
Modélisation haptique de formes organiques par métaballes assistée de périphériques de réalité virtuelle

Renaud Ott — Daniel Thalmann — Frédéric Vexo

*Virtual Reality Laboratory (VRlab)
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Suisse CH-1015
{renaud.ott, daniel.thalmann, frederic.vexo}@epfl.ch*

RÉSUMÉ. Cet article présente un système d'esquisse de modèles géométriques utilisant des périphériques de réalité virtuelle. D'un point de vue matériel, ce dernier est composé d'une Station Haptique, permettant un retour d'effort sur les deux bras, et d'un casque de réalité virtuelle. Ceux-ci permettent ainsi de créer un environnement de modélisation intuitif, car le paradigme d'interaction tire parti de ces périphériques. D'un point de vue logiciel, les primitives de dessins sont les surfaces implicites, et notamment les métaballes, des objets mathématiques appréciés pour leur capacité à fusionner et donc à produire des formes arrondies utiles pour modéliser des êtres vivants comme les humains. Une utilisation soutenue de ce système montre qu'il est rapide et efficace pour créer des modèles géométriques exportables vers d'autres systèmes.

ABSTRACT. This paper describes a modeling system for prototyping geometric shapes. It uses a variety of virtual reality devices. On the hardware side, there is a Haptic WorkstationTM, a device conveying force feedback to the upper limbs, and a Head-Mounted Display. VR devices help creating an intuitive modeling interface. On the software side, we use implicit surfaces and metaballs. Designers appreciate this kind of geometric primitives because they are particularly useful to create organic shapes such as humans. Our system provides an efficient and rapid alternative for producing advanced prototypes of geometric shapes. The resulting prototypes can be further refined within other modeling systems.

MOTS-CLÉS : Modélisation, Retour d'effort, Environnement virtuel, Métaballes, Interfaces immersives.

KEYWORDS: Modeling, Designing, Bimanual Haptic feedback, Virtual Environment, Metaballs, Immersive interfaces.

1. Introduction et Motivations

Aujourd'hui, le travail de modélisation est devenu une des plus importantes activités dans le cycle de production d'applications liées à la réalité virtuelle, au développement de jeux vidéos, aux effets spéciaux pour le cinéma ou même dans certains secteurs de l'industrie qui nécessitent la conception de nouvelles pièces. Les infographistes ou les ingénieurs travaillant dans ce domaine sont impliqués dans une série d'étapes allant du tracé d'une esquisse sur un papier jusqu'au rendu final, virtuel ou réel. Des recherches ont montré que les premières étapes de conception sont les plus coûteuses dans le processus de création [LIP 98].

Ainsi, nous nous intéressons à un système de modélisation qui permettrait de réduire le temps nécessaire à ces premières étapes. Pour nous aider dans cette tâche, nous présentons un logiciel assisté par des périphériques de réalité virtuelle. En effet, ceux-ci autorisent la création d'interfaces bien plus intuitives que le traditionnel couple clavier/souris en permettant d'utiliser un paradigme d'interaction plus proche de la réalité.

Cet article présente dans une première partie un état de l'art de la conception de formes assistée par ordinateur ainsi que les observations qui nous ont conduit au concept de modélisation géométrique de métaballes à l'aide de périphériques haptiques. Puis, dans une seconde partie nous décrivons le système proposé. Enfin, nous terminerons par l'évaluation de celui-ci afin d'en déduire les avantages et limitations.

2. Interfaces pour la modélisation

Dans un premier temps, nous allons examiner le processus de modélisation couramment utilisé par les professionnels, puis nous allons voir quelles améliorations peuvent être apportées pour l'optimiser, notamment en se servant d'un périphérique haptique fonctionnant avec les deux mains.

Lors de la création d'un modèle 3D, plusieurs étapes peuvent être délimitées. La première est toujours la même quelque soit le type de production visé ; il s'agit d'une étape de prototypage, pendant laquelle la plupart des acteurs du processus de développement donnent leur avis sur le modèle désiré ou les problèmes inhérents à certains choix de conception. Généralement, c'est une esquisse réalisée à la main avec un papier et un crayon [DEI 00], ou une maquette en argile, mais on peut aussi citer certains systèmes, comme celui de Alexe et al. ou de Perry et al., permettant de réaliser cette fonction électroniquement, donc généralement plus rapidement et avec un droit à l'erreur [ALE 04] [PER 01]. Vient ensuite la phase de modélisation géométrique qui contient elle-même plusieurs raffinements, puis le placage de textures si nécessaire et enfin l'animation. Ces étapes ne sont pas forcément dans l'ordre indiqué ; elles sont fonction de la méthodologie préférée par l'infographiste [LAW 97]. Tout au long de cet article, nous nous sommes focalisés sur le début de ce processus, c'est-à-dire la création de l'esquisse, puis d'un premier modèle géométrique destiné à être raffiné. En effet, Lipson et al., dans [LIP 98], précisent qu'une interface efficace et intuitive

pendant ces étapes permet non seulement d'économiser du temps, mais aussi de simplifier le raffinement futur de l'objet. Cependant, Tovey et al. ont montré qu'il est techniquement difficile de créer un tel système, qui soit suffisamment généraliste pour être utilisé dans la plupart des secteurs de l'industrie [TOV 02]. Nous ne nous penchons donc pas sur ce problème global, mais nous examinerons plutôt un cas particulier de la modélisation : la création d'humains virtuels.

Lors de nos recherches nous avons souvent été conduit à modéliser des humains virtuels [THA 02] [THA 03] [MAG 04]. Bien que nous avons l'habitude de travailler avec des outils classiques (Esquisses papier, 3D Studio Max [DIS], Maya [ALI]) pour créer ces personnages, nous avons aussi cherché à simplifier ce travail de création. Dans [CAL 02], nous avons ainsi proposé un logiciel permettant de modéliser une tête virtuelle grâce à un maillage déformable. Celui-ci a la particularité de fonctionner avec un gant de réalité virtuelle (en anglais, *DataGlove*, périphérique permettant de capturer la posture de la main) et un traqueur magnétique (pour récupérer la position et l'orientation de la main). L'utilisateur peut alors manipuler avec ses doigts la surface du modèle (qui est initialement une sphère, la forme géométrique simple se rapprochant le plus d'une tête) pour déplacer les sommets du maillage. Le paradigme d'interaction est en réalité celui de la pâte à modeler (en anglais, *Clay Modeling*). En fait, ce paradigme s'appuie sur la théorie des modèles déformables (en anglais, *Free-Form Deformation, FFD*, [ALL 89] [BLO 90] [GAL 91] [HSU 92]). Cependant, lors de l'utilisation de notre système, plusieurs limitations sont rapidement mises en évidence.

Premièrement, c'est l'absence de retour d'effort qui trouble le plus l'utilisateur. En effet, utiliser une main pour déformer un objet qu'on ne sent pas physiquement déstabilise. Pourtant, FFD est une technique qui s'associe de manière simple avec des périphériques haptiques, puisqu'elle permet d'interagir directement avec les sommets d'un modèle. D'ailleurs, de nombreuses recherches ont étudié la faisabilité d'un tel système ; on peut notamment citer le travail de Foskey et al. [FOS 02], de McDonnell et al. [MCD 02] ou encore de Kuo-Cheng et al. [KUO 03] : les deux groupes présentent un logiciel de déformation de modèles 3D utilisant *FreeForm™* [SEN] et le PHANToM [SAL 97], un exosquelette haptique avec six degrés de liberté qui n'exerce le retour d'effort que sur un seul doigt, ce qui nous amène à la seconde limitation.

Deuxièmement, le fait de n'interagir qu'avec une seule main semble rendre l'interface moins fonctionnelle [BER 03]. En effet, cela implique nécessairement d'avoir un mode de visualisation qui permet de tourner le modèle pour voir les faces non-visibles à l'écran, obligeant l'utilisateur à stopper son travail de modélisation pour replacer le modèle dans l'espace. Foskey et al., ainsi que McDonnell et al. réalisaient cette opération en déplaçant le PHANToM sur un des boutons de l'interface. D'autre part, le travail d'un objet mou avec une seule main est délicat dans certaines situations. Si l'on fait l'analogie avec la poterie, on se rend compte que la main qui façonne la forme ne bouge pas : en réalité c'est le modèle qui bouge. Dans cette situation par exemple, la seconde main pourrait être celle qui tourne l'objet. D'une manière générale, l'utilisation de deux mains est bien plus naturelle et semble simplifier l'interface.

Troisièmement, la technique des Free-Forms Deformation pour modéliser un humain nécessite du temps : en réalité, cette technique est efficace lorsqu'il s'agit de raffiner un modèle existant car il faut forcément débiter avec une forme de base pour pouvoir travailler, et si cette dernière est trop différente de celle que l'on cherche à obtenir, il y a énormément de modifications à apporter. Or, comme nous cherchons à réaliser un système qui trouve son utilité plus tôt dans le processus de modélisation, c'est-à-dire à un moment où le modèle n'existe pas encore, nous avons besoin d'un moyen plus adapté. Ainsi, une technique utilisant des surfaces implicites [ANG 01] nous est apparue comme étant indiquée, notamment parce qu'il ne suffit que de quelques primitives pour approximer une forme complexe [SHE 95] et aussi parce que celles-ci sont très adaptées à la représentation des formes organiques comme les humains ou les animaux [MAG 97]. Un sous-ensemble particulier de surfaces implicites, communément connu sous le nom de métaballe [BLI 82] est devenu un concept très en vogue en informatique (en anglais, *metaball* ; on utilise aussi le terme *blob*, du nom d'un film des années 50 où l'on a pu voir pour la première fois ce genre de formes). Dans cette étude, nous nous sommes donc limités à la modélisation par métaballes, même si cette technique n'est pas sans contraintes : en effet, nous désirons un système qui permette de visualiser directement le résultat final, sans passer par une phase de rendu (en anglais on le qualifierait de système *wysiwyg*, "*What you see is what you get*", "Ce que vous voyez, c'est ce que vous obtiendrez au final"). Il faut donc pouvoir créer et visualiser des métaballes en temps réel, alors même que cette technique est très coûteuse en calculs. Nous verrons dans la section 3.4 de ce document comment nous avons contourné ce problème.

En définitive, nous venons de voir trois points à prendre en compte pour la création de notre système de modélisation, à savoir, une interface immersive, intuitive, qui utilise les métaballes comme primitive de dessin. La partie suivante expose l'architecture du système que nous avons réalisé, qui comprend un périphérique haptique avancé, la station haptique, un système de visualisation immersif, grâce à un casque de réalité virtuelle, et l'interface logicielle pour la modélisation à l'aide de métaballes.

3. Description du système

Cette partie décrit l'architecture logicielle et matérielle du système de modélisation à l'aide de métaballes. La première sous-partie présente le périphérique principal avec lequel l'utilisateur interagit, c'est-à-dire la station haptique, tandis que dans les sous-parties suivantes, nous décrirons les différents modules qui ont servi à l'implémentation du système.

3.1. Les périphériques matériels : station haptique et casque de réalité virtuelle

Notre station haptique est un produit vendu par ImmersionTM et commercialisé sous le nom de Haptic WorkstationTM [Imm a]. Cette station a l'avantage unique de pouvoir créer un retour d'effort sur les deux membres supérieurs permettant ainsi à

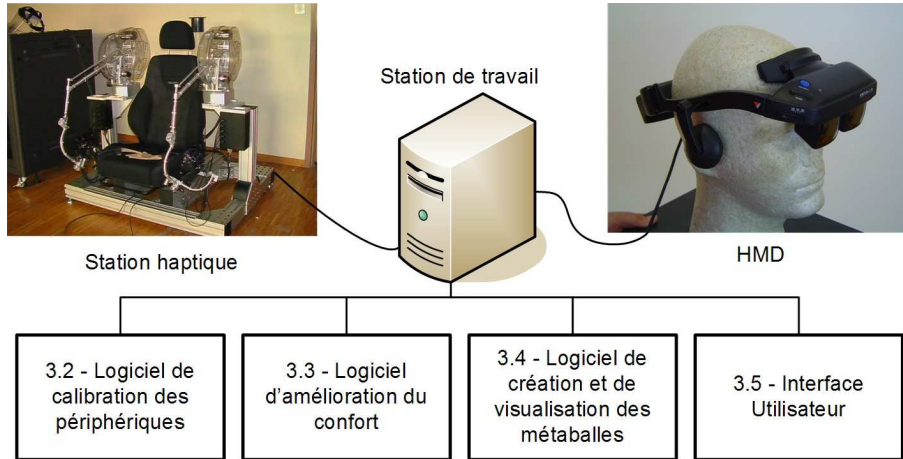


Figure 1. Architecture logicielle et matérielle du système de modélisation

l'utilisateur de manipuler des objets avec ses deux mains, ce qui semble, au premier abord, naturellement plus intuitif.



Figure 2. Les périphériques de la station haptique : à gauche, le CyberGlove™ ; au centre, le CyberGrasp™ ; à droite, le CyberForce™ et le CyberTrack™

Comme le montre la figure 2, pour parvenir à ce niveau d'immersion, la station haptique intègre en fait plusieurs périphériques devenus aujourd'hui classiques en réalité virtuelle, à savoir :

- Une paire de CyberGlove™ qui sont des gants servant à capturer la posture des mains. Par main, vingt-deux angles entre phalanges et métacarpes sont évalués à l'aide de jauges de contraintes, permettant ainsi de connaître la position des doigts.

- Une paire de CyberGrasp™, des exosquelettes se positionnant sur la main, permettant de tendre les doigts à l'aide de câbles (on ne peut pas contraindre l'utilisateur de fermer son poing). Ce périphérique est utilisé habituellement pour donner l'impression à l'utilisateur qu'il tient un objet dans sa main, l'exosquelette l'empêchant

alors de fermer le poing. Sur chacun des doigts, on peut définir une force maximale d'environ $10N$.

- Deux CyberForce™ qui sont le pendant du CyberGrasp™ pour les bras. Ils servent à créer une force tridimensionnelle (maximum $80N$) au niveau du poignet, modifiant ainsi la position de la main de l'utilisateur si celui-ci ne résiste pas. Cependant, ce système ne peut pas être utilisé pour le contraindre à modifier l'orientation de sa main.

- Enfin, deux CyberTrack™, intégrés dans l'exosquelette des CyberForces™ permettent de connaître la position et l'orientation du poignet. La précision et la fréquence de rafraîchissement de ces valeurs est très largement supérieure aux systèmes magnétiques ou optiques (précision de l'ordre du dixième de millimètre en position et de l'ordre du dixième de degré en orientation ; ces valeurs étant mises à jour à une fréquence d'environ $1000Hz$).

Notre application de modélisation n'emploie pas tous ces périphériques selon leur mode d'utilisation classique. En effet, nous verrons dans la partie 3.5 que nous avons fait le choix de créer une interface dont tous les éléments servent : en d'autres termes, nous utilisons des périphériques uniquement si ceux-ci apportent une information à l'utilisateur du système. La manière dont l'utilisateur interagit avec l'interface ne nécessite pas que l'on représente sa main réelle par une main virtuelle fidèle à la réalité (une paume, cinq doigts). Les gants servent donc à détecter des postures particulières correspondant à des actions et non à modéliser une main. Ainsi, le retour d'effort sur chacun des doigts, créé par les CyberGrasp™, ne sert pas à simuler la saisie d'objets, mais à donner des initiatives à l'utilisateur pour que celui-ci adopte une posture de doigts particulière en fonction du contexte (cf. §3.5).

Habituellement, la station haptique est programmée à l'aide de la bibliothèque VHT [Imm b] fournie par Immersion™. Cette bibliothèque comprend de nombreuses fonctions permettant d'organiser un environnement virtuel composé d'objets simples (sphères, cylindres, cubes et volumes convexes) et de les manipuler à l'aide de la station. Malheureusement dans notre cas, il s'agit d'objets dont la forme est dynamique, ce qui dépasse les possibilités de VHT. Nous avons donc choisi d'utiliser notre propre bibliothèque développée de manière interne.

Enfin, le système matériel ne serait pas complet sans un périphérique de visualisation. Nous avons choisi d'utiliser un casque de réalité virtuelle (*Head-Mounted Display* en anglais, noté *HMD* dans la suite de ce document) qui est, au choix, semi-transparent, ce qui signifie que les images virtuelles se superposent sur ce que l'on verrait sans casque, ou fermé (seules les images virtuelles sont visibles). En outre, ce dernier possède un traqueur inertiel et magnétique intégré qui permet d'évaluer son orientation et sa position.

Ces périphériques étant indépendants les uns par rapport aux autres, nous avons du mettre en place un protocole de calibration comme décrit dans la sous-partie suivante.

3.2. *Le logiciel de calibration*

Un point très important pour favoriser l'immersion de l'utilisateur dans un système haptique avec HMD est de coordonner ses mouvements avec la visualisation. C'est d'autant plus important lorsque le HMD fonctionne en mode semi-transparent, puisqu'il faut que le casque superpose alors les mains virtuelles sur les mains réelles.

Pour atteindre ce but, nous avons présenté dans [OTT 05] un mécanisme permettant de placer dans un même espace les coordonnées renvoyées par les traqueurs des deux mains. En effet par défaut, la station haptique n'est pas du tout calibrée. Cela se manifeste lorsqu'on joint les mains, par exemple, on obtient deux mains virtuelles très éloignées l'une de l'autre. Dans notre système de modélisation par métaballes, nous avons réutilisé ce module, tout en y rajoutant selon le même mécanisme la calibration du traqueur du HMD.

L'environnement visuel ainsi construit est donc relativement intuitif pour l'utilisateur : en effet, la caméra virtuelle bouge en accord avec les mouvements de sa tête, et les déplacements de ses mains sont conformes à ceux qu'il effectue réellement. De ce point de vue, l'immersion visuelle est grandement facilitée. Cependant, dans cette situation l'utilisateur est contraint dans un exosquelette dont le poids, bien qu'il soit compensé mécaniquement, n'est pas négligeable. De plus cette surcharge n'est pas constante, elle est fonction de la configuration spatiale de l'exosquelette et donc de la position de la main. La partie suivante montre comment nous avons surmonté ce problème de confort.

3.3. *Le logiciel d'amélioration du confort*

L'utilisation prolongée de cette station pour accomplir une tâche se révèle être anormalement fatigant. En effet, les muscles des bras perçoivent immédiatement la surcharge causée par l'exosquelette (environ 600g sur chaque main lorsque les bras sont tendus), ce qui a pour conséquence d'épuiser l'utilisateur en une dizaine de minutes. Bien entendu, cela nuit considérablement à l'immersion.

Dans [OTT 05], nous avons proposé une solution consistant à compenser le poids de l'exosquelette, voir même celui des bras de l'utilisateur, en se servant du retour d'effort. Cette méthode consiste à utiliser un champ de force précalculé qui contient les valeurs des forces de compensation, puis de les appliquer en permanence. L'environnement virtuel gagne ainsi en réalisme. En effet, bien que ce ne soit pas une amélioration visuelle, l'utilisateur ne cherche plus à s'expliquer une surcharge qu'il ne percevait pas dans la réalité. Bien sûr, la sensation d'être "emprisonné" dans un exosquelette ne disparaît pas, mais ce dernier ne le fatigue plus autant. Nous avons démontré que ce système accentue de manière significative le confort, et en définitive l'immersion. Dans la sous-section suivante nous présentons l'interface de modélisation avec laquelle l'utilisateur interagit.

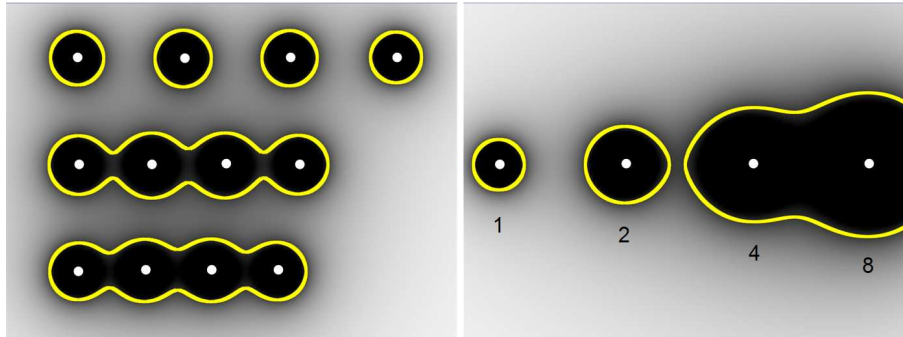


Figure 3. Métaballes dans un espace bidimensionnel. Les points clairs représentent le centre des métaballes. Le champ scalaire est caractérisé par le dégradé. Les lignes sont les courbes implicites, résultats de l'algorithme ; en 3D, ces courbes deviennent des surfaces. A gauche, plusieurs métaballes se mélangeant ; à droite, des métaballes régulièrement disposées dont on fait varier uniquement le rayon

3.4. Création et visualisation des métaballes

Cette section décrit l'algorithme utilisé pour le calcul et la visualisation des métaballes. On peut découper cet algorithme en deux parties, la première étant la création du champ scalaire en lui-même, tandis que la seconde correspond au calcul de la surface frontière.

L'espace de travail est un champ scalaire où chaque point P est caractérisé par une valeur x_P . Dans la figure 3, une valeur élevée du champ est représenté par une couleur sombre, et une valeur faible est caractérisée par une couleur claire. Dans notre implémentation, cette valeur x est toujours positive.

Chaque métaballe M_i est caractérisée par un point, son centre, noté C_i , visible sur la figure 3, et par une valeur d'influence, notée T_i .

La formule [1] décrit la fonction $f_i(P)$ qui, pour une métaballe donnée M_i , exprime l'influence de cette dernière sur le champ. La formule [2] montre que pour chaque point du champ, nous trouvons sa valeur en sommant les influences de toutes les métaballes.

$$f_i(P) = \frac{T_i}{(1 + \text{distance}(P, C_i)^2)} \quad [1]$$

$$x_P = \sum_{M_i} f_i(P) \quad [2]$$

Une métaballe est donc un point (1D) qui influence un champ. Par analogie, nous pouvons aussi imaginer des courbes (2D) voire même des volumes (3D) qui influencent ce dernier. Ainsi, notre implémentation contient aussi une primitive *métaligne*, caractérisée par deux points définissant un segment, qui s'avère très utile pour les objets allongés comme les jambes d'un humain virtuel.

Ce que nous cherchons ensuite à visualiser est une surface implicite consistant en un ensemble de points satisfaisant la fonction $f(P) = K$, où K est un paramètre dont le choix de la valeur est laissé à l'utilisateur. Ce paramètre influence l'allure générale de la surface. Une valeur élevée de K tend à rapprocher la surface vers le centre des métaballes, tandis qu'une valeur faible a tendance à englober un plus grand volume.

En l'état, cette surface ne peut pas être rendue graphiquement, puisqu'elle est décrite par une fonction continue. Il faut donc polygonaliser cette surface. Ceci passe forcément par une discrétisation de l'espace, car on ne peut naturellement pas calculer la valeur du champ en tous points de l'espace, de même qu'on ne peut pas trouver tous les points P satisfaisant la fonction $f(P) = K$. La solution telle qu'elle a été implémentée en 1982 par Blinn, ne pourrait toujours pas aujourd'hui être calculée en temps réel. L'algorithme de discrétisation utilisé est celui des "*Marching Cubes*" [LOR 87] (voir figure 4, pour un champ 2D). Le principe est de découper le champ en sous-parties cubiques, puis d'analyser les valeurs du champ en chaque sommet de ces cubes (notre implémentation utilise une discrétisation de $64 \times 64 \times 64$ cubes). Si un cube ne contient que des sommets dont les valeurs sont supérieures à K , ce dernier est à l'intérieur d'une métaballe, alors que si les valeurs sont toutes inférieures à K , il est en dehors. Cependant dans ces deux cas, le cube n'intersecte pas la surface implicite, donc il ne participe pas pour la création de celle-ci. En fait, les seuls cubes participant sont ceux dont les valeurs des sommets encadrent K . Ensuite il y a deux manières de construire la surface : soit en n'utilisant que le fait qu'un sommet est à l'intérieur ou à l'extérieur de la métaballe, soit en interpolant la surface par rapport à la vraie valeur du champ. La première présente l'avantage d'être un peu plus rapide tandis que la seconde, celle que nous avons choisie, approxime plus fidèlement la surface et est donc visuellement plus lisse. La figure 4 illustre cette différence sur un champ bidimensionnel.

3.5. *Le paradigme d'interaction : L'interface du logiciel*

Cette partie présente le paradigme d'interaction qui permet de créer un modèle. Nous avons vu dans la partie précédente comment les métaballes étaient paramétrées. Voici donc une liste récapitulative des actions possibles que notre interface doit au minimum pouvoir effectuer :

- Créer une métaballe.
- Créer une métaligne.
- Faire varier le rayon d'une métaballe.
- Faire varier le rayon d'une métaligne.

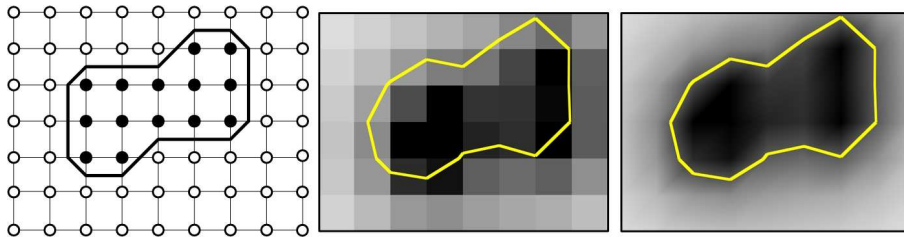


Figure 4. Principe des "Marching Cubes" avec une discrétisation d'un champ 2D en 8×6 carrés. A gauche, les sommets de chaque carré sont évalués de manière binaire (i.e. est "en dedans" ou "en dehors" de la métaballe). Au centre et à droite, la courbe résulte d'une interpolation entre les valeurs réelles de chaque sommet

Nous avons choisi d'utiliser des postures de mains pour déclencher les actions. Nous avons fait ce choix pour des questions de rapidité. En effet, l'utilisateur peut facilement ouvrir ou fermer un doigt à tout moment, sans avoir à déplacer la main. Il y a trois postures différentes : le poing fermé, l'index tendu, et l'index et le majeur tendus. Ces postures sont suffisamment différentes pour être discriminées aisément. Il n'y a donc pas besoin de mettre en place un complexe algorithme de reconnaissance de geste. L'automate de la figure 5 présente le principe pour effectuer les actions ci-dessus. Le retour d'effort sur les doigts grâce aux CyberGraspTM est utilisé pour aider l'utilisateur à changer d'état. Par exemple, si il est dans l'état de repos et qu'il ouvre les deux doigts de la main droite, les CyberGraspTM vont légèrement tirer les doigts de la main gauche pour l'aider à passer dans l'état "Prévisualise une métalligne".

La figure 6 présente la vue de l'interface par l'utilisateur dans le HMD. Comme nous l'avons déjà vu, le modèle qui représente les mains n'est pas réaliste, afin de ne pas faire croire à l'utilisateur qu'il peut interagir avec ses doigts.

Le principe pour ajuster la taille d'une métaballe ou d'une métalligne est très simple. Une fois que celle-ci est créée (on vient donc de définir le centre), le rayon de la métaballe varie en fonction de la distance entre le doigt et ce centre, la visualisation restant en temps réel. Cependant, il apparaît difficile lorsque l'on veut faire une métaballe de petit rayon par exemple, et que l'on s'est déjà beaucoup éloigné du centre de s'en rapprocher. C'est justement dans ce genre de situations que le retour d'effort est très efficace. L'idée sous-jacente est de définir le centre de la métaballe comme attracteur de la main, et de faire varier la force de manière proportionnelle au rayon. Ainsi, si l'utilisateur souhaite créer une métaballe ayant un grand rayon, donc un fort volume, il doit exercer une force importante, tandis qu'il peut se relâcher pour laisser la station haptique ramener sa main vers le centre s'il en veut une de faible rayon. Dans notre implémentation finale, nous avons choisi une force de $0,2 N/cm$ de rayon. Étant donné que le volume de travail dont la forme est plus ou moins cubique mesure environ $80 cm^3$, cela représente une force maximale de $16 N$ (pour une métaballe de $1,6 m$ de diamètre).

Nous venons de voir l'architecture générale du système, composée de périphériques de réalité virtuelle, d'un logiciel de calibration et d'amélioration du confort, ainsi que d'un système de création et de visualisation des métaballes. Dans la partie suivante, nous présenterons son évaluation.

4. Résultats et évaluation des tests

Dans cette section nous présentons l'évaluation du système de modélisation par métaballes. Nous allons notamment voir en quoi l'utilisation d'une interface immersive et des technologies haptiques contribuent à l'amélioration du système. Enfin, nous allons étudier son comportement global lors d'une utilisation soutenue.

D'une manière générale, les personnes ayant testé cette interface sont des infographistes professionnels (3 personnes), qui connaissent parfaitement un logiciel en particulier. Mais, nous avons aussi retenu l'avis d'informaticiens (3 personnes aussi) ne connaissant la modélisation que dans le principe. Les infographistes à qui nous nous sommes adressés maîtrisent 3D Studio Max® ou Maya®. Bien entendu, nous ne leur avons pas demandé de comparer leur logiciel favori au notre, mais seulement d'évaluer l'interface en terme de rapidité de prise en main, de facilité d'utilisation, et enfin d'évaluer l'intérêt de celle-ci par rapport au but fixé : "créer un système permettant de réaliser une esquisse d'un modèle géométrique de forme humaine ou animale

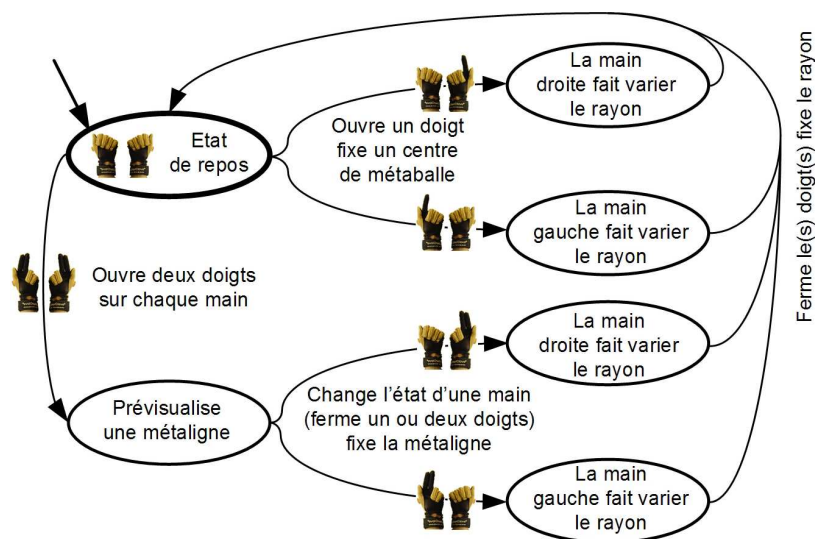


Figure 5. Automate régissant l'interface utilisateur. La partie supérieure sert à créer des métaballes et la partie inférieure des métalignes

qui pourrait aussi être raffiné ultérieurement à l'aide d'autres logiciels". Voici les questions posées auxquelles ils devaient répondre :

- Le paradigme d'interaction à l'aide d'interfaces de réalité virtuelle est-il intuitif ?
- Avez-vous l'esquisse souhaitée après utilisation ? Est-elle vraiment fonctionnelle en vue d'un futur raffinement ?
- Donnez une évaluation du temps de création de l'esquisse ?

D'après les testeurs, le paradigme d'interaction n'est pas complètement intuitif, particulièrement pour la posture de la main lors de la création d'une métaligne. En effet, s'il paraît logique d'avoir à tendre un doigt (comme pour montrer quelque chose) pour placer un centre de métaballe, puis de le fermer pour valider la création de la métaballe, il n'en est pas de même pour la ligne : tendre l'index et le majeur sur les deux mains n'a pas de réelle signification. Cependant, une fois ces postures connues et comprises, la souplesse d'utilisation représente un gros avantage par rapport aux interfaces classiques : la plupart des testeurs trouvant particulièrement appréciable de rester en permanence à proximité de leur modèle, c'est-à-dire de ne pas avoir à déplacer ses mains pour actionner des composants d'interfaces (boutons, ascendeurs, etc.).

D'autre part, en ce qui concerne l'interface immersive, utilisant le HMD, les traqueurs sur les mains et le retour haptique, plusieurs remarques ont été faites. En premier lieu, ce mécanisme est intuitif : les testeurs réalisent rapidement que les mains réelles déplacent les mains virtuelles, et qu'un mouvement de la tête oriente et bouge

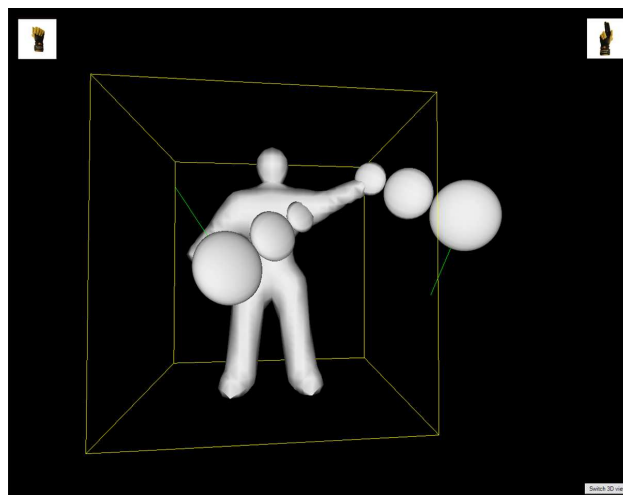


Figure 6. *Vue de l'interface utilisateur à travers le HMD. Les mains sont composées des disques permettant de symboliser les doigts. Le modèle a été créé en quatre minutes à l'aide de 5 métaballes et de 5 métalignes*

la caméra comme dans la réalité. En outre, le retour d'effort notamment pour paramétrer le rayon des métaballes offre une réelle aide à l'utilisation. En effet, le fait d'attirer la main vers un point en fonction de sa distance permet de donner très rapidement une information de position relative. Cette remarque n'est pas surprenante, P. Fuchs et al. l'énonce déjà dans [FUC 04] en faisant remarquer que le traitement d'une information visuelle est plus complexe et long que celui d'une information proprioceptive.

Enfin, le système vu d'une manière générale est intéressant par rapport aux buts fixés. On peut très rapidement dessiner et visualiser des formes, mais il faut admettre qu'un certain temps d'apprentissage (environ dix minutes) est nécessaire pour arriver à modéliser ce que l'on désire. Cela découle directement du fait que les surfaces implicites fonctionnent avec des structures de contrôle, comme les splines. C'est-à-dire que l'endroit où l'on place les points de contrôles ne correspond pas à l'endroit où la surface passe : il faut extrapoler la forme désirée, pour trouver le bon centre des métaballes. Cependant, en ce qui concerne le problème de la modélisation interactive qui n'était pas forcément assurée à cause de la nature même des surfaces implicites, le système reste en temps-réel jusqu'à une centaine de métaballes, ce qui suffit largement pour réaliser l'esquisse d'un modèle géométrique tel que celui de la figure 6. D'autre part, une fois créée, les testeurs ont tous trouvé que leur esquisse répondait à leurs attentes.

En définitive, notre système semble présenter des fonctionnalités intéressantes pour la modélisation. Il s'avère effectivement rapide pour créer un modèle simple d'humain virtuel, alors qu'il s'agit normalement d'une tâche délicate. Les personnes novices en modélisation trouvent donc là un système particulièrement intuitif qui ne nécessite pas d'apprentissage particulier, et les infographistes professionnels relèvent plusieurs points, principalement dans l'utilisation des technologies haptiques, qu'ils aimeraient bien voir intégrés dans leurs interfaces de modélisation.

5. Conclusion et perspectives

Tout au long de cet article, nous avons étudié comment réaliser un système de modélisation géométrique avec une interface immersive créée à l'aide de périphériques de réalité virtuelle. Les principaux atouts de cette interface sont l'utilisation des deux mains, du retour d'effort, possible grâce à la station haptique, et du HMD, qui permettent ainsi d'avoir un paradigme d'interaction très intuitif. La durée d'apprentissage, principale barrière à l'utilisation des systèmes de modélisation classiques, s'en trouve donc considérablement réduite. Cet article montre aussi que les métaballes sont une primitive de dessin efficace lorsqu'il s'agit de réaliser des esquisses. En effet, les machines d'aujourd'hui permettent d'effectuer une implémentation en temps réel d'un algorithme de visualisation de surfaces implicites, ce qui en atténue le principal défaut (le contrôle global de la forme). Dans un futur proche, nous souhaitons intégrer les périphériques utilisés et le paradigme d'interaction dans une extension (en anglais, *plug-in*) d'un logiciel de modélisation qui propose des métaballes comme primitives.

Remerciements

Ce travail a été partiellement supporté par l'Office Fédéral Suisse pour l'Éducation et la Science dans le cadre du Réseau d'Excellence de l'Union Européenne IST - ENACTIVE¹ et du Réseau d'Excellence IST - AIM@SHAPE².

6. Bibliographie

- [ALE 04] ALEXE A., GAILDRAT V., BARTHE L., « Interactive modelling from sketches using spherical implicit functions », *Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa, AFRIGRAPH '04*, Stellenbosch, South Africa, 2004, ACM Press, p. 25–34.
- [ALI] ALIAS®, « Maya 6®, 3D modeling, animation, effects, and rendering solution, <http://www.alias.com/eng/products-services/maya/index.shtml> ».
- [ALL 89] ALLEN J. B., WYVIL B., WITTEN I. H., « A Method for Direct Manipulation of Polygon Meshes », *Proceedings of Computer Graphics International '89*, 1989, p. 451–469.
- [ANG 01] ANGELIDIS A., CANI M.-P., « Surfaces implicites multi-résolutions », Master's thesis, Institut National Polytechnique de Grenoble, France, juin 2001.
- [BER 03] BERNSTEIN N. L., LAWRENCE D. A., PAO L. Y., « Design of a uniaxuated bi-manual haptic interface », *Proceedings of Haptic Interfaces for Virtual Environment and Teleoperator Systems, 11th Symposium*, 2003, p. 310–317.
- [BLI 82] BLINN J. F., « A Generalization of Algebraic Surface Drawing », *ACM Trans. Graph.*, vol. 1, n° 3, 1982, p. 235–256, ACM Press.
- [BLO 90] BLOOMENTHAL J., WYVILL B., « Interactive techniques for implicit modeling », *Dans les Proceedings of the 1990 symposium on Interactive 3D graphics, SI3D '90*, Snowbird, Utah, United States, 1990, ACM Press, p. 109–116.
- [CAL 02] CALINON S., VEXO F., « WiCKed : Sculpture sur tête virtuelle », Rapport Interne EPFL, juin 2002.
- [DEI 00] DEISINGER J., BLACH R., WESCHE G., BREINING R., SIMON A., « Towards Immersive Modeling - Challenges and Recommendations : A Workshop Analyzing the Needs of Designers », *Dans les Proceedings of ISATA'00*, Dublin, Ireland, 2000.
- [DIS] DISCREET®, « 3D Studio Max®, animation, modeling and rendering software, <http://www.discreet.com/products/3dsmax/> ».
- [FOS 02] FOSKEY M., OTADUY M. A., LIN M. C., « ArtNova : touch-enabled 3D model design », *Dans les Proceedings of the IEEE Virtual Reality*, 2002, p. 119–126.
- [FUC 04] FUCHS P., MOREAU G., *Le Traité de la Réalité Virtuelle, deuxième édition*, vol. 1, Les Presses de l'École des Mines de Paris, 2004.
- [GAL 91] GALYEAN T. A., HUGHES J. F., « Sculpting : an interactive volumetric modeling technique », *Dans les Proceedings of SIGGRAPH'91*, ACM Press, 1991, p. 267–274.

1. <http://www.enactivenetwork.org>

2. <http://www.aim-at-shape.net>

- [HSU 92] HSU W. M., HUGHES J. F., KAUFMAN H., « Direct manipulation of free-form deformations », *Dans les Proceedings of SIGGRAPH '92*, ACM Press, 1992, p. 177–184.
- [Imm a] IMMERSION CORPORATION™, « Haptic Workstation™ description, http://www.immersion.com/3d/products/haptic_workstation.php ».
- [Imm b] IMMERSION CORPORATION™, « Virtual Hand SDK datasheet, http://www.immersion.com/3d/products/virtualhand_sdk.php ».
- [KUO 03] KUO-CHENG W., FERNANDO T., TAWFIK H., « FreeSculptor : a computer-aided freeform design environment », *Dans les Proceedings of the 2003 International Conference on Geometric Modeling and Graphics*, Juillet 2003, p. 188–194.
- [LAW 97] LAWSON B., *How Designers Think : The Design Process Demystified*, 3rd ed., Butterworth-Heinemann, Oxford., 1997.
- [LIP 98] LIPSON H., SHPITALNI M., *Product Development and CAD/CAM*, Verlag, Berlin, F.L. Krause, 1998.
- [LOR 87] LORENSEN W. E., CLINE H. E., « Marching cubes : A high resolution 3D surface construction algorithm », *Dans les Proceedings of SIGGRAPH '87*, ACM Press, 1987, p. 163–169.
- [MAG 97] MAGNENAT-THALMANN N., PANDZIC I. S., MOUSSALY J.-C., « The Making of the Terra-Cotta Xian Soldiers », *In proceedings of Digital'97, Digital Creativity*, Genève, Suisse, juillet 1997, p. 66–73.
- [MAG 04] MAGNENAT-THALMANN N., THALMANN D., *Handbook of Virtual Humans*, John Wiley and Sons, 2004.
- [MCD 02] MCDONNELL K. T., QIN H., « Virtual Clay : Haptics-based Deformable Solids of Arbitrary Topology », *Dans les Proceedings of the Second International Workshop on Articulated Motion Deformable Objects, Lecture Notes in Computer Science*, 2002.
- [OTT 05] OTT R., GUTIERREZ M., THALMANN D., VEXO F., « Improving User Comfort in Haptic Virtual Environments through Gravity Compensation », *Dans les Proceedings of WORLD HAPTICS'05*, 2005.
- [PER 01] PERRY R. N., FRISKEN S. F., « Kizamu : a system for sculpting digital characters », *Dans les Proceedings of SIGGRAPH '01*, New York, USA, 2001, ACM Press, p. 47–56.
- [SAL 97] SALISBURY J. K., SRINIVASAN M. A., « Phantom-Based Haptic Interaction with Virtual Objects », *IEEE Comput. Graph. Appl.*, vol. 17, n° 5, 1997, p. 6–10, IEEE Computer Society Press.
- [SEN] SENSABLE®, « SensAble Technologies Inc., FreeForm™ modeling system, <http://www.sensable.com/products/3ddesign/freeform/hardware.asp> ».
- [SHE 95] SHEN J., THALMANN D., « Interactive Shape Design Using Metaballs and Splines », *Proceedings of Implicit Surfaces '95, Grenoble, France*, Grenoble, France, 1995, p. 187–196.
- [THA 02] THALMANN D., « Simulating a Human Society : the Challenges », *Dans les Proceedings of Computer Graphics International, CGI'02*, 2002.
- [THA 03] THALMANN D., VEXO F., « MPEG-4 Character Animation. », *KI*, vol. 17, n° 4, 2003.
- [TOV 02] TOVEY M., PORTER S., NEWMAN R., « Investigating automotive sketches as a source of geometry for 3D models », *Dans les Proceedings of the Tools and Methods of Competitive Engineering 2002 Symposium, Wuhan, Hubei, Chine*, 2002.