme lorsque les sommets du maillage avaient des poids de *skinning* adaptés. Malgré tout, on peut constater lors de l'animation du cylindre que celle-ci est particulièrement saccadée car nous avons très peu de *keyframes* d'animation et qu'elles sont très éloignées les unes des autres. Ceci rend donc le mouvement du cylindre très peu naturel. Ainsi pour résoudre un tel problème, deux solutions sont envisageables : la première consisterait à créer une liste de *keyframes* contenant les positions du squelette pour tout instant et les afficher les unes après les autres rapidement, comme un dessin qui s'anime lorsqu'il est dessiné sur plusieurs feuilles de papiers et que l'on les alterne rapidement. Cette méthode est particulièrement difficile à mettre en oeuvre surtout si l'on souhaite avoir des scènes complexes où des animations durent plusieurs minutes voire heures (dans les dessins animés par exemple) et est particulièrement coûteuse en mémoire. La seconde méthode, qui est celle que nous développerons ici, consiste à n'avoir que quelques *keyframes* comme c'est le cas pour notre cylindre ici, puis de faire une interpolation du repère d'une *keyframe* au repère de la suivante. Ainsi, la modification engendrera la création d'un nouveau squelette avec les repères interpolés à l'instant t et sera complètement détruit à l'instant $t+1$ ce qui permet d'économiser énormément de mémoire pour une augmentation minime, si ce n'est négligeable, du temps de calculs.

En effet, l'interpolation des repères est une opération qui ne demande pas nécessairement beaucoup de temps de calculs. Il suffit pour cela d'interpoler d'un côté les translations (locales) des joints d'une *keyframe* à l'autre, et d'interpoler les orientations, c'est à dire les quaternions. Attention cependant, s'il est facile d'interpoler une translation à l'aide de l'équation 15, il est moins évident d'interpoler des quaternions. Pour ce faire, nous allons utiliser la méthode dite SLERP (*Spherical Linear intERPolation*) qui consiste non pas à interpoler linéairement chaque composante du quaternion $q = xi + yj + zk + w$ mais bien de les interpoler en suivant la courbe formée par une sphère (d'où la mention *Spherical* qui est primordiale dans cette méthode. L'équation 16 résume mathématiquement l'interpolation en question.

$$t_{12} = (1 - \alpha) * t_1 + \alpha * t_2 \quad (15)$$

avec t_{12} la translation interpolée entre la translation t_1 du quaternion q_1 et la translation t_2 du quaternion q_2

$$q_{12} = \frac{\sin((1 - \alpha)\theta)}{\sin(\theta)} q_0 + \frac{\sin(\alpha\theta)}{\sin(\theta)} q_1 \quad (16)$$

Aussi, en appliquant cette interpolation sur le squelette du cylindre, il est possible alors d'interpoler leurs repères d'une *keyframe* à l'autre et donc de rendre le mouvement beaucoup plus fluide. Même si le résultat est beaucoup plus visible en direct, en exécutant le code que nous fournissons dans l'archive, la Figure 11 permet de voir que le mouvement n'est pas saccadé d'une position à l'autre mais que l'on a bien un mouvement continu du squelette. Il semble important de noter que cette figure n'affiche que deux des multiples positions interpolées. Et on peut constater à l'aide de la Figure 12 que le *Skinning* calculé en

Section 3.2 est parfaitement compatible avec l'interpolation et que le mouvement des sommets du cylindre est lui même fluide.



Figure 11: À gauche et à droite, le squelette respectif de la *keyframe* 0 (la *bind pose*) et de la *keyframe* 1. Au milieu, deux des positions calculées par interpolation des deux *keyframes* mentionnées ci-avant. Ces positions interpolées évoluent au cours du temps.

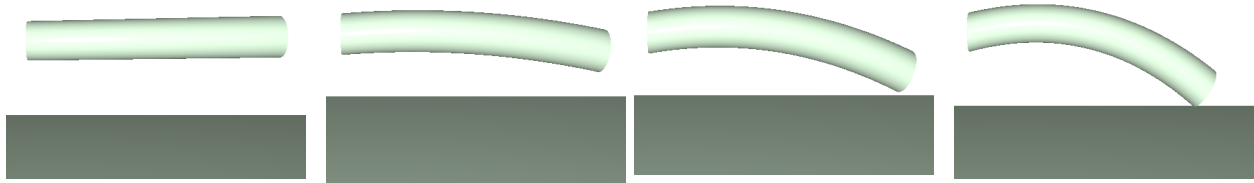


Figure 12: À gauche et à droite respectivement les positions des sommets du cylindre affectés par le *skinning* selon la *keyframe* 0 et 1 de l'animation. Au milieu, deux positions des sommets du cylindre dont le *skinning* est déterminé à partir du squelette interpolé (cf Figure 11) des squelettes des deux *keyframes* 0 et 1 au cours du temps.

4 Skinning d'un monstre

4.1 Généralisation à un objet complexe

Nous allons chercher ici à appliquer la technique de *skinning* évoquée précédemment sur un objet plus complexe : un monstre. Nous n'allons pas redétailler ici les nombreuses étapes de calculs en ce qu'elles sont identiques aux calculs effectués avec le cylindre. De plus l'ensemble des caractéristiques des sommets (positions, normales, textures, poids de *skinning*, ...) sont déjà déterminées dans le fichiers *.obj* chargé. De plus, les positions des joints de la *bind pose* et des différentes *keyframes* de l'animation du monstre peuvent également être directement chargés depuis les fichiers *.skeleton* et *.animations*.

Aussi en chargeant ces données il nous est possible de visualiser que l'ensemble des étapes précédentes fonctionne toujours à l'aide de cette forme complexe. Ces étapes sont résumées en image sur la Figure 13

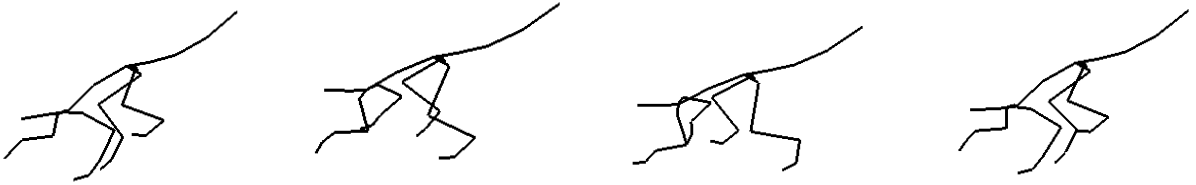
Figure 13: Visualisation des différentes étapes évoquées dans le cylindre dans les sections précédentes et appliquées à un objet complexe : un monstre



(a) Position au repos des sommets du monstre



(b) Position au repos du squelette du monstre



(c) Visualisation de plusieurs des *keyframes* du squelette avec interpolation entre deux *keyframes* pour améliorer la fluidité des mouvements du squelette.



(d) Visualisation du comportement des sommets du monstre sur plusieurs *keyframes* après application du *skinning*.

Suite à cette étude nous pouvons aisément constater que la méthode de création de squelette et de *skinning* est particulièrement efficace pour la déformation d'objets complexes tels que le monstre. En effet en partant uniquement de sommets définis, d'un squelette et des animations du squelette, il est possible de bien faire mouvoir l'ensemble des sommets du monstre afin que le mouvement soit fluide et naturel.

4.2 Visualisation des poids de *skinning*

Sachant que nous n'avons pas choisi par nous même les poids de skinning du monstre, il peut être intéressant de les visualiser en changeant la couleur des sommets. Ainsi, sachant que jusqu'à présent les sommet n'avaient qu'un attribut de texture mais une couleur par défaut, il nous est possible dès l'initialisation du monstre de rajouter pour chacun de ses sommets un attribut de couleur dépendant du poids et du joint auquel il est attribué.

Aussi nous avons choisi de ne considérer dans la couleur que les deux joints avec les plus gros poids. Notons

également que si un sommet n'a qu'un seul joint de référence il aura une couleur différente. Ainsi, pour un sommet donné, ses coordonnées dans l'espace colorimétrique seront telles que présentées dans l'équation 17

$$\begin{aligned} R &= W_{j_0} && \text{si au moins 2 joints associés} \\ G &= W_{j_1} && \text{si au moins 2 joints associés} \\ B &= W_{j_0} && \text{s'il n'y a qu'un joint d'associé} \end{aligned} \quad (17)$$

avec R, G, B les différentes couleurs, W_j le poids associé au joint j et j_0 et j_1 les joints associés au sommet courant et enregistrés par ordre croissant d'ID. Ceci signifie que même si le poids de j_1 est supérieur à celui de j_0 , il sera tout de même enregistré en deuxième place. Ce qui permet de garder une cohérence et une continuité d'un sommet à l'autre.

Ainsi donc en appliquant cette formule à notre monstre, il est possible d'obtenir la répartition des poids illustrée en Figure 14. Attention il est important de savoir que si deux sommets éloignés ont la même couleur, cela ne signifie pas qu'ils ont le même joint de référence, cela signifie simplement que le poids de leur premier joint de référence est prépondérant devant le poids du second joint de référence.



Figure 14: Coloration du monstre en fonction de ses poids de *skinning*. En rouge les sommets dont le poids associé au premier joint est le plus fort, en vert si le poids associé au deuxième joint est plus fort et en bleu les sommets où il n'y a qu'un seul joint d'associé.

Ainsi si un sommet est rouge ou vert il est respectivement plus pondéré par l'action de son premier ou son deuxième joint de référence. S'il est jaune, dans ce cas les joints ont une action similaire sur le sommet en question. Mais si au contraire il est bleu cela signifie qu'il n'a qu'un seul et unique joint de référence. On peut alors régulièrement constater une alternance rouge/jaune/vert/bleu/rouge qui s'explique par la position des joints du monstre : lorsque l'on se déplace le long d'un os du squelette, considérant le joint 0 et le joint 1 les extrémités de l'os :

- Rouge : le sommet est pondéré en grande partie par le joint 0
- Jaune : en se déplaçant le long de l'os, le poids du joint 0 diminue et du joint 1 augmente
- Vert : en continuant de se déplacer, le joint 1 fini par être prépondérant
- Bleu : le joint 1 est le seul joint à impacter le sommet
- Rouge : on se déplace le long d'un nouvel os, le joint 0 n'est plus impactant, désormais seul le joint 1 et le joint 2 le sont. Donc la mécanique recommence avec les joints 1 et 2 comme extrémités d'os.

4.3 Mise en mouvement du monstre

Enfin, afin de donner une cohérence à l'animation du monstre, il semble nécessaire de le faire avancer lorsque l'animation est jouée et reculer lorsque l'animation est jouée en sens inverse. Pour ce faire peu de modifications doivent être apportées. Il est nécessaire de renseigner une variable s'incrémentant ou se décrémentant à chaque pas de temps et permettant de faire avancer ou reculer le monstre en fonction du sens dans lequel est jouée l'animation. Puis en se rappelant que le squelette de l'animation courante du monstre permet de renseigner sur les caractéristiques du monstre dans l'espace ainsi que sa position et que tous les joints de ce squelette sont créés dans l'espace local excepté le premier joint qui est renseigné dans l'espace global, il suffit donc d'appliquer le mouvement à la translation de ce premier joint afin de mettre en mouvement notre monstre dans l'espace. Le résultat est ainsi visible en Figure 15 où l'on voit trois temps où le monstre se déplace devant le mur en avançant. Notons que l'implémentation est telle qu'en exécutant le code, il alterne entre une phase d'avancée et de recul mais que la distinction ne serait pas pertinente à afficher visuellement dans ce rapport.



Figure 15: Affichage de la position du monstre dans le monde à différents instants. De gauche à droite le temps croît. On peut bien constater que le monstre est dans une phase où il avance vers la gauche de l'écran.

On peut alors bien constater ici, à l'aide d'une application concrète, à quel point l'utilisation d'un squelette est pratique puisqu'il suffit de ne modifier la position que d'un seul et unique point du squelette pour pouvoir faire bouger l'ensemble des sommets du maillage ce qui permet de faciliter grandement l'écriture du code mais également d'accélérer le processus puisque l'opération est propagée dans le maillage à partir de l'opération appliquant le *skinning*. Ce qui permet également de gagner beaucoup de temps de calcul.

5 Conclusion

Au cours de ce TP, nous avons pu constater la puissance et la praticité de l'utilisation d'un squelette pour modéliser le comportement d'un ensemble de sommets. En effet, en réduisant N sommets en K joints avec $K \ll N$, nous avons alors pu constater qu'opérer une déformation de l'objet ou encore le mouvoir, était bien simplifié à l'aide de l'opération de *skinning* en comparaison avec traiter chaque sommet du maillage indépendamment les uns des autres. Il est aussi notable que la théorie des quaternions est très utile dans ce problème car non seulement la création des matrices de rotation s'en retrouve très simplifiée mais en plus les calculs sont beaucoup moins coûteux et les quaternions permettent facilement l'interpolation des angles, ce qui pourrait être compliqué et coûteux à gérer avec des matrices de rotations. De plus cette méthode est à la fois applicable à des objets très simples comme un cylindre ou beaucoup plus complexes tels qu'un monstre tant que les animations et le squelette au repos ont été déterminés au préalable par un graphiste.