



**MIMESIS, un environnement de conception et de
simulation de modèles physiques particuliers
masses-interactions CORDIS-ANIMA pour l'animation :
du mouvement généré à l'image du mouvement**

Matthieu Evrard

► **To cite this version:**

Matthieu Evrard. MIMESIS, un environnement de conception et de simulation de modèles physiques particuliers masses-interactions CORDIS-ANIMA pour l'animation : du mouvement généré à l'image du mouvement. Synthèse d'image et réalité virtuelle [cs.GR]. Institut National Polytechnique de Grenoble, 2009. Français. <tel-01199192>

HAL Id: tel-01199192

<https://hal.archives-ouvertes.fr/tel-01199192>

Submitted on 15 Sep 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque

|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

THESE

pour obtenir le grade de

DOCTEUR DE L'Institut polytechnique de Grenoble

Spécialité : « *Ingénierie de l'interaction, de la cognition, de l'apprentissage et de la création* »

préparée au laboratoire « Informatique et Création Artistique » de l'Ecole Nationale Supérieure d'Informatique et de Mathématiques de Grenoble, et au sein de l'Association pour la Création et la Recherche d'Outils d'Expression
dans le cadre de **l'Ecole Doctorale** « *Ingénieries de la santé, de la cognition et environnement* »

présentée et soutenue publiquement

par

Matthieu Evrard

le 25 septembre 2009

TITRE

MIMESIS, un environnement de conception et de simulation de modèles physiques particuliers masses – interactions CORDIS-ANIMA pour l'animation : du mouvement généré à l'image du mouvement

DIRECTEUR DE THESE

Claude Cadoz

CO-DIRECTRICE DE THESE

Annie Luciani

JURY

Mme. Marie-Paule Cani
M. Jean-Pierre Jessel
M. Marc Neveu
M. Daniel Barthélémy
M. Claude Cadoz
Mme. Annie Luciani

, Présidente
, Rapporteur
, Rapporteur
, Examinateur
, Directeur de thèse
, Co-directrice de thèse

Statut du manuscrit de thèse au regard de la propriété intellectuelle

Du point de vue de l'édition, un manuscrit de thèse entre dans la catégorie de la littérature grise qui selon l'AFNOR regroupe les « documents dactylographiés ou imprimés, produits à l'intention d'un public restreint, en dehors des circuits commerciaux de l'édition et de la diffusion et en marge des dispositifs de contrôle bibliographiques ». Or cette définition ne prend pas en compte la situation nouvelle apportée par la numérisation et la mise en ligne publique des manuscrits de thèse sur Internet. Ce flou actuel nous oblige à mettre en garde le lecteur sur la question de la propriété intellectuelle du présent manuscrit, question qui n'était pas si pesante lorsque les manuscrits de thèse restaient dans un cadre de diffusion relativement restreint. Nous rappelons donc ici qu'un manuscrit de thèse ne devrait pas faire l'objet de citations dans le cadre de publications normalisées notamment de publications scientifiques. En outre, ne constituant pas une publication normalisée, tout ou partie de ce document pourront faire l'objet de publications futures dont il nous faut préciser les droits d'utilisation.

Ce document a été rédigé pour servir de support à l'obtention du grade de docteur. Ce sont les travaux qui y sont rapportés et ma participation à ces travaux qui justifient l'obtention de ce grade et non le manuscrit en lui-même. Il serait inconvenant que par extrapolation, ce texte puisse être considéré comme un ouvrage dont je puisse être le seul auteur, ôtant du même coup aux autres participants l'usage de leurs apports. Il s'agit en effet d'une production écrite polyphonique dont j'ai été le maître d'ouvrage, qui rend compte d'un travail d'équipe, et qui a fait l'objet de nombreux rapports et publications intermédiaires, ainsi que de corrections, de réajustements, de réécritures, parfois même d'insertion d'écrits d'autres membres de l'équipe. Ce fait ne constitue pas une exception, il en est de même pour une grande majorité des écrits de thèse fondés sur des travaux collaboratifs.

En conséquence, les publications ou usages ultérieurs, qui pourraient se baser sur ou reprendre tout ou partie de ce travail, se devront de respecter les participations de chacun, présentes et passées, comme ce fut d'ailleurs le cas dans les publications qui ont eu lieu durant la thèse. Il nous semble naturel de considérer que les personnes les mieux placées pour se porter garantes de la bonne gestion de la propriété intellectuelle dans les publications ou usages futurs de ces travaux sont naturellement Annie Luciani, ma directrice de thèse et responsable de l'équipe dans laquelle se sont effectués ces travaux, et moi-même, dans la mesure où nous connaissons au plus près la participation en nature et en quantité de tous.

Table des matières

Liste des figures	9
Remerciements	13
AVANT-PROPOS	15
- CHAPITRE 1 - DES MODELES PHYSIQUES CORDIS-ANIMA, DE LEURS SPECIFICITES EN ANIMATION, ET DE LA NECESSITE D'UN ENVIRONNEMENT INTERACTIF DE CONCEPTION ET DE SIMULATION POUR CES MODELES	21
I. Nouvelle pratique de l'animation : de la description explicite du mouvement à sa génération	23
I.1. Brève histoire de l'animation : du théâtre d'ombres à l'ordinateur	23
I.2. Modèles avec description explicite du mouvement	25
I.3. Modèles générateurs	26
II. Modèles physiques pour l'animation : critères de comparaison	29
II.1. Généricité	29
II.2 Contrôle du mouvement	33
III. CORDIS-ANIMA	36
III.1. Principes fondateurs	36
III.2. Un formalisme modulaire	36
III.3. Une base de paramètres compréhensible	38
III.4. Un formalisme orienté temps-réel	38
III.5. Un formalisme pour la multi-sensorialité	39
IV. La nécessité d'un environnement de conception interactif	41
IV.1 Etat des lieux des réseaux masses - interactions dans les systèmes d'animation	41
IV.2 Un environnement de conception interactif de modèles masses - interactions n'existe pas encore	44
V. Bilan.....	47
- CHAPITRE 2 - LES FONCTIONNALITES D'UN ENVIRONNEMENT DE CONCEPTION INTERACTIF ET DE SIMULATION DE MODELES CORDIS-ANIMA.....	49
I. Différentes phases dans la conception d'un modèle CORDIS-ANIMA	51
II. Phase « PSQL » : Conception de la topologie du réseau CORDIS-ANIMA	55
II.1. La complexité topologique dans les réseaux CORDIS-ANIMA	55

II.2. Conception textuelle	56
II.3. Conception graphique interactive	60
II.4. Manipulation de l'organisation du réseau	64
III. Phase « PSQN » : Edition des paramètres physiques	69
III.1. Une phase essentielle dans la constitution du modèle	69
III.2. Les différents type de paramètres physiques	70
III.3. Mise en œuvre de procédures d'édition des paramètres physiques	72
IV. Édition des variables d'état initiales	75
IV.1. Variables extensives et variables intensives.....	75
IV.2. Les trois niveaux dans l'édition des positions initiales.....	76
IV.3. Similitudes et différences avec un logiciel de modélisation de formes.....	77
IV.4. Fonctionnalités pour l'organisation grossière des positions initiales	78
IV.5. Placement à partir de données provenant d'un environnement de modélisation de formes .	80
IV.6. Le placement par simulation physique	81
IV.7. Les vitesses initiales.....	81
V. Simuler un modèle CORDIS-ANIMA	83
V.1. Le théorème de Shanon et les sorties du processus de simulation physique dans MIMESIS	83
V.2. L'utilisation par un système aval du signal calculé :	84
V.3. La discrétisation des équations continues et les problèmes liés à la divergence.....	85
V.4. Le temps réel et le temps non-réel.....	85
V.5. Recommandations	86
VI. Visualisation des variables d'état de la simulation.....	89
VI.1. Visualiser le résultat de la simulation pour mieux le comprendre et le contrôler	89
VI.2. Visualisation de la simulation différente de la représentation dans la fenêtre de modélisation	90
VI.3. Plusieurs visualisations pour un même mouvement	91
VII. Bilan.....	93
 - CHAPITRE 3 - INTERACTIONS NON LINEAIRES DANS UN ENVIRONNEMENT DE CONCEPTION INTERACTIF.....	 95
I. Des interactions non linéaires pour augmenter la complexité des comportements dynamiques générés	97
I.1. Une condition sine qua non pour l'obtention de comportements complexes en animation.....	97
I.2. S'affranchir partiellement de la pratique des sciences physiques	98
I.3. Un corpus d'interactions aux origines diverses.....	100
II. Typologie des interactions non linéaires pour les modèles masses - interactions	101
II.1. Quelle typologie et selon quels critères ?.....	101
II.2. Interactions non linéaires potentielles	102
II.3. Interactions non linéaires dissipatives	107
II.4. Modification de paramètres en fonction des variables de la simulation	111
III. Modalités de représentation au sein d'un environnement de conception interactive	113
III.1. Un outil de conception et de représentation générique pour les interactions ?	113

III.2. Interactions non-linéaires potentielles	114
III.3. Interactions non linéaires visqueuses	121
III.4. Manipulation graphique des paramètres	126
III.5. Mise en œuvre de la modification de paramètres en fonction des variables de la simulation	131
IV. Bilan.....	133
- CHAPITRE 4 - INSERER MIMESIS DANS UNE CHAINE DE PRODUCTION GLOBALE D'ANIMATION.....	135
I. Du mouvement du point au mouvement de la forme	137
I.1. L'habillage du modèle physique particulière : d'une topologie physique vers une topologie spatiale.....	137
I.2. Les <i>mappings</i> possibles entre le modèle du mouvement et le modèle de la forme	140
I.3. Logiciels de synthèse et d'animation de formes utilisés comme moyen d'habillage	149
I.4. Vers de nouvelles pratiques de l'habillage	153
II. Communication entre systèmes par signaux de geste et de mouvement	155
II.1. Définitions.....	156
II.2. Caractéristiques des signaux de geste et de mouvement.....	159
II.3. Un format de bas niveau, générique et minimal n'existe pas encore	163
II.4. Format proposé : le format TM GMS.....	167
II.5. Les sorties des réseaux CORDIS-ANIMA sont des signaux de geste et de mouvement : les modules enregistreurs.....	171
II.6. Signaux de geste de position en entrée des réseaux CORDIS-ANIMA.....	173
III. Bilan.....	177
- CHAPITRE 5 - REALISATIONS ET VALIDATIONS.....	179
I. Objectifs.....	181
II. La version actuelle de MIMESIS.....	183
II.1. Le modeleur	183
II.2. L'interface graphique	185
II.3. Le simulateur.....	186
III. La bibliothèque de modèles.....	189
III.1. Pourquoi une bibliothèque ?	189
III.2. Réimplantation des modèles CORDIS-ANIMA du passé.....	190
III.3. Nouveaux modèles validant l'introduction d'interactions non linéaires dans MIMESIS ...	193
III.4. Validation de l'outil par des modèles créés par d'autres usagers.....	206
IV. Utilisations en situation pédagogique	215
IV.1. Principes de la pédagogie autour de MIMESIS	215
IV.2. Mise en situation	216
CONCLUSIONS ET PERSPECTIVES	221
Bibliographie	227

Liste des figures

Figure 1 : Extraits du film « Modèles physiques et animation » [Luc90]	27
Figure 2 : Schéma du module MAT.	37
Figure 3 : Schéma du module LIA	37
Figure 4 : Un réseau CORDIS-ANIMA simple	38
Figure 5 : Premières versions d'une interface interactive pour le simulateur ANIMA.....	57
Figure 6 : construction du réseau de type agglomérat pour un modèle de fumée	59
Figure 7 : Modèle de marionnette dans la fenêtre 3D de modélisation.	63
Figure 8 : Structuration possible d'un modèle CORDIS-ANIMA de boules suspendues	65
Figure 9 : Structuration possible d'un modèle CORDIS-ANIMA de boules suspendues construit par duplication de réseau.	66
Figure 10 : Graphe de structuration d'un modèle CORDIS-ANIMA de boules suspendues.....	67
Figure 11 : Liste de labels possibles pour certains modules et sous-réseaux.	68
Figure 12 : Placement automatique selon un rectangle.	79
Figure 13 : Outils de placement par mapping de positions provenant d'un fichier.....	80
Figure 14 : Habillage sommaire d'un modèle CORDIS-ANIMA	91
Figure 15 : Modèle du double pendule.....	91
Figure 16 : Habillage d'un module de double pendule par une ligne brisée symbolisant le lien de cohésion entre les masses.	92
Figure 17 : Habillage d'un module de double pendule par deux traits reliant les masses en interaction de butée.	92
Figure 18 : Fonction d'interaction de Lennard-Jones pour $n=12$ et $m=6$	104
Figure 19 : Interactions de cohésion irréversible.	105
Figure 20 : Comportement d'un matériau subissant une déformation plastique.....	106
Figure 21 : Modèle élasto-plastique parfait et élasto-plastique avec écrouissage.....	106
Figure 22 : Interaction non linéaire visqueuse par morceaux intervenant dans la locomotion.	108
Figure 23 : Composante d'accrochage de l'interaction de frottement sec.	110
Figure 24 : Linéarisation par morceaux de l'interaction cohésive de Lennard-Jones.....	115
Figure 25 : Interaction de répulsion en $1/r^2$ et linéarisations par morceaux possibles.	115
Figure 26 : Le choix des paramètres pour des interactions simples.	116
Figure 27 : L'interaction de cohésion et ses paramètres K_1 , K_2 , S et L	117
Figure 28 : Interaction de répulsion à 3 zones et sa base de paramètres K_1 , K_2 , S_1 et S	118
Figure 29 : Attractions et répulsions à 3 et 4 paliers dans MIMESIS.	119
Figure 30 : Représentation fonctionnelle de l'interaction de cohésion irréversible linéaire par morceaux avec sa base de paramètres.....	120
Figure 31 : Schéma de l'algorithme du module élasto-plastique.....	121
Figure 32 : 5 combinaisons possibles d'interaction non linéaires viscoélastiques simples pour l'animation.....	122
Figure 33 : Représentation fonctionnelle des interactions d'attraction et de cohésion avec adjonction de viscosité non linéaire.	123

Figure 34 : Schéma de l'interaction visqueuse linéaire par morceaux quelconque avec seuils en vitesse.....	123
Figure 35 : Construction du vecteur $\overrightarrow{T(N)}$ tangent au mouvement relatif.	124
Figure 36 : Schéma de l'algorithme du module FROS.	126
Figure 37 : Modification de raideur de la part de l'utilisateur conduisant à un changement de catégorie de comportement pour l'interaction ATR3.....	127
Figure 38 : Manipulation graphique des paramètres de raideur de l'interaction COH3.	128
Figure 39 : Manipulation graphique des paramètres de raideur de l'interaction REP3.	129
Figure 40 : Manipulation graphique du paramètre L de l'interaction COH3.....	129
Figure 41 : Manipulation graphique des seuils de distance de l'interaction REP3.....	130
Figure 42 : Représentation fonctionnelle de l'interaction LLM.....	131
Figure 43 : Editeur d'interactions linéaires par morceaux quelconques.	131
Figure 44 : Module de modification paramétrique en ligne.....	132
Figure 45 : Deux types d'habillage pour un même modèle particulière.	138
Figure 46 : Chronogrammes avec deux habillages différents pour la même simulation.....	139
Figure 47 : Clignotement de l'habillage du modèle « boules suspendues » par une triangulation de Delauney.....	141
Figure 48 : Définition d'un point intermédiaire par une relation géométrique.....	142
Figure 49 : Le mouvement généré peut imposer des contraintes géométriques sur l'habillage d'objets rigides multiples en interaction.....	143
Figure 50 : Visualisation de la trace d'une simulation masses - interactions de mouvement de foule.....	144
Figure 51 : Du modèle de mouvement simulé par un réseau masses - interactions à son habillage à l'aide de l'écran d'épingles.....	147
Figure 52 : Interactions entre épingles voisines en 1D.	148
Figure 53 : Schéma de présentation générale du fonctionnement de l'écran d'épingles.	148
Figure 54 : L'animation dans Blender.....	151
Figure 55 : Fenêtre de mapping dans MAYA implémentée dans le langage de script MEL.....	152
Figure 56 : contrôle de l'évolution d'une forme dans Blender.	153
Figure 57 : Signaux mis en jeu en entrée et en sortie d'une simulation physique.	158
Figure 58 : Signaux mis en jeu dans une situation d'interaction avec une simulation par l'intermédiaire d'un transducteur gestuel rétroactif.....	158
Figure 59 : Situations variées où les gestes et les mouvements interviennent dans des dimensionnalités différentes.....	160
Figure 60 : Comparaison de la bande passante des signaux de geste et de mouvement avec d'autres types de signaux.	162
Figure 61 : Situation gestuelle multidimensionnelle complexe	169
Figure 62 : Implantation du format TM GMS.	170
Figure 63 : Connexion d'un enregistreur de position (EP) à un module MAT.....	171
Figure 64 : Découpage temporel d'une simulation.....	174
Figure 65 : Module générateur de position dans un réseau CORDIS-ANIMA.	175
Figure 66 : Différents habillages dans MIMESIS pour une même configuration spatiale de points.....	184
Figure 67 : Aperçu de l'environnement graphique actuel de MIMESIS.....	185
Figure 68 : Architecture globale de la chaîne simulation-affichage-sauvegarde dans MIMESIS.	188

Figure 69 : La roulette et la boule, solides en mouvement.	190
Figure 70 : La chaîne.	191
Figure 71 : Le drapeau.	191
Figure 72 : Comportements intermédiaires de la matière.	192
Figure 73 : Croisement entre deux groupes d'individus dans un couloir étroit.	192
Figure 74 : Le modèle « Libération ».	194
Figure 75 : Interaction de cohésion sans libération.	194
Figure 76 : Interaction de cohésion avec libération.	195
Figure 77 : Auto-organisation de 100 masses dans un espace pavé.	196
Figure 78 : Topologie du réseau masses - interactions du modèle boule contre mur.	196
Figure 79 : Autre représentation de la topologie du réseau masses - interactions du modèle boule contre mur.	197
Figure 80 : Simulation d'une fracture par le passage d'une masse ponctuelle en interaction avec de masses ponctuelles formant un mur cohésif capable de se fracturer.	197
Figure 81 : Anticipation lors d'un croisement de deux sujets.	198
Figure 82 : Modèle du personnage avec masse ponctuelle visqueuse pour éviter les rotations brusques.	198
Figure 83 : Modélisation de la situation de croisement avec anticipation.	199
Figure 84 : Simulation du phénomène d'anticipation lors du croisement de deux sujets.	200
Figure 85 : Modèle 2D de véhicule sur sol meuble.	201
Figure 86 : Simulation d'un véhicule sur sol meuble.	202
Figure 87 : Modèle du pavé posé sur un plan inclinable.	203
Figure 88 : Simulation de l'inclinaison d'un plan jusqu'à l'inclinaison limite permettant à une masse ponctuelle de commencer à glisser.	203
Figure 89 : Modèle de roue en interaction de frottement sec.	204
Figure 90 : Simulation d'une situation d'accrochage.	205
Figure 91 : Forme caractéristique du phénomène de striction	206
Figure 92 : Les phases successives du phénomène modélisé.	207
Figure 93 : Modification du paramètre K1 de l'interaction de cohésion.	207
Figure 94 : Modèle de striction habillé par des surfaces implicites et rendus à l'aide de Povray [AEL07]	208
Figure 95 : Les trois temps de l'interface « Ecran d'épingles ».	209
Figure 96 : Modèle de fumée habillé à l'aide du logiciel « Ecran d'épingles ».	210
Figure 97 : Travaux de modélisation de Marouane Cherkaoui.	211
Figure 98 : Modèle de structure frissonnante utilisant des signaux TMGMS.	211
Figure 99 : Extraits de fluente réalisé par Chimin Hsieh et Annie Luciani.	213
Figure 100 : Modèle de pendule de torsion avec fort amortissement.	217
Figure 101 : Modèle de proie prise dans une toile d'araignée.	218

Remerciements

Je tiens à remercier en tout premier lieu Annie Luciani, administrativement ma co-directrice de thèse, mais dans les faits la véritable directrice ces travaux. Je la remercie de m'avoir fait l'honneur de m'avoir ouvert à une approche de l'art du mouvement qu'elle défend depuis près de 30 ans et de m'avoir permis de collaborer à sa vision. Je ne saurais assez dire combien ces années de travail avec Annie ont changé ma conception des sciences, et de leur rapport à la société. Cette thèse lui doit presque tout. Je lui dois, quant à moi, de sortir de cette période différent du jour où j'y suis entré.

Je tiens à remercier ensuite Claude Cadoz, mon directeur de thèse, qui, s'il a joué un rôle administratif et a suivi de plus loin ces travaux, m'a toujours soutenu. Il est certainement aussi responsable, par son souci du partage des savoir et savoir-faire, autant qu'Annie en ce domaine, mais avec un style qui lui est propre, de mon envie grandissante de diffuser par la pédagogie la pratique de l'animation défendue dans cette thèse.

Je souhaite remercier chaleureusement Nicolas Castagné qui a supervisé, pendant les premiers temps de ces travaux, le développement de MIMESIS. Nicolas m'a certainement plus appris en matière de génie logiciel que mes années d'étude en école d'ingénieur. Je le remercie pour m'avoir sollicité pour les questions de conception alors que j'avais rarement des solutions à apporter aussi bonnes que les siennes. Nicolas m'a tout simplement appris un métier.

Je remercie Ali Allaoui et Kevin Sillam que j'ai co-encadrés avec Annie Luciani pendant leur stage de Master II pour m'avoir fait confiance malgré ma jeune expérience et pour avoir poursuivi l'aventure en thèse en constituant cette nouvelle « équipe image » au sein de l'ACROE-ICA. Certains des résultats de cette thèse leur sont dus.

Je remercie tous les utilisateurs de MIMESIS durant ces cinq années, plus particulièrement Daniel Barthélémy. Plus qu'un utilisateur, Daniel en est un défenseur plus ancien que moi des problématiques de l'ACROE-ICA. Il a su créer en moi plus d'assurance pour ce sujet grâce à son engouement. Je remercie aussi tous les étudiants du Master « Art, Science, Technologie » qui ont participé aux TP MIMESIS et plus particulièrement ceux qui ont continué en stage de Master. Merci donc à Maria Christou et à Marouane Cherkaoui d'avoir été des utilisateurs talentueux du logiciel, et d'être venus augmenter la bibliothèque des modèles. Je remercie énormément Chimin Hsieh, pour avoir été l'utilisateur de MIMESIS pendant ses 3 années de thèse et d'avoir engagé des discussions sur l'environnement et ses fonctionnalités à la lumière de problématiques de recherche et de création passionnantes. Le succès de ses travaux m'apporte aussi des résultats concluants.

Je remercie Geoffrey Cerna pour avoir été la mémoire technique de MIMESIS et avoir fait la transition entre le développement des versions passées et celui des versions réalisées pendant mes années à l'ACROE-ICA.

Je remercie Aurélie Arliaud pour avoir rédigé avec rigueur une grande partie de la documentation de MIMESIS, pour avoir mis en place des archives qui ont permis à tout le laboratoire de pouvoir

facilement accéder aux travaux du passé et pour avoir installé le studio image sans lequel le reste de la chaîne de production d'images animées par MIMESIS ne pourrait exister.

Je remercie Martin Mahl pour avoir entrepris avec Annie Luciani, Nicolas Castagné et moi-même le portage de MIMESIS sous PC-LINUX et initier ainsi un nouveau cycle dans la vie de cet environnement.

Je remercie tous les membres de l'ACROE-ICA qui ont partagé mon quotidien, ceux qui sont encore là et ceux qui ont quitté le laboratoire. Parmi eux, je remercie particulièrement Jean-Loup Florens pour ses discussions d'un haut niveau quel que soit le sujet, ses conseils et ses aides techniques, et pour susciter en moi le regret d'ignorer encore de multiples domaines scientifiques et techniques. Je remercie aussi Maria Guglielmi pour porter haut le métier d'administratif, pour sa bonne humeur et aussi ses coups de gueules aux moments opportuns. Je remercie tous ceux qui furent doctorants en même temps que je l'ai été. Qu'ils aient commencé avant ou après moi, qu'ils aient fini ou non, nous avons partagé le même radeau à un moment ou à un autre. Merci à Olivier Tache, Julien Castet, François Poyer, Sylvain Marlière, Damien Couroussé, Diani Herrera.

Je remercie Swan, Max 2 et Iann, les membres de Blackrain, mon groupe de rock avec lequel j'ai vu beaucoup de pays lors de nos concerts et qui m'a empêché de m'enfermer dans les seules études. Ce groupe a nourri et continuera à nourrir ma vie.

Je remercie Edwige Dessemond pour avoir partagé ma vie pendant cinq années même si les derniers mètres nous ont séparés.

Il va sans dire que je remercie ma famille pour son soutien. Je pense particulièrement à mes parents qui ont su respecter mon choix de privilégier la recherche à l'argent et de faire une thèse après mon diplôme d'ingénieur. Issu d'un milieu populaire, je sais la fierté que cette thèse leur procure tout autant que les doutes que ces années ont pu provoquer.

Je dédie ce manuscrit à deux de mes grands-parents, décédés aujourd'hui. Odette Gerdil-Margueron ma grand-mère maternelle, femme pieuse, ayant subi les affres du déclassement social, qui s'est battue seule pour élever sa fille pendant une grande partie de sa vie et dont cette thèse aurait fait le bonheur le plus parfait, étant elle-même la fille d'une scientifique astronome. Georges Evrard, mon grand-père paternel, immigré polonais, atterri dans les bidonvilles de Paris, devenu mécanicien au sortir de la guerre et dont je tiens certainement la méfiance vis-à-vis de ce qu'il appelait les intellectuels, même si j'en ai toujours été un à ses yeux.

Avant-propos

Les travaux de cette thèse portent sur MIMESIS, un environnement de conception interactif et de simulation de réseaux CORDIS-ANIMA pour l'animation. Ces travaux s'inscrivent dans le programme scientifique du groupe ACROE-ICA créé il y a plus de trente ans par Annie Luciani, Claude Cadoz et Jean-Loup Florens. Ce groupe a pour vocation de concevoir, de développer, d'expérimenter et de diffuser des outils d'expression pour le son, l'image, le geste et plus généralement pour les arts instrumentaux du temps dans le contexte des technologies de l'information et de la communication.

Dans le domaine de l'animation, les travaux d'Annie Luciani, initiés il y a près de 30 ans, ont cette originalité d'être tous fondés sur le formalisme des réseaux CORDIS-ANIMA mettant en jeu des systèmes physiques de masses en interaction. Ce formalisme conçu et utilisé au sein de l'ACROE-ICA de manière quasi exclusive, fait de ce groupe un expert effectif de la modélisation physique particulière.

Le milieu des années 90 et les années 2000 ayant vu fleurir des environnements logiciels interactifs dans des domaines aussi divers que la bureautique, le traitement de l'image, le traitement du son, la mécanique, le groupe ACROE-ICA s'est naturellement posé la question du développement d'un tel environnement pour faciliter la conception des réseaux CORDIS-ANIMA. Voici donc la raison initiatrice de ses travaux.

Dans ce contexte, j'ai intégré le groupe en tant qu'ingénieur en 2004, sur financement en partie de l'ACROE, en partie du réseau européen d'excellence Enactive Interfaces et dans le cadre du projet ARCAD en collaboration avec l'école européenne supérieure de l'image de Poitiers/Angoulême.

L'environnement MIMESIS a été développé par une équipe de l'ACROE-ICA au sein de laquelle j'ai pris une part de plus en plus importante avec les années. Je suis plus précisément arrivé à un moment où MIMESIS était au début d'une reconception totale du noyau fonctionnel et d'une partie de l'interface, après avoir déjà connu des versions antérieures, certes peu stables mais ayant permis des premiers tests en situation d'utilisation, notamment lors des journées de Rencontres en Informatique et Création Artistique en 2001 (RICA2001). J'ai effectué initialement ces travaux sous la direction d'Annie Luciani et de Nicolas Castagné, responsable technique, en collaboration avec Geoffrey Cerna et un petit nombre de stagiaires. Au bout d'une année de travail, nous sommes parvenus à une version stable que j'ai prise en main pour en augmenter les fonctionnalités : entrées/sorties de signaux gestuels, interactions non linéaires. J'ai ensuite encadré les développements d'Ali Allaoui sur le contrôle en ligne des paramètres physiques de la simulation, les entrées et le *mapping* de fichiers de positions, les entrées par capteur.

Parallèlement à cela, j'ai participé aux discussions de conception de la refonte du noyau fonctionnel de l'environnement GENESIS, environnement de conception pour la synthèse sonore à l'aide de modèles CORDIS-ANIMA, et plus précisément à la spécification du langage de modélisation. Annie Luciani et Claude Cadoz souhaitent en effet que ce noyau fonctionnel soit le plus possible compatible voire commun à MIMESIS et à GENESIS dans le futur, puisque ces deux environnements sont fondés sur le même formalisme de modélisation et de simulation CORDIS-ANIMA.

Finalement j'ai contribué à superviser le portage de MIMESIS sur des plateformes PC-LINUX afin de ne plus rendre tributaire cet environnement des stations SILICON GRAPHICS sur lesquelles il avait été développé et qui sont aujourd'hui peu utilisées par la communauté de l'informatique graphique au regard de leur succès initial dans les années 90.

J'ai eu l'occasion, finalement et à de multiples reprises, d'accompagner des utilisateurs dans leur pratique de MIMESIS, dans des situations d'initiation à la modélisation mais aussi de recherche et de création.

M'en étant tenu pendant un long moment aux développements de MIMESIS, l'intérêt d'une thèse ne m'est pas apparu immédiatement. Ma motivation première était de travailler au développement des activités du groupe, activités que je trouvais relativement uniques. Depuis 30 ans, le programme scientifique, technique et artistique du groupe était en effet parfaitement cohérent, menait à des développements technologiques conséquents, des savoirs-faire innovants et des productions qui me touchaient. Il ne s'agissait pas seulement de rêver à de nouvelles pratiques de l'art par l'intermédiaire des nouvelles technologies de l'information mais de les rendre possible. Il m'était apparu en outre qu'un travail très conséquent restait à faire dans le cadre de la production d'images animées et que surtout, j'arrivais à l'époque où il était temps de prendre ces questions en main, bien plus que dans le cas de la synthèse sonore par modèle physique qui était déjà relativement installée.

Or, je me suis rendu compte, en me confrontant aux autres travaux dans ce domaine, qu'il n'était pas seulement nécessaire de développer de nouveaux outils pour répondre aux problèmes qui se posaient au sein du groupe ACROE-ICA, mais qu'un travail de clarification de l'intérêt de ces outils était nécessaire pour que les travaux qui étaient réalisés soient mieux diffusés. J'étais en effet surpris du peu de références qui pouvaient leur être fait et je commençais moi-même à me heurter à certaines incompréhensions des travaux de l'ACROE-ICA de la part de certains chercheurs en informatique graphique.

Pourtant, en considérant uniquement les développements de MIMESIS qui avaient amené à l'obtention d'un logiciel stable, mis de plus en plus souvent en situation d'utilisation, il me semblait anormal que l'approche qu'il défendait ne soit pas plus largement connue et comprise.

Il m'a semblé que le cadre d'une thèse était un moyen d'expliquer l'intérêt du développement d'une telle technologie et les nouvelles pratiques que cette technologie était en mesure de créer. Cette thèse dépasse donc mon intérêt propre en espérant ouvrir les esprits à ce qu'Annie et tous les gens qui ont travaillé avec elle, moi y compris, considérons comme des pratiques encore confidentielles mais ouvrant vers un avenir où elles pourraient prendre une plus grande importance. Cette posture devenait une posture de communication sur mes activités et d'explicitation de ses visées. Une technologie est une histoire des activités de l'homme, mais une histoire tronquée, imparfaite, lorsqu'elle ne va pas avec l'explication de ses finalités. Inscrire un développement technologique dans le cadre d'une recherche, c'est donc permettre cette explication en même temps que la technologie se développe et est mise en pratique.

Une majorité des travaux publiés dans la communauté de l'informatique graphique met l'accent sur l'obtention d'un effet particulier, l'animation d'un objet bien défini (une chevelure, un vêtement, etc.). Lorsque ces travaux s'intéressent à l'utilisateur, c'est souvent pour lui fournir des outils qui permettent de contrôler un modèle « clé en main ». Plus rarement, il est considéré que l'utilisateur doit avoir à faire au processus global de modélisation. D'une certaine manière, il est rarement considéré que l'établissement d'un modèle soit une opération de création à la portée d'un animateur ou que la connaissance qu'il devrait acquérir pour être capable de cela passe par des connaissances en sciences qui ne lui sont pas accessibles. La manière dont de nombreux chercheurs voient les pratiques futures

de l'animation par modèle physique est celle d'un duo programmeur-chercheur/animateur au sein duquel le second « consomme » les travaux du premier.

Au contraire, Annie Luciani et son équipe ont plutôt à cœur de fournir à l'animateur, un animateur certes d'un nouveau type, avec de nouvelles connaissances, des outils pour modéliser lui-même des comportements dynamiques. Cela peut être imaginé grâce à la simplicité, la généricité, la modularité et la compréhensibilité des réseaux CORDIS-ANIMA. La thèse soutenue ici est qu'un environnement de conception pour ces réseaux est alors nécessaire pour qu'émerge une réelle communauté de praticiens de ce type de modélisation.

La rédaction de ce manuscrit a donc été soumise à la difficulté de soutenir une approche plutôt non-conventionnelle dans le cadre de l'animation par modélisation physique. En acceptant, et en revendiquant la nécessité de produire une thèse sur une technologie, il a fallu en accepter les codes qui sont les codes de la recherche. Dès lors, il était obligatoire de placer ces travaux par rapport à des travaux du même domaine en explicitant les différences que nous avons pointées puis ensuite expliciter les verrous qui apparaissent en choisissant cette approche.

Ce manuscrit s'appuie sur une technologie existante et utilisée. Nous soutenons l'idée qu'il s'agit là d'une validation de la pertinence de cette thèse et nous rendrons compte de cette réalité en évitant de transformer ce manuscrit en une documentation technique, ce qu'il n'a pas lieu d'être.

Ce manuscrit s'organise donc de la manière suivante. Le premier chapitre s'attache à justifier la thèse défendue qui est la nécessité de l'existence d'un environnement de conception de modèles CORDIS-ANIMA pour l'animation. Le second chapitre présente alors globalement la manière dont un tel environnement doit se structurer, les fonctionnalités, au moins de base, qu'il doit proposer et les verrous auxquelles la mise en œuvre de ces fonctionnalités doit faire face. Le troisième chapitre entre dans le détail des interactions non linéaires et de leur mise en œuvre dans un environnement interactif. Les liaisons non linéaires sont en effet certainement le point fondamental pour l'obtention d'une réelle diversité de comportements dynamiques visibles. Le quatrième chapitre pose le problème de l'insertion du logiciel dans un ensemble d'outils pour la production globale d'images animées. Le cinquième et dernier chapitre rend finalement compte de la mise en œuvre de l'environnement MIMESIS et de son utilisation dans des situations de pédagogie, de recherche et de création.

- Chapitre 1 -

**Des modèles physiques CORDIS-
ANIMA,
de leurs spécificités en animation,
et de la nécessité d'un
environnement interactif de
conception et de simulation pour
ces modèles**

I. Nouvelle pratique de l'animation : de la description explicite du mouvement à sa génération

I.1. Brève histoire de l'animation : du théâtre d'ombres à l'ordinateur

Il faut remonter au VI^{ème} siècle, en Chine, pour retrouver la première trace connue à ce jour d'une pratique mettant en œuvre des images en mouvement : le théâtre d'ombres. Des marionnettes articulées éclairées par une lanterne projettent leur ombre sur une toile pour créer un spectacle de silhouettes animées. L'animation est donc celle que les mouvements du marionnettiste impriment à la marionnette. Or il y a un couplage mécanique, un lien matériel, nous dirons une interaction physique entre les deux corps, celui animé, vivant, du marionnettiste et celui inanimé de la marionnette. Le marionnettiste est celui qui donne vie à ce corps qui sans lui est comme mort.

À partir du XVII^{ème} siècle, les techniques qui permettent d'obtenir des images animées connaissent un développement foisonnant en Europe. De l'ombre, nous passons au dessin, donc à la mise en avant des traits et des couleurs. La projection d'images dessinées devient possible avec l'invention de la Lanterne Magique que l'on attribue au savant allemand Athanase Kircher, invention décrite dans son *Ars Magna Lucis et Umbrae* (1646). Cet appareil, constitué d'une bougie dont la lumière traverse une plaque transparente peinte, projette une image sur une surface extérieure à l'aide d'une lentille convergente. À la même époque, le physicien néerlandais Christiaan Huygens remplace la plaque peinte par une plaque animée représentant les mouvements d'un squelette (1659). À l'aide de petits dispositifs mécaniques commandés par des ficelles, les dessins projetés par la Lanterne Magique s'animent. Le XVIII^{ème} siècle voit alors fleurir nombre de spectacles reprenant en l'améliorant cette technique. On retiendra notamment les fameuses fantasmagories d'Etienne Robertson (1764-1837), spectacles itinérants en France, entre science et charlatanisme, qui projetaient au public des images animées de morts ressuscitant et de fantômes effrayants, souvent accompagnées de musique, de récits et même d'ambiances olfactives. Le corps à corps qui pouvait exister entre le marionnettiste et la marionnette est remplacé ici par un dispositif mécanique. C'est l'esprit du siècle, qui annonce l'industrie, la mécanique, et surtout l'automatique, ce rêve de comprendre le vivant comme une

machine, machine toujours imparfaite, mais qui, lorsqu'elle arrive à nous tromper, nous donner une illusion de vie, nous dit bien quelque chose du vivant. Jusqu'ici, le mouvement des images provient encore d'un mouvement matériel qui en est directement la cause. C'est un mouvement qui se produit, dans le temps de l'image, quelque soit la cause matérielle qui le produit, une intervention humaine ou un mécanisme.

Ce n'est qu'avec la découverte d'un phénomène longtemps appelé persistance rétinienne et attribuée au docteur John Ayrton Paris qu'une première rupture de paradigme est introduite dans la pratique de la création d'images animées. En faisant défiler rapidement deux images présentant deux objets différents, l'observateur a l'illusion que ces deux objets se superposent dans une même image. Paris crée vers 1820-1825, le Thaumatrope, un jouet composé d'un disque accroché à deux ficelles. Les deux faces de ce disque sont illustrées par deux objets différents. En faisant tourner les ficelles entre le pouce et l'index, on obtient une illusion de superposition. Au cours du XIX^{ème} se développent alors de nombreux dispositifs tirant partie de cette découverte, non plus pour superposer deux images en une seule mais pour créer une illusion de mouvement en faisant défiler plusieurs instantanés d'une scène. On peut citer parmi ces dispositifs technologiques pionniers le Phénakistiscope de Joseph Plateau (1831), le Zootrope de William George Horner (1834), le Praxinoscope d'Emile Reynaud (1877), ancêtre de son complexe Théâtre Optique (1889) capable de projeter des scènes animées de plus de 10 minutes. Ce dernier deviendra une attraction à succès du musée Grévin à Paris jusqu'à la perte croissante de l'intérêt du public, intérêt qui se portera plutôt vers le cinématographe des frères Lumière (1895).

Du mouvement continu des plaques animées, nous sommes donc passés à un mouvement discret, découpé en un ensemble d'instantanés, d'images « clés ». Cette illusion de mouvement obtenue par la projection d'images se succédant n'est plus aujourd'hui expliquée par des propriétés de rémanence des capteurs de la rétine de l'œil mais plutôt par une reconstruction directe du mouvement par le cerveau, phénomène appelé effet phi. Il n'en reste pas moins que la découverte de cette illusion donne véritablement naissance à une pratique artistique nouvelle, différente de celle du marionnettiste ou de l'automaticien qui fabrique ses dispositifs. Car les instantanés du mouvement peuvent être produits dans un temps différent de celui du mouvement que verra le spectateur. Un nouveau type d'art est né. Cet art est celui de l'animateur, art multiforme, car les techniques de production peuvent être aussi variées que les techniques picturales. Les images peuvent être dessinées, peintes, gravées. L'animateur doit être capable de retranscrire le mouvement en une suite d'instantanés. Ainsi, si le marionnettiste se cache pendant la scène qu'il anime, si le cinéaste se cache derrière la caméra pendant la scène qu'il filme, l'animateur lui, en quelque sorte, se cache entre les instantanés de la scène. L'animation, jusqu'ici, requiert donc la capacité de rendre le mouvement; mais avec la difficulté de le dessiner, de le retranscrire dans un temps qui n'est pas le sien.

Au cours du XX^{ème} siècle, la production de *films d'animation* connaît un développement industriel conséquent. De grands studios américains tels que Walt Disney (1923) ou Pixar (1986) apparaissent et produisent des films d'animation longs atteignant les niveaux de diffusion des films « classiques ». Il n'est pas rare que des équipes de 200 à 300 personnes soient mobilisées pendant plusieurs années pour créer ces films. En France, au Japon, en Belgique, en Russie, en Chine, des studios se créent aussi et

remportent d'importants succès. Partout des artistes cherchent à inventer des techniques nouvelles. Malgré cela, l'animation reste un art peu reconnu. Il n'apparaît pas dans la classification des six arts classiques et rarement on se donne la peine de lui attribuer une place à côté de ceux-ci comme le cinéma (septième art) et d'autres arts ont su le faire. Mais il possède son public, ses festivals (le festival du film d'animation d'Annecy ou du Zagreb entre autres).

Dans ce contexte, l'apparition de l'outil informatique permet deux choses : les images peuvent être synthétisées et non plus produites à la main ; et l'animation, elle, peut être obtenue par un calcul. Dans le cas de la synthèse de l'image, l'ordinateur permet de mettre à disposition du créateur d'image un ensemble d'outils lui permettant de créer des formes puis de les rendre visible en les coloriant, en y appliquant des textures, des reflets... Des outils de modélisation et de rendu de forme se développent à partir des années 70 et les années 90 voient fleurir des environnements de conception permettant la création et la manipulation de formes spatiales de manière interactive. Parallèlement à cela, dans le cadre de l'animation, l'ordinateur sert aussi à animer ces formes, gérer leur évolution dans le temps et potentiellement produire cette évolution à l'aide de calculs.

I.2. Modèles avec description explicite du mouvement

Au début de l'histoire de l'animation par ordinateur, le concept d'images « clés », comme le pratiquait l'animation « à la main », est conservé. L'animateur fournit une première série d'images « clés », beaucoup moins nombreuses et plus espacées dans le temps que celles de la séquence finale. Lorsque l'animation se faisait intégralement image par image, une équipe de dessinateurs, de « petites mains », les intervallistes, étaient chargés de produire les images intermédiaires. Les travaux de recherche en animation par ordinateur tentent pour leur part de remplacer le travail des intervallistes en calculant ces images intermédiaires par des procédés automatiques comme des interpolations ou des transformations géométriques dans le temps (rotations, translations). On citera ici les travaux pionniers du réalisateur Peter Foldes secondé par Nestor Burtnyk qui développa avec son équipe l'un des premiers systèmes d'animation par ordinateur [BW76]. On peut concevoir que ces pratiques de l'animation soient réductrices par rapport à un véritable art du mouvement. En effet, le travail des intervallistes est un véritable travail d'interprétation des mouvements intermédiaires. Si les techniques ont évolué aujourd'hui en permettant un meilleur contrôle des interpolations et des fonctions d'évolution décrivant le mouvement, si ces techniques sont largement utilisées dans les films d'animation, les intervallistes interviennent toujours aujourd'hui dans le réajustement des séquences pour leur donner un naturel que les techniques automatiques enchaînant les positions clés n'ont pas su produire. Les travaux originaux de Dominique Bechmann et al [BD93] sur les objets 4D apporte quant à eux une vision où l'espace et le temps sont représentés par un même objet. Dominique Bechmann propose une description des objets à animer dans un espace 4D qui incluse le temps comme une dimension similaire aux dimensions de l'espace à travers une topologie volumique. Les animations correspondantes sont obtenues en déformant ces objets « espace-temps ». De cette manière il est possible d'obtenir des changements topologiques au cours du temps, comme des déconnexions, la formation de trous, aussi bien que des modifications « classiques » de géométrie, comme les translations ou les rotations.

I.3. Modèles générateurs

Lorsque nous disons des méthodes de description du mouvement qu'elles sont explicites, c'est que le temps est, pour ces méthodes, une donnée explicite. L'ordinateur n'est utilisé que pour remplacer une activité où la description de chaque position est une fonction du temps. Ces modèles, qui peuvent être nommés « modèles descriptifs du mouvement » comme le propose Luciani en 1984 dans [LC84] consistent à définir un mouvement par une méthode externe à l'objet animé. Au contraire de ces approches, des chercheurs ont opté pour une autre démarche visant à modéliser des systèmes dynamiques capables de produire par eux-mêmes des mouvements. Ces modèles, que Luciani propose de nommer « générateurs de mouvement » [LC84], sont constitués d'entités dont le comportement dans le temps n'est pas décrit de manière explicite mais de manière implicite. Suivant cette approche, l'animateur construit un système qui va ensuite évoluer au cours d'une simulation, cette dernière pouvant être sous contrôle de données temporelles externes et produire en sortie les mouvements visuels. Ce paradigme d'animation rompt de manière assez radicale avec le paradigme des modèles descriptifs de mouvement. Cette pratique de l'animation est totalement différente dans le sens où l'animateur devient un modélisateur du mouvement. Il lui est nécessaire de s'interroger sur les causes capables de produire – de générer – le mouvement qu'il souhaite réaliser, sur ce qui peut mettre en mouvement la scène animée qu'il désire obtenir. Il ne travaille plus sur les fonctions d'évolution du mouvement, sur le signal du mouvement, mais il crée un système capable de générer ces fonctions, ce signal.

Les principaux formalismes de génération du mouvement ont vu le jour à peu près simultanément au début des années 80 en particulier par les travaux de Reynolds, de Reeves, de Luciani ou de Sims.

Reynolds [Rey82, Rey 87] introduit très tôt le concept d'entités appelées agents ou acteurs capables de réagir en fonction de données provenant de l'environnement suivant un comportement intrinsèque décrit par l'utilisateur dans un langage procédural. Ces agents suivent en outre des règles d'interaction mutuelle (éviter un autre agent par exemple). La somme de ces agents autonomes crée des mouvements émergents qui ne sont pas décrits explicitement par l'animateur. On peut retrouver de tels systèmes aujourd'hui dans des logiciels comme MASSIVE ayant servi à l'animation des foules dans la trilogie de Peter Jackson *Le Seigneur des anneaux*. Les systèmes multi-agents sont généralement utilisés pour la synthèse de comportements sociaux et sortent rarement de ce cadre. Ils n'ont pas trouvé d'applications par exemple dans la génération de phénomènes naturels. Les systèmes de particules de Reeves [Ree83] n'entrent pas initialement dans la catégorie des formalismes de génération de mouvement. Les particules de Reeves servent à modéliser des objets aux contours flous (nuages, feu, explosion). Ces particules sont soumises à des champs dont l'évolution temporelle est explicite et dont la nature dépend du comportement à obtenir. Elles furent enrichies par la suite de propriétés physiques (inerties) qui leur permirent d'être soumises à des forces et d'augmenter les possibilités d'animation comme on peut, par exemple, le trouver dans [Sim90]. Ces réalisations spectaculaires sont à imputer au développement des machines massivement parallèles qui permirent de gérer un très grand nombre de particules, mais les systèmes de particules n'ont pas apporté de solution générale au problème de la génération de mouvements complexes.

A la même époque, Luciani introduit, dans le cadre de l'animation, le modèle physique particulaire et le contrôle gestuel temps réel à retour d'effort [LC84, LC86]. Sont ainsi produits les premiers modèles physiques particuliers de marionnettes ou de surfaces déformables manipulées en temps réel par des sticks, avec ou sans retour d'effort. Ces travaux obtiennent en 1990 le premier prix du Video Show d'Eurographics à Montreux avec le film « Modèles physiques et animation »

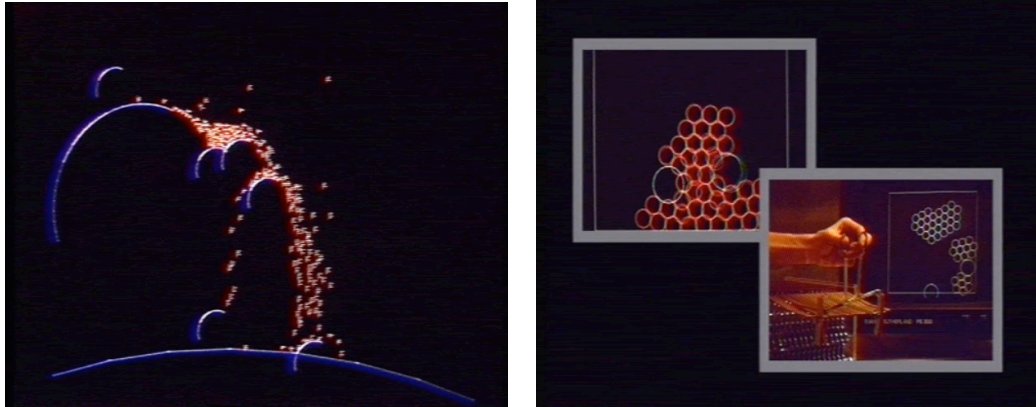


Figure 1 : Extraits du film « Modèles physiques et animation » [Luc90]

Ces travaux s'inscrivent dans une phase de développement des modèles physiques pour l'animation, à la fin des années 80 et le début des années 90, avec deux postures principales, marquant le début d'une grande diversification des modèles pour l'animation. La première posture veut qu'un type de modèle souvent issu des sciences physiques serve à produire un type de comportement dynamique. Les formalismes peuvent alors être divers pour réaliser ce but. Nous citerons comme travaux pionniers ceux de Terzopoulos et al. dans [TF88]. Ils constituent tout particulièrement des travaux pionniers des méthodes de résolution numérique des équations de la mécanique dans le cadre de l'animation. À l'inverse de cette posture, Miller et Pearce [MP89], Terzopoulos et al. [TPF89], Tonnesen [Ton91] ou encore Luciani et al. [LJF*91] qui développèrent le langage de modélisation et de simulation CORDIS-ANIMA sur lequel sont bâtis nos travaux de thèse, mirent en lumière le fait qu'une très grande variété de comportements pouvait être générée à l'aide d'un formalisme basé sur l'idée de masses ponctuelles en interaction. En effet, ce genre de formalisme a permis de modéliser avec le même type de savoir-faire une très vaste plage de comportements dynamiques : des comportements rigides ou déformables aux fractures et plasticité [LJR*91] en passant par des comportements collectifs comme ceux observés dans les mouvements de foules [HLT*03] ou encore des effets de turbulence [LHV*95].

Les modèles générateurs ont permis de produire des mouvements complexes qu'un animateur, même chevronné, ne pouvait pas obtenir par les méthodes d'animation par images clés successives. Parmi ces méthodes génératives, les modèles physiques sont reconnus, en outre, pour le naturel des mouvements qu'ils produisent. Il est intéressant de définir dès lors des critères pour comparer les différents modèles physiques existants. Nous traitons cette question dans la partie suivante.

II. Modèles physiques pour l'animation : critères de comparaison

II.1. Généricité

II.1.1. La généricité d'une méthode : avantages et inconvénients

La généricité d'une méthode réfère au fait que cette méthode puisse être utilisée dans différents contextes et en l'état. Dans le cadre de l'animation, on dira qu'une méthode est générique si elle permet d'obtenir un très vaste éventail de comportements dynamiques de natures différentes (solides, fluides, collectifs, etc.) sans que des développements exogènes à la méthode soient nécessaires. Le caractère générique d'une méthode possède de nombreux avantages. Un des plus importants est la créativité que la généricité permet : plus l'éventail des possibles est large et ouvert, plus l'animateur pourra créer de nouveaux comportements dynamiques sans se sentir piégé par un « style » prédéfini. Travailler à partir d'une méthode générique, c'est aussi explorer des terrains vierges et par conséquent créer un savoir-faire qu'il peut être intéressant de transmettre aux autres pour qu'eux-mêmes continuent le travail de recherche relatif à cette pratique. Un second avantage concernant une méthode générique, c'est donc le lien communautaire qu'elle peut créer. Un logiciel de montage vidéo comme Final Cut, par exemple, permet à une communauté de monteurs de se constituer. Cette communauté discutera des pratiques du montage relatives à l'environnement et s'échangera savoir-faire et astuces. En revanche, un outil de choix aléatoire de séquences vidéo permettra un montage très particulier dont le processus de création sera généralement peu discuté. L'utilisation sera plus une sorte de *gadget* ; on parlera rarement de processus de création et on mettra plus l'accent sur la notice d'utilisation. Mais une méthode générique nécessite généralement un temps de maîtrise plus long qu'une méthode spécifique permettant d'obtenir un effet donné. L'animateur devra souvent se confronter au problème de la « page blanche », page d'autant plus blanche que la méthode est nouvelle et inexplorée et que les savoir-faire ne sont pas encore construits. L'appel à l'enseignement et à la pédagogie est donc essentiel lorsqu'on doit se confronter à des méthodes génériques.

Au début de l'histoire de l'animation par modèles physiques, de nombreux travaux comme ceux de Miller et Pearce [MP89], de Terzopoulos et al. [TPF89], de Tonnesen [Ton91] ainsi que ceux de Luciani et de l'ACROE [Luc84, Raz86, LJC*91, LJR*91], ont montré le caractère générique de leur

méthode. Faute d'une confrontation aux questions essentielles apportées par ces nouvelles méthodes, faute aussi d'une patience suffisante nécessaire à la maîtrise de ces méthodes et passée la première phase d'engouement, ces modèles furent souvent abandonnés. Historiquement, les travaux de recherche en animation par modèle physique sont revenus à des méthodes plus traditionnelles de résolution d'équations physiques (physique du solide, physique des milieux continus, éléments finis) et vers des modèles programmés et calculés au cas par cas [NFJ02, EMF02, FSJ01]. Il en résulte la floraison d'effets dynamiques souvent tous semblables issus d'un même modèle particulier. Ces effets, bien qu'on puisse les voir aujourd'hui dans des séquences animées, restent encore relativement minoritaires par rapport à l'ensemble des pratiques d'animation. Dans un film d'animation commercial, le modèle physique ne représente en effet, encore aujourd'hui, guère plus que 5% de l'animation totale. En revanche, le développement d'effets dynamiques particuliers a enrichi grandement l'animation des jeux vidéo qui s'intéressent souvent moins à l'expressivité des mouvements produits qu'au caractère interactif du jeu. Mais là encore ce sont souvent le même type d'effets que l'on retrouve (solides en mouvement, tissus...).

II.1.2. Généricité selon le formalisme considéré

Au cours de l'histoire de la recherche en animation par modèle physique, de nombreux formalismes ont vu le jour. Nous aimerions ici les passer rapidement en revue, non pour en faire un Etat de l'Art détaillé mais pour les étudier à la lumière de la question de la généricité dont nous avons mis en avant l'extrême importance notamment au regard de la question de la créativité et de la fonction sociale.

Physique du solide

Le comportement de solides en mouvement est obtenu aujourd'hui principalement en appliquant à des volumes, les principes de la physique du solide : application de forces et de couples sur un volume auquel on associe un moment d'inertie et un centre de gravité [AG85]. Pour plus de précision, on pourra se référer par exemple aux cours SIGGRAPH de Witkin et al. dans [WBK01]. La modélisation et la simulation des collisions constituent le coût principal en temps de calcul des méthodes de simulation basées sur la physique du solide. Pour plus de précisions, [TKH*04] dresse l'état de l'art assez complet des méthodes de simulation de collision entre objets. Il s'agit, pour chacune d'entre elles de détecter de manière géométrique, dans un premier temps, les pénétrations à venir puis, dans un second temps, d'empêcher ces pénétrations par l'application de forces de contraintes.

Du point de vue de la généricité, les méthodes basées sur la physique du solide ne sont adaptées qu'à la simulation de comportement d'objets parfaitement rigides et ne peuvent pas s'appliquer à tout autre type de comportements dynamiques (objets déformables, fluides, phénomènes collectifs).

Modèles masses - ressorts

Les modèles masses - ressorts ont connu un grand succès dans la modélisation d'objets déformable notamment dans la simulation de mouvements de tissus ([BHW94, Pro95, FWC*98, ME98]). Ces modèles proposent de mettre des masses ponctuelles en interaction viscoélastique afin de simuler un comportement élastique de l'objet plus ou moins déformable. Le modèle masses -ressorts est déduit du

maillage géométrique représentant l'objet. On place, aux sommets du maillage, des masses reliées par des ressorts lorsque les sommets correspondants sont liés par une relation géométrique d'adjacence définie par la maille. Dans un souci d'optimisation du nombre de masses utilisées, il arrive que le modèle masses - ressorts soit plus raffiné dans les zones de plus forte déformation ([HPH96, CHC*02]) qu'ailleurs, selon des principes inspirés du raffinement géométrique des maillages [Sha00]. Des coupes peuvent être apportées dans le maillage et répercutées sur le modèle masses - ressorts en supprimant des ressorts. Cette suppression peut avoir lieu suivant un critère purement géométrique ou selon qu'un seuil de tension soit atteint.

Du point de vue de la généricité, ces modèles ne s'appliquent généralement qu'à la simulation des comportements d'objets déformables pouvant éventuellement subir des changements topologiques par des méthodes ad hoc.

Méthodes héritées de la mécanique des milieux continus

La mise en œuvre d'un modèle physique requiert une compétence particulière qui, jusqu'ici, est restée l'apanage des physiciens. La communauté de l'informatique graphique reprend donc à son compte les équations mises en place par les physiciens, notamment celles issues de la mécanique des milieux continus. L'hypothèse des milieux continus consiste à considérer des milieux dont les propriétés caractéristiques, comme la densité, l'élasticité, etc., sont continues. Une telle hypothèse permet d'avoir recours aux outils mathématiques reposant sur les fonctions continues et/ou dérivables [Ger73]. Les chercheurs en animation physique à l'aide de l'ordinateur tentent alors de discrétiser et de résoudre ces équations, pour obtenir des simulations efficaces pour la génération d'effets visuels.

Parmi l'ensemble de ces méthodes, la méthode des éléments finis consiste à subdiviser un domaine étudié, souvent un solide déformable, en un assemblage de sous régions contiguës appelées éléments finis, dont le comportement physique est calculable, car ils sont mécaniquement homogènes et de géométrie simple, et calculé aux sommets de ces éléments appelés points nodaux. Il s'agit de discrétiser les équations régissant par exemple les lois d'élasticités des objets déformables. Cette technique a été utilisée par Terzopoulos [TPB*87] pour intégrer la formulation lagrangienne des comportements élastiques lors de l'animation de corps déformables. La méthode initiale de modélisation basée sur la loi de Hooke, modélisation d'une élasticité parfaite, a été ensuite élargie par les auteurs pour modéliser des comportements non linéaires comme la plasticité ou les fractures [TF88]. D'autres méthodes ont été proposées pour modéliser la déformation de la peau de personnages virtuels lors de la réalisation de certaines tâches [GMT89, MT95], ou pour modéliser des déformations d'organes [CDA99]. On peut aussi citer certains travaux qui ont entrepris de simuler des phénomènes de fractures grâce à la méthode des éléments finis [OH99, OBH02] en ajoutant une modification géométrique du réseau contrôlée par un seuil sur les contraintes. Neveu et al. [NFD95] proposent de remplacer quant à eux les formes nodales d'un élément par des fonctions B-splines d'un morceau de surface de sorte que le calcul des équations du mouvement se fasse à partir des points de contrôles. Ils réduisent du fait de la représentation surfacique plus économe des B-splines le nombre de points nécessaires au calcul du mouvement.

De nombreux travaux traitent de la résolution de l'équation de Navier-Stokes qui régit le comportement des fluides pour obtenir des animations de liquides, fumées, et même de sables [ZB05].

Il s'agit alors, pour résoudre de telles équations, de disposer d'un système de représentation des variables. Deux grands systèmes de représentation sont généralement utilisés, la représentation eulérienne comme on peut la trouver dans [FM96] et la représentation lagrangienne comme on peut la trouver dans [Sta99], qui sont parfois combinées pour former des systèmes de représentation hybrides. La représentation eulérienne discrétise l'espace de manière à calculer en tout point de cet espace la valeur de certaines variables (vitesses, densités) à chaque instant de la simulation alors que la représentation Lagrangienne met en œuvre des particules se déplaçant dans l'espace au cours du temps.

Dans [DG96], Desbrun et al. introduisent l'idée d'utiliser la méthode des *Smooth particules* pour la résolution d'équations de la physique des milieux continus qui est héritée de l'astrophysique et consiste à discrétiser un objet (solide, liquide [HM01, MCG03], gaz...) dont on connaît l'équation du comportement à un niveau macroscopique, en une somme de particules représentant une partie de la matière et ayant une extension spatiale. Chaque particule possède un noyau représentant la distribution de la matière autour de celle-ci ainsi que son rayon d'action sur les autres particules. On retrouve une telle méthode dans [MKN*04] pour la modélisation des solides déformables. Ces travaux proposent de s'affranchir du maillage géométrique en contrôlant une représentation par points de l'objet à l'aide d'un plus petit nombre de particules de la même nature que les *smooth particules* et qu'ils nomment éléments physiques ou *phixels*.

Du point de vue de la généralité, ces méthodes sont très peu génériques. Elles ne s'appliquent très bien qu'aux comportements dynamiques particuliers que les équations de la mécanique des milieux continus décrivent. Elles ne sont d'ailleurs utilisées qu'à cette fin et non pour un large éventail de comportements dynamiques. On parlera volontiers pour les décrire de modèles *one shot*. Ces modèles sont par ailleurs difficiles à combiner entre eux. La question des conditions aux limites et de la combinaison de différents modèles reste une question ouverte et qui commence à être étudiée aujourd'hui [RSG*08, Coq08]. De plus, du point de vue strict des formalismes, on peut remarquer que les méthodes de résolutions intégreraient mal de potentielles interactions à distance qui sont pourtant courantes dans la nature (attraction gravitationnelle, attraction électrique, etc.). Ainsi, par exemple, les *Smooth Particules* n'interagissent qu'avec leurs voisines.

Modèles masses - interactions

Les modèles masses - interactions implantent la physique newtonienne, formalisme des masses ponctuelles en interaction. Les premiers travaux de modélisation et de simulation à partir de ces modèles sont ceux de Greenspan dans les années 70. Greenspan remet en cause la légitimité et l'efficacité de la représentation continue des phénomènes physiques en faisant remarquer que d'une part, ceux-ci émergent d'ensemble de particules en interaction, que d'autre part le processus d'observation et de mesure est discret et qu'enfin, ces représentations continues permettent difficilement de représenter des effets non linéaires, pourtant majoritaires dans la nature. Dans son ouvrage « Particle Modeling », Greenspan modélise un nombre impressionnant de phénomènes physiques par des ensembles de masses ponctuelles en interaction de type Lennard-Jones. Ces modèles ont inspiré par la suite les travaux en synthèse d'images de Miller et Pearce [MP89], de Terzopoulos et al. [TPF89] ou encore de Tonnesen [Ton91]. Partant de ce type de modèles masses – interactions,

Luciani et al. ont introduit une notion nouvelle, celle de réseaux masses – interactions, qui consistent en l'écriture des principes de la physique newtonienne sous la forme de réseaux de composants élémentaires communiquant. Ces réseaux ont été nommés « réseaux newtoniens » dans [Luc04]. Leur instanciation dans un contexte discret et algorithmique est à la base du formalisme CORDIS-ANIMA développé par l'équipe de l'ACROE à Grenoble [LJC*91, LJR*91].

L'approche de la modélisation par modèles masses - interactions est sensiblement différente de celle des modèles masses - ressorts. Une différence fondamentale réside dans le fait que les premiers sont par nature parfaitement indépendants d'un maillage géométrique. Le réseau n'est pas déduit de la topologie d'une maille. Une masse ponctuelle peut représenter un objet, plusieurs objets ou une partie d'un objet. Une masse peut aussi ne représenter aucun objet. Une interaction entre deux masses peut être une interaction entre deux objets, deux parties de deux objets différents ou du même objet. Elle peut représenter une relation physique matérielle comme un contact tout autant qu'immatérielle comme la relation d'évitement entre deux individus dans une foule [LTE06]. Une deuxième différence avec les modèles masses - ressorts est que les interactions peuvent être bien plus complexes qu'une simple interaction linéaire du premier ordre, la non-linéarité étant plus souvent la loi que l'exception.

Les modèles masses – interactions, puis leur généralisation sous la forme des réseaux masses - interactions et enfin leur formulation discrète ont montré leur grande généricité pour l'animation notamment à travers les travaux du groupe ACROE-ICA. Ils sont capables de générer des phénomènes dynamiques naturels (volutes, vortex, empilements, solides déformables ou non, fractures, plasticité) autant que des phénomènes dynamiques autonomes (comportements de foules, motricité) dont on imagine généralement plus difficilement que le modèle physique puisse s'y appliquer. Le fait que des comportements dynamiques différents peuvent être obtenu à partir du même formalisme assure une facilité à combiner les modèles pour que différents objets de nature différente entre en interaction. C'est cette généricité acquise qui, entre autres éléments, a motivé les travaux de cette thèse pour poursuivre la mise à disposition de ces réseaux masses - interactions pour l'animation.

II.2 Contrôle du mouvement

Le paradigme de l'animation par modèle physique, nous l'avons dit, est radicalement différent de la description explicite des trajectoires et des mouvements. Les modèles physiques sont en effet des modèles à temps implicite. L'animateur ne décrit plus, dans ce genre de procédé, la trajectoire des objets, mais contrôle le mouvement uniquement à partir des paramètres physiques, des conditions initiales (position, vitesse) ou encore des forces appliquées au cours de la simulation sur le système. Ces notions, nouvelles dans l'animation, impliquent un décadre du savoir-faire. Pour obtenir un mouvement souhaité, quelles valeurs de forces et de paramètres utiliser ? L'intuitivité de la relation entre les valeurs des paramètres et les comportements obtenus est une question particulière à ces modèles, et aussi leur principale difficulté. De même la notion de valeur numérique de la force est peu intuitive. Savons-nous dire combien de Newtons sont mis en jeu lorsque nous marchons par exemple ? Cette rupture dans le paradigme d'animation a amené certains chercheurs à penser qu'il était nécessaire de réintroduire une part de description explicite des trajectoires et des déformations de type

cinématique pour contrôler les modèles physiques. Ce faisant, l'animation gagnerait en naturel grâce à l'utilisation de modèles physiques tout en permettant à l'animateur de ne pas avoir à trop changer ses pratiques et ne pas entrer dans le formalisme physique réputé difficile d'accès pour lui.

Pour résoudre le problème de la détermination des forces à appliquer sur un objet physique simulé pour obtenir un mouvement souhaité, Barzel et al. proposent de résoudre un problème de dynamique inverse dans [BB88]. Ces forces sont calculées en tenant compte de contraintes à maintenir comme par exemple la contrainte que deux solides restent accrochés en un point, ou qu'un objet se déplace vers un point donné. L'utilisateur se charge de définir ces contraintes et les algorithmes résolvent alors un système d'équations linéaires sous contraintes suivant une des nombreuses méthodes fournies par la discipline du calcul numérique. Witkin et al. proposent, quant à eux, une méthode dite de « *space-time constraints* » [WK88,PW99], pour laquelle les contraintes ne sont plus simplement spatiales mais aussi temporelles. L'utilisateur fournit un ensemble de positions « clés » à réaliser à un moment donné. Les forces à appliquer sont alors calculées de façon à minimiser l'énergie nécessaire pour satisfaire ces contraintes. Une telle approche de la dynamique inverse a pu être utilisée pour calculer les forces à appliquer pour obtenir un mouvement entièrement défini [HMT94].

Les forces produites par le modèle peuvent mener à des comportements non souhaités, que l'utilisateur peut vouloir modifier. Ainsi, un modèle masses - ressorts peut, par exemple, selon le maillage géométrique dont il est déduit, exhiber des comportements anisotropiques anormaux. Dans [BC00], Bourguignon et Cani cherchent à obtenir des comportements isotropes ou anisotropes à anisotropie contrôlée à partir d'un maillage quelconque du solide déformable. Pour cela, ils rompent le principe d'action et de réaction et modifient les directions d'applications des forces d'interactions en fonction du critère décidé par l'utilisateur. Nous pouvons aussi citer certains travaux dont l'idée est de coupler un système d'intelligence artificiel à un objet physique afin de déterminer les forces internes à appliquer à cet objet pour qu'il réalise un objectif codé comme une fonction but : atteindre un lieu, par exemple. Nous pouvons citer comme pionnier de ce genre de pratiques les travaux de Van de Panne utilisant un réseau de neurones pour contrôler un objet physique [PF93].

La détermination des paramètres physiques permettant d'obtenir un comportement souhaité est quant à elle une opération difficile. Les paramètres physiques ont en effet un rôle fondamental dans l'obtention du comportement dynamique généré par le modèle physique. Pour un même modèle, des paramètres différents peuvent, en effet, générer des comportements de nature totalement différente [MP89, Luc00]. Une solution partielle à ce problème peut être obtenue lorsque les paramètres physiques d'un objet sont connus dans la littérature. Van Gelden [Gel98] propose par exemple une méthode de calcul des paramètres d'un modèle masses - ressorts modélisant un solide déformable à partir des données expérimentales du module de Young du matériau. Choi et al. [CSH*04] proposent, quant à eux, de déterminer des paramètres d'un tel réseau par une méthode d'optimisation sous contrainte du comportement produit par un solide déformable modélisé par une méthode des éléments finis. Si on simule le matériau par la technique des éléments finis et si on met en entrée d'un tel système les paramètres physiques du matériau, puisqu'on détermine les paramètres du modèle masses - ressorts à partir de la réponse à des forces externes de l'objet simulé par la méthode des éléments finis, indirectement, les paramètres du modèle masses - ressorts sont obtenus à partir d'un mouvement en

entrée (celui du solide simulé par FEM). Dans [PSE*00], Popovic et al. proposent quant à eux une méthode de détermination des paramètres physiques de la simulation en fonctions de positions clés à atteindre à un moment donnée. Contrairement à la méthode des *spacetime constraints*, aucune force extérieure n'est appliquée mais l'idée est aussi de résoudre un problème d'optimisation pour la résolution d'un système linéaire sous contraintes.

Les méthodes de contrôle du modèle physique citées connaissent encore aujourd'hui assez peu de mise en œuvre dans le cadre des logiciels d'animation faisant intervenir le modèle physique. Une des raisons possible qui peut-être invoquée est la lourdeur des calculs. Au-delà de quelques objets, la complexité algorithmique explose. De plus les méthodes de contrôles sont peu génériques et souvent appliquées à un problème bien particulier. A ce titre, la plateforme de réalité virtuelle LIVE développé par une équipe de l'IRIT, propose d'utiliser de manière concurrente plusieurs méthodes pour le contrôle de mouvement.

Cette tendance vers l'hybridation peut être mise en perspective par une autre approche que le groupe ACROE-ICA soutient quant à lui: les modèles physiques proposés sont obtenus par un acte de conception qui consiste à fabriquer le modèle à l'aide des briques élémentaires d'un formalisme donné, à trouver les conditions de sa résolution numérique et enfin à ajuster ses paramètres à la main. L'activité de modélisation physique est donc une pratique qui requiert un niveau d'expérience original par rapport à l'animation par positions « clés ». Ainsi, il est nécessaire de produire et de transmettre un savoir-faire de la modélisation relatif à cette nouvelle activité. Un modèle n'est en effet pas vu ici comme une formulation pré-existante, par exemple fournie par la littérature des sciences physiques et à implanter, mais comme un processus en cours, une ré-invention, un acte de re-crédation qui débouche sur un « savoir - modéliser » transmissible. C'est en tout cas le parti qu'a pris le groupe ICA-ACROE tout au long de ses travaux de modélisation à l'aide du formalisme CORDIS-ANIMA notamment ceux d'Annie Luciani en animation auxquels nous avons collaboré. En effet, rarement les travaux publiés par le groupe n'ont eu à faire à des méthodes exogènes pour déterminer un modèle. Les modèles ont été mis en place par des chercheurs, des étudiants, des artistes, à partir de l'exploration des possibilités offertes par des réseaux discrets newtoniens.

III. CORDIS-ANIMA

III.1. Principes fondateurs

Le moteur de simulation par réseaux masses - interactions CORDIS-ANIMA a fait l'objet de développements depuis plusieurs années [CLF93, LJC*91, LJR*91]. Il est constitué aujourd'hui d'un ensemble de fonctionnalités de simulation particulière et a été validé scientifiquement par la création d'une vaste panoplie de modèles en images, en sons et en interaction gestuelle.

Trois contraintes ont été définies pour le formalisme CORDIS-ANIMA :

- La technologie des transducteurs impose la discrétisation et la finitude des communications entre l'ordinateur et l'univers réel.
- Les objets physiques imposent de traiter la corrélation entre deux variables duales, l'une extensive, l'autre intensive.
- Les créateurs du formalisme ont choisi de respecter la modularité et la physicalité des objets aussi élémentaires soient-ils.

On en déduit que les communications physiques entre un opérateur et un objet simulé sur ordinateur sont, différemment aux communications physiques entre objets :

- discrètes en ce qui concerne leur localisation
- finies en ce qui concerne leur nombre
- orientées et à variables duales séparées

En revanche, la communication entre l'opérateur et l'objet simulé est de même nature que la communication inter objets. Les objets sont donc constitués d'éléments fondamentaux, les modules.

III.2. Un formalisme modulaire

La communication à l'intérieur de la machine se fait entre des entités, les modules, au comportement bien déterminé. Ces modules sont connectables entre eux grâce à des points de communication. Ces éléments qui réalisent la communication soit pour l'objet soit pour l'opérateur sont donc de deux types :

- les points M pour lesquels l'entrée est une variable intensive et la sortie une variable extensive.
- les points L pour lesquels c'est l'inverse.

Les règles de connectivité sont les suivantes :

- La connexion entre deux points consiste à utiliser la sortie de l'un comme entrée de l'autre. En conséquence, ne sont connectables que des points de type opposés.
- On peut connecter plusieurs points L à un même point M. La force entrante dans le point M est la somme des forces sortant des points L auxquels il est connecté.
- Il est interdit de connecter plusieurs points M à un même point L car l'entrée de position d'un point de type L ne peut provenir de plusieurs positions de points M différents.

La base constituée d'un atome à deux points L, et d'un atome à un point M permet d'obtenir toutes les figures possibles de réseaux topologiques. Il en résulte la définition de deux types de modules élémentaires, les modules MAT et les modules LIA. Le module à un point M est le module MAT. Il modélise un élément matériel. Soumis à une force, dans un champ gravitationnel par exemple, il se déplacera en fonction de paramètres tels que sa masse et de la valeur de ce champ. L'élément matériel reçoit donc une force et retourne un déplacement.

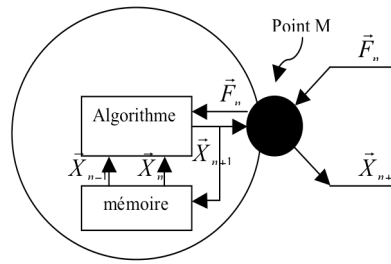


Figure 2 : Schéma du module MAT.

Le module à deux points L est le module LIA. C'est un élément de liaison, qui permet de relier deux masses entre elles, et qui modélise la composante viscoélastique d'une structure physique. Il permet de déterminer les forces d'interaction entre deux éléments matériels. Il prend en entrée les déplacements de deux modules MAT via leur point M, et retourne deux forces d'interaction. Ce type catégorise donc tous les éléments de connexion d'un objet physique, soit des élasticités, des viscosités et autres, entre modules MAT.

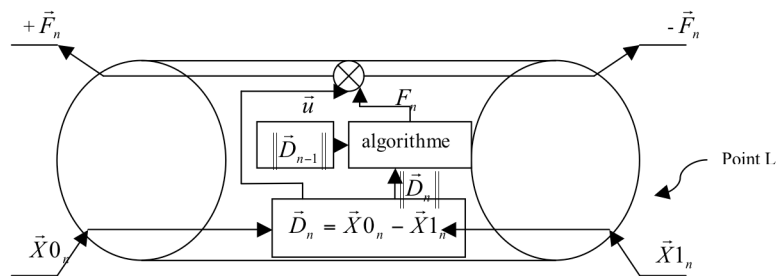


Figure 3 : Schéma du module LIA

Les objets CORDIS-ANIMA servant à la modélisation seront construits par l'assemblage d'éléments matériels et de liaisons basés sur les types MAT et LIA. Les éléments construits selon le type MAT peuvent être comparés aux particules élémentaires d'un objet physique et ceux construits sur le type LIA aux interactions qui les connectent. C'est à ce titre que nous parlons de réseaux masses - interactions (cf. figure 4).

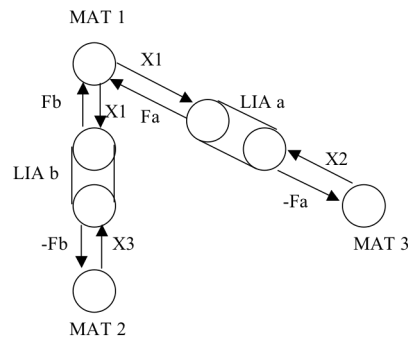


Figure 4 : Un réseau CORDIS-ANIMA simple

Ainsi, la pratique de l'animation par réseaux CORDIS-ANIMA a la particularité d'être une approche constructiviste en partant de modules élémentaires pour arriver à des réseaux de plus en plus complexes, une sorte de LEGOTM dynamique.

III.3. Une base de paramètres compréhensible

Pour une bonne compréhension de la fonction de chaque module, il est nécessaire que les paramètres qui permettent d'en contrôler le comportement soient le plus compréhensibles possible pour l'utilisateur. Le choix a été fait de proposer une base de paramètres faisant référence à des grandeurs physiques comme la masse, l'élasticité ou la viscosité. Il est assez facile, en effet, de se faire une représentation de ces grandeurs. Une forte valeur d'élasticité produira des comportements très rigides, à l'inverse une faible valeur produira des comportements plutôt mous. Ces paramètres physiques font en réalité référence à notre expérience du quotidien comme, par exemple, lorsque nous manipulons des objets plus ou moins lourds.

III.4. Un formalisme orienté temps-réel

Tous les algorithmes des modules CORDIS-ANIMA sont des algorithmes de calcul élémentaires faisant intervenir très peu de multiplications, d'additions, de calcul de racines carrées et de branchements conditionnels. Comparé à un grand nombre de calculs numériques effectués aujourd'hui sur les ordinateurs, le calcul d'un pas de temps de simulation au sein d'un module CORDIS-ANIMA est très élémentaire et très peu coûteux. De plus la modularité du formalisme en fait un bon candidat

pour le calcul sur des machines fortement parallèles. Il en résulte que par nature, on peut dire de CORDIS-ANIMA qu'il est orienté temps réel. Au sein du groupe ACROE-ICA, il a donc été possible d'implanter plusieurs versions du simulateur CORDIS-ANIMA dans une version temps réel synchrone permettant de faire calculer des modèles à plusieurs milliers de Hertz [FCL98,Gir99].

III.5. Un formalisme pour la multi-sensorialité

Une démarche originale du groupe ACROE-ICA a été de développer l'ensemble de ses technologies à partir du seul formalisme CORDIS-ANIMA utilisé pour la génération de mouvements visuels, la génération de sons et l'interaction gestuelle. Grâce à des transducteurs gestuels rétroactifs, appelés également interfaces à retour d'effort ou interfaces haptiques, il est possible que l'utilisateur interagisse avec la simulation c'est-à-dire qu'il agisse sur les objets simulés et que ces objets lui retournent une information de force. Dans ce cadre, l'utilisateur se retrouve dans une situation nouvelle d'interaction avec des objets simulés capables de produire du son et de l'image grâce au même formalisme. Cette approche d'une instrumentalité dans les arts dynamiques est une approche relativement nouvelle pour l'animation là où elle l'est moins pour la musique par exemple. En effet, la musique a toujours utilisé des instruments faisant intervenir le geste dans une relation intime et physique.

Comme nous avons pu le voir, le formalisme CORDIS-ANIMA semble avoir de nombreux avantages tant en matière de simplicité qu'au niveau des possibilités créatives qu'il promet grâce à sa forte généricité. Cependant, dans la pratique du formalisme notamment pour l'animation, un verrou important a longtemps limité sa facilité d'accès et d'utilisation. En effet, la plupart des modèles réalisés ont été programmés directement en faisant appel à la bibliothèque de modules. Dans la version de CORDIS-ANIMA développée par Jimenez [Jim93], le modèle se présentait sous la forme d'un script qui contenait à la fois la structure du modèle, c'est-à-dire les modules et leur connexion, et les valeurs des paramètres physiques. Dans ce cas, gérer un modèle complexe est très peu commode et prend du temps car d'une part le texte à plat illustre mal la complexité et les valeurs quantitatives sont dispersées tout au long du programme. Il faut donner à voir le modèle, permettre à l'utilisateur d'en manipuler ses éléments. Il faut surtout permettre de manipuler de grosses quantités de modules et des fonctions d'interaction complexes. La toute première version d'une interface interactive pour le simulateur ANIMA (version de CORDIS-ANIMA pour l'animation) de l'environnement de conception appelé ANIMUS a été développée très tôt par Annie Luciani [Luc85] et A. Razafindrakoto [Raz86]. Cependant, elle ne permettait pas de manipuler que des modèles très simples de l'ordre de quelques dizaines de masses et quelques dizaines d'interactions.

Grâce au développement récent des interfaces interactives performantes, et des nombreux outils facilitant ce développement, il nous est aujourd'hui possible de remettre cette nécessité sur le chevalet. La motivation de cette thèse est donc de donner à l'utilisateur tout l'environnement de conception interactif nécessaire pour accéder à la création de modèles CORDIS-ANIMA mettant en valeur toutes leurs possibilités.

IV. La nécessité d'un environnement de conception interactif

IV.1 Etat des lieux des réseaux masses - interactions dans les systèmes d'animation

Les modèles masses - interactions sont aujourd'hui mis à disposition d'utilisateurs suivant différentes modalités que nous allons expliciter ici.

IV.1.1. Environnements de conception non interactifs

Excepté la librairie CORDIS-ANIMA, peu de librairies exclusivement centrées sur les modèles masses - interactions ont été mises à disposition des utilisateurs avec un éventail aussi large de modules. En revanche, il existe depuis une quinzaine d'années, d'autres librairies basées sur des concepts différents ou similaires qui ont été mise à disposition des développeurs. Une des plus utilisées aujourd'hui notamment dans le domaine des jeux vidéo est la librairie Havok. Cette librairie était initialement basée sur un modèleur pour la physique du solide capable de donner un comportement de solides ou de solides articulés à des objets prédéfinis de manière géométrique. En outre, la notion de contrainte géométrique (point fixe, contrainte sur une trajectoire) ou physique (ressort) est originellement présente. Depuis les premières versions, la librairie Havok s'est enrichie d'autres méthodes de simulation et notamment de modèles masses - ressorts pour la simulation de vêtements ou encore la gestion de fractures au sein de ces modèles tout en conservant la possibilité aux objets déformables obtenus d'entrer en collision avec les autres objets de la scène. Depuis la mise à disposition de cette librairie, il est devenu courant de voir dans les jeux vidéo, une multitude d'objets déformables ou non ainsi que de personnages réagissant dynamiquement à des chocs. Il reste que ces librairies mélangent la description physique et la description géométrique des objets et de ce fait ne permette pas au concept propre à la modélisation masses - interactions de s'y insérer de manière générique.

Suivant une autre logique de modélisation, Maya a développé un noyau pour la simulation de modèles masses - interactions. Ce projet baptisé Nucleus a été développé par Jos Stam qui travaillait préalablement sur des modèles fondés sur la mécanique des milieux continus [Sta99]. La description

du projet Nucleus est motivée par les propriétés de généricité des modèles masses - interactions ainsi que la compatibilité entre des modèles simulant des comportements dynamiques différents. Selon les propres termes de Jos Stam : « traditionnellement les phénomènes tels que le vent, les vêtements, les solides ont été traités de manière séparée dans des systèmes d'animation distincts. L'ambition du projet Nucleus est de fournir un environnement de simulation unifié capable de traiter des interactions entre différents types d'éléments comme les liquides, les vêtements, les cheveux ou les solides avec un unique système dynamique. [...] Le concept clé de Nucleus est que les interactions entre les différents éléments peuvent être représentées par des modèles simples basés sur des particules en interaction ». Néanmoins, dans la mise en œuvre, J. Stam suit la philosophie de l'animation par modèle physique consistant à ne pas laisser l'utilisateur modéliser les différents comportements dynamiques mais bien plutôt à lui fournir des outils de contrôle de plus haut niveau sur des modèles « clés en main ». L'utilisateur n'a pas la possibilité de concevoir lui-même ses modèles puisque l'API de Nucléus n'a pas été, en tout cas pour l'instant, rendue libre d'accès et qu'aucune interface de conception n'a été prévue. Il s'en suit que le résultat actuel est plutôt décevant par rapport aux objectifs initialement énoncés. Le noyau Nucléus a bien été utilisé dans les versions récentes de Maya, mais uniquement pour modéliser des objets déformables et des liquides. Les outils de contrôle de haut niveau se sont quant à eux restreint à ceux qui existaient déjà dans de nombreux logiciels (modification globale de la raideur, prévention des collisions, etc.) avec tout de même une possibilité de combiner les modèles, comme de placer des liquides dans des objets déformables par exemple.

IV.1.2. Logiciels de démonstration

Certains logiciels permettent à l'utilisateur d'expérimenter les concepts de modélisation physique sur des exemples simples, préalablement construits ou qu'ils peuvent construire eux-mêmes. Il existe ainsi un certain nombre d'applications sur Internet permettant d'expérimenter le frottement entre deux solides (solide sur un plan incliné).

L'application JAVA Sodaplay se rapproche plus particulièrement des modèles masses - interactions dans une version masses - ressorts où le principe constructiviste de la modélisation est mis en œuvre. L'utilisateur peut concevoir des « créatures » composées de ressorts visqueux qui sont capables de se mouvoir par elles-mêmes ou par adjonction d'énergie à l'intérieur des ressorts. De manière évidente, le but d'un tel projet n'est pas la production professionnelle d'animation et elle ne dispose que d'un petit nombre de fonctionnalités : mêmes paramètres de raideur et de viscosité pour tous les ressorts, visualisation sommaire, fonctions d'interaction linéaires.

D'une manière plus évoluée, plusieurs logiciels d'aide à l'apprentissage de la physique ont fait leur apparition récemment, comme le logiciel Interactive Physics [IPhys] ou le logiciel PHUN [Phun]. Ils permettent à l'utilisateur de créer des modèles physiques à partir de formes (formes dessinées, ...) ou d'objets simples (sphère, pavé, ...). Par leur interactivité, de tels logiciels pourraient se rapprocher de la philosophie de MIMESIS, mais ils ne cherchent pas le même degré de généricité car ils sont pensés uniquement comme un soutien à l'apprentissage des savoirs de la physique ou à la création récréative de scènes dynamiques simples. Ainsi, il n'existe par exemple aucune entrée et aucune sortie. Les modèles sont uniquement des modèles 2D. Les principes mis en jeu sont surtout ceux de la physique générale centrée sur la physique du solide, la physique du point n'en étant qu'une petite partie. Les

briques fondamentales sont en effet surtout des formes et des objets en relation physique et non des masses ponctuelles. Les interactions ne sont pas considérées comme des briques de bases. Ainsi, excepté les frottements qui sont un classique du savoir physique, il n'y a pas d'autres interactions complexes.

Pour autant, ces logiciels attestent d'une volonté d'inscrire l'utilisateur dans une optique de modélisation et ce peut être pour un novice de la modélisation physique un premier contact par l'expérience avec ce type de pratique.

IV.1.3. Composant matériel

Le calcul physique comme d'autres peut être gourmand en temps de calcul. C'est notamment le cas pour les modèles masses - interactions lorsqu'un grand nombre de masses et d'interactions forment le modèle comme dans certains modèles de tissus présents aujourd'hui dans les jeux vidéos ou dans les films d'animation. Pour accélérer ces calculs, une piste autre que la solution de l'optimisation algorithmique peut être de faire appel à des calculateurs dédiés. Ainsi, la société AGEIA a développé un processeur physique (PPU), nommé PhysiX, dont les opérations systématiquement utilisées par le calcul physique sont implantées « en dur ». Il n'a pas été possible d'obtenir d'informations techniques sur l'architecture de ce composant matériel. En revanche, son histoire nous permet de faire l'hypothèse qu'il existe un réel enjeu autour du développement de tels processeurs. En effet, dès sa sortie, le processeur PhysiX a connu une réelle pression de la part de la société NVidia développant des processeurs graphiques (GPU) qui y a vu un concurrent potentiel. Aux vues du marché fleurissant des GPU, ces calculateurs ont atteint des performances telles que des chercheurs dans des domaines variés voulant réaliser des calculs lourds ont utilisé leurs propriétés de parallélisme. La réponse de Nvidia a donc été de concurrencer AGEIA en proposant de réaliser les calculs physiques directement sur le GPU. Au final, cette concurrence s'est soldée par l'intégration du processeur PhysiX dans les nouvelles cartes de NVidia et un *middleware* commun permet aujourd'hui d'opérer les calculs au choix sur l'une ou l'autre des unités.

Puisque aucune information n'est encore disponible sur l'architecture de ce processeur, la question reste ouverte pour le futur sur son utilité pour le calcul de réseaux masses - interactions. Des *benchs* restent à réaliser pour savoir si des modèles de références sont ou non effectivement calculés plus rapidement et dans de bonnes conditions, respectant par exemple à chaque instant la physicalité des simulations. Des effets non désirés peuvent en effet apparaître sournoisement, liés au format des nombres, au découpage spatial de l'espace de simulation, effets non visibles dans les simulations actuelles à très grand nombre de particules et relativement simples du point de vue des interactions et de la dynamique.

IV.1.4. Environnement de conception interactif

Nous ne connaissons pas d'environnements interactifs capables de concevoir des réseaux masses - interactions. En revanche, certains projets se sont attachés à fournir un environnement de conception de modèles à partir de bibliothèques hétérogènes de modélisation et de simulation. C'est le cas

notamment du projet SOFA [ACF*07], initialisé par Stéphane Cotin et repris dans un projet commun entre plusieurs équipes de recherche en simulation chirurgicale. Au sein de SOFA, plusieurs formalismes coexistent dont la plupart sont issus de la mécanique des milieux continus ou de la mécanique des solides rigides. Mais on trouve aussi des systèmes masses - ressorts. Ainsi, il est possible de simuler des objets déformables mais aussi des fluides. L'originalité de SOFA est de proposer un environnement de développement interactif pour gérer des scènes complexes faisant intervenir plusieurs objets dont les comportements sont simulés par des formalismes différents à l'aide d'un graphe de scène. Il est possible à l'utilisateur de définir un objet selon plusieurs points de vue en même temps: par exemple un point de vue géométrique, un point de vue physique, un point de vue interaction avec d'autres objets. Ainsi, les collisions peuvent être modélisées par un formalisme différent de celui utilisé pour le modèle du comportement propre de l'objet. Libre alors à l'utilisateur de choisir un modèle physique particulier ou d'implanter le sien pour une partie définie dans le graphe de scène.

Très clairement, le parti pris de SOFA est donc de permettre de poursuivre les efforts sur la recherche de collaboration entre modèles différents, contrairement à celui de Nucleus qui prônait l'abandon de ces recherches pour revenir à un formalisme plus générique et modulaire, position pour laquelle les recherches du groupe ACROE-ICA optent.

IV.2 Un environnement de conception interactif de modèles masses - interactions n'existe pas encore

Aujourd'hui, certains membres de la communauté des chercheurs en informatique graphique redécouvrent avec enthousiasme la richesse du modèle physique particulière. C'est le cas, par exemple, des chercheurs de la société ILM [KNB03]. Mais leur développement se fait toujours dans un contexte très spécifique et aucun environnement logiciel n'est mis à la disposition d'une communauté importante. A part les quelques logiciels de démonstration de type « Sodaplay » dont nous avons parlé ci-dessus ou les implantations partielles de fonctionnalités physiques dans des logiciels de synthèse d'images existants, il n'existe pas à ce jour d'environnement de conception générique offert au modélisateur de mouvement par modèle physique. Dans la pratique, les entreprises utilisant des modèles physiques implantent au cas par cas leur méthode en offrant quelque fois une interface à l'animateur lui permettant de manipuler certains paramètres. Le but est alors de paramétrer un comportement défini à l'avance.

En parallèle, les gros logiciels d'animation (MAYA, Softimage, etc.) incorporent au compte-goutte et de manière rudimentaire (pas d'interaction non linéaires, fonctionnalités de conception limitées, édition globale des paramètres) des fonctionnalités de modélisation physique particulière. On y trouve généralement peu d'interactions complexes. Les fonctionnalités de conception sont limitées et il est seulement permis à l'utilisateur d'éditer des paramètres globaux. Plus généralement, la présence de modèles physiques au sein d'environnements de conception interactifs est tributaire de la géométrie. On décrit en effet d'abord les objets d'un point de vue géométrique puis on définit un comportement physique pour ces objets.

Dans tout ce contexte, la modélisation physique masses - interactions a fait l'objet de développements assez limités. En effet, bien qu'il soit plus facile à implémenter et moins coûteux en temps de calcul, par rapport aux autres modèles physiques (méthode des éléments finis, physique du solide...), elle suppose des décadres cachés à la fois conceptuels et technologiques, comme en particulier le développement d'un savoir-faire / savoir-modéliser qui ne découle pas directement des démarches « objets 3D » ou « solides » définissant en premier lieu une forme. Or, moyennant de tels décadres, elle offre une panoplie d'effets et surtout une compatibilité entre effets tout à fait remarquables.

On peut donc considérer que la modélisation et notamment la modélisation masses - interactions est un enjeu toujours actuel pour la production d'images animées. Or, de manière surprenante, aucun environnement n'existe encore pour la conception de modèles masses - interactions et encore moins de réseaux discrets masses - interactions qui en sont la version la plus générique du point de vue des sciences de l'information. Le fait qu'aucun environnement de conception n'existe implique de plus qu'il n'existe pas non plus de savoir-faire coordonnés et accessibles, soutenus par des interfaces utilisateurs adaptées, savoir-faire nécessaires à la mise au point de modèles physiques adéquats et efficaces par des usagers nouveaux.

V. Bilan

Un des problèmes majeurs de la modélisation physique masses - interactions est donc de ne pas disposer à l'heure actuelle d'un outil de conception interactif, proposant un cadre et des méthodes de conception. En effet, au-delà des principes physiques plus ou moins bien connus et maîtrisés « scolairement » ou « artisanalement » qui ne permettent pas d'aller bien loin dans l'activité de modélisation, il n'existe aucun cadre de conception pour la modélisation physique masses – interactions. Or, cette modélisation est d'un type de difficulté vraiment différente de la modélisation géométrique pour la synthèse du mouvement. En effet la modélisation géométrique est basée sur la description des formes des objets alors que notre approche propose en premier lieu de modéliser des comportements dynamiques. Cette approche est relativement nouvelle, c'est celle de la dynamique initiée notamment par Newton, discipline nouvelle au regard de l'invention de la géométrie. Si un support comme la feuille de papier suffit à la représentation explicite, sensible et externe de la géométrie, la dynamique nécessite une représentation du temps que la simulation à l'aide de l'ordinateur permet de donner de manière explicite et sensible. La simulation physique est donc une pratique très neuve dans l'histoire des techniques humaines.

Un outil complet de modélisation dans lequel coopèreraient différents composants de conception, structurelle, paramétrique, de simulation et de visualisation, adaptés à la conception et à la mise au point des modèles physiques masses - interactions, est donc nécessaire. Voici deux exemples liminaires, les autres nécessités étant développées tout au long de ce document :

- On appelle souvent « modèle physique » la structure d'un ensemble masses - interactions. Cela est insuffisant car une difficulté essentielle est la mise au point des valeurs de paramètres physiques. Celles-ci dépendent de la structure du modèle et inversement. De plus, ces valeurs ont des incidences importantes sur le comportement algorithmique de la simulation numérique. Enfin, certaines variables essentielles pour ces réglages sont des variables cachées comme par exemple la valeur du pas de simulation.
- La mise au point du modèle dynamique nécessite de disposer d'outils d'observation tels que des possibilités de visualisations élémentaires adaptées, qui ne sont pas à confondre avec les représentations visuelles finales.

Cela signifie qu'il est essentiel de disposer d'un outil de conception interactif, complet, si possible léger, proposant des outils adaptés aux types de difficulté de la modélisation physique particulière.

Nous touchons ici au point crucial qui motive les travaux de cette thèse. La difficulté dans le modèle physique particulière ne réside pas tant dans l'écriture des algorithmes élémentaires mais dans la modélisation elle-même, qui consiste à trouver le bon modèle pour le bon effet (structures, paramètres, pas de simulation, valeurs initiales, modalités de contrôle). Il faut constituer en quelque sorte un profil de compétence « génie empirique en modélisation physique » qui sera précisément favorisé par la mise à disposition d'un environnement de conception interactif. Le travail qui a accompagné cette thèse est donc de proposer un environnement de conception pour les réseaux CORDIS-ANIMA répondant aux problématiques propres de ce type de formalisme afin d'initier le développement d'une nouvelle génération de logiciels pour l'animation pour laquelle l'animateur devient un véritable modélisateur.

- Chapitre 2 -

**Les fonctionnalités d'un
environnement de conception
interactif et de simulation de
modèles CORDIS-ANIMA**

I. Différentes phases dans la conception d'un modèle CORDIS-ANIMA

Le chapitre introductif précédent, incluait une présentation du concept d'animation par modélisation et simulation de réseaux masses - interactions CORDIS-ANIMA et situait ce concept par rapport à l'état des lieux en animation par ordinateur. Il nous a permis de mettre en évidence que le principe d'animation proposé rompt de manière assez radicale avec l'idée de l'objet tel que la géométrie peut la poser, c'est-à-dire en opérant une réduction spatiale dans la représentation des phénomènes observés ou observables. À l'inverse des environnements usuellement développés par la synthèse d'images, l'animateur utilisant le formalisme CORDIS-ANIMA ne commence donc pas par définir une représentation géométrique des phénomènes observés ou souhaités, visions où l'objet serait en premier lieu défini par un contour spatial, mais se pose plutôt la question du comportement dynamique, du mouvement qu'il veut obtenir, c'est-à-dire de l'évolution temporelle de critères visuels. Par conséquent, ce n'est que dans une autre phase ou plus précisément lors d'une autre activité, qu'il devra révéler le mouvement généré par un habillage qui lui donnera une forme spatiale. Un environnement de conception de réseaux CORDIS-ANIMA doit donc être d'abord centré sur la conception du réseau lui-même, en ce qu'il permettra à l'utilisateur, avant toutes autres choses, de générer le mouvement qu'il souhaite. Il est donc nécessaire d'analyser les étapes de l'élaboration d'un modèle pour une telle pratique afin que l'interface porte les concepts propres à ce type de conception. La conception d'un modèle physique CORDIS-ANIMA pour la génération de mouvements visuels, allie plusieurs activités de nature différentes propres à la modélisation physique et en particulier la modélisation physique particulière. Il est naturellement nécessaire de catégoriser ces activités afin de mieux les spécifier et les coordonner. Pour cela, une méthode consiste à les considérer au point de départ de la conception, dans l'étape de la « page blanche ». Dans cette situation où l'utilisateur doit définir un modèle, il traverse ces activités de manière séquentielle. Par la suite, bien entendu, il sera souvent nécessaire de revenir sur chacune d'entre elles, dans un processus itératif de conception, tant que le modèle n'est pas conforme à ses attentes.

Cinq phases peuvent alors être identifiées lors de la conception d'un modèle CORDIS-ANIMA :

Phase de « Préstructuration Qualitative » ou PSQL:

La mise en place d'un modèle CORDIS-ANIMA nécessite en premier lieu de définir la topologie du réseau masses – interactions qui permettent de définir la structure physique de l'objet ou du

phénomène à simuler. La topologie d'un réseau est la configuration des connexions entre modules du formalisme, comme les modules MAT et LIA présentés dans le chapitre précédent. Lors de cette phase, il s'agit donc de définir le nombre de modules utilisés, leur type et les connexions entre ces modules. Dans la mesure où le réseau et sa topologie conditionnent la classe de mouvements productibles, cette phase a été nommée « Préstructuration Qualitative », qualitative signifiant qu'aucune donnée quantitative n'est ici fixée.

Phase de « Préstructuration Quantitative » ou PSQN:

Une fois le réseau structurellement défini, on peut alors, et seulement alors, donner une valeur à chacun des paramètres des modules qui le composent. Le nom de « Préstructuration quantitative » s'impose alors pour identifier cette phase.

Phase « Conditions Initiales » ou CI :

On peut alors, et seulement alors, comme pour tous les systèmes dynamiques qui sont des systèmes sensibles aux conditions initiales, donner les valeurs initiales des variables du modèle.

Phase « simulation » ou « jeu » :

Alors, et seulement alors, Le modèle, c'est-à-dire le réseau de composants masses - interactions spécifique aux mouvements souhaités, est simulable, puisqu'ont été définies toutes les données nécessaires et suffisantes à la génération temporelle des mouvements. On peut remarquer que si le modèle ainsi défini, contient des entrées en provenance de l'extérieur, via des capteurs ou même des données capteurs enregistrés, la simulation correspond à une activité de jeu.

Phase « habillage spatial et visualisation » :

Le mouvement pouvant alors être généré et bien qu'il soit totalement calculable, si l'on veut l'observer visuellement, il est nécessaire de passer par une étape d'habillage spatial puis de visualisation de cet habillage, à partir de toutes les données précédentes (structure, paramètres, conditions initiales).

Ces différentes phases ont été définies de manière formelle et avant toute expérimentation dès la première spécification du modeleur CORDIS-ANIMA [CLF84, CLF93, LJC*91]. Puis d'autres systèmes ont été réalisés qui ne sont pas fondés sur cette organisation des activités de conception. En particulier, dans la proposition de Jimenez [JL93], les phases de préstructuration qualitative et quantitative n'étaient pas disjointes. D'une manière générale, dans les systèmes de synthèse d'images, ces phases ne sont pas disjointes. Il semble en effet naturel de concevoir dans le même temps la topologie du réseau et la donnée des valeurs des paramètres. Le système CORDIS-ANIMA développé par Jimenez a été très utilisé et durant cette utilisation, il est apparu en effet clairement que sa qualité essentielle a été de pouvoir développer très rapidement des modèles de manière très intégrés. Deux limitations essentielles, non triviales, se sont cependant présentées, sur lesquelles se fondent le travail de la thèse proposée ici, et qui tiennent par des raisons profondes au principe du modèle physique lui-même.

Tout d'abord, il s'est avéré difficile de dépasser un certain stade de simplicité – donc d'atteindre à un certain stade de complexité. Cela tient au fait que le réglage des paramètres d'un système dynamique est une activité très complexe, souvent sous-estimée au profit de la forme du réseau (ou dans d'autres types de modèles non modulaires, à la forme de l'équation régissant les phénomènes). Notre propre expérience, puisque nous avons commencé cette thèse par l'examen et la réimplantation de tous les modèles précédemment disponibles, nous a par exemple montré, que pour une topologie donnée, le travail paramétrique nécessitait au moins 10 passes. Ceci est une spécificité du modèle physique qui n'existe pas – ou est moins critique - en modélisation géométrique. Ensuite, cette manière de penser altère la généricité du formalisme. En effet, et pour les mêmes raisons, une topologie – ou une formule – est un principe permettant de générer plusieurs classes de phénomènes, et ce selon les paramètres choisis. Un exemple bien connu est celui de la turbulence, où celle-ci dépend non seulement de l'équation mathématique, par exemple l'équation de Navier-Stokes, mais également de conditions paramétriques, par exemple le nombre de Reynolds. Par conséquent, le fait de lier de manière native, la structure à la valeur des paramètres tend à limiter l'expérimentation extensive de ceux-ci. Enfin, cela nous ramène à une manière de penser de type « objet », bien adapté à la représentation géométrique, et dont nous avons montré dans le chapitre précédent, qu'elle était dans le fond très différente de l'observation physicienne, puisqu'en physique, les entités – « les objets » - constituant un système ne sont pas séparables, si ce n'est à la frontière, par la définition du système fermé.

Ce chapitre a pour vocation de présenter les problématiques relatives à chacune des étapes de la conception d'un modèle CORDIS-ANIMA, telles que nous les avons revisitées et généralisées à la suite de nos expériences et de proposer des outils de conception adaptés à chacune de ces problématiques et à leur coordination, pour concevoir des modèles CORDIS-ANIMA au degré de complexité que nous souhaitons aujourd'hui.

II. Phase « PSQL » : Conception de la topologie du réseau CORDIS-ANIMA

II.1. La complexité topologique dans les réseaux CORDIS-ANIMA

Une propriété essentielle des réseaux CORDIS-ANIMA, et qui fait leur force majeure est la possibilité de disposer d'une grande variabilité topologique. Il est par conséquent nécessaire de tenir compte en premier lieu de cette variabilité dans la spécification d'un outil de conception de cette topologie, efficace et adapté.

Différemment de ce qui est habituellement utilisé en modèle physique particulière, comme par exemple dans les logiciels « jouets » de type Sodaplay ou même dans beaucoup de modèles réalisés dans le passé avec CORDIS-ANIMA, une étude des modèles existants récents (après 2000) nous montre qu'un nombre important de masses, de l'ordre de plusieurs centaines au millier, est la norme plutôt que l'exception pour obtenir une grande partie des comportements dynamiques naturels comme, par exemple les phénomènes collectifs dont les piles, les vorticités, les accumulations, les effondrements de surface sont des comportements émergents. Ce nombre de masses n'est pas uniquement nécessité, par les grandes tailles spatiales des objets ou phénomènes à modéliser. D'ailleurs, la discrétisation de la matière en masses ponctuelles pour représenter le mouvement réclame en général un nombre de points inférieur à celui nécessaires à la visualisation de l'objet. Mais c'est avant tout pour la complexité des dynamiques elles-mêmes que le nombre des masses ponctuelles doit augmenter. La variabilité topologique relève beaucoup plus encore de la configuration des interactions reliant les masses entre elles : de n masses reliées en chaîne (nombre d'interactions de l'ordre de n) pour un modèle très structuré comme une corde, à n masses interconnectées (nombre d'interactions de l'ordre de n^2), pour un modèle où tous les éléments sont indifférenciés comme dans un fluide, en passant par toutes les structures intermédiaires : structures tenseurs volumiques pour modéliser des barres rigides, structures planes complexes pour des comportements surfaciques de matériaux plastiques, etc. Le comportement de l'objet simulé dépend aussi de la nature des interactions choisies. Une interaction élastique linéaire permettra, par exemple, de simuler des comportements élastiques voir rigides si le paramètre d'élasticité est élevé. Mais une collision, une plasticité, une striction, des frictions, ne pourront pas être rendues grâce à ce type d'interaction. Il faut,

dans ce cas, nécessairement se tourner vers l'utilisation d'interactions non linéaires. Ainsi la variabilité topologique des réseaux CORDIS-ANIMA dépend du nombre de masses, de la manière dont elles sont liées entre elles par des interactions et de la nature des modules d'interaction utilisés.

Un exemple de modèle complexe est peut-être le meilleur moyen d'illustrer la grande variabilité potentielle de la topologie d'un réseau CORDIS-ANIMA. Prenons l'exemple de la modélisation d'un véhicule sur sol meuble comme décrit par Chancelou dans [Cha96] que nous avons réimplanté dans MIMESIS comme il le sera montré au chapitre 5, paragraphe III.3.4. Le sous-réseau global propre au seul véhicule est obtenu à l'aide d'un petit réseau plutôt hétérogène (masses placées à certains sommets du véhicule et reliées par des interactions élastiques permettant d'assurer la cohérence). En revanche, le sol est lui modélisé à l'aide d'un réseau comportant un grand nombre de masses (deux couches, une supérieure modélisant le sol meuble lui-même et une autre inférieure formées de masses fixes permettant de le « fixer » dans l'espace). Les masses voisines de cette sous-partie du réseau global formant le modèle sont reliées entre elles suivant un motif plutôt homogène (interaction plastiques entre les deux couches, interactions plastiques entre les masses voisines de la couche supérieure). Finalement ces deux parties de réseaux sont en interactions de butée. Toutes les masses des roues du véhicule sont en interaction avec toutes les masses de la couche supérieure du sol meuble.

Cet exemple exhibe bien deux grands pôles de configuration pour les réseaux CORDIS-ANIMA :

- Réseaux de grande taille plutôt homogènes comme le sous-réseau propre au sol meuble du modèle. Les interactions sont toutes du même type et l'on retrouve des grands motifs dans l'organisation des connexions (entre masses voisines, toutes interconnectées)
- Réseaux de taille moyenne ou petite et plutôt hétérogènes, comme le sous-réseau propre au véhicule. Les interactions sont semblables ou non et surtout on ne retrouve pas de grands motifs de connexions. Bien souvent aussi dans des modèles complexes, il s'agit de combiner ces deux types de réseaux comme l'auteur le fait par exemple à l'aide d'interactions non linéaires viscoélastiques dites butées viscoélastiques.

La suite de cette partie traite donc des fonctionnalités les plus adaptées pour chaque grand pôle de configuration. La conception textuelle étant plutôt appropriée pour l'édition de réseaux de grande taille et homogènes bien qu'elle le soit aussi pour l'édition de réseaux de petite taille et hétérogènes. La conception graphique interactive est une fonctionnalité, quant à elle, réservée plutôt pour la seule deuxième catégorie. Dans tous les cas, la conception est facilitée par une structuration des réseaux qui doit être la plus libre possible.

II.2. Conception textuelle

Pour les raisons introduites au paragraphe précédent, il est primitivement (ou absolument) nécessaire de disposer d'un outil d'édition de la topologie des réseaux CORDIS-ANIMA, qui respecte les axes de complexité des réseaux CORDIS-ANIMA, et en particulier qui permettent de créer des topologies diverses, de manière à être capables d'explorer toute la variabilité des comportements dynamiques autorisée par le formalisme, et dont on souhaite qu'ils soient le plus nombreux possible : de la rigidité

ou élasticité simple à des comportements turbulents, des changements d'états, des comportements à plusieurs échelles temporelles, des comportements collectifs émergents, etc. En effet, ce vaste éventail de comportements implique des topologies de réseaux CORDIS radicalement différentes. Un comportement rigide, rigide articulé ou faiblement déformable nécessitera que la topologie spatiale du réseau reste fixe. Les masses seront plutôt en interactions avec leurs voisines proches. Les interactions entre les masses doivent alors assurer une cohésion forte du réseau. Il est donc nécessaire que les fonctions d'interactions assurent des zones de cohésion fonctionnelle. L'interaction élastique linéaire du premier ordre en est un exemple. En revanche un comportement collectif, dans lequel les éléments sont relativement indifférenciés, sera plutôt modélisé à l'aide d'un réseau où chaque masse est en interaction variable avec toutes les autres.

L'approche permettant le mieux de répondre à cette contrainte de variabilité est l'approche basée sur un langage textuel. Le langage textuel permet en effet la création de topologies très différentes grâce à la définition de structures de contrôles telles que les conditions (structures alternatives), les boucles (structures répétitives), branchements conditionnels ou non, etc. Nous devons faire remarquer ici que, de ce fait, nous nous dissociions des tendances contemporaines basées sur la conception graphique interactive. Nous nous dissociions également des premières implantations de modeleurs interactifs de CORDIS-ANIMA pour la synthèse de mouvements visuels, effectués dans les premiers systèmes développés par le laboratoire. En effet, dès 1985-1986, la première implantation du modeleur ANIMA effectuée par Annie Luciani [Luc85] et par Razafindrakoto [Raz86] mettait en œuvre une conception graphique interactive. Ces expériences ont montré aussi bien la force que les limitations de ce type de conception pour un outil générique puissant de modèles de génération de mouvements.

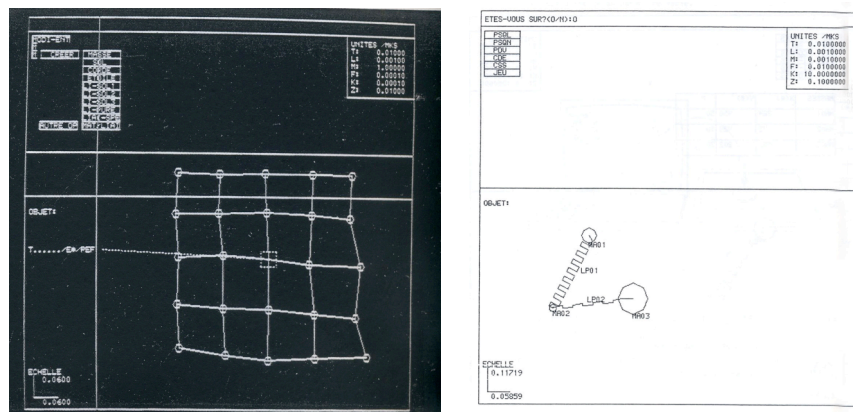


Figure 5 : Premières versions d'une interface interactive pour le simulateur ANIMA

Nous donnons ci-après un exemple de modèle pour lequel la conception graphique interactive serait inadaptée, « le modèle de fumée », au profit d'une conception textuelle.

Deux grandes structures principales peuvent illustrer la nécessité de l'utilisation d'un langage, en particulier pour des modèles présentant un grand nombre de modules, en interaction relativement homogène. Il s'agit de la structure d'agglomérat dans laquelle chaque masse est en interaction avec

toutes les autres et de l'interaction entre deux sous-réseaux (toutes les masses d'un sous-réseau sont en interaction avec toutes les masses d'un autre sous-réseau). Le modèle de fumée développé par Vapillon et décrit dans [HLV96] présente à lui seul ces deux structures. Ce modèle est constitué de deux réseaux, l'un représentant la fumée et l'autre le milieu dans lequel le jet de fumée est injecté. Chacun des deux réseaux est composé de masses en agglomérat. Chaque masse de la fumée est en interaction avec toutes les autres, de même que pour chaque masse du milieu. La construction graphique interactive d'un tel réseau, entrant dans la catégorie de motif des agglomérats (tous les MAT sont connectés avec tous les autres) est pour le moins titanesque et peu adapté à l'objectif. Un langage permet une description relativement minimale d'un tel réseau.

Soit les n masses constituant la fumée. On peut les indexer à l'aide d'un tableau :

MAT fumee[n]

De même, on peut indexer les $n*(n-1)/2$ interactions entre ces masses.

LIA interFumee[n*(n-1)/2]

Deux boucles imbriquées permettent alors de définir le réseau de masses en interaction constituant la fumée.

```
k=1
Pour i de 1 à n-1
  Pour j de i+1 à n
    Connecter(fumee[i], interFumee[k], fumee[j])
    k=k+1
```

De la même façon, on peut décrire l'agglomérat constituant le milieu. Si ce milieu est composé de m masses, ces m masses vont alors être en interaction avec les n masses de la fumée. Soit milieu[m], ces m masses et interFumeeMilieu[m*n] les $m*n$ interactions. L'interaction entre les deux sous-réseaux peut se décrire de la manière optimale suivante :

```
k=1
Pour i de 1 à n
  Pour j de 1 à m
    Connecter(fumee[i], interFumeeMilieu[k], milieu[j])
    k=k+1
```

La figure 6 illustre la construction du réseau à l'aide de la conception textuelle. Les particules constituant la fumée sont représentées dans une couleur plus claire que celles constituant le milieu, et les liens respectivement inter-particules, inter-milieu et particule milieu en rouge, vert et bleu. Un nombre très restreint de particules est représenté car le diagramme est fait à la main, de manière à expliciter les algorithmes décrits précédemment, et nous l'avons dit, une conception graphique est mal

appropriée pour la conception d'un tel réseau avec un grand nombre de particules. En haut à gauche de la figure, les MAT sont déclarés. En haut à droite, il s'agit du début de la boucle de connexion entre les MAT constituant la fumée (pour $i=0$, $j=1$ et $j=2$). En bas à droite les fumée[0] est déjà connecté à tous les autre MAT « fumée ». Finalement, en bas à gauche, le réseau global a été obtenu.

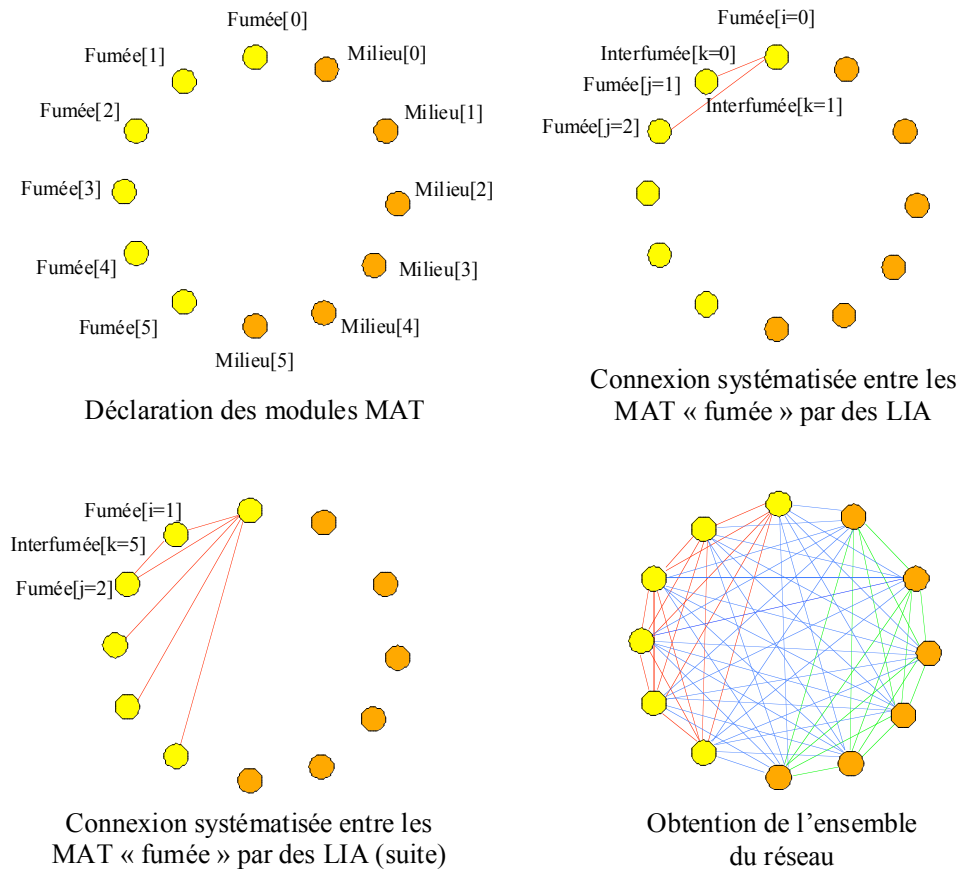


Figure 6 : construction du réseau de type agglomérat pour un modèle de fumée

D'autres cas remarquables de topologies peuvent être cités. Ainsi, la conception d'un modèle de surface illustre bien les différences fondamentales de conception entre un modèle géométrique et modèle physique particulière, et par conséquent la différence entre les outils de conception appropriés. Ainsi, si la conception graphique interactive se prête assez bien à la conception d'un modèle géométrique d'un morceau de surface, puisque on peut y définir la notion de maille et concevoir le modèle comme un ensemble de mailles dupliées, la notion de maille, en revanche, n'est pas adaptée au modèle physique. En effet, il est généralement faux d'identifier une maille et un réseau physique du type de CORDIS-ANIMA. Un objet déformable peut par exemple comporter beaucoup moins de masses que sa description ne comporte de sommets. Un bon modèle physique tentera d'ailleurs de trouver le nombre minimal de masses ponctuelles pour un comportement dynamique souhaité alors que le modèle géométrique de l'objet, modèle visuel ne fait pas la même économie de sommets. Le nombre de sommets correspondra à la finesse, de l'irrégularité de la forme de l'objet, alors que le nombre de masses correspondra plus au nombre de degrés de liberté du mouvement. De plus, une arête reliant deux sommets est un élément géométrique à différencier d'une interaction entre deux masses

ponctuelles. En considérant que la topologie du réseau CORDIS-ANIMA suit la configuration de la maille, le phénomène simulé qui en résulte peut exhiber des comportements dynamiques très particuliers et non souhaitables comme des anisotropies (cf. [BC00]). En conséquence, il peut y avoir une masse là où il n'y a pas de sommet, une interaction là où il n'y a pas d'arête.

En conséquence, différemment de la plupart d'autres outils de conception, et différemment des conceptions précédentes, le choix nominal pour la conception de topologies de réseaux CORDIS-ANIMA, s'est porté en premier lieu sur un outil de conception textuelle, qui permet de concevoir tout types de structures, et donc est bien à même de couvrir la variété et la complexité des réseaux CORDIS-ANIMA. Le choix d'une conception graphique, voire d'une génération automatique d'un réseau topologique comme dans la plupart des logiciels d'animation actuels, est un choix s'inspirant d'une vision géométrique du modèle physique trop souvent encore méconnu même par des personnes qui le pratique. Ce choix restreint énormément les possibilités créatives offertes par la flexibilité des réseaux CORDIS-ANIMA.

II.3. Conception graphique interactive

II.3.1. Une fonctionnalité complémentaire de la conception textuelle

Bien qu'un langage soit un outil permettant de concevoir tout type de structures et bien qu'il soit incontournable pour l'édition de topologies faisant intervenir un très grand nombre de modules, il est, en revanche, beaucoup moins nécessaire pour la description de réseaux simples servant à l'élaboration de modèles bien structurés (marionnette, objets rigides, etc.). La conception graphique interactive, nommée « manipulation directe » dans le domaine des interfaces homme-ordinateur, par construction graphique du réseau peut prendre dans ce cas toute son utilité. En effet, la connexion d'un module d'interaction entre deux modules de masse à l'aide de la souris dans une fenêtre 3D peut être une action de l'utilisateur plus rapide que la saisie d'une ligne de texte décrivant l'opération équivalente. Elle ne nécessite pas que l'utilisateur connaisse le nom des modules qu'il connecte en les désignant à l'aide par exemple de la souris. D'une manière plus générale, la manipulation directe est optimale pour toute construction impliquant des relations points à points, qui par conséquent sont nécessairement en nombre assez petit. En outre, la manipulation directe permet à l'utilisateur d'organiser spatialement et de lui-même son espace de travail en même temps qu'il élabore le réseau.

C'est le choix qui a été fait dans l'environnement de conception GENESIS [CC02], environnement permettant de concevoir des modèles physiques CORDIS-ANIMA pour la création de structures instrumentales et compositionnelles musicales. Or les problématiques de GENESIS, même s'il propose aussi de créer dans un premier temps la topologie d'un réseau CORDIS-ANIMA, ne sont pas les mêmes. En effet, les masses de GENESIS seront amenées à évoluer lors de la simulation dans un espace adimensionnel. L'utilisateur, place donc interactivement des masses sur un espace 2D appelé « établi » et connecte ces masses à l'aide d'interactions (ces interactions sont orientées contrairement à MIMESIS). La position de ces masses sur l'établi n'a aucun effet sur le son produit, elles peuvent être

disposées librement. Dans le cas de MIMESIS, dans un premier temps, les masses peuvent elles aussi être disposées librement, mais, puisqu'elles seront amenées à évoluer dans un espace 3D lors de la simulation, il est possible que l'utilisateur construise son réseau au regard de futurs critères spatiaux. Par exemple, pour créer une surface déformable, il pourra avoir envie de relier par des interactions viscoélastiques des masses conduites à rester proche lors de la simulation alors que les masses éloignées ne le seront pas.

En complément de la conception textuelle, l'environnement MIMESIS doit également proposer à l'utilisateur de pouvoir construire son modèle graphiquement en manipulant des modules qu'il connecte entre eux. Cette fonctionnalité n'est pas implantée à l'heure actuelle. En effet, l'ajout de la possibilité de conception graphique, qui n'est nominale qu'une commodité, soulève des problèmes complexes de double représentation du réseau (graphique et textuelle) et de tous ses sous-composants, et d'une collaboration entre ces deux représentations. Cette question complexe fait l'objet de la thèse en cours de Ali Allaoui.

II.3.2. Une nécessaire représentation graphique des réseaux CORDIS-ANIMA

Cependant, en dehors des possibilités de construction, la représentation graphique d'un réseau CORDIS-ANIMA, est néanmoins nécessaire et donc implantée de manière nominale, et cela, pour trois de ses propriétés bien connues. Tout d'abord, si la conception textuelle permet de concevoir et de manipuler tout type de réseau, elle n'en permet pas une appréhension complète. C'est pourquoi, même sans aller jusqu'à une conception graphique interactive de tout ou partie de réseau, une représentation visuelle est nécessaire, pour compléter la représentation textuelle et donner une vue d'ensemble du réseau. Une deuxième propriété inhérente à la représentation graphique réside dans les possibilités de désignation interactive, soit pour éditer, soit pour informer, de la nature des constituants ou l'édition des données quantitatives (paramètres et valeurs initiales). Ce point sera traité dans le paragraphe suivant. Enfin, une troisième propriété de la représentation graphique consiste dans le fait de disposer d'un plongement dans l'espace géométrique du réseau. En effet, une fois définie la topologie du réseau, il est nécessaire de définir des données quantitatives dont certaines sont des données spatiales, comme les positions initiales et les longueurs au repos. Ici, un plongement géométrique du réseau dans un espace 3D métré est nécessaire, pour apprécier globalement ces différentes grandeurs. Par conséquent, et avant même de disposer d'une conception graphique interactive complémentaire de la conception textuelle, une représentation graphique du réseau s'impose.

II.3.3. Quelle représentation graphique des réseaux CORDIS-ANIMA ?

Dans le paragraphe précédent, deux fonctionnalités duales de la représentation graphique ont été amenées :

- l'une permettant la représentation de la topologie du réseau. On pourrait parler de représentation graphique fonctionnelle. Un espace 2D tel que celui de GENESIS peut très bien être suffisant pour représenter un réseau.
- l'autre permettant la représentation, et éventuellement la manipulation de données spatiales 2D ou 3D (positions et longueurs).

On peut d'ores et déjà imaginer, compte tenu de la variété et de la quantité de données à fournir indépendamment (topologie du réseau, paramètres physiques de chaque composant, valeurs initiales, habillage, visualisation, simulation et vision du phénomène simulé), toutes très consommatrices d'espaces « écran », que nous sommes confronté à une optimisation de l'espace de travail. Dans ce contexte, bien que ces deux représentations graphiques soient différentes, il nous a semblé nécessaire, tout au moins pour l'instant, c'est-à-dire tant que l'expérience des usagers le tolère, de regrouper ces deux fonctions en une seule.

Quelle représentation graphique choisir alors pour les modules dans une fenêtre 3D ? En effet, les modules représentés doivent faire l'objet d'une représentation fonctionnelle, mais plongés dans un espace 3D métré. Nous avons vu qu'un module MAT est un module avec un point de connexion M. Ce point de connexion possède une entrée en force et une sortie en position. Le module LIA lui est connecté à deux MAT. Il possède deux points L qui, symétriquement aux points M, sont constitués d'une entrée en position et d'une sortie en force. La représentation minimale pour ces deux modules est donc une représentation ponctuelle dans l'espace 3D pour les modules MAT et un trait reliant deux MAT pour le LIA. La représentation ponctuelle rappelle la masse ponctuelle. Il faut cependant bien garder à l'esprit que la représentation d'un MAT est une représentation fonctionnelle et non géométrique. A ce titre, ce serait induire en erreur l'utilisateur que de lui proposer comme représentation du module MAT une sphère avec une taille fixe dans une fenêtre 3D, sphère qui semblerait grossir au fur et à mesure que l'on s'approche d'elle. Par conséquent, nous avons choisi de représenter les modules MAT par des disques dont la taille est fixe à l'écran et indépendante du point de vue de la caméra. De même l'épaisseur des traits représentant les modules LIA dans une fenêtre 3D est elle aussi fixe à l'écran. Cette représentation fonctionnelle est réalisée dans une fenêtre 3D de modélisation, fenêtre qui servira par la suite aussi à la sélection des modules pour l'édition des paramètres physiques, à l'édition des valeurs initiales des modules et à la désignation des modules pour l'habillage par les formes. La figure 7 montre un modèle de marionnette construit dans la fenêtre 3D de modélisation sous deux points de vue différents.

L'espace de représentation est ici métré (il dispose d'une échelle, de grilles, de repères). Si la taille des modules ne change pas avec le zoom, le métrage, lui suit donc le zoom. Ainsi, la place des modules dans cet espace correspondra à leurs valeurs initiales pour la simulation. Les longueurs initiales (distance initiale entre les modules dans l'espace métré) et longueurs au repos (paramètre physique de certaines interactions) seront elles aussi représentées dans cet espace.

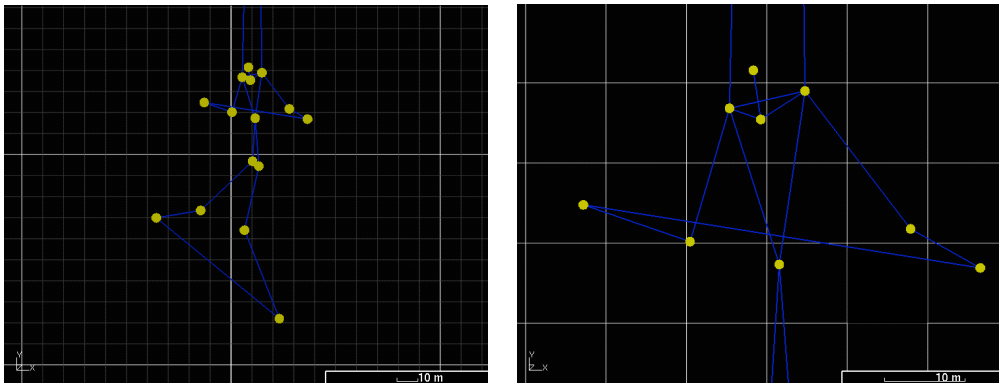


Figure 7 : Modèle de marionnette dans la fenêtre 3D de modélisation.

À gauche : le modèle global. À droite : gros plan sur la partie supérieure. La taille des modules à l'écran ne change pas (représentation fonctionnelle).

II.3.4. Dissocier espace topologique et espace 3D ?

Nous l'avons dit, nous avons fait le choix de représenter à la fois l'espace fonctionnel donnant la topologie du réseau et l'espace 3D permettant de travailler sur les valeurs initiales dans une même fenêtre 3D pour des raisons d'optimisation de l'espace de travail. Dès que les systèmes permettront efficacement la gestion multi écran, en particulier pour donner toute sa dimension à l'espace de rendu final, nous serons naturellement amenés à dissocier les représentations graphiques fonctionnelles et les représentations 3D. Cela ne sera pas alors sans poser de problèmes, mais apportera aussi de nouvelles possibilités de conception interactive.

Le premier problème sera de réfléchir au choix de la dimension de l'espace fonctionnel. La pratique des utilisateurs de GENESIS nous apprend qu'un espace 2D est suffisant pour la conception de réseaux CORDIS-ANIMA pour le son. Or il n'est pas certain que la représentation 2D d'un réseau complexe conduit à être plongé dans un espace 3D soit la solution la plus adaptée. Or si le choix se portait sur un espace 3D, la conception graphique se verrait plus compliquée. En effet, à quelle profondeur placer les modules à la main lorsque l'utilisateur interagit à l'aide d'une souris 2D ? Un second problème, si l'on fait le choix de deux espaces, est d'offrir des outils permettant de faire le lien entre un module représenté dans l'espace fonctionnel et son plongement dans l'espace géométrique. Le module se trouvant représenté à deux endroits différents, il faudra s'assurer que l'utilisateur puisse l'identifier en même temps dans les deux fenêtres parmi les autres.

Ces difficultés résolues, l'utilisateur disposera alors d'un environnement plus puissant pour la modélisation de réseaux CORDIS-ANIMA puisque cet environnement séparera la phase de préstructuration qualitative, de la phase d'édition des conditions initiales. L'environnement portera donc de manière plus forte le concept propre de la modélisation CORDIS-ANIMA. Si nous reprenons l'exemple du réseau CORDIS-ANIMA pour la modélisation de fumée et représenté fonctionnellement par la figure 6. L'utilisateur pourra conserver la représentation en cercle des MAT bien adaptée au travail sur le réseau topologie tout en plongeant les masses ponctuelles dans un espace 3D différent où

elles auront une disposition différente et répondant au phénomène visuel à modéliser (masses en ligne pour former des jets par exemple).

Au final, un espace de représentation fonctionnelle pur pourra s'affranchir de la représentation de tous les modules à l'écran et par exemple les empaqueter dans des modules plus particuliers avec un grand nombre de points de connexions. Toujours dans notre exemple, la représentation fonctionnelle minimum de notre réseau de masses « fumées » suivant un motif « d'agglomérat » pourra être représenté par un seul module. C'est d'ailleurs une pratique déjà sommairement en cours dans le logiciel GENESIS qui propose un module CEL, cellule constituée d'une masse fixe et d'une masse mobiles reliées par une interaction viscoélastique permettant de simuler une sinusoïde amortie donnant le phénomène sonore, le plus minimaliste (une note avec une seule fréquence s'atténuant dans le temps).

II.4. Manipulation de l'organisation du réseau

II.4.1. Une labellisation et non un graphe de scène

Les paragraphes précédents adressent deux des premiers aspects liés à la complexité du modèle : le choix d'outils de conception et le choix d'outils de représentation. Par ces outils, il est possible de concevoir et de représenter n'importe quel type de réseau CORDIS-ANIMA. Cependant, la construction même d'un tel réseau ne peut pas être une opération linéaire. Il est alors nécessaire de disposer d'autres outils qui permettront de concevoir et de manipuler ce réseau par parties, d'accéder à ses composantes, de travailler par zones homogènes, etc.

Un problème quelque peu similaire s'est déjà présenté dans le milieu des années 90 pour le rendu de scènes géométriques complexes. Les développeurs n'avaient alors en effet à leur disposition que des bibliothèques de création de formes élémentaires et de rendu. Silicon Graphics met alors le premier à disposition de ces développeurs Open Inventor [Wer93], une bibliothèque logicielle permettant de décrire la scène complexe à l'aide d'un graphe acyclique. De nombreuses bibliothèques suivront, qui toutes utiliseront le concept de graphe orienté acyclique pour décrire une scène, tirant ainsi profit de spécificités des scènes géométriques qui sont d'une part la contiguïté spatiale des éléments (descriptible sous la forme de maillages) et leur relation unilatérale (descriptible sous la forme d'arborescences univoques d'appartenance).

Le cas des réseaux CORDIS-ANIMA est sensiblement différent. La structure topologique des réseaux CORDIS-ANIMA est spécifique, en ce sens qu'elle ne peut être limitée par des considérations exogènes à sa syntaxe, comme par exemple la contiguïté spatiale propre à un objet géométrique ou une structuration arborescente d'éléments dépendant unilatéralement les uns des autres. Tout élément peut a priori être connecté à tout autre, pouvant aller jusqu'à un réseau à interconnexion bilatérale complète. Il est par conséquent impossible de définir de manière unique et définitive des dépendances entre des

sous-parties du réseau. Ces dépendances ne pourront être créées qu'à la discrétion du modélisateur, de manière tout à fait libre et flexible, pour l'aider localement dans sa tâche de modélisation.

Nous allons illustrer ces différents points dans l'exemple qui suit appelé « Boules suspendues ».

Soit le réseau de la figure 8 simulant le comportement de deux boules suspendues chacune à deux fils qui vont entrer en collision. Les fils sont constitués d'une chaîne de masses en interactions viscoélastiques. Ils sont fixés à des masses immobiles servant de points de fixation. On peut alors définir des sous-ensembles du réseau pour chacun des fils. Mais, de la même manière, on peut définir l'ensemble de tous les fils. On est bien dans ce cas face à une structure hiérarchique arborescente allant du modèle global, à l'ensemble des fils composé de quatre fils, similaire aux structures hiérarchiques usitées en représentations géométriques de scènes. Qu'en est-il cependant des éléments « boules suspendues » ? Chacune des boules peut être prise dans un ensemble formé des deux fils qui les suspendent et d'eux mêmes. On est ici dans une structuration hiérarchique. Mais ces boules sont également en interaction entre elles par des butées viscoélastiques. Elles constituent donc également une entité, au même titre que les fils. Il est alors légitime d'inclure les deux boules dans un même ensemble. Par conséquent, chaque boule peut légitimement appartenir alors à deux ensembles ayant des parties disjointes. Nous ne sommes plus dans le cas d'une structure d'arbre.

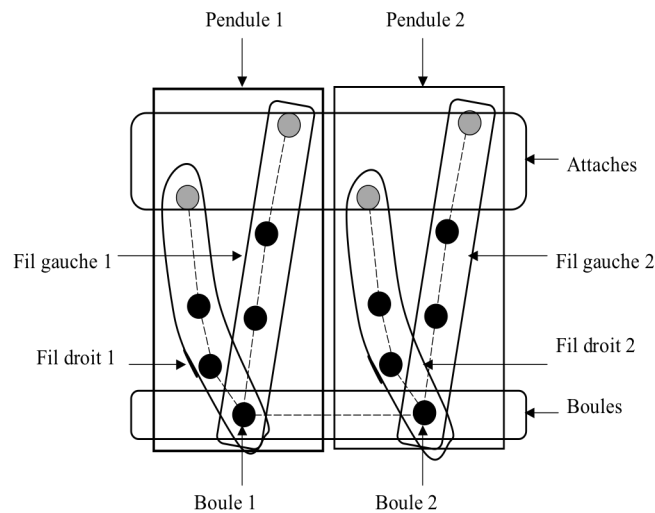


Figure 8 : Structuration possible d'un modèle CORDIS-ANIMA de boules suspendues

Pour construire le modèle l'utilisateur peut aussi choisir de créer un premier pendule, « pendule 1 », puis de le dupliquer en « pendule 2 ». Par conséquent, il est aussi intéressant de considérer qu'une structuration peut aussi permettre de dupliquer des réseaux. La figure 9 présente une telle structuration du réseau ainsi construit :

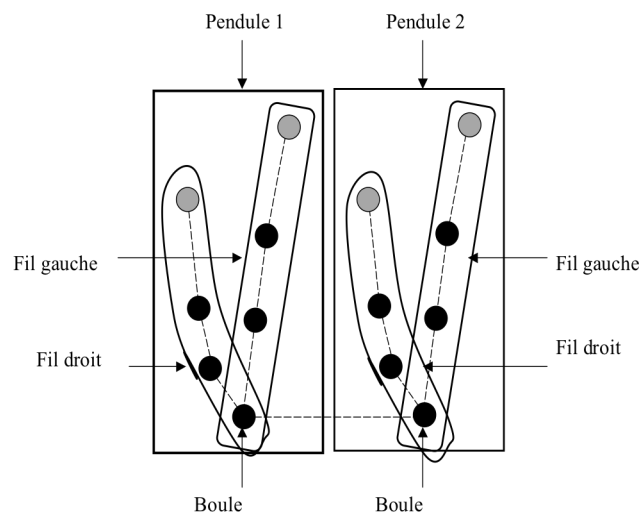


Figure 9 : Structuration possible d'un modèle CORDIS-ANIMA de boules suspendues construit par duplication de réseau.

Nous voyons, à partir de cet exemple, que la structure de description que l'on souhaiterait mettre en place fait intervenir des sous-ensembles d'éléments de même nature alors que les graphes de scènes, eux, font intervenir des éléments de nature différente (éléments géométriques, matériaux, lumières...). De plus, très souvent, les graphes de scène possèdent une structure hiérarchique d'arbre. Des graphes de scènes peuvent alors notamment servir au traitement cinématique d'un objet articulé. La structure d'arbre ne correspond pas à une description fonctionnelle des parties d'un réseau. On peut vouloir faire dépendre un ensemble de modules de plusieurs ensembles. Un module lui-même peut avoir plusieurs fonctions dans le modèle. L'idée même de localisation spatiale bien définie que l'on peut retrouver dans la description d'objet de topologie fixe n'est plus valable dans le cas général d'un réseau quelconque.

II.4.2. Mise en œuvre de la structure de donnée pour la labellisation

Pour organiser le réseau, il est donc ici question de nommer de diverses manières à l'aide de différents « labels » des modules ou des ensembles de modules, sous-parties de réseau, idée différant de la définition d'un graphe de scène. Il reste qu'une structure de graphe acyclique orienté qui n'est pas forcément un arbre (un nœud peut avoir plusieurs pères) est une description adaptable à notre problème si elle est utilisée de manière sous-jacente. Puisqu'elle bénéficie de nombreux exemples d'implantations, cela en fait une structure efficace.

L'outil de structuration au sein de l'environnement MIMESIS peut donc être basé sur un tel graphe. Les nœuds de ce graphe sont de deux types : des modules CORDIS-ANIMA ou des sous-ensembles de réseau CORDIS-ANIMA. Un module est alors un nœud terminal du graphe alors qu'un sous-réseau en

est un nœud interne. Si on labellise chaque lien orienté du graphe, les modules et les sous-réseaux peuvent non seulement être désignés par des noms automatiquement donnés (dans la figure 10, les noms précédés de #) mais être aussi désignés à l'aide du label résultant du parcours dans le graphe allant de la racine au module concerné, noms construits alors par l'utilisateur. Un tel label ne peut exister que parce que la structure du graphe est acyclique. De plus, pour que le parcours ait un début, il faut convenir que le graphe ait une racine unique que l'on peut définir à partir d'un graphe orienté.

Pour dupliquer un sous-réseau, on peut faire appel à un de ses labels. Le sous-réseau résultant aura alors le premier père désigné par le label du sous-réseau d'origine. Il conviendra alors seulement de donner un nom au lien ainsi créé. La structure acyclique du graphe nous permet de dupliquer en outre un sous-réseau sans avoir à renommer les liens qui en sont issus. La figure 10 nous donne un exemple de graphe de structuration pour le réseau présenté précédemment. Elle est suivie d'un tableau recensant l'ensemble des labels pour certains modules et sous-réseaux. Le réseau a été obtenu, dans l'exemple suivant, en dupliquant successivement FILGAUCHE en FILDROIT et PENDULE 1 en PENDULE 2. Mais on l'aurait tout aussi bien obtenu en créant FILGAUCHE, en le faisant dépendre de PENDULE 1 et en dupliquant PENDULE1.FILGAUCHE en FILDROIT qui aurait alors dépendu de PENDULE 1.

La structure de graphe sous-jacente à la labellisation proposée par la figure 8 peut donc être :

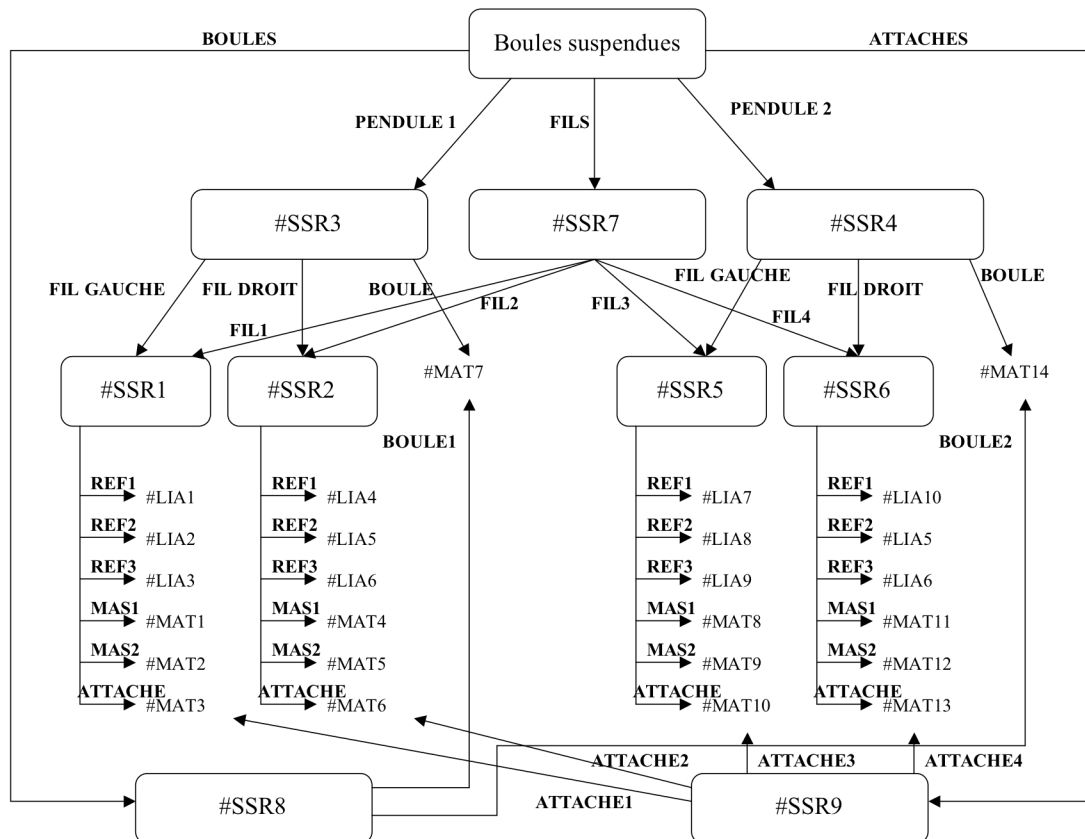


Figure 10 : Graphe de structuration d'un modèle CORDIS-ANIMA de boules suspendues.

L'utilisateur a bien dans ce cas plusieurs manières d'appeler certains modules comme le montre la figure 11 :

Module ou sous-réseau	Label possible
#SSR1	PENDULE1.FILGAUCHE FILS.FIL1
#SSR5	PENDULE1.FILGAUCHE FILS.FIL3
#MAT3	PENDULE1.FILGAUCHE.ATTACHE ATTACHES.ATTACHE1
#MAT7	PENDULE1.BOULE BOULES.BOULE1

Figure 11 : Liste de labels possibles pour certains modules et sous-réseaux.

L'environnement MIMESIS dispose d'une possibilité de labellisation des éléments du réseau CORDIS-ANIMA. Or cette structure a fait l'objet de plusieurs conceptions successives au cours de nos travaux de thèse pour notamment être incluse dans un langage de modélisation commun aux logiciels GENESIS et MIMESIS et développé au sein du groupe ACROE/ICA. Nous avons pris part aux phases de spécifications de ce langage sur la question de la labellisation bien que notre version actuelle de MIMESIS ne dispose pas de ce langage plus évolué.

III. Phase « PSQN » : Edition des paramètres physiques

III.1. Une phase essentielle dans la constitution du modèle

La structure actuelle proposée de l'environnement de création sépare clairement la conception de la topologie du réseau CORDIS-ANIMA de la donnée de ses paramètres quantitatifs.

Rappelons tout d'abord ce que nous entendons précisément par paramètres physiques. Les paramètres physiques ne sont pas l'ensemble des variables contenues dans le programme, ni même l'ensemble des variables nécessaires à la simulation. Il s'agit strictement des constantes physiques associées à chaque module. Des exemples en sont la valeur de l'inertie d'un module MAT élémentaire, la valeur de l'élasticité et/ou de la viscosité linéaire pour un module LIA élémentaire, mais également, les différentes valeurs des conditions et dispositions entrant dans la définition des modules d'interaction non linéaires. Ce ne sont en aucun cas les données qui circulent dans le réseau, données de forces, de vitesses, de positions, d'accélération, qui sont en entrée et en sortie des modules et qui portent le nom usuel de variables d'état.

Compte tenu de l'importance de la valeur des paramètres dans un système dynamique, la donnée et l'expérimentation de ces valeurs doivent pouvoir être faites avec le plus grand soin. En effet, dès le premier abord, ce sont les paramètres qui définiront les zones de convergence et de divergence du système dynamique. Ces zones sont faciles à définir dans des cas très simples, d'un système du second ordre par exemple [Inc96]. Mais elles ne sont pas calculables a priori dans un réseau de topologie et de taille quelconque, en particulier lorsque ce réseau contient des non-linéarités. De plus, ce sont les paramètres qui définissent in fine le comportement du réseau. Or comme dans tout système dynamique, plusieurs types de comportements différents, c'est-à-dire appartenant à des états comportementaux non nécessairement contigus, peuvent émerger d'une même structure topologique. Ainsi, des bifurcations peuvent apparaître, comme des bifurcations de phases dans les systèmes réels. Par conséquent, la classe de comportements générés dépend autant de la structure topologique que des zones de paramètres physiques. Ceci est à mettre en vis-à-vis de la très grande généricité du formalisme CORDIS-ANIMA, par rapport à des représentations formelles plus proches d'un type de comportement donné. C'est cette généricité qui permet de parler d'outil de création, l'artiste ne pouvant se contenter d'une librairie de phénomènes a priori typés. En contrepartie, c'est à l'environnement – et non à la formule - de lui permettre d'explorer correctement tous les

comportements possibles, leurs limites, leurs similarités, leur dissemblances. Dans cette exploration comportementale, la valeur grossière des paramètres pour circonscrire les zones de convergence/divergence mais également bien souvent leur précision parfois à 10^{-6} ou plus, pour catégoriser des états différents, sont d'égale importance. Enfin, bien que le nombre de paramètres par module soit restreint, le nombre total de paramètres physiques à gérer peut devenir vite très important, augmentant la difficulté des questions dont on vient de parler.

Pour toutes ces raisons, qui sont intimement liées au modèle physique et plus particulièrement au modèle physique masses - interactions, il est donc nécessaire de proposer l'expérimentation paramétrique comme une activité à part entière. L'importance de cette phase est souvent sous-estimée au profit de la formule, de ses procédés de résolution numérique, et dans le cas de CORDIS-ANIMA au profit de la conception de la topologie du réseau. C'est mal connaître à la fois le principe et la réalité du modèle physique que de sous-estimer le rôle des paramètres physiques dans l'obtention d'une classe de comportements.

III.2. Les différents type de paramètres physiques

On peut classer en deux grandes catégories les paramètres physiques du modèle. Ceux qui ne réfèrent pas à la spatialité et ceux qui y réfèrent. Pour chacun de ces deux types les procédures d'édition auxquelles nous allons apporter une réponse sont soumises à des problématiques différentes.

III.2.1. Les paramètres non spatiaux

Les paramètres non spatiaux sont ceux qui relient des variables intensives (forces, etc.) avec des variables extensives (positions, vitesses, accélération). Ce sont les inerties, les élasticités, les viscosités. Leur dimension dans les équations sera du type $[N/m]$, $[N/(m/T)]$, $[N/(m/T^2)]$.

L'inertie relie les forces avec les accélérations suivant la loi fondamentale de la dynamique énoncée par Newton :

$$M^* a = \sum \vec{F}$$

Dans le formalisme CORDIS-ANIMA, le paramètre d'inertie est porté par les modules MAT et entre dans le calcul des nouvelles variables de position pour ces modules en fonctions de la somme des forces en entrée. Dans un schéma de discrétisation faisant intervenir des accélérations avancées, la nouvelle position sera donnée par :

$$\overrightarrow{X_{n+1}} = \frac{\overrightarrow{F_n}}{M^* F_e^2} + 2\overrightarrow{X_n} - \overrightarrow{X_{n-1}}$$

L'élasticité K , corrèle des forces d'interaction potentielle avec la distance entre deux masses ponctuelles :

$$\vec{f}_{e_{1 \rightarrow 2}} = K * (d\|\vec{x}_1 - \vec{x}_2\| - L)\vec{u}_{1 \rightarrow 2}$$

Une telle interaction fait que la distance entre deux masses ponctuelles tend à être ramenée à une valeur de L . Plus K est important, plus on dira que l'interaction est dure ou rigide. Il s'agit d'un paramètre d'interaction qui sera donc porté par les modules de type LIA dans le formalisme CORDIS-ANIMA.

La viscosité Z , corrèle des forces d'interaction dissipatives avec la vitesse relative entre deux masses ponctuelles, suivant :

$$\vec{f}_{v_{1 \rightarrow 2}} = Z * \frac{d\|\vec{x}_1 - \vec{x}_2\|}{dt} \vec{u}_{1 \rightarrow 2}$$

Une telle interaction fait que les vitesses entre deux masses ponctuelles tendent à devenir les mêmes vectoriellement. Plus Z est important, plus on dira que l'interaction est visqueuse ou amortie. Il s'agit d'un paramètre d'interaction qui sera donc porté lui aussi par les modules de type LIA dans le formalisme CORDIS-ANIMA.

Les valeurs de ces paramètres non spatiaux peuvent varier du « tous semblables » au « tous différents » suivant le type de modèle mis en œuvre. Le comportement dynamique d'un objet largement homogène modélisé par un grand nombre de masses en interaction pourra comporter des paramètres physiques non spatiaux relativement semblables. Au contraire, le réseau modélisant un objet hétérogène pourra présenter des paramètres différents.

III.2.2. Les paramètres spatiaux

Les paramètres spatiaux sont les seuils en distance et les distances au repos intervenant dans les algorithmes des interactions. Ce sont donc des valeurs scalaires auxquelles sont comparées les distances entre masses ponctuelles ou qui peuvent entrer directement dans le calcul des forces à appliquer.

La distance au repos L de l'interaction élastique du premier ordre est telle qu'aucune force ne s'applique entre deux masses ponctuelles lorsque leur distance est égale à la valeur de L comme le montre l'expression de la force appliquée :

$$\vec{f}_{e_{1 \rightarrow 2}} = K * (d\|\vec{x}_1 - \vec{x}_2\| - L)\vec{u}_{1 \rightarrow 2}$$

Les seuils en distance quant à eux délimitent des espaces en distance. L'expression de la force d'interaction est différente pour chacun de ces espaces. Un tel seuil intervient par exemple dans l'algorithme de la plus simple des interactions non linéaires qu'est la butée élastique. Si la distance entre deux masses ponctuelles est inférieure à un seuil S , une interaction du premier ordre s'applique entre les deux masses. Les deux masses butent entre elles. Si, en revanche, cette distance est supérieure à S , aucune force ne s'applique. L'introduction de seuils en distance permet donc par exemple l'exclusion mutuelle d'espaces.

Dans tous les cas, les paramètres physiques spatiaux ont une influence sur la configuration spatiale des masses au cours de la simulation. Ils ont donc un lien étroit avec une éventuelle géométrie construite et contrôlée par la simulation pour créer des objets visuels en mouvement. Leur valeur peut dépendre d'une géométrie prédéfinie pour un objet tout autant que les paramètres d'une géométrie contrôlée par la simulation peuvent avoir un lien avec ces paramètres comme nous le montrerons dans le paragraphe I.2.2. du chapitre 4 de ce manuscrit intitulé « la forme conditionnée par les paramètres spatiaux des interactions du modèle masses-interactions ». Faisant référence à la spatialité du modèle CORDIS-ANIMA, ces paramètres peuvent aussi avoir un lien avec les valeurs initiales de la simulation. Évoluant potentiellement du « tous semblables » au « tous différents », la similitude influera moins sur le caractère homogène ou hétérogène du comportement dynamique que sur l'organisation spatiale du modèle.

III.3. Mise en œuvre de procédures d'édition des paramètres physiques

La distinction entre deux types de paramètres physiques, spatiaux et non spatiaux, pose des problèmes différents particulièrement lorsque le nombre de ces paramètres devient conséquent ce qui est plutôt la norme que l'exception puisque le nombre de paramètres est directement lié au nombre de modules. Dans le cas des paramètres non spatiaux, les questions adressées seront relatives à l'édition d'un grand nombre de paramètres potentiellement tous différents ou similaires. Dans le cas des paramètres spatiaux, la prise en compte de la spatialité du modèle se surajoute à ces problématiques. Cela nous conduit à définir plusieurs procédures complémentaires pour l'édition des paramètres physiques.

Éditer des paramètres tous différents

Lorsque les paramètres doivent être tous différents, il est généralement nécessaire de les éditer à un moment ou à un autre à la main. Une liste à plat des paramètres, par exemple dans un tableau, facilite une vue d'ensemble. Mais le lien entre une liste à plat de tous les paramètres de tous les modules et les modules correspondant dans le modèle est difficile à faire. Pour renforcer ce lien, on peut imaginer offrir à un utilisateur la possibilité de naviguer entre la représentation graphique du réseau topologique du modèle et le tableau des paramètres physiques en permettant une désignation par sélection interactive de modules particuliers dans l'un ou l'autre des deux environnements avec une mise à jour automatique de cette sélection dans l'environnement où elle ne se déroule pas. De nombreux outils d'édition de liste de paramètres existent en outre comme le tableur Excel. Il est intéressant de

permettre à un environnement de conception de prendre en entrée les sorties de ces outils au moins par l'intermédiaire de l'importation de fichiers de paramètres créés à l'extérieur de l'environnement.

Éditer des paramètres par zone homogène

Il se peut que l'on affecte à un grand nombre de modules la même valeur de paramètre. Cela définit des zones dans le modèle pour lesquelles ces paramètres sont homogènes. Ce type d'édition requiert des outils de sélection d'un ensemble de modules pour donner à tous ces modules les mêmes valeurs de paramètre. La sélection interactive dans la fenêtre 3D de modélisation permet une telle pratique notamment par la possibilité de sélectionner tous les modules d'un certain type apparaissant dans une fenêtre définie à l'aide de la souris. Or ces zones homogènes ne sont pas nécessairement des zones bien définies dans l'espace de modélisation 3D. Des interactions éloignées dans cet espace de modélisation peuvent constituer un même ensemble aux vues du critère d'homogénéité des paramètres. Dès lors, il faudra plutôt faire appel à des outils de sélection d'un grand nombre de modules par l'intermédiaire d'outils travaillant sur une organisation structurale des réseaux comme, par exemple, la labellisation définie dans la partie II.4 « Manipulation de l'organisation du réseau » peut le permettre.

Définition automatique des paramètres spatiaux

Les paramètres spatiaux peuvent dépendre des valeurs initiales du modèle. Cela est d'autant plus vrai lorsque l'on souhaite que l'objet simulé soit à l'équilibre en début de simulation. Pour réaliser ce projet, il est nécessaire que les paramètres de distance au repos des interactions aient pour valeur la distance initiale entre les masses ponctuelles reliées par les interactions dont ils sont les paramètres.

Que l'on souhaite l'équilibre ou pas, cette distance doit être calculée et accessible à l'utilisateur pour qu'il puisse l'utiliser s'il le souhaite comme valeur de paramètres spatiaux.

Édition de paramètres à partir de règles de calcul

Il est possible, en outre, d'imaginer que les paramètres soient définis par une règle de calcul comme une valeur aléatoire dans une plage donnée. Dans ce cas, il faut faire appel à un langage. Or nous n'avons pas souhaité offrir à l'utilisateur une telle fonctionnalité car cela fait difficilement sens pour les paramètres physiques dont il est rarement possible de donner une formule permettant de les déterminer. L'activité de paramétrisation est bien au contraire une activité d'ajustement fin des paramètres qui a peu avoir avec des formules. Dans certains cas très particuliers, de telles formules existent notamment lors de la détermination des valeurs d'un paramètre de viscosité permettant d'obtenir un amortissement critique lorsque le modèle est simple (uni-dimensionnel, peu couplé) comme peut le montrer Incerti dans [Inc96]. L'édition à partir de formules ne doit donc pas être écartée mais introduite à un utilisateur novice avec la plus grande précaution.

IV. Édition des variables d'état initiales

IV.1. Variables extensives et variables intensives

Nous avons distingué, dans la partie précédente, les paramètres physiques de variables physiques d'état. Les systèmes dynamiques sont des systèmes sensibles aux conditions initiales, c'est-à-dire aux valeurs initiales de ses variables physiques. Il faut donc initialiser ces variables physiques comme il convient pour le phénomène à obtenir. Dans le cadre des modèles masses - interactions, ces variables sont de deux types : les variables extensives (positions et vitesses) et les variables intensives (forces).

En ce qui concerne les variables extensives, nous pourrions proposer à l'utilisateur soit la possibilité de rester dans une vision « positions » et lui donner, comme variables initiales à définir, les positions et les positions retardées, soit lui proposer de définir les conditions initiales en terme de positions et de vitesses initiales. Bien que les positions retardées interviennent directement dans l'algorithme des modules et bien que cette notion se prête bien à la compréhension des trajectoires, donc à la compréhension d'un mouvement, il nous a semblé préférable, pour des questions de facilité de compréhension des conditions initiales, c'est-à-dire avant que le mouvement ait commencé, de manipuler des vitesses initiales plutôt que des positions retardées. En effet, la vitesse, qui se définit comme la différence entre la position et la position retardée dans le schéma de discrétisation « retard », usuellement appelé « schéma d'Euler », est en effet une variable dont il est plus facile de se faire une représentation mentale, tout au moins en début de mouvement, et qui peut se représenter par un vecteur. On différencie alors, de manière catégorielle, deux variables qui auraient appartenu à la même catégorie de représentation (le point dans l'espace) si le choix avait été fait de garder la position retardée comme variable apparente dans l'environnement MIMESIS.

En ce qui concerne les variables intensives, c'est-à-dire dans CORDIS-ANIMA, uniquement les forces, il est délicat de laisser à l'utilisateur la possibilité de déterminer des valeurs pour celle-ci au début de la simulation. En effet, l'intensité d'une force est une donnée difficile à déterminer. De surcroît, le modélisateur n'aura pas à travailler avec cette notion lors de son élaboration du modèle CORDIS-ANIMA et pendant la simulation puisque les forces résultent des interactions et sont donc entièrement déterminées à chaque instant de la simulation par les positions et vitesses des masses ponctuelles en interaction ainsi que des valeurs des paramètres physiques de ces interactions. Or, si les positions initiales, les positions retardées initiales (déduites à partir de la donnée des vitesses initiales) ainsi que les paramètres physiques sont donnés par l'utilisateur, il est parfaitement possible de déduire les forces résultantes de l'interaction en début de simulation directement par le calcul d'un pas de simulation des algorithmes des LIA.

IV.2. Les trois niveaux dans l'édition des positions initiales

L'édition des valeurs initiales des variables de la simulation est une opération radicalement différente de celle des paramètres physiques. La métrique qu'il faut définir pour l'utilisateur est en effet, dans le cas de valeurs initiales, une métrique spatiale (des mètres) absolue (des positions et non des distances) et spatio-temporelle (des mètres par seconde). De plus, si les paramètres physiques peuvent être les mêmes pour certains modèles, si cela peut encore être vrai pour les valeurs initiales des vitesses dont nous parlerons à la fin de cette partie, les valeurs initiales des positions sont majoritairement différentes pour chaque module. La différence essentielle réside dans le fait que ces variables sont des valeurs vectorielles et non scalaires, ce qui fait intervenir nécessairement un problème de représentation de nature spatiale.

C'est donc la seconde fois que la spatialité intervient dans ce chapitre, la première fois étant le cas des paramètres physiques spatiaux. La problématique n'est maintenant plus de définir des seuils et des distances mais des positions dans un espace 2D ou 3D avec potentiellement un très grand nombre de ces positions. Dans un tel contexte, la saisie de toutes les valeurs dans un tableau de valeurs alphanumérique devient une tâche impossible dès que leur nombre explose. La mise en place d'outils interactifs de placement dans l'espace est donc nécessaire. Plus précisément, il va s'agir de placer dans un espace 3D de modélisation, des modules MAT dont les positions correspondront aux valeurs initiales de position de ces modules.

Dans ce cadre, il est alors possible de définir trois niveaux de pratique du placement. Ces trois niveaux ont chacun leur propriété et leur utilité propres. Ils se succèdent généralement. Il est néanmoins possible de revenir sur l'un ou l'autre de la même manière que les différentes phases de la construction globale d'un modèle ne sont pas des phases strictement successives :

- L'organisation grossière des MAT dans l'espace de modélisation permet d'ordonner grossièrement dans l'espace 3D les modules MAT. C'est un préambule souvent nécessaire pour travailler plus en précision sur la position de ces modules. Il s'agit de mettre de l'ordre dans les modules avant de commencer la seconde phase.
- L'organisation complexe des MAT permet de placer de manière plus précise ces modules suivant des structures spatiales complexes.
- L'affinage des positions est un niveau d'édition des positions initiales essentiel. Il peut constituer à travailler à des niveaux de précision élevés par rapport à l'échelle générale de l'organisation des MAT. Les réseaux CORDIS-ANIMA sont en effet des systèmes très sensibles aux conditions initiales et une petite modification dans la valeur d'une position initiale pourra amener le réseau à se comporter de manière totalement différente au cours de la simulation. Cela peut être par exemple l'adjonction d'un infime déséquilibre par rapport à une position d'équilibre instable.

IV.3. Similitudes et différences avec un logiciel de modélisation de formes

La tentation est grande de prendre exemple sur les logiciels de modélisation de forme 3D, qui gèrent depuis plusieurs années et de manière interactive l'édition d'un grand nombre de positions, pour réimplanter ces fonctionnalités dans le cadre de l'édition des positions initiales des modules MAT de CORDIS-ANIMA. Or la réimplantation de ces fonctionnalités est une opération lourde, puisque les logiciels de modélisation de formes 3D sont de très gros logiciels. Mais des considérations plus profondes posent des limites à la tentation de réimplanter cet éventail de fonctionnalités issues des environnements interactifs de la synthèse de formes. La précision de la position d'un sommet ou d'un point de contrôle d'une forme n'est pas aussi sensible dans le cadre d'une modélisation géométrique que dans le cadre de la simulation de réseaux CORDIS-ANIMA. En effet, une variation de position qui ne se voit pas à l'œil a peu d'intérêt pour la modélisation de formes. En revanche, cela peut avoir, comme nous l'avons déjà dit, un impact décisif sur le mouvement généré par la simulation d'un réseau CORDIS-ANIMA. En outre, l'adjonction d'un grand nombre des fonctionnalités issues de la synthèse de forme peut déplacer le problème de la modélisation du mouvement à l'aide des réseaux CORDIS-ANIMA vers un problème de modélisation de forme qui est un problème radicalement différent du premier. N'oublions pas qu'il s'agit ici de ne donner que des valeurs d'état extensives initiales pour un système dynamique. La plupart des outils de la synthèse de forme incluent quant à elle l'édition de positions de points dans une problématique plus vaste de gestions des formes : recollements, découpages, raffinements... Il nous faut remarquer de plus que les caractéristiques topologiques des réseaux CORDIS-ANIMA sont différentes de celles des formes géométriques qui peuvent être mises en mouvement en contrôlant ces formes à l'aide des sorties d'une simulation CORDIS-ANIMA. Il est à noter que le nombre de masses est bien souvent inférieur au nombre de sommets d'une maille lorsque c'est une maille dont l'évolution est contrôlée au cours du temps.

Ainsi, plutôt que de faire rentrer un logiciel de modélisation de formes dans un logiciel de modélisation de réseaux CORDIS-ANIMA, il semble plus judicieux d'utiliser les sorties de ces logiciels comme une entrée possible de l'activité d'édition des positions initiales. Ce faisant, la question de la difficulté de la gestion d'un grand nombre de positions n'est que partiellement réglée. Certaines fonctionnalités relatives à l'édition de la spatialité doivent pourtant trouver leur place notamment pour permettre l'organisation grossière des MAT et la saisie fine des positions initiales. La méthode la plus rudimentaire pour saisir des valeurs initiales est de toutes les saisir manuellement, par exemple à l'aide d'un tableau des valeurs initiales avec des champs de saisie de valeur. Nous voyons difficilement d'autres méthodes possibles pour une telle activité.

Les deux parties suivantes discutent en revanche des fonctionnalités nécessaires à l'organisation grossière et à l'organisation complexe des positions initiales des modules MAT.

IV.4. Fonctionnalités pour l'organisation grossière des positions initiales

IV.4.1 Manipulation directe

Les concepts de manipulation directe pour le placement et le déplacement dans un espace 3D n'ont rien de nouveau. Depuis l'apparition des interfaces interactives fondées sur le principe du *what you see is what you get*, la plupart des logiciels de synthèse et d'animation de formes 3D proposent des outils de placement interactif dans un univers 3D. Ce type de fonctionnalité est bien adapté pour l'organisation grossière de points dans l'espace et convient donc à notre problème.

Il peut s'agir notamment :

- D'un déplacement d'un MAT ou d'un ensemble de MAT sélectionnés, à l'aide de la souris, déplacement s'effectuant dans le plan de l'écran : un tel déplacement agit potentiellement sur plusieurs coordonnées des MAT si le plan de l'écran n'est pas orthogonal à l'un des trois axes du repère de l'espace. Une telle fonctionnalité n'est efficace que dans le cadre d'un déplacement grossier qui peut être utile lorsque l'utilisateur souhaite organiser son espace.
- De translations d'un ou plusieurs MAT et de rotations par rapport à un axe, un point : le déplacement en translation et en rotation sont deux fonctionnalités plus précises de positionnement et de déplacement dans l'espace d'éléments géométriques. En effet, ces méthodes ne dépendent pas entièrement de la position de l'observateur puisque ces déplacements sont contraints. Mais il s'agit toujours d'une opération grossière par rapport à la saisie alphanumérique.
- De redimensionnement : la modification globale de la dimension d'une forme est un outil classique des environnements de modélisation de forme. Mais contrairement à ces environnements, le redimensionnement des positions initiales d'un réseau CORDIS-ANIMA a une incidence sur le comportement du modèle lors de la simulation. Prenons un modèle fait de deux masses en interaction élastique dans une configuration spatiale initiale telle qu'elles sont à l'équilibre au début de la simulation. Un redimensionnement relatif des positions initiales de ces masses va les mettre nécessairement dans un état de déséquilibre. Pour que le redimensionnement n'ait pas d'incidence sur le comportement de l'objet simulé, il convient donc de modifier les paramètres physiques spatiaux suivant la même échelle. Il faut aussi modifier les vitesses initiales suivant cette échelle. Or même, une fois prises toutes ces précautions, rien n'assure que les comportements dynamiques générés par la simulation soient nécessairement les mêmes à deux échelles différentes puisque le codage des nombres n'est pas nécessairement semblable à deux échelles de valeurs différentes et que ce codage influe sur le comportement de la simulation sensible aux conditions initiales.

IV.4.2 Placement initial automatique suivant une configuration spatiale prédéterminée

Les fonctionnalités de manipulation directe des positions initiales sont des fonctionnalités nécessaires mais qui ne résolvent pas entièrement la gestion d'un grand nombre de positions initiales. Il est nécessaire de définir aussi un certain nombre de fonctionnalités de placement initial automatique qui permettent d'automatiser certaines actions de l'utilisateur en plaçant initialement les modules MAT dans une configuration spatiale donnée. Il existe une infinité de possibilités, de la configuration spatiale simple à la plus élaborée. Mais de la même manière que les logiciels de création de forme 3D proposent un ensemble de primitives géométriques dont l'assemblage et la déformation permettra la création de formes complexes, il est possible de proposer un ensemble de configurations simples permettant de construire des configurations plus complexes en combinant les fonctionnalités tout en évitant d'en proposer un trop grand nombre qui risque de porter l'attention sur une activité de définition de configurations spatiales complexes calquée sur la synthèse de forme, activité différente de la modélisation par réseau CORDIS-ANIMA.

Les deux configurations les plus simples sont la configuration en ligne et la configuration en cercle. De ces deux configurations en dérivent d'autres lorsqu'on augmente la complexité. De la configuration en ligne, on passe en 2D au pavage régulier suivant un plan et en 3D au pavage régulier suivant une grille. De la configuration en cercle, on passe en 3D de la configuration sphérique. Il n'est pas nécessaire de disposer d'un grand nombre de catégories de configuration pour pouvoir réaliser des placements complexes. Il convient plutôt de trouver le juste nombre suffisant, minimal et assez générique. La version actuelle de MIMESIS supporte par exemple des configurations en ligne, en pavant un rectangle, en pavant un parallélépipède, en cercle, en pavant la surface d'une sphère.

Il convient dans ce cas de permettre à l'utilisateur de définir la forme sous-jacente à la configuration, par exemple les extrémités du segment de droite portant la ligne de modules.

La figure suivante est une capture de l'outil de placement pavant un rectangle :

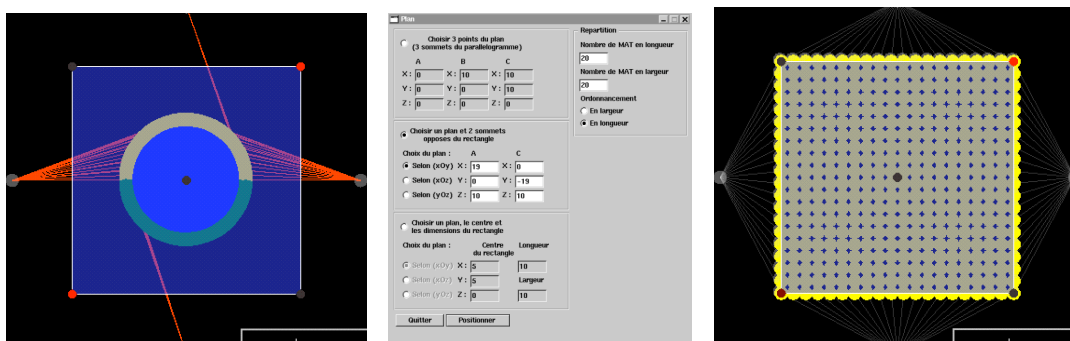


Figure 12 : Placement automatique selon un rectangle.

A gauche : un réseau généré par compilation d'un texte. Au centre : la fenêtre d'outils de placement automatique selon un rectangle. A droite : les modules MAT positionnés par l'outil.

IV.5. Placement à partir de données provenant d'un environnement de modélisation de formes

La plupart des environnements de modélisation de forme sont capables de sortir des fichiers contenant un ensemble de points situés dans un espace 2D ou 3D correspondant aux points définissant les formes modélisées. On peut réutiliser ce fichier pour mapper les positions qu'il contient sur les positions initiales des masses du modèle CORDIS-ANIMA.

La figure 13 est une capture de l'outil de placement d'un fichier de positions sur les positions initiales d'un réseau CORDIS-ANIMA dans notre environnement de conception interactif. La fenêtre interactive en haut à gauche représente un outil de *mapping* des positions provenant d'un environnement de modélisation de formes vers un ensemble de positions initiales d'un réseau CORDIS-ANIMA alors que la fenêtre en haut à droite représente l'ensemble des modules MAT du réseau CORDIS-ANIMA au fur et à mesure du placement. Le réseau est donné dans sa configuration spatiale courante puis les modules MAT sélectionnés se placent aux positions choisies par l'utilisateur. Le modèle CORDIS-ANIMA ainsi obtenu pourra par la suite contrôler l'évolution temporelle de la forme qui a été utilisée pour produire les sorties de position. Ce pourra être le cas lors de la modélisation d'objets déformables. Mais cette forme pourra aussi ne pas être utilisée. Dans ce cas, l'environnement de modélisation de forme aura été détourné de sa fonction première pour servir d'outils de placement externe à notre environnement de conception de réseaux CORDIS-ANIMA.

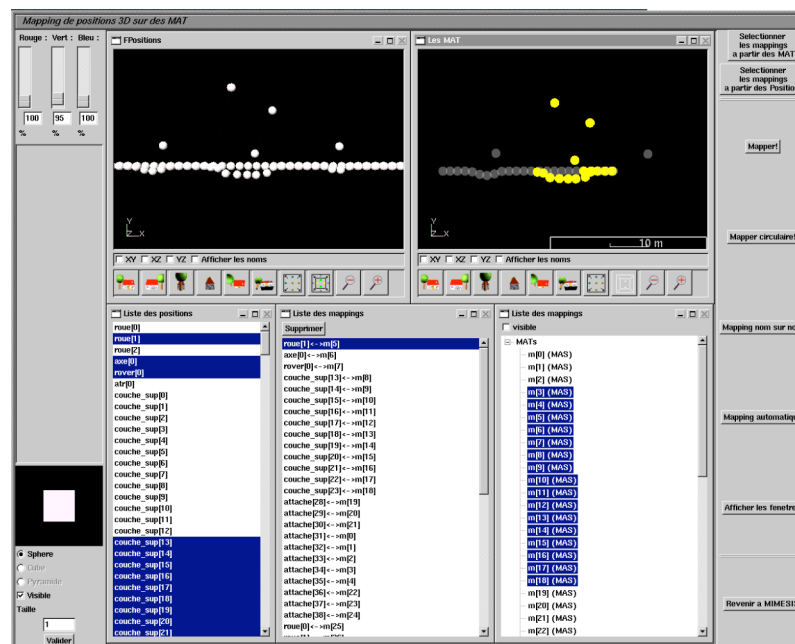


Figure 13 : Outils de placement par mapping de positions provenant d'un fichier.

IV.6. Le placement par simulation physique

Une approche originale du placement, complémentaire aux approches précédentes et particulières à la modélisation physique consiste à utiliser la configuration spatiale d'un modèle CORDIS-ANIMA obtenue à un moment donné, comme par exemple à un moment où le réseau a atteint un état d'équilibre stable, comme ensemble de positions initiales pour un autre modèle. Suivant ce processus, l'utilisateur ne définit pas de manière explicite les positions initiales du modèle qu'il construit mais il crée un modèle CORDIS-ANIMA permettant de définir un ensemble de positions initiales. Il modélise les conditions, les causes de l'obtention d'une configuration spatiale donnée. Ce processus est largement utilisé par Luciani et al. dans [LG97]. Dans cet article, l'organisation spatiale initiale de réseaux est obtenue par la simulation d'un réseau CORDIS-ANIMA pour obtenir certaines configurations comme des structures en hélice ou encore cubiques à faces centrées.

IV.7. Les vitesses initiales

Le cas des vitesses initiales est un cas intermédiaire entre les positions initiales et les paramètres physiques. En effet, de la même manière que les paramètres physiques, les vitesses peuvent aussi bien être toutes semblables que toutes différentes. Mais elles s'expriment dans un espace à plusieurs dimensions comme les positions initiales.

Deux choix sont donc possibles pour déterminer ces vitesses. Le premier est de définir leurs différentes coordonnées dans l'espace. Ce choix est le plus proche de la définition des vitesses initiales comme différence entre positions initiales et positions retardées initiales données par un système de coordonnées. Mais ce n'est pas le plus pratique. Il est intéressant de considérer qu'une vitesse peut être donnée par son intensité et son orientation. Pour l'utilisateur, cela revient à penser la vitesse suivant deux modalités, la direction initiale du mouvement et l'intensité de l'initiation de ce mouvement. Si, dans la phase de détermination des vitesses initiales, l'utilisateur se rend compte que l'orientation de la vitesse est la bonne mais que la norme de celle-ci n'est pas assez grande, il semble plus intéressant pour lui de modifier une unique valeur scalaire plutôt que trois coordonnées.

Une différence de plus par rapport aux paramètres physiques est que les vitesses initiales peuvent être représentées par des vecteurs dans un espace à plusieurs dimensions, le même que l'espace où sont représentés les modules MAT avec leurs positions initiales. Il est donc aussi possible de définir des fonctionnalités interactives d'édition de ces vitesses par manipulation directe d'un ou plusieurs vecteurs en norme et/ou en orientation.

V. Simuler un modèle CORDIS-ANIMA

Une fois définis les éléments des trois étapes nécessaires à la modélisation physique : (1) la structure topologique du réseau CORDIS-ANIMA (Phase PSQL), (2) les valeurs de chacun des paramètres physiques des modules élémentaires impliqués (Phase PSQN) et (3) leurs valeurs initiales (VI), qui correspondent au modèle recherché, le modèle est entièrement calculable. Dans le cas d'une simulation à temps discret effectuée automatiquement par un ordinateur, plusieurs cas peuvent se présenter. Dans la mesure où il ne s'agit pas du cœur de mon activité ni de ma compétence, j'en rappellerai ici seulement quelques principes et les choix effectués dans l'outil de modélisation et de simulation MIMESIS. Pour de plus amples informations, je renvoie avec modestie à mes collègues du laboratoire, ainsi qu'aux diverses études externes sur les questions largement cités dans la littérature du modèle physique en informatique graphique. Je me permettrai simplement d'alerter sur le fait qu'il s'agit d'une question complexe, que toutes les disciplines impliquées (analyse numérique, automatique, traitement du signal) n'épuisent pas à ce jour et qu'il s'agit par conséquent encore d'un sujet de recherche, en tout cas très actif au laboratoire.

V.1. Le théorème de Shanon et les sorties du processus de simulation physique dans MIMESIS

Le théorème de Shanon nous démontre qu'un signal temporel discrétisé peut être entièrement reconstruit sous forme analogique sans perte d'information dès lors qu'on le discrétise en échantillons temporels à pas de temps constant T_e si ce pas de temps T_e est de valeur 2 fois plus faible que le pas de temps de coupure T_c , correspondant à la fréquence de coupure du signal analogique d'origine. Dit autrement, si on l'échantillonne à une fréquence $F_e = 2 * F_c$, F_c , fréquence de coupure du signal, est une caractéristique du signal considéré, par le fait que tout signal réel présente un spectre fréquentiel (calculé par sa transformée de Fourier) bornable, c'est-à-dire dans lequel les composantes fréquentielles de valeur de fréquence supérieures à cette fréquence de coupure sont peu significatives et peuvent être ignorées.

En conséquence, tout processus de simulation visant à produire ce signal, doit calculer le nombre d'échantillons par seconde préconisé par le théorème de Shanon. Il s'agit donc d'un paramètre du système de simulation qui doit être accessible aux utilisateurs, ce qui est le cas dans l'environnement

MIMESIS. Cependant, on estime que pour la majorité des phénomènes mécaniques donnant lieu à des effets visuels pertinents, une fréquence de simulation de l'ordre de quelques kHz est en général suffisante. La valeur par défaut dans MIMESIS est de 1050 Hz. En nominal, MIMESIS fournit donc en sortie 1050 échantillons par seconde pour chaque donnée de sortie des modules élémentaires calculés faisant partie du modèle. Cette valeur de la fréquence de simulation peut être modifiée pour la simulation de phénomènes présentant des composantes temporelles plus rapides, dans le cas d'objets audio-visibles très rigides ou exhibant de fortes non linéarités émergentes, comme des fractures ou des effets d'auto organisation. En effet, ces phénomènes présentent des raies spectrales à haute fréquence non nécessairement visibles qu'il est nécessaire de restituer sous peine de ne pas obtenir ces discontinuités pourtant visibles.

V.2. L'utilisation par un système aval du signal calculé :

Lorsqu'un phénomène est utilisé en entrée d'un dispositif aval, celui-ci se caractérise par sa propre fréquence de coupure. Si celle-ci est inférieure à celle du phénomène observé, alors le phénomène observé est filtré, ce dispositif se comportant comme un filtre passe-bas, c'est-à-dire ne restituant (au mieux) que les fréquences du signal d'origine inférieures à sa propre fréquence de coupure. Ainsi le système perceptif visuel humain n'est pas sensible à des fréquences de l'ordre au maximum de quelques centaines de Hz alors que le système perceptif auditif est sensible jusqu'à quelques dizaines de kHz. Quant au système haptique, on admet aujourd'hui qu'il soit sensible à quelques kHz. Par conséquent, lorsqu'il s'agit uniquement de « voir » le signal ainsi produit, il est suffisant d'effectuer un sous échantillonnage. Usuellement les écrans de visualisation admettent des signaux à 25 ou 50 Hz. Mais cela ne signifie pas qu'il faille le produire à cette fréquence. La première raison essentielle est qu'un phénomène visible peut être produit par des phénomènes mécaniques présentant des composantes fréquentielles de valeur plus élevées et non visibles. Prenons un exemple simple dans le cas du son : deux signaux sinusoïdaux non audibles, l'un à 100 000 Hz et l'autre à 100 100 Hz, produisent lorsqu'on les superpose en les additionnant, un signal sinusoïdal audible de battement à 100 Hz. Des phénomènes similaires existent visuellement où des signaux à fréquences visibles sont provoqués par des signaux à fréquences non visibles. C'est le cas de phénomènes transitoires comme des fractures. Le changement d'état visible « non fracturé / fracturé » est causé par des phénomènes mécaniques non visibles, parfois audibles et même parfois à fréquences plus élevées que les fréquences audibles. Il est par conséquent nécessaire d'effectuer la simulation à la fréquence de Shanon correspondant à la bande passante des composantes fréquentielles nécessaires à la création des phénomènes. Une deuxième raison est que cela reviendrait à limiter l'usage aval de ces signaux. Cela est le cas, par exemple, lorsque l'on veut effectuer des simulations physiques en cascade. Cela est également le cas, lorsque le dispositif de sortie est un système haptique ou un système audio. Cela conduit à définir la simulation comme un ensemble de boucles temporelles imbriquées : une boucle de base tournant à une fréquence de simulation centrale judicieusement choisie par rapport au phénomène à simuler, et un ensemble de boucles de calcul produisant des signaux sous ou sur échantillonnés. En pratique, un ensemble de fréquences de simulation sont préconisées dans le système MIMESIS, qui sont celles à la fois multiples de 25 Hz et sous-multiples de 44,1 kHz (fréquence d'échantillonnage standard des signaux audio). Le cas par défaut étant 1050 Hz. Ceci garantit l'usage de l'environnement

MIMESIS dans diverses situations (simulations par parties ou simulations réparties, simulations hapto-visuelles, simulations audio-visuelles, simulations hapto-audio-visuelles, chacune d'elles pouvant être intégrales, par parties ou réparties), pour d'autres usages que la consommation visuelle immédiate (par exemple pour l'analyse, ou pour alimenter d'autres modèles et simulations, à l'instar d'un système de motion capture simulé).

V.3. La discrétisation des équations continues et les problèmes liés à la divergence

Dans la transformation d'une représentation continue d'un système à entrée / sortie à une représentation à temps discret de ce même système, un problème advient qui est supplémentaire aux questions traitées dans le point 1 et 2 liées à la représentation et à l'usage des signaux. Il s'agit du problème de la divergence : les équations continues sont transformées en suites récurrentes dont l'ordre ainsi que les conditions de convergence dépendent des schémas de discrétisation adoptés pour les variables traitées (positions, vitesses et accélérations) et des causalités entrées / sorties introduites par l'algorithmique. En tout état de cause, les systèmes à temps discrets ne convergent pas dans tout l'espace des valeurs des paramètres physiques (M, K, Z), les schémas de discrétisation à position retardée et à vitesse retardée (dits « schémas d'Euler »), présentant la zone de convergence la plus étroite [Inc96]. Plusieurs solutions mathématiques développées en analyse numérique et de plus récentes solutions algorithmiques développées en informatique graphique existent pour augmenter la zone de convergence de ces représentations à temps discrets [DSB99, PGM*98]. Elles ont toutes le désavantage qu'elles altèrent les propriétés dynamiques du système modélisé, et par conséquent l'idée que ce fait le concepteur du comportement du modèle ; le comportement après simulation pouvant être notablement différent de celui imaginé, en particulier dans les cas complexes de phénomènes non linéaires à forts transitoires. Certaines d'entre elles introduisent ces biais implicitement comme c'est le cas de la méthode proposée par Desbrun et al. [DSB99]. Une méthode plus directe mais qui a l'avantage de ne pas altérer le comportement du système modélisé, consiste à augmenter la fréquence de simulation. C'est la méthode utilisée dans les simulateurs encapsulés dans MIMESIS, nommés TELLURIS et CORDIS-OFF. L'inconvénient n'est pas si grand qu'il n'y paraît si on garde à l'esprit les arguments développés dans la partie V.2 ci-dessus qui explique que la simulation doit la plupart du temps être effectuée à fréquence supérieure à celle de la visualisation, pour des raisons liées à la structure même du phénomène dynamique à restituer.

V.4. Le temps réel et le temps non-réel.

On ne peut pas définir le temps réel indépendamment d'une référence temporelle absolue, c'est-à-dire donc d'une horloge de référence qui d'une part soit *externe* au processus lui-même, ici au processus de simulation et d'autre part, cadence *régulièrement* les signaux produits (et donc les processus de leur production et de leur consommation). Dans le cas présent, cette horloge peut être calée sur l'échantillonnage régulier de signaux de visualisation, à 25 Hz par exemple. Elle peut aussi être calée

sur l'échantillonnage des signaux en provenance ou/et à destination des systèmes haptiques. La simulation sera temps réel si tous les échantillons absorbés par la simulation en provenance des systèmes amont ou/et produits par la simulation et absorbés par les systèmes aval, sont traités entre deux tops d'horloge. Ils pourront être « temps réel au regard de la visualisation » dans le premier cas, ou « temps réel au regard de l'haptique » dans le second cas. Et ainsi de suite, pour tous les autres cas imaginables. Autrement dit, si le temps de calcul d'un état complet de la scène à l'instant nT_e est inférieur à T_e :

$$T_e = \text{Pas de l'horloge de référence} = \text{Pas de la simulation} = T_c/2$$

Où T_e est le temps d'échantillonnage et T_c est l'inverse de la fréquence de coupure

Dans le cas où cette contrainte ne peut être tenue, la simulation est dite à temps relâché eu égard à l'horloge considérée (cadencant les signaux visuels, et/ou cadencant les signaux haptiques, etc.) lorsque le pas de temps moyen est proche du temps réel et qu'il n'est pas strictement synchrone, par exemple lorsque les échantillons sont consommés ou produits autour de la fréquence temps réel. La simulation est dite à temps différé lorsque, pour des questions liées au rapport Puissance de calcul du processeur / Complexité de la scène, le pas de temps de calcul est supérieur au pas de temps de la simulation choisie (cf. points 1 et 2) pour une bonne restitution du phénomène simulé. Alors, il n'y a pas d'autres solutions que d'effectuer la simulation totalement hors-ligne et de stocker les échantillons ainsi produits dans un fichier, dont on espère qu'il sera lu par un processus de consommation de ces données (par exemple un processus de visualisation) temps réel synchrone. Le cas inverse où le temps de calcul d'une scène est bien inférieur à la bande passante du phénomène simulé et/ou à la fréquence de visualisation, peut se présenter aussi. On peut alors avoir intérêt à calculer au plus vite et observer le résultat dans une phase postérieure. Tous ces cas sont prévus dans MIMESIS.

V.5. Recommandations

Tout d'abord, il est nécessaire de calculer la simulation au plus près des caractéristiques temporelles (i.e. fréquentielles) du phénomène à restituer, en se plaçant au plus près de la fréquence de Shannon nécessaire à la représentation discrète du phénomène. Cette fréquence n'est souvent pas connue. La simulation peut alors être un outil permettant de l'estimer empiriquement. Ensuite, il ne faut pas hésiter, et ce avant toute étape d'optimisation calculatoire ou algorithmique, de se placer dans de bonnes situations de convergence de l'algorithme discret, en n'hésitant pas d'augmenter cette fréquence de simulation si nécessaire. Puis, lorsque l'on souhaite observer et valider visuellement le comportement du modèle, il est absolument impératif de garantir le temps réel synchrone dans le processus de visualisation. Dans le cas où le temps de calcul est inférieur au temps de Shannon souhaité, le processus de visualisation peut être effectué en ligne en incluant un processus de sous-échantillonnage des signaux produits par la simulation. Dans tous les autres cas, soit on simplifie le modèle, lorsque le phénomène à modéliser s'y prête, jusqu'à ce que son temps de calcul soit compatible avec une observation visuelle temps réel, soit on travaille en temps différé à la fréquence de simulation nécessaire à la dynamique du phénomène à restituer (voir partie V.1). Puis dans une deuxième étape, les signaux ainsi générés peuvent être sous-échantillonnés à la fréquence de visualisation, et le processus de visualisation effectué de synchronisation et de régularité

correspondant aux rendus de signaux temporels discrets. De même, dans le cas d'une production / consommation de données haptiques, il est impératif de garantir le temps réel synchrone. Cela signifie que la scène doit être calculée entre deux échantillons d'entrée ou de sortie depuis / vers les dispositifs haptiques, ces deux signaux étant eux-mêmes synchrones et décalés du plus petit nombre d'échantillons possibles (ce plus petit nombre étant 1). On ne saurait trop mettre en garde, dans la phase de mise au point du modèle, contre l'usage de procédés de visualisation ou d'haptisation dits « fluide », qui, s'ils peuvent être plaisants, ne garantiraient pas forcément une représentation temporelle fidèle des signaux discrétisés, en introduisant par exemple des fluctuations temporelles des signaux objectivement produits par le modèle et sa simulation correcte. Libre à l'utilisateur de les pratiquer une fois obtenu le modèle physique voulu.

Enfin, une fois le bon modèle et les bonnes simulations obtenues, dans de bonnes conditions de simulation et de visualisation garantissant la fidélité objective de la simulation et de son observation, on peut alors entrer dans une phase d'analyse conduisant à des optimisations éventuelles. Il peut s'agir soit de l'optimisation du modèle lui-même (diminution du nombre de masses ou d'interactions), soit des conditions numériques de sa simulation (diminution du pas de temps de la simulation, utilisation d'algorithmes améliorant la convergence), soit des conditions de sa visualisation (fluidisation d'un fort sous-échantillonnage).

VI. Visualisation des variables d'état de la simulation

VI.1. Visualiser le résultat de la simulation pour mieux le comprendre et le contrôler

Les modèles CORDIS-ANIMA produisent des fonctions d'évolution lors de la simulation, fonctions constituées de l'ensemble des positions des masses évoluant au cours du temps. Il est nécessaire de représenter un tel signal temporel multidimensionnel dont la raison d'être est la production d'images animées. Il faut insister sur les deux points : la représentation d'un signal, et son utilisation pour la production d'images animées. Le premier point nous conduit à la réalisation d'outils de mesures de l'évolution des variables de la simulation. Il nous oblige à fournir des données de quantification du mouvement. Les outils correspondant à cette quantification sont les outils de représentation du signal suivant un graphe prenant en abscisse le temps et en ordonnées les valeurs des variables de la simulation. Il s'agit non seulement de représenter l'évolution des variables extensives positions 3D (X, Y, Z) tout autant que les vitesses, l'accélération, mais aussi les variables intensives comme la force. Les valeurs vectorielles doivent être certes représentées mais elles ne sont pas toujours les plus significatives. Ainsi une variable scalaire comme une distance entre deux masses peut avoir plus de sens que l'évolution de deux positions vectorielles. Le second point nous mène à développer parallèlement des outils pour qualifier le mouvement. Le modélisateur ne juge alors plus sur des nombres mais sur des phénomènes visuels produits à partir des variables de la simulation. Du signal généré par un modèle CORDIS-ANIMA émerge des structures qui intéressent le modélisateur et qu'il a besoin d'explicitier graphiquement. Ces structures émergentes peuvent être temporelles comme la formation de vortex, les effondrements de surface mais aussi topologiques, comme l'émergence de comportements rigides ou déformables. Ce sont ces structures émergentes que le modélisateur a en tête lorsqu'il crée son modèle. Mais pour contrôler le fait qu'elles apparaissent effectivement, il faut mettre à sa disposition des outils de représentation du comportement global de la simulation. La

fenêtre 3D est un bon moyen de représenter cela, mais la simple représentation ponctuelle ne suffit pas. Il faut habiller les fonctions d'évolution à l'aide de formes, pour « donner forme » au signal.

La partie I du chapitre 4 se consacre exclusivement à cette question qui est celle de l'habillage de modèles physiques particuliers. Il propose une analyse des manières de passer du point se déplaçant dans l'espace à la forme se déplaçant dans l'espace du mouvement généré à l'objet en mouvement révélé à l'écran. Si nous pouvons renvoyer le lecteur à ce chapitre dont les préoccupations ont une partie commune avec notre question nous devons ici mettre le doigt sur la différence fonctionnelle que nous pointons. La question n'est pas ici d'obtenir une œuvre finale. Il s'agit de donner à voir le mouvement à l'utilisateur, de manière minimale, juste nécessaire pour qu'il puisse travailler dessus, un peu à l'image du modélisateur en géométrie qui ne travaille que sur une représentation « fil de fer » de sa forme, cela, pour des raisons d'optimisation du fonctionnement du système tout autant que de l'économie d'informations qui ne serviront pas à l'utilisateur dans sa phase de conception du mouvement. Le placage d'un matériau sur une forme ne sert à rien pour comprendre le mouvement par exemple. Si le bénéfice de l'utilisation de logiciels tiers est énorme pour la qualité du rendu de la scène, un habillage optimisé ou adéquat est nécessaire au sein d'un outil d'animation pour la compréhension du mouvement. Par ailleurs, la méthode de représentation que l'utilisateur choisira pour visualiser la simulation pourra être similaire, raffinée ou différente du choix de la méthode du rendu final.

VI.2. Visualisation de la simulation différente de la représentation dans la fenêtre de modélisation

Le choix de l'habillage géométrique peut être très différent de la représentation du réseau topologique lors de la phase de modélisation. Ce n'est pas, par exemple, parce qu'il y a une interaction entre deux masses qu'il est intéressant de tracer un trait pendant la simulation. À l'inverse il n'est pas nécessaire qu'il y a une interaction entre deux masses pour tracer un trait qui donne forme à un objet.

Pour illustrer ce propos étudions le modèle de mouvement dansé de valse proposé par Hsieh & al. dans [HL05]. Ce modèle est composé de 8 masses dont certaines sont en interactions viscoélastiques très rigides pour symboliser le corps du danseur. D'autres interactions viscoélastiques entre les masses représentant les mains et les masses représentant les pieds permettent une synchronisation entre les mouvements des bras et des jambes. Un système d'interaction composé d'interactions de butées viscoélastiques permettent d'imprimer le mouvement à 4 temps de la valse au corps du danseur. À partir des positions des masses, Hsieh contrôle une structure filaire qui va lui permettre de visualiser le résultat de sa simulation. On peut remarquer que les interactions servant à la génération du mouvement de valse (interactions servant à la synchronisation et au pas de 4) ne sont pas symbolisées par des traits. À l'inverse, Hsieh tire un trait entre les deux masses représentant les épaules alors qu'aucune interaction n'existe entre ces deux masses. Ce qui intéresse le modélisateur dans le choix de cet habillage, c'est de matérialiser le corps pour en étudier le mouvement.

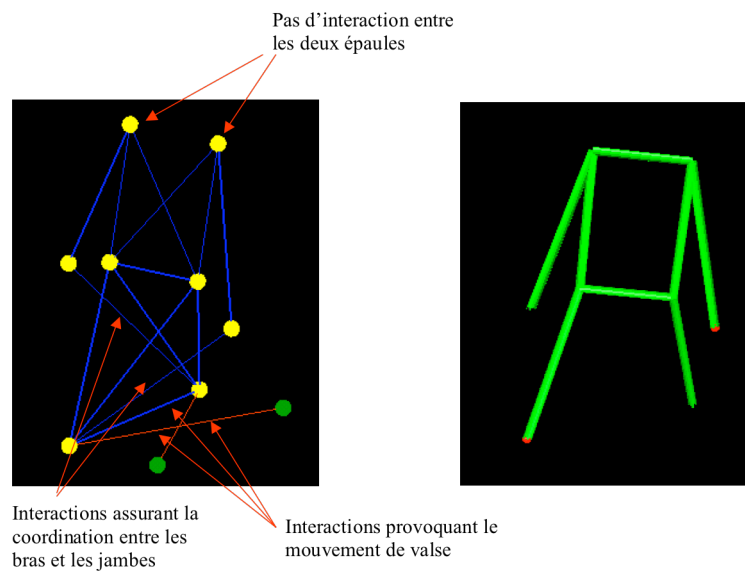


Figure 14 : Habillage sommaire d'un modèle CORDIS-ANIMA

À gauche : modèle CORDIS-ANIMA de valse. À droite : matérialisation visuelle du corps dansant.

VI.3. Plusieurs visualisations pour un même mouvement

Il est possible de visualiser de plusieurs manières une même simulation CORDIS-ANIMA pour faire apparaître des informations différentes concernant le mouvement généré. Prenons l'exemple d'un modèle simple de double pendule. Ce double pendule est constitué de deux pendules eux-mêmes constitués de trois masses reliées en chaîne par une interaction viscoélastique. Les deux pendules ont une masse en commun qui est fixe et modélise le point d'attache de ces deux pendules. Les masses en vis-à-vis des deux pendules sont elles en interaction de butée viscoélastique comme le montre la figure 15.

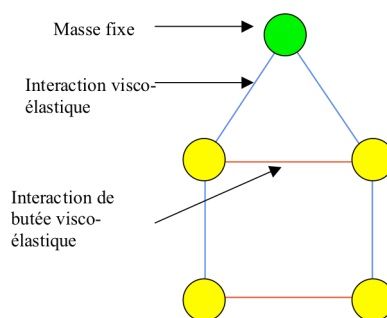


Figure 15 : Modèle du double pendule

L'élasticité assurant une cohésion, une visualisation qui peut faire sens pour l'utilisateur est de représenter durant la simulation les deux chaînes formées de 3 masses par une ligne brisée comme le montre la figure 16 :

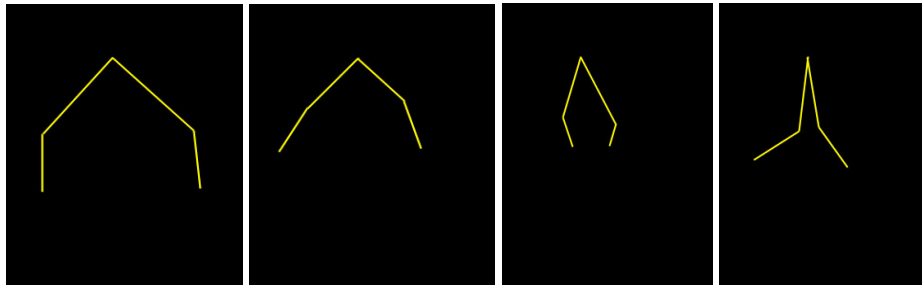


Figure 16 : Habillage d'un module de double pendule par une ligne brisée symbolisant le lien de cohésion entre les masses.

Ces différents exemples illustrent la variabilité des méthodes permettant de révéler le mouvement généré par un modèle CORDIS-ANIMA. Mais nous pouvons faire la remarque qu'un même mouvement peut-être lui-même révélé de plusieurs manières. Et que ces différentes manières ne disent pas la même chose sur celui-ci. Des habillages d'une fonction d'évolution révèlent différentes faces du mouvement. Si on reprend le modèle de pendule, on peut choisir de visualiser à l'aide de traits les masses en interaction de butée. Dans ce cas, la visualisation focalise le point d'attention sur l'évolution des collisions comme le montre la figure suivante présentant la même simulation aux mêmes instants avec ce nouvel habillage :

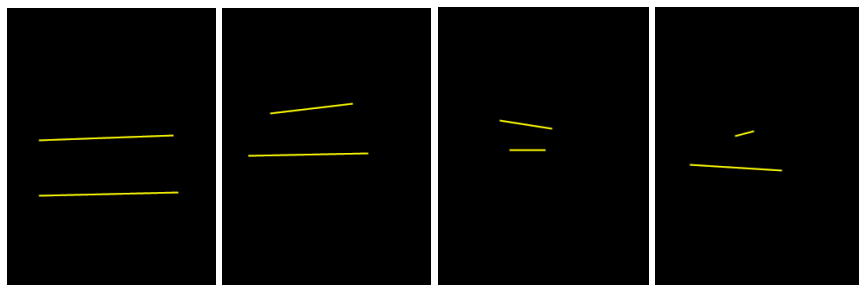


Figure 17 : Habillage d'un module de double pendule par deux traits reliant les masses en interaction de butée.

VII. Bilan

Nous avons présenté, dans ce chapitre, les fonctionnalités de base d'un environnement de conception interactif de réseaux CORDIS-ANIMA. Les deux chapitres suivant vont plus précisément présenter l'extension nécessaire des fonctionnalités de cet environnement pour qu'il puisse prétendre à être utilisable dans un contexte assez générique. Un point fondamental traité dans le chapitre suivant est de définir, d'implanter et de mettre à disposition un nombre suffisant de types de modules d'interaction. L'histoire de la modélisation par réseaux CORDIS-ANIMA pour l'animation montre en effet que de simples interactions linéaires sont loin d'être suffisantes pour obtenir effectivement un vaste éventail de comportements dynamiques. Le chapitre 4 sera alors consacré à la question de la communication de MIMESIS avec d'autres systèmes pour l'insérer dans une chaîne globale de production d'images animées.

- Chapitre 3 -

Interactions non linéaires dans un environnement de conception interactif

I. Des interactions non linéaires pour augmenter la complexité des comportements dynamiques générés

I.1. Une condition sine qua non pour l'obtention de comportements complexes en animation

Dans l'histoire des modèles générateurs pour la synthèse d'image, un nombre très important de comportements ou d'effets ont été obtenus : du solide rigide au solide déformable, du liquide à la fumée, du mouvement d'un individu humain au mouvement d'une foule, et bien d'autres encore. Mais parmi la variété de techniques utilisées, les formalismes masses - interactions comme les réseaux CORDIS-ANIMA sont très certainement ceux qui ont permis d'obtenir le plus de comportements dynamiques différents. En revanche, de tels formalismes, s'ils sont restreints aux « modèles masses - ressorts » déduits de la description morphologique des objets, ne sont pas assez génériques pour fournir cette vaste palette de comportements. Tout au plus permettent-ils d'obtenir des comportements d'objets faiblement déformables. Pour obtenir une véritable variété de comportements dynamiques, il convient de faire appel d'une part à des modèles de type réseau qui se détachent de la représentation morphologique de l'objet et d'autre part à des interactions non linéaires complexes.

La mise en œuvre de réseaux à topologie complexe a été discutée dans le chapitre 2, partie II. Elle implique la mise à disposition de fonctionnalités capables de gérer la grande variabilité des réseaux possibles. Nous avons notamment vu que l'élaboration d'un langage textuel dédié à la conception de réseaux CORDIS-ANIMA est une condition sine qua non pour qu'un environnement interactif de conception puisse prétendre à une véritable genericité. Le chapitre suivant, quant à lui, s'intéresse à la mise en œuvre d'outils spécifiques, au sein du même environnement de conception, dédiés à la création et à la gestion d'interactions non linéaires.

Rappelons ici la signification précise du terme « interaction » dans un formalisme de type masses - interactions. Il s'agit d'un composant, (on pourrait dire également une fonctionnalité) représentable par une équation reliant deux variables extensives de même nature (des positions par exemple, ou des

vitesse) à deux variables intensives de force, égales et opposées pour satisfaire le principe d'action - réaction. Les interactions de base sont alors naturellement les interactions linéaires de type élastique ou visqueux, puisqu'elles relient des forces à des différences de positions (plus généralement de distance) pour les premières ou de différences de vitesse pour les secondes. Cependant, ces interactions ne sont, bien entendu, pas suffisantes pour représenter la panoplie de phénomènes dynamiques qu'un modéleur générique pour le mouvement souhaite donner accès. Une des principales limitations est en effet la capacité de restituer des phénomènes faisant appel à des comportements non linéaires des systèmes générateurs qui les produisent.

Mais quelle est la nature de ces interactions ? Doit-on s'intéresser à la littérature des sciences physiques pour les trouver ? Suffit-il d'implanter ce qui serait potentiellement déjà pratiqué par une large communauté de physiciens, ou au contraire, est-on face à une pratique relativement nouvelle de la modélisation physique qui demande au préalable d'en proposer une classification et d'analyser leurs modalités de représentation pour les mettre à disposition des animateurs ?

I.2. S'affranchir partiellement de la pratique des sciences physiques

Les physiciens se sont emparés de l'outil informatique dans la pratique de leur discipline, comme ce fut le cas pour d'autres sciences. Bien souvent, cette communauté a utilisé l'informatique pour résoudre numériquement des équations complexes de formulation continue (équations de Navier-Stokes, de Maxwell, etc.). De plus, dans un grand nombre de cas, ces modèles et théories physiques ne se réfèrent pas toujours explicitement à une vision interactionnelle.

Or, l'approche de l'animation par modèle physique masses - interactions que nous proposons ne consiste précisément pas à obtenir une représentation discrète des formulations continues développées par la physique. Au contraire, il s'agit plutôt de proposer une panoplie de fonctions d'interaction de base qui, reliant deux masses ponctuelles, permettrait de représenter par construction d'assemblage des comportements observables. Au plus simple, un système composé d'une telle interaction de base reliant deux masses ponctuelles pourrait lui-même représenter un comportement dynamique observable, voire connu, d'un système composé de deux corps, de deux particules, voire de deux individus en interaction. Nous verrons par exemple qu'une fonction d'interaction de frottement sec peut-être définie entre deux masses ponctuelles alors que le frottement sec, dans la littérature physique, se réfère presque toujours à deux surfaces. Ces comportements peuvent d'ailleurs indifféremment être des comportements microscopiques ou macroscopiques. Ainsi, nous verrons qu'une interaction de type « Lennard-Jones » est à même de représenter l'interaction entre deux molécules mais également des interactions cohésives observables entre deux corps à notre échelle ou deux individus dans une foule. À ce sujet, Luciani dans [Luc00] montre que l'approche de modélisation soutenue par le formalisme CORDIS-ANIMA peut être qualifiée de mésoscopique dans le sens où les masses ponctuelles et les interactions qui les relient, se situent à un niveau de modélisation juste suffisant

entre une approche microscopique et une approche macroscopique pour reproduire avec le plus petit nombre de paramètres possibles les phénomènes désirés.

Cette philosophie fait écho aux travaux de Greenspan dans les années 70. Greenspan remet en cause la légitimité et l'efficacité de la représentation continue des phénomènes physiques en faisant remarquer que d'une part, ceux-ci émergent d'ensemble de particules en interaction, que d'autre part le processus d'observation et de mesure est discret et qu'enfin, ces représentations continues permettent difficilement de représenter des effets non linéaires, pourtant majoritaires dans la nature. Dans son ouvrage intitulé *Particle Modeling*, Greenspan modélise un nombre important de phénomènes physiques par des ensembles de masses ponctuelles en interaction de type Lennard-Jones : Ces modèles ont inspiré par la suite les travaux en synthèse d'images de Miller et Pearce [MP89], de Terzopoulos et al. [TPF89], de Tonnesen [Ton91] ainsi que ceux de Luciani et al. [LJC*91, LJR*91].

Or la modélisation particulière proposée par un environnement tel que MIMESIS est consubstantielle à la nature discrète des calculs opérés par l'ordinateur et ne pouvait en quelque sorte pas exister avant l'ordinateur. Les modules représentent des fonctionnalités qui opèrent des calculs discrets de nature physique liant les variables extensives et les variables intensives du modèle. Ces calculs sont très simples mais très nombreux, ce qui nécessite l'usage de l'informatique. Par la construction de réseaux, l'utilisateur de CORDIS-ANIMA, représente directement dans l'ordinateur les hypothèses qu'il fait sur les causes du mouvement qu'il souhaite obtenir, et visualise, grâce à la simulation et l'affichage à l'écran, le mouvement généré par ce réseau. Il peut alors comparer le mouvement obtenu à ce qu'il en attendait.

Ce qu'il attendait du mouvement, il doit le trouver par l'observation. C'est ici que se fait le point de jonction avec les sciences physiques. L'animateur peut observer lui-même les mouvements dans la nature, mais il peut aussi faire appel aux observations des physiciens. Cette observation peut faire référence à la mesure de phénomène, mais peut aussi ne pas y faire référence. On peut vouloir obtenir des vortex sans faire référence à un vortex mesuré, mais le physicien peut nous apprendre que dans une fumée il y a des vortex. L'animateur doit donc chercher à obtenir un comportement dynamique qu'il a correctement réussi à formuler à partir de toute source qu'il a à sa disposition, son observation et son intuition ou alors des observations opérées par d'autres que lui.

Pour réaliser ce projet, CORDIS-ANIMA met à sa disposition des modules dont il doit comprendre préalablement la fonctionnalité. Or la difficulté ne réside pas dans la compréhension du comportement des modules de masse, compréhension relativement aisée. La difficulté réside bien d'avantage dans les comportements des modules d'interaction qui peuvent être très divers. Ces modules, comme nous l'avons dit, peuvent faire référence à des comportements physiques sans nécessairement avoir été utilisés par les physiciens. D'où proviennent alors les algorithmes de ces modules ?

I.3. Un corpus d'interactions aux origines diverses

La pratique de la modélisation et de la simulation de réseaux CORDIS-ANIMA construits pas à pas n'est pas une pratique très courante au sein des sciences physiques. Les sources qui nous ont permis de répertorier les différentes interactions sont donc bien plus diverses. Ainsi, nous nous sommes basés à la fois sur certains phénomènes physiques connus, sur l'expérience de modélisation masses - interactions développées par le laboratoire depuis une trentaine d'années ainsi que sur l'observation des attitudes et demandes des praticiens lors de l'usage de CORDIS-ANIMA pour la création artistique ou la pédagogie.

En ce qui concerne les phénomènes physiques, la seule approche de la physique ne suffit pas et nombre d'idées riches peuvent venir d'ouvrages et d'études de systèmes mécaniques, qui ont un point d'entrée différent de celui de la connaissance des phénomènes de la nature : ainsi au côtés des interactions bien connues telles que l'interaction intermoléculaire de Van der Waals, ou les modifications non linéaires de l'élasticité conduisant à la plasticité et à des phénomènes d'hystérésis dans le matériau, nombre d'autres comportements proviennent de la maîtrise de systèmes artificiels comme le jeu mécanique ou les fluctuations en fonction de la charge.

En ce qui concerne l'expérience acquise par le laboratoire, voici qu'en plus de 30 ans, un nombre important de modèles ont été produits qui ont fait intervenir des masses en interaction. Sur une telle période, nombre de comportements dynamiques très variés ont été simulés. Bien que ne nous puissions affirmer que tout ait été modélisé, il est légitime de partir d'une revue la plus exhaustive possible des modèles développés et des interactions qu'ils mettent en jeu. Le temps qui a passé nous assure que cette revue définit un nombre relativement important d'interactions pour que la modélisation avec celles-ci permette d'obtenir une grande variété de comportements dynamiques, certainement même, la plus riche jamais obtenue avec un même formalisme.

En ce qui concerne les attitudes et demandes des praticiens, incluant ici non seulement les usagers externes au laboratoire (artistes et étudiants principalement) mais également les praticiens du laboratoire, chercheurs et étudiants dont les demandes et attentes n'en sont pas moins légitimes, il nous a fallu nous pencher et trancher sur le dilemme sans fin entre « généralité et ouverture extrême » et « compréhensibilité et guide d'usage plus contraignant ». Dans les parties III de ce chapitre, nous expliquerons chaque fois les raisons de nos choix pour ce critère.

Compte tenu de ces différents savoirs et pré-requis d'origine et d'objectifs divers, nous avons donc adopté une attitude pragmatique qui nous a amené à élaborer une typologie des fonctions d'interaction non linéaires de CORDIS-ANIMA ainsi que les modalités de leur représentation et de leur utilisation au sein d'un environnement de conception interactif. En conséquence, les parties qui suivent sont organisées de la manière suivante. Nous proposons dans un premier temps une typologie des fonctions d'interaction non linéaires. Dans un second temps, nous discutons des modalités de représentation possibles pour ces interactions ainsi que des choix que nous avons effectués dans le cadre d'un environnement de conception interactif.

II. Typologie des interactions non linéaires pour les modèles masses - interactions

II.1. Quelle typologie et selon quels critères ?

Il existe, comme nous l'avons dit, plusieurs origines qui nous ont permis de répertorier les divers types d'interaction pour CORDIS-ANIMA. Rappelons qu'une interaction entre deux points peut être décrite comme une fonction plus ou moins complexe donnant la valeur de la force appliquée sur les masses ponctuelles en fonction de leur distance et de leur vitesse relative. Formellement, les interactions peuvent être mise en parallèle. Il donc est possible de distinguer deux classes d'interaction, les interactions potentielle et interactions dissipatives. Il s'agit du critère le plus simple pour nous permettre un premier classement de ces interactions.

Puis au sein même de ces catégories existent plusieurs familles d'interactions qui correspondent à des grandes familles de comportements de la matière que la physique a largement étudiées. Le second critère de classification à l'intérieur de nos deux grandes catégories sera donc celui du comportement unitaire des interactions, comportement dont nous emprunterons souvent le vocabulaire aux sciences physiques.

Nous terminerons sur une présentation des interactions non linéaires faisant intervenir des variations en ligne des paramètres physiques de la simulation. Il s'agit d'une catégorie à part qui est pourtant essentielle pour l'obtention de certains comportements dynamiques. Elle est encore en cours d'exploration notamment dans les travaux de thèse d'Ali Allaoui avec qui nous avons collaboré.

II.2. Interactions non linéaires potentielles

II.2.1. La force en fonction de la distance

Les interactions non linéaires potentielles sont les interactions dont la fonction dérive d'un potentiel. Elles sont, par conséquent, non-dissipatives. De plus, on peut les représenter sous la forme d'un diagramme faisant intervenir la force appliquée en fonction de la distance entre les masses ponctuelles qui interagissent. Ces interactions peuvent alors appliquer, lors de la simulation et en fonction de la distance entre les masses ponctuelles, des forces répulsives (que nous compterons comme positives dans un diagramme force-distance) ou des forces attractives (que nous compterons comme négatives dans ce même diagramme).

Nous avons vu en introduction de cette thèse, l'interaction de butée. Nous en avons dit l'extrême importance pour l'animation puisqu'il s'agissait de l'interaction la plus simple faisant émerger un espace du fait de l'exclusion spatiale entre les masses qu'elle induit. C'est aussi la plus simple des interactions non linéaires potentielles. Or les interactions non linéaires potentielles ont été utilisées depuis longtemps. C'est même historiquement le premier type que l'on peut retrouver dans l'histoire de la physique du point avec une complexité plus importante que notre butée. En effet, l'interaction de gravitation, dont la fonction est inversement proportionnelle au carré des distances entre les masses ponctuelles, et que Newton introduit pour résoudre le problème du mouvement relatif des corps entre dans la catégorie des interactions non linéaires potentielles. Ces interactions ont ensuite connu leur homologue en électromagnétisme (forces de Coulomb d'interaction entre deux particules chargées par exemple). Nous parlerons de ce genre d'interaction comme des interactions d'attraction ou de répulsion à distance. Nous recensons ensuite dans la littérature, et à la suite des travaux de Greenspan [Gre73], certains travaux autour de l'interaction dérivant du potentiel de Lennard-Jones. Cette interaction est augmentée par Luciani et al. [LG97] ou encore Reynolds [Rey97] pour rendre des phénomènes de fracture. Au final, nous présenterons les interactions faisant intervenir en sus une hystérésis permettant de modéliser la plasticité de la matière comme on peut la trouver dans les études de rhéologie.

II.2.2. Interactions d'attraction et de répulsion à distance

De façon consécutive à l'énonciation des lois fondamentales de la dynamique, Newton (1642,1727) chercha à unifier par une loi, la corrélation des mouvements de deux corps matériels. Partant des travaux de Kepler (1591,1630) sur le mouvement des planètes du système solaire et des travaux de Huygens (1629,1695) sur la force centrifuge, Newton avance l'hypothèse que les mouvements relatifs de deux corps inertes de masses respectives m_1 et m_2 peuvent être décrits en faisant appel à une force

s'exerçant entre eux, force dépendante de chacune de leur masses m_1 et m_2 et inversement proportionnelle au carré de leur distance d:

$$F = G \cdot \frac{m_1 m_2}{d^2}$$

Une telle loi faisant intervenir l'inverse du carré des distances est ensuite observée par Coulomb à partir de 1750 dans le cas de deux corps électriquement chargés puis de manière analogue entre deux aimants. Ces forces, contrairement à la force de gravitation entre deux corps, peuvent être non seulement attractives mais aussi répulsives. Ainsi deux charges de même signe, ou deux pôles similaires de deux aimants, se repoussent-ils. Dans le cas du formalisme CORDIS-ANIMA, permettant de construire des modèles à constantes réparties, la masse est portée uniquement par les modules de type MAT. Un LIA ne connaît donc pas la valeur des deux masses des deux modules MAT qu'il relie. Par conséquent, le formalisme CORDIS-ANIMA ne peut simuler ce type de comportements attractifs et répulsifs qu'en introduisant comme paramètre de l'interaction portée par le LIA le produit de valeurs des deux masses qu'il relie. Ainsi, pour rendre de tels comportements, l'algorithme du LIA devrait être :

$$F_n^1 = -F_n^2 = \frac{K}{d^2(X_{n-1}^1, X_{n-1}^2)} u_{2 \rightarrow 1}, \text{ en donnant à K une valeur proportionnelle à } m_1 \cdot m_2.$$

Heïgeas et al. [HLT*03] proposent d'utiliser ces forces répulsives et attractives à distance dans une modélisation du comportement collectif de foule par des fonctions linéaires par morceaux, le nombre de morceaux faisant partie de la modélisation de l'interaction. Chaque intervalle sur lequel la fonction d'interaction élastique est linéaire correspond dans le modèle des auteurs à une « sphère mentale » d'évitement ou de déroutement. À faible distance, la pente de la fonction localement linéaire est importante. Cela modélise un évitement urgent pour prévenir les collisions. À longue distance, cette pente est plus faible. Cela modélise une anticipation à longue distance, moins « dure » qu'un évitement à courte distance. Les auteurs expliquent qu'il suffit de 3 à 4 portions linéaires de la fonction d'interaction élastique pour modéliser un très grand nombre de comportement. La formulation de la force est alors :

$$\begin{aligned} \text{Si } D < D_1 \text{ alors } F &= (K_1 D + Z_1 V) \vec{u} \\ \text{Si } D_i < D < D_{i+1} \text{ alors } F &= (K_{i+1} D + Z_{i+1} V) \vec{u} \text{ pour } i=1 \text{ et éventuellement } 2 \\ \text{Si } D_f < D \text{ alors } F &= \vec{0} \end{aligned}$$

II.2.3. Interactions de Lennard-Jones

De telles interactions d'attraction ou de répulsion peuvent être vues comme un cas particulier de la famille des interactions potentielles non dissipatives du type Lennard-Jones, famille introduite explicitement par Greenspan [Gre73] puis de Miller et Pearce [MP89], de Terzopoulos et al. [TPF89], de Tonnesen [Ton91] et qui a permis de modéliser de nombreux états de la matière (gazeux, liquide,

déformable) à l'aide d'un seul type d'interactions. L'expression générale de la fonction d'interaction de cette famille est :

$$F(D) = \frac{B}{D^n} - \frac{A}{D^m}$$

Dès lors quand les paramètres m et n sont positifs, si $A=0$ et la force est purement répulsive ($F(D) = \frac{B}{D^n}$). Mais si $B=0$, la force est purement attractive ($F(D) = -\frac{A}{D^m}$). On peut remarquer que l'élasticité linéaire s'obtient en prenant $n = -1$ et $A = 0$.

La combinaison des deux termes offre de nouvelles fonctions d'interaction extrêmement intéressantes. La figure 18 représente une interaction de Lennard-Jones pour $n=12$ et $m=6$ comme on peut en trouver dans la littérature sur la physique moléculaire ou encore dans [Ton91]. Cette interaction exhibe une partie répulsive à courte distance et une partie attractive à plus longue distance suivie d'une partie à très longue distance où la force est nulle. Elle fait apparaître, entre les deux zones, attractive et répulsive, une zone qui, combinant les deux effets, représente l'état cohésif. Entre les parties attractive et répulsive, la distance r_0 est une distance d'équilibre où aucune force ne s'applique, la distance entre les particules tendant à être ramenée à r_0 si une perturbation est appliquée. La cohésion est rompue si la distance entre les masses ponctuelles est supérieure à $2r_0$. Cela correspond à une libération comme la physique moléculaire en observe. Cette libération ne peut pas être obtenue à l'aide d'une simple interaction linéaire.

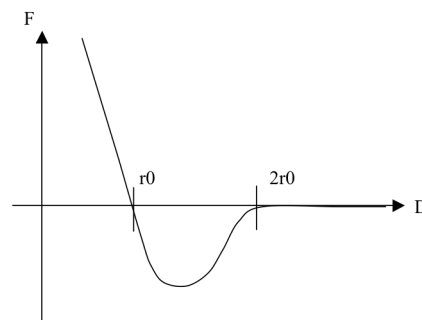


Figure 18 : Fonction d'interaction de Lennard-Jones pour $n=12$ et $m=6$.

II.2.4. Interactions cohésive irréversible ou de fracture

Derrière l'idée de fracture, il y a l'idée de rupture irréversible d'un lien entre deux entités. À ce titre, l'interaction du type Lennard-Jones est à même de rendre compte de ce type de comportement de libération : pour un certain seuil dépassé, plus aucune force n'est appliquée. Pour rendre des phénomènes de fracture, la littérature en animation par modèle physique s'est souvent contentée de la confondre avec l'idée de libération. Dans les travaux de Terzopoulos et Fleisher [TF88,TF'88], les auteurs montrent comment leurs modèles peuvent représenter des fractures en mettant à zéro le

coefficient d'élasticité entre deux nœuds adjacents quand un certain seuil de force d'un modèle linéaire élastique. Une idée similaire est utilisée par Norton et al. [NTB*91] qui utilisent un réseau de masses en interaction viscoélastique pour leurs modèles. Pour simuler une fracture, les auteurs suppriment les ressorts liant deux masses quand la distance entre ces deux masses a , elle aussi, dépassé un certain seuil. Pour créer une fracture plus complexe, ils répercutent, selon une analyse géométrique, cette rupture à certaines interactions voisines.

Mais d'un point de vue dynamique, une fracture n'est pas simplement la libération d'une liaison entre éléments de matière. A la limite de son seuil d'élasticité, la fracture résiste. Il convient donc d'introduire une nouvelle famille d'interactions présentant des paliers répulsifs à la suite de zones de cohésion. Ce palier répulsif agit alors aussi comme « un mur » nécessitant un effort de résistance qui empêche les deux masses ponctuelles de se désolidariser avant que la rupture adienne brusquement. Ce terme supplémentaire empêche de surcroît que les deux masses se solidarissent à nouveau lors d'une rencontre. La libération est donc en quelque sorte irréversible ce qui nous mène à appeler cette interaction une interaction cohésive irréversible. On peut retrouver dans la littérature de telles interactions comme par exemple dans Luciani et al. [LG97] ou dans Reynolds [Rey97]. Reynolds propose de rajouter plusieurs paliers successivement répulsifs et attractifs (cf. figure 19a) pour modéliser plusieurs états irréversibles dans la fracture. Luciani montre en fait qu'une famille de fonctions présentant un seul palier répulsif est largement suffisante pour modéliser la plupart des fractures (cf. figure 19b).

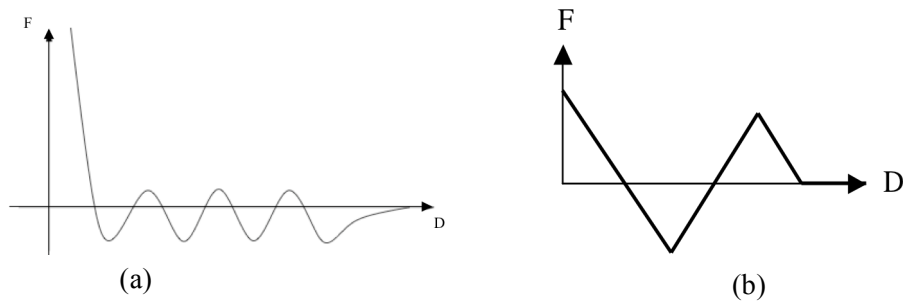


Figure 19 : Interactions de cohésion irréversible.
(a) selon Reynolds [Rey97] et (b) selon Luciani [LG97]

II.2.5. Interactions à hystérésis modélisant une plasticité

Un grand nombre de matériaux (solides, granulaires, etc.) soumis à une contrainte présentent un domaine de réversibilité limité c'est-à-dire qu'au delà d'un certain seuil de contrainte, le matériau ne revient pas à sa forme initiale et présente une déformation plastique. Ce seuil est un seuil limite d'élasticité pour un comportement élastique du matériau. Les courbes de charge et de décharge ne sont plus les mêmes et la caractéristique contrainte en fonction de la déformation présente alors une hystérésis. Par la suite, si on effectue de nouvelles contraintes qui ne dépassent pas le seuil d'élasticité

(qui peut être différent ou similaire au précédent) le comportement du matériau est à nouveau élastique. Le matériau présente aussi une zone appelée zone d'écrouissage fortement non linéaire qui, si elle est, dépassée amène à sa rupture. La figure 20 présente le profil contrainte(Q)-déformation(q) d'un tel comportement.

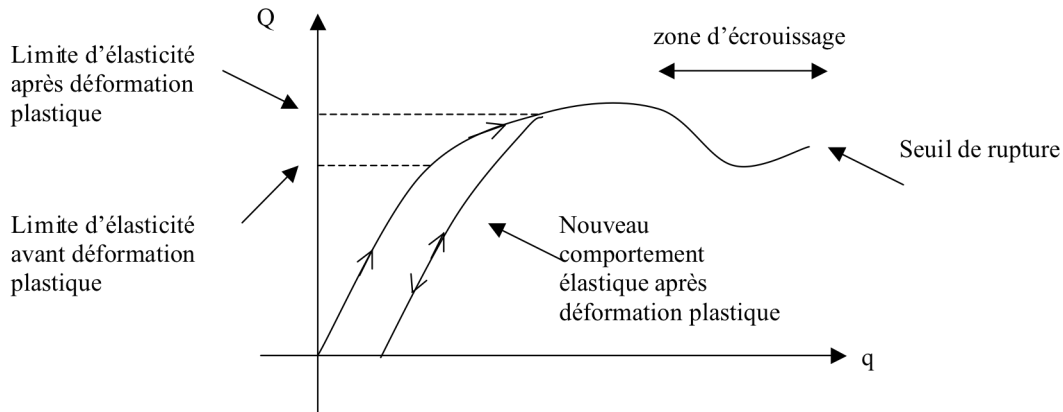


Figure 20 : Comportement d'un matériau subissant une déformation plastique.

Dans le cadre d'un formalisme masses - interactions, Jimenez [Jim93] développe une interaction rendant compte de ce phénomène dynamique restreint à sa partie purement plastique (pas de phénomène d'écrouissage ni de rupture). Il utilise l'image d'un patin, solide pesant frottant sur son support mis en série avec deux ressorts de raideurs k et k' comme le montre la figure 21.b. Dans le cas où il n'y a pas de ressort k' , l'interaction est une plasticité parfaite ne faisant pas intervenir de modification du seuil d'élasticité (cf. figure 21.a). Visuellement, cette plasticité parfaite est suffisante pour rendre un comportement plastique. Ce modèle simple est linéaire lorsque le patin ne glisse pas. Dès lors qu'il glisse, la longueur au repos du système global équivalent est modifiée. Algorithmiquement, cela implique la modification d'un paramètre de longueur au repos pendant la simulation. L'auteur ne propose pas de modèles faisant intervenir un phénomène d'écrouissage. Un tel modèle sera obtenu dans le chapitre 5 au paragraphe II.3.4, en faisant intervenir une modification de paramètres en ligne.

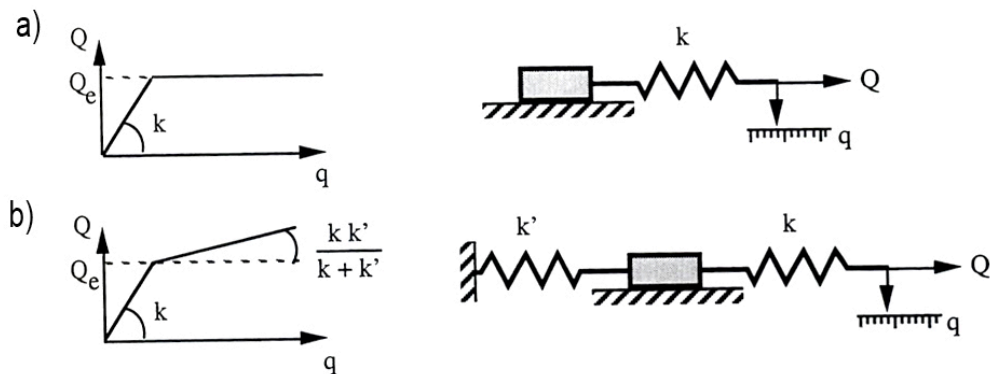


Figure 21 : Modèle élasto-plastique parfait et élasto-plastique avec écrouissage.

II.3. Interactions non linéaires dissipatives

II.3.1. Des interactions dissipatives

Les interactions non linéaires visqueuses appliquent des forces qui sont fonctions de la différence des vitesses entre les masses ponctuelles. Ce sont des interactions dissipatives puisqu'elles font perdre au système de l'énergie cinétique. Il est souvent nécessaire d'ajouter un terme dissipatif à une interaction potentielle pour modéliser la plupart des phénomènes réels. Dans la littérature de la modélisation physique pour l'animation, le rôle de la viscosité est souvent sous-estimé et traité rapidement. Elle a tendance à être considérée et utilisée pour améliorer la convergence des algorithmes discrets. Cela est vrai d'ailleurs dans d'autres cadres que celui de la modélisation particulières. Ainsi certaines méthodes d'intégration numérique comme les méthodes implicites vont permettre d'augmenter le pas temporel de la simulation tout en conservant la convergence des algorithmes. Mais ces méthodes vont introduire une viscosité cachée. Certains auteurs comme [DSB99] considèrent qu'il n'est pas très important d'évaluer et de contrôler cette viscosité puisqu'elle est nécessaire à un comportement réaliste et se retrouve dans tous les phénomènes naturels.

Dans une approche très radicale de la modélisation particulière telle que celle de Greenspan, la viscosité ne semble pas nécessaire à un niveau microscopique puisqu'il s'agit d'un phénomène émergent au niveau macroscopique. Or pour que ce phénomène émerge, un très grand nombre de particules est nécessaire. Greenspan introduit donc une viscosité ambiante pour limiter le nombre de particules. Certaines approches en modélisation particulières ont utilisé la viscosité comme une donnée de base de la modélisation de phénomènes dynamique. Les travaux de Tonnesen [Ton91] proposent d'adjoindre à une interaction de type Lennard-Jones, un terme visqueux dépendant de la vitesse relative des particules. Dans [MP89], Miller et al., le terme visqueux est de plus non linéaire et dépend d'un seuil en distance. Mais c'est certainement l'approche proposée par Luciani et al. [LJF*91, Luc00] qui porte le plus d'attention à l'introduction de la viscosité dans le processus de modélisation. Dans la mesure où l'interaction visqueuse est une interaction générique de base, par définition, du formalisme CORDIS-ANIMA, le praticien peut décider de l'introduire entre deux particules, modélisant ainsi une interaction visqueuse interparticulaire ou entre une particule et son milieu, modélisant ainsi une viscosité ambiante. Les auteurs ont pu ainsi montrer par exemple que la viscosité interparticulaire jouait peu dans les comportements de matériaux granulaires (au profit d'une viscosité ambiante) et inversement qu'elle jouait un rôle prépondérant dans les comportements de turbulences fluides. Le traitement générique de la viscosité comme un composant au même titre que l'élasticité permet ainsi au concepteur de modèle d'adapter son modèle au phénomène recherché.

De la même manière que la seule interaction potentielle linéaire (interaction élastique) ne suffit pas à la modélisation de phénomènes complexes, une interaction linéaire dissipative, ne suffit pas à la modélisation de comportements dissipatifs complexes. La non-linéarité peut être tout autant nécessaire. Mais, les phénomènes dissipatifs étant traités par les sciences physiques comme des phénomènes macroscopiques, il est encore moins courant pour eux d'en présenter une vision particulière et d'en proposer une typologie basée sur une littérature. Nous décidons donc de distinguer parmi ces

interactions celles qui font intervenir des critères de non linéarité portant sur les vitesses ou sur les distances. Cette différenciation qui a des implications sur les effets obtenus est aussi une différenciation dans la manière de représenter ces interactions. Pour finir et de manière un peu à part, nous présenterons les interactions de frottement sec, phénomène émergeant très présent dans les sciences physiques qui ne peuvent avoir son équivalent au sein d'un module d'interaction dans un formalisme particulière qui si on rompt le principe de force axiale qui participait à la garantie du physicalisme du formalisme CORDIS-ANIMA.

II.3.2. Interactions non linéaires visqueuses avec seuils en vitesse

Les interactions non linéaires visqueuses avec seuils en vitesse peuvent se représenter sous forme d'un graphe similaire au graphe force-distance des interactions non linéaires potentielles. Ce graphe donne la valeur de la force en fonction de la vitesse relative des masses ponctuelles (cf. figure 22). La modélisation physique particulière pour la génération de phénomènes sonore en fait un grand usage. On retrouve des interactions non linéaires visqueuses faisant intervenir des seuils en vitesse notamment dans la modélisation du frottement d'archet [FH01]. On retrouve pourtant dans la littérature certains cas d'utilisation de telles interactions. Dans [DLC93], Delnondedieu et al. montrent que l'utilisation d'interaction non linéaires visqueuses est une condition nécessaire pour permettre la locomotion de modèles moteurs minimaux se déplaçant dans un milieu visqueux. L'interaction avec le milieu et ses propriétés dissipatives non linéaires sont des composants fondamentaux de la locomotion qui déterminent sa nature et sans lequel elle ne peut avoir lieu. Le milieu visqueux doit en effet fournir moins de résistance à la vitesse en début de mouvement pour permettre une impulsion nécessaire à l'initiation d'un mouvement de locomotion.

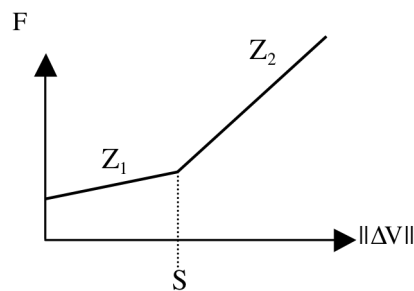


Figure 22 : Interaction non linéaire visqueuse par morceaux intervenant dans la locomotion.

II.3.3. Interactions non linéaires visqueuses avec seuils en distance

Nous avons vu qu'une interaction non linéaire visqueuse représentable par un graphe force-vitesse n'avait pas le même niveau d'importance qu'une interaction non-linéaire potentielle pour la génération de comportements dynamiques visuels, puisqu'elle n'adresse pas la question de la spatialité. Or ce n'est pas le cas des interactions non linéaires visqueuses présentant des valeurs de viscosité dépendant

de la distance entre les masses ponctuelles donc de la spatialité. De telles interactions ne peuvent plus être représentées par un graphe 2D car elles font intervenir différents états qui dépendent des distances entre les masses ponctuelles.

Dans [Luc00], l'auteur propose de modéliser 5 états de la matière : l'état gazeux, liquide, solide mais aussi granulaire et semi-solide (gels, pâtes...) à partir du formalisme CORDIS-ANIMA. Pour réaliser cela, l'auteur montre que la combinaison d'une interaction élastique seuillée en distance avec une interaction visqueuse elle aussi seuillée en distance mais pas nécessairement avec le même seuil permet d'obtenir ce vaste domaine de comportements. En jouant par exemple sur le rapport entre les deux seuils et pour des valeurs d'élasticité et de viscosité différentes, l'auteur passe de l'état de sables ou de fluides à l'état de pâtes et de crèmes. Le critère de distance au sein d'une interaction non linéaire visqueuse allié à une interaction non linéaire potentielle permet donc à l'auteur d'obtenir un nombre de comportements dynamiques très vaste allant des phénomènes collectifs d'empilement aux comportements fluides et turbulents.

Dans [HLT*03], déjà évoqué au paragraphe I.2.2, les auteurs adjoignent aussi aux interactions potentielles non linéaires d'attraction et de répulsion des interactions non linéaires visqueuses avec seuils en distance. En effet, le comportement de foules obtenus ne dépendent pas uniquement de la fonction d'interaction élastique non linéaire mais également et de manière importante de la fonction de viscosité associée portant les variations de vitesse relative de chaque individu en interaction.

II.3.4. Interactions de frottement sec

Le frottement sec est un phénomène complexe qui apparaît, dans le mouvement de deux surfaces en contact. De nombreux travaux, encore aujourd'hui, étudient son comportement au niveau microscopique. Il ressort de ces études que différents phénomènes microscopiques peuvent être observés que nous classons tous, au niveau macroscopique, sous le terme générique de frottement sec. Au XVIIIème siècle, Charles de Coulomb propose déjà un modèle de comportement global de ces phénomènes. Les forces de frottement entre deux surfaces sont proportionnelles à la réaction normale des surfaces et indépendantes de l'aire de contact. On considère alors que ces forces sont dues à l'enchevêtrement des aspérités microscopiques des surfaces de contact. Le modèle fait intervenir deux états du frottement, l'un statique et l'autre dynamique. Deux objets sont dans un état de frottement statique lorsque les deux surfaces sont immobiles, l'une par rapport à l'autre. La force permettant cet « accrochage » est inférieure à un seuil limite lui-même proportionnel à la force normale N maintenant les corps l'un contre l'autre.

$$F_{stat} < c_s \cdot N$$

Si le seuil limite en force est dépassé, les corps glissent l'un sur l'autre. Le frottement est alors dynamique mais indépendant de la vitesse de glissement, proportionnel à la force normale et dans la direction de la vitesse tangentielle relative \vec{V}_{tg} entre les deux surfaces.

$$\vec{F}_{dyn} = c_d \cdot N \cdot \frac{\vec{V}_{tg}}{\|\vec{V}_{tg}\|}$$

Cette caractérisation du phénomène est donc basée sur un critère géométrique : dans le régime statique, les positions relatives des surfaces en contact doivent rester constantes. Ce type de critère ne peut pas s'intégrer directement dans le cadre d'une simulation dynamique de réseaux masses - interactions au sein duquel la notion de surface n'a pas nativement sa place. Les travaux réalisés par Baraf dans [Bar91] qui vise à prendre en compte, par des méthodes analytiques, ce phénomène de frottement sec entre solides rigides, sont une illustration de la difficulté à mêler les deux approches.

Pour résoudre cette question, Jimenez, développe dans [Jim93] un module de frottement qui, dans l'état statique, calcule une force venant s'opposer aux mouvements relatifs tangentiels entre deux masses. Ce modèle de liaison non-axiale a été mis au point pour décrire de façon « macroscopique » une interaction qui au niveau microscopique met en jeu un grand nombre de points et de liaisons axiales entre ces points (modèles d'ailleurs testés dans [CLH96]). On pourrait rapprocher ce procédé de modélisation d'un procédé de variation structurelle de changement d'échelle : remplacement d'un modèle microscopique complexe par un modèle macroscopique simple par souci de l'économie de la représentation.

Pour cela, l'auteur adjoint au module de type butée, c'est-à-dire à une interaction non linéaire potentielle simple avec un seuil en distance (comme présenté au paragraphe III.2.1), une force d'interaction tangentielle F_{acc} (force d'accrochage) s'appliquant quand l'interaction de butée applique des forces non nulles entre deux masses ponctuelles. Tout se passe comme si l'interaction de butée définissait deux sphères avec deux points virtuels de contact appartenant chacun à l'une de ces sphères. L'interaction de frottement sec ainsi obtenue peut se trouver dans 3 états différents : un état libre dans lequel la distance entre les masses est supérieure au seuil de la butée, un état de frottement statique et un état de frottement dynamique pour lesquels cette distance est inférieure au seuil. L'interaction de frottement applique une force tangentielle s'apposant aux mouvements relatifs des points virtuels et du type le plus simple, c'est-à-dire viscoélastique (cf. figure 23). Lorsqu'un certain seuil de force est dépassé, la force appliquée devient \vec{F}_{dyn} .

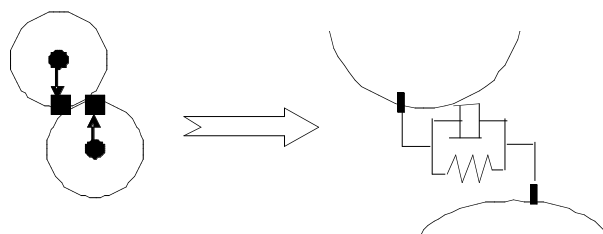


Figure 23 : Composante d'accrochage de l'interaction de frottement sec.

II.4. Modification de paramètres en fonction des variables de la simulation

Nous avons vu que l'utilisation de fonctions non linéaires complexes est une fonctionnalité cruciale pour qu'un formalisme masses - interactions puisse prétendre à une véritable généralité. La plupart des fonctions de ces interactions calculent des forces en fonction d'une condition sur les variables d'entrée (positions, vitesses). Mais jusqu'ici, les paramètres de ces interactions sont fixés une fois pour toute avant la simulation. La modification des paramètres en ligne est une fonctionnalité fondamentale qui permet d'obtenir de nombreux phénomènes dynamiques complexes : oscillations non linéaires comme les oscillateurs de Van der Pol, changement d'état [TPK89], phénomènes de relaxation, de motricité [Mil88] ou encore de croissance [BL03]. Dans [TPK89], les auteurs proposent de modifier le changement d'état d'un objet déformable modélisé par un système masses - interactions et soumis à une quantité de chaleur qui le fait fondre. La fonte de l'objet est obtenue en faisant décroître le paramètre de raideur des ressorts jusqu'à un certain seuil pour lequel le ressort est supprimé et remplacé par une interaction de cohésion de type Lennard-Jones. Les modèles de serpents et de vers de Miller [Mil88] sont mis en mouvement en modifiant en ligne la valeur des longueurs à vide des ressorts entrant dans la modélisation des créatures. La longueur à vide, paramètre de la simulation, devient une fonction du temps, une fonction sinusoïdale par exemple. La motricité est ici générée par une variation interne de l'énergie du système (variation apportée à l'intérieur de l'interaction). On retrouve ce principe de modélisation dans des logiciels comme Sodaplay qui permet de construire des créatures du même type. Les phénomènes de croissance sont également modélisés par Boulenguez et al. dans [BL03]. On y observe notamment des changements de taille, des changements d'état après croissance, des accrochages entre éléments. La modélisation physique d'un tel processus fait intervenir deux types d'interaction: les interactions intracellulaires, qui vont déterminer la croissance de la cellule et les interactions intercellulaires, qui vont déterminer l'accrochage entre cellules et donc influencer sur leur croissance. Les longueurs au repos des interactions intracellulaires permettent de régler l'évolution de l'énergie interne de la cellule. Modifier un paramètre durant la simulation peut en revanche poser des questions non triviales. En effet, une telle modification conduit à une variation de l'énergie interne du système dont il faut avoir conscience et qu'il s'agit d'être capable de quantifier. Il se peut aussi qu'une modification de paramètres en ligne conduise à franchir les limites de convergence des algorithmes du réseau masses - interactions. Ainsi, permettre la modification des paramètres au cours de la simulation peut rendre les modèles instables et incontrôlables pour le modélisateur novice.

III. Modalités de représentation au sein d'un environnement de conception interactive

III.1. Un outil de conception et de représentation générique pour les interactions ?

Incrémenter les fonctionnalités d'un logiciel interactif pour la création nécessite certaines précautions. Il convient de ne jamais perdre le caractère générique de l'interface au risque de créer un logiciel « fourre-tout » où les fonctionnalités s'empilent, les métaphores de manipulations et d'interactivités foisonnent. Car le danger est alors de perdre l'utilisateur dans des méandres de comportements d'interface qu'il pourra à juste titre trouver déroutants voire rebutants ou qu'il n'explorera tout simplement pas. Or, notre analyse précédente des possibilités offertes par la modélisation de réseaux masses - interactions nous montre qu'il est nécessaire d'inclure des interactions complexes dans le cœur du processus de création et que ces interactions ont des natures assez dissemblables (hystérésis, forces non-axiales, seuils en vitesse ou en distance) même si elles ont un point topologique commun, celui de produire des forces en sortie, fonctions des entrées de position fournies par deux masses ponctuelles.

Devant cette diversité, on peut être amené à considérer qu'il est nécessaire de permettre à l'utilisateur de décrire lui-même des interactions à l'aide de composants fonctionnels de base à assembler pour créer une interaction particulière. Le module d'interaction participant au réseau masses - interactions devient alors une boîte programmable. Ce type d'approche a déjà été proposé par Pommereuil et al. [PL99]. Or il nous faut remarquer que les outils développés par ces auteurs, bien que disponibles au sein de l'ACROE-ICA, n'ont jamais été intégrés dans le développement des environnements de conception du laboratoire. On pourrait trouver des raisons conjecturelles à cela. Il est en effet assez fréquent que des développements de laboratoire ne conduisent à aucune utilisation en l'état puisqu'ils sont souvent des éléments de validation de concepts avant une mise aux usages longtemps après, de l'ordre de la dizaine d'années. Mais il existe une raison beaucoup plus profonde. En effet, si une certaine variété de fonctions d'interaction est nécessaire et doit être mise à disposition des utilisateurs sous peine de disposer d'un formalisme trop pauvre pour générer une variété de comportements suffisamment riche, cette variété est limitée en nombre de catégories ou, en tout cas, de catégories connues à ce jour et dont l'utilité a été validée.

Ainsi une interface de conception d'interactions quelconques ne peut être proposée qu'à un utilisateur expert de ce type de modélisation et dont l'expertise lui permettra d'argumenter la nécessité de la création éventuelle d'un nouveau type de modules d'interaction. A l'inverse, un utilisateur moins expert ou en cours d'apprentissage du processus de modélisateur pourrait être amené vers de fausses pistes de modélisation. En effet, un outil évolué de conception d'interactions complexes au cœur du modèleur peut l'amener à croire que le seul cœur de la modélisation de comportements complexes réside dans la définition d'interactions complexes. Un tel outil pourrait donc le détourner d'une confrontation nécessaire à d'autres points critiques de la modélisation comme par exemple la définition d'une topologie adéquate et ou encore le réglage des paramètres physiques. Voilà plutôt notre utilisateur dessinant des fonctions d'interactions fortement non linéaires, avec des automates d'états complexes, ne se préoccupant plus de la signification physique de ce qu'il conçoit et n'obtenant très certainement au final pas grand-chose. De plus, et Pommeureuil le souligne, un outil générique de conception d'interactions peut aussi créer des interactions ne garantissant plus la physicalité du module.

De manière complémentaire à cette approche et sur la base de la typologie que nous avons présentée dans la partie précédente, il semble donc nécessaire de proposer à l'utilisateur une base d'interactions déduite de notre typologie. Il convient, pour les familles qui la composent, de trouver la modalité de représentation la plus compréhensible pour l'utilisateur. Cela passe par une réflexion sur la manière de présenter les fonctions d'interactions, sur le choix des paramètres à proposer et sur la manière de manipuler ces paramètres. Il convient de surcroît de nous assurer que les choix des représentations, des paramètres et de leur manipulation n'affectent pas la possibilité de la mise en œuvre du corpus des modèles existant déjà. Notre environnement doit en effet permettre de mettre en œuvre les modèles du passé tout en permettant une exploration de nouveaux modèles.

Cette partie reprend donc la typologie établie dans la partie précédente et discute des questions de représentation et de mise en œuvre de chaque catégorie de cette typologie. Nous montrerons notamment la possibilité de choisir de travailler avec des fonctions linéaires par morceaux et l'intérêt que cela implique sur la compréhensibilité de paramètres bien choisis. Nous verrons aussi en quoi la manipulation graphique de tels paramètres aide à renforcer cette compréhensibilité. Nous présenterons de plus notre choix de variétés d'interactions mises à disposition de l'utilisateur.

III.2. Interactions non-linaires potentielles

III.2.1. Linéarisation par morceaux

La plupart des interactions potentielles non linéaires que nous avons présentées exhibent une fonction d'interaction complexe (inverse du carré de la distance, voire puissance supérieur). Cela amène à des profils continus pour les fonctions. Mais est-ce que ces profils continus sont véritablement nécessaires pour l'obtention des comportements dynamiques souhaités. Prenons l'exemple du cas de l'interaction

de Lennard-Jones dans une configuration d'interaction cohésive. L'information pertinente est-elle dans le profil continu et complexe de la fonction d'interaction ou dans le fait qu'elle présente une zone de répulsion suivie d'une zone d'attraction ainsi que deux seuils, l'un de cohésion, l'autre de libération ? Du point de vue de la dynamique recherchée, nous penchons pour le second point. Dès lors, si la linéarisation par morceaux est bien choisie, la fonction d'interaction obtenue suffit à générer le comportement dynamique de cohésion. Bien choisir cette linéarisation dans ce cas revient à ce que l'interaction conserve une zone répulsive suivie d'une zone attractive et que les seuils soient préservés comme le montre la figure suivante :

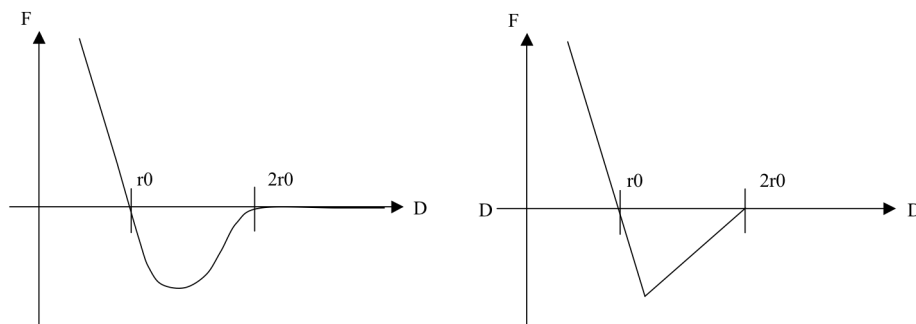


Figure 24 : Linéarisation par morceaux de l'interaction cohésive de Lennard-Jones.

À gauche : l'interaction cohésive de Lennard-Jones. À droite : la linéarisation par morceaux de cette interaction. Ses propriétés sont conservées.

De manière similaire, on peut imaginer une linéarisation par morceaux d'autres fonctions d'interaction potentielles comme les interactions gravitationnelles ou les interactions électriques entre particules chargées. Prenons le cas par exemple de l'interaction électrique répulsive. On peut considérer qu'au-delà d'un certain seuil en distance, la force appliquée à une influence négligeable sur le mouvement. De manière similaire, on peut considérer que pour des distances très petites, cette force sensée tendre vers l'infini peut avoir une valeur finie. Ainsi la linéarisation la plus simple serait l'interaction de butée. Une linéarisation aussi grossière peut être suffisante dans certains cas (rendre une répulsion), Mais on peut aussi souhaiter raffiner cette linéarisation en augmentant le nombre de morceaux, de manière à définir des zones plus ou moins répulsives, comme le proposent les modèles de phénomènes collectifs de foules dans [HLT*03].

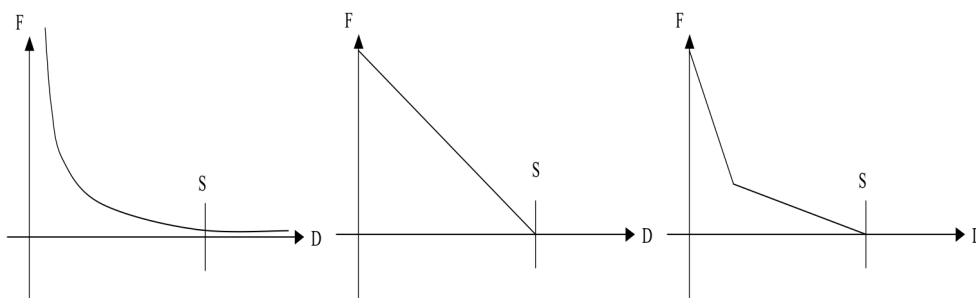


Figure 25 : Interaction de répulsion en $1/r^2$ et linéarisations par morceaux possibles.

Au centre : l'interaction de butée

De manière générale, l'utilisation de fonctions non linéaires potentielles mais linéaires par morceaux est suffisante pour modéliser les mêmes phénomènes dynamiques que leur version fortement non linéaire à condition que cette linéarisation soit conçue en respectant la fonctionnalité de l'interaction. Mais le fait de travailler avec de telles fonctions d'interaction à un avantage supplémentaire, celui de proposer des profils de fonctions similaires aux interactions de base de CORDIS-ANIMA que sont la butée et l'interaction linéaire élastique. Nous allons voir dans le paragraphe suivant comment il est alors possible de choisir une base de paramètres pour ces interactions dont la signification est similaire aux paramètres de ces interactions de base.

III.2.2. Le choix des paramètres offerts à l'utilisateur

Quels paramètres choisir pour définir une fonction d'interaction de manière à ce que ces paramètres aient un sens fort, dont un utilisateur puisse se faire une représentation mentale ? Prenons les cas de base de CORDIS-ANIMA que sont la butée et l'interaction linéaire élastique. Dans les deux cas, les pentes de ces fonctions d'interaction donnent le paramètre d'élasticité K (voir figure 26). qui fait référence au concept de raideur comme on peut l'expérimenter dans la vie quotidienne. Dans le cas de la butée, le second paramètre est le paramètre de seuil en distance S au-delà duquel les forces appliquées sont négligeables. Une distance a elle aussi un sens fort et significatif. De même, le second paramètre de l'interaction linéaire élastique est une distance au repos, référent à un critère spatial et significatif. Ces paramètres suffisent à définir ces interactions. On aurait pu choisir de représenter ces fonctions à l'aide de points dans l'espace bidimensionnel force-distance. Mais il aurait alors fallu faire intervenir un paramètre référant à une intensité de force. Un tel paramètre est lui beaucoup moins compréhensible car nous parlons rarement en terme de forces dans le langage courant. On dit d'un choc qu'il est dur ou mou plutôt que de forte intensité à une distance donnée de pénétration. Le choix des bases de paramètres (K, S) ou (K, L) sont donc des choix judicieux pour permettre à l'utilisateur de se faire une représentation mentale de leur signification. A l'inverse de ce choix, si l'on se réfère au choix des paramètres m et n pour définir une interaction de Lennard-Jones comme présentée au paragraphe précédent, il est très difficile de comprendre la signification de tels paramètres et leur influence sur le comportement de l'interaction.

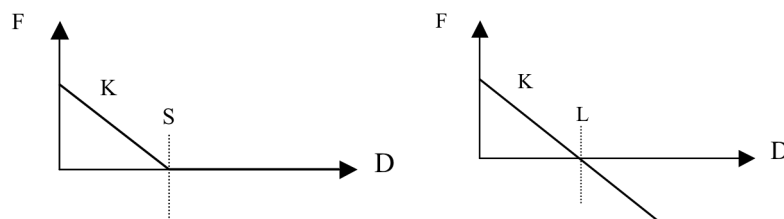


Figure 26 : Le choix des paramètres pour des interactions simples.

L'interaction de butée (à gauche, K et S) et l'interaction linéaire élastique (à droite, K et L)

En faisant le choix de proposer à l'utilisateur des fonctions d'interaction potentielles linéaires par morceaux, les différents morceaux peuvent être définis par leur pente et des points leurs appartenant. Il devient alors possible de définir une base de paramètres similaire à la butée ou à l'interaction linéaire élastique, c'est-à-dire de parler en termes d'élasticité, de seuil de distance, de distance au repos.

Étudions, pour nous en convaincre, la formalisation de l'interaction de cohésion que nous proposons au paragraphe III.2.1, linéarisation par morceaux de l'interaction de Lennard-Jones. Les pentes de cette interaction peuvent être associées à des paramètres d'élasticité. Plus la pente de la partie répulsive sera importante, plus l'interaction sera dure et résistera à la pénétration. Mais quels points de la courbe choisir pour faire office de paramètres spatial ? Ce qui importe dans cette interaction, nous l'avons vu, ce sont les deux distances l'une de libération et l'autre de cohésion. La première correspond à un seuil au-delà duquel les forces appliquées sont nulles. Sa fonction est donc similaire au seuil S de la butée. La seconde correspond à une distance d'équilibre. Sa fonction est elle similaire à la distance au repos de l'interaction élastique linéaire L . Nous disposons donc avec les paramètres S , L et les deux élasticités K_1 et K_2 correspondant aux pentes des deux portions linéaires, d'une base de paramètres compréhensible faisant référence à des comportements que l'utilisateur a pu déjà expérimenter en utilisant des interactions plus simples (cf. figure 27). Ces paramètres réfèrent, de plus, au comportement de l'interaction. Nous aurions pu choisir, à la place du seuil de libération, la distance D pour laquelle la force est la plus attractive (cf. figure 27) mais ce paramètre est beaucoup moins significatif de la cohésion.

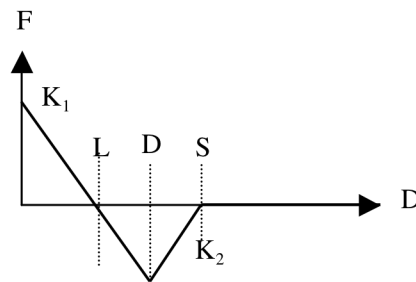


Figure 27 : L'interaction de cohésion et ses paramètres K_1 , K_2 , S et L .

Le paramètre D aurait pu être choisi, mais il est beaucoup moins significatif du comportement de cohésion.

Un deuxième exemple différent, ne faisant pas intervenir de point d'équilibre peut être pris. Considérons l'interaction de répulsion à 3 zones proposée par [HLT*03] (cf. figure 28). Cette interaction présente deux pentes non nulles que nous pouvons donc considérer comme représentant une fonction d'élasticité. De la même manière que pour l'interaction de cohésion, au-delà d'un seuil S , les forces appliquées sont nulles. Mais il nous manque un paramètre pour définir totalement la fonction d'interaction. Nous pourrions choisir F_{max} la force maximale de l'interaction (cf. figure 28), mais nous avons vu qu'il était difficile de raisonner numériquement en termes de forces. En revanche, l'abscisse du point faisant la jonction entre les deux parties de la courbe séparent deux zones répulsives de raideur différentes. La fonction de ce paramètre de distance est donc celle d'un seuil que nous nommons S_1 .

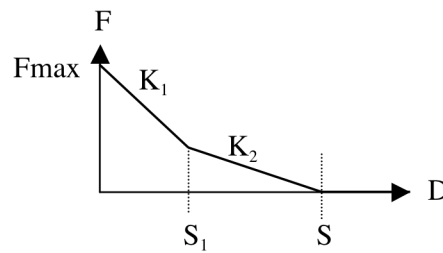


Figure 28 : Interaction de répulsion à 3 zones et sa base de paramètres K_1 , K_2 , S_1 et S .

Nous voyons donc qu'il est possible de définir une base de paramètres compréhensibles par l'utilisateur, faisant référence à un vocabulaire du quotidien.

III.2.3. Familles d'interactions non linéaires potentielles choisies et mise à disposition de l'utilisateur

Nous avons vu qu'une interaction non linéaire potentielle pouvait se représenter la plupart du temps à l'aide d'un graphe force-distance. Dans un souci de balayer le plus grand nombre de cas possibles, on pourrait imaginer proposer à un utilisateur de construire lui-même le graphe de son interaction. Or, pour les mêmes raisons qui nous ont amené à mettre en garde l'utilisation d'un outil d'édition général d'interactions notamment pour les utilisateurs novices en modélisation masses - interactions, un outil général d'édition d'interactions non linéaires potentielles n'a de sens que pour un utilisateur expert en modélisation qui sait justifier de la nécessité d'un nouveau type d'interaction. Un tel outil n'ayant pas été implanté par le passé, nous en proposons ici l'étude au paragraphe III.4.3. Mais il est de la première importance de proposer à l'utilisateur, notamment novice, des familles d'interactions dont l'utilité a été démontrée par l'expérience et qui ont été décrites dans notre typologie. La partie suivante présente donc les familles d'interactions que nous mettons à disposition de l'utilisateur (attractions, répulsions, cohésions) avant de présenter l'interaction non linéaire potentielle générale. N'y est pas présente l'interaction plastique dont l'hystérésis n'est pas représentable par une fonction linéaire par morceaux.

Modules d'interaction d'attraction et de répulsion

Comme nous l'avons vu, une simple interaction linéaire par morceau et à deux morceaux du type de la butée, n'est pas forcément suffisante pour rendre correctement des phénomènes dynamiques faisant intervenir des phénomènes d'attraction et de répulsion. Une linéarisation faisant intervenir de 3 à 4 morceaux linéaires avec des pentes différentes est généralement largement suffisante pour rendre des attractions ou des répulsions complexes. Nous introduisons donc dans MIMESIS des interactions à 3 morceaux, une d'attraction (ATR3) et une de répulsion (REP3) et de même des interactions à 4 morceaux (ART4, REP4).

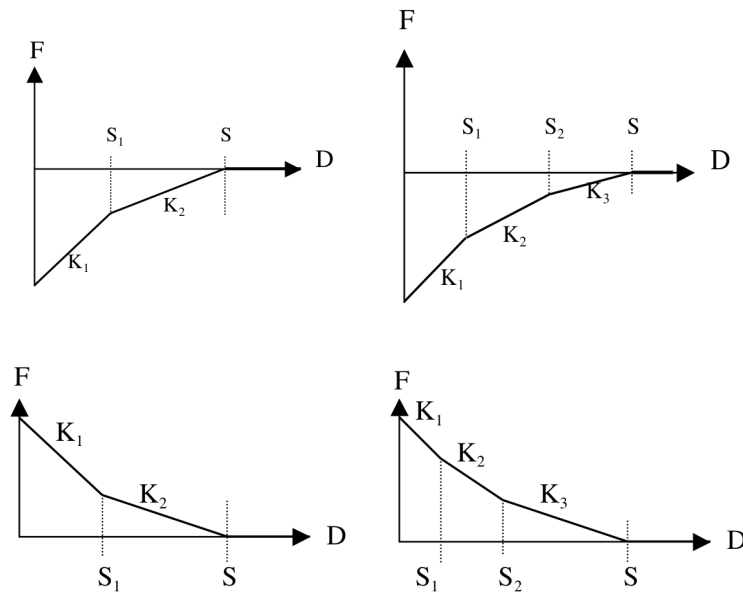


Figure 29 : Attractions et répulsions à 3 et 4 paliers dans *MIMESIS*.

En haut : les attractions. En bas : les répulsions.

Nous avons vu aussi qu'en choisissant comme paramètres les K_i , Z_i , S_i , et en nommant le dernier S_i , S , on conserve une référence au modèle de paramètres de la butée. La liaison répulsion peut, en effet être vue comme une « butée progressive » présentant des paliers, les seuils de distances. Entre chaque palier, la répulsion n'est pas aussi « raide », c'est en quelque sorte un module viscoélastique différent.

Modules d'interaction de cohésion

Nous déduisons de l'analyse présentées dans les paragraphes II.2.3 et II.2.4 expliquant l'utilité de deux familles d'interactions cohésives (cohésion de Lennard-Jones et cohésion irréversible), dont les versions linéaires par morceaux sont suffisantes à l'obtention des comportements dynamiques cohésifs, la mise en œuvre de telles familles d'interaction au sein de notre environnement interactif de modélisation.

La première famille présente donc trois zones : une première zone de répulsion, puis une zone d'attraction et enfin une zone libre. Nous avons présenté cette interaction au paragraphe II.2.4.

En rajoutant une zone répulsive après la zone attractive, nous obtenons une interaction de cohésion irréversible dont la mise en œuvre de la base de paramètres est calquée sur la cohésion de Lennard-Jones linéarisée par morceaux.

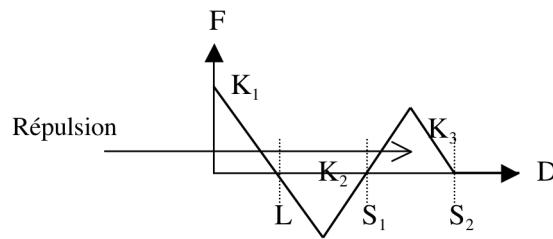


Figure 30 : Représentation fonctionnelle de l'interaction de cohésion irréversible linéaire par morceaux avec sa base de paramètres.

Module d'interaction potentielle linéaire par morceau quelconque

Notre analyse préalable fondée sur un état de l'art nous a amené à proposer un certain nombre d'interactions non linéaires potentielles. Rien ne peut nous le certifier qu'un utilisateur n'aura jamais besoin de construire une interaction n'appartenant pas aux familles que nous proposons.

Nous avons vu comment la linéarité par morceau et le choix d'une bonne base de paramètres, suit une logique globale qui se réfère directement aux modules de base de MIMESIS que sont la butée et l'interaction linéaire élastique. On définit des segments de droites (entre deux valeurs de distances) dont les pentes sont des paramètres de raideur. Créer une interaction potentielle linéaire par morceaux quelconque est donc un niveau conceptuel « juste un peu plus complexe » que l'utilisation de familles prédéfinie. Cela implique d'avoir intégré la logique des familles particulières pour créer sa propre interaction dont on est certain de la nécessité. Les interactions proposées ont appris à l'utilisateur ce qu'est une raideur, un seuil en distance et comment ces interactions peuvent être utiles lorsqu'elles sont un peu plus complexes que la butée. Proposer des interactions potentielles linéaires par morceaux et quelconques ne fait pas appel à d'autres concepts si ce n'est que l'espace est plus partitionné.

Par similitude avec les familles d'interactions prédéterminées, de telles interactions quelconques sont donc données par N triplets de paramètres (K_i, Z_i, S_i) , pour $i \in 1..N$ et avec N le nombre de segments. Par la donnée de ces N triplets, nous définissons un ensemble d'interactions dont les fonctions sont toutes les mêmes à une translation près, selon l'axe des forces (l'axe des ordonnées). Il manque un paramètre supplémentaire pour que la fonction représentant l'interaction soit unique. Au niveau de l'implémentation du module dans CORDIS-ANIMA, ce paramètre est la valeur F_0 de la force en $D=0$. Mais nous discuterons dans le paragraphe III.4.3, la manière dont nous permettons à l'utilisateur de construire sa propre interaction.

Le cas particulier de la plasticité

Nous avons vu que l'interaction élasto-plastique proposée par Jimenez dans [Jim93] était le modèle minimal permettant de rendre un phénomène d'hystérésis comme une déformation plastique. Cette interaction pouvait être vue comme une interaction linéaire dont le paramètre de longueur au repos L_{rep} varie quand un seuil en force ($SeuilPlast$) est atteint. Initialement, cette longueur au repos a une valeur L_0 . Le seuil est atteint pour des valeurs positives c'est-à-dire répulsives, la longueur au repos diminue. Il y a compression irréversible. Dans le cas où cette valeur est atteinte pour des valeurs de

force négatives c'est-à-dire attractives, la longueur au repos augmente. Il y a une extension irréversible. La figure 31 représente de manière fonctionnelle, le comportement d'une telle interaction. Nous avons représenté le cas où le seuil de plasticité aurait été dépassé dans le cas de forces positives et négatives. Dans ces 2 cas, nous avons dessiné en pointillés la courbe représentative de ce que serait l'interaction linéaire de longueur au repos L_{rep} correspondante. Tant que nous nous trouvons en régime saturé, c'est-à-dire tant que la force qu'appliquerait un module L_{rep} ne se trouve pas dans la plage $[-SeuilPlast, SeuilPlast]$, l'interaction subit une elongation d . La nouvelle valeur de L_{rep} est $L_{rep} = d - L_{plast}$ si la force est attractive et $L_{rep} = d + L_{plast}$ si la force est répulsive, avec $L_{plast} = SeuilPlast/K$.

Quels paramètres offrir alors à l'utilisateur ? Suivant le raisonnement déjà effectué pour d'autres interaction qu'il est difficile de penser en termes de valeur de forces, nous préférons le paramètre L_{plast} plutôt que le paramètre $SeuilPlast$. Le paramètre L_{plast} représente la longueur maximale au-delà de laquelle l'interaction modifie sa longueur au repos.

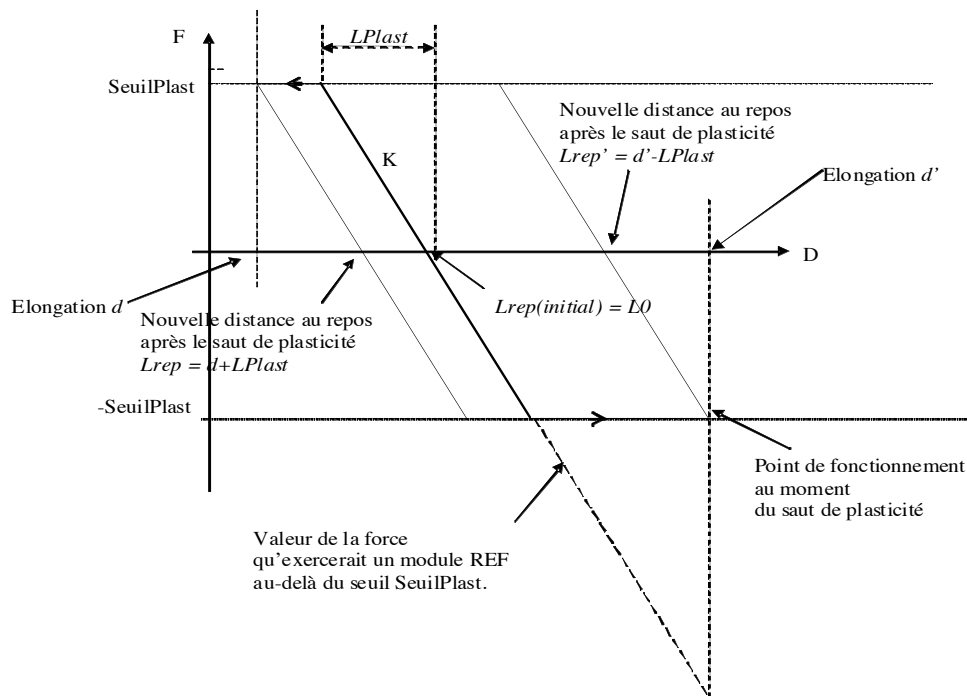


Figure 31 : Schéma de l'algorithme du module élasto-plastique.

III.3. Interactions non linéaires visqueuses

III.3.1. Module d'interaction visqueuse avec seuil en distance

Nous avons vu qu'il était essentiel de traiter la viscosité de la même façon que l'élasticité dans le cadre d'interactions présentant explicitement ce paramètre. Nous avons vu de plus la nécessité d'une viscosité non linéaire et le fait que pour l'animation, cette non-linéarité dépende en premier lieu de

seuils en distances plutôt que de seuils en vitesse. De telles interactions ne peuvent alors se représenter qu'en termes d'automates d'états.

Une interaction visqueuse non linéaire de ce type peut être utilisée en elle-même pour obtenir des phénomènes de turbulences. Dans [Luc00], l'auteur montre que l'interaction non linéaire visqueuse avec un simple seuil en distance est suffisante pour obtenir des tourbillons de Kelvin-Helmholtz. Mais l'auteur montre qu'en combinant une telle interaction non linéaire visqueuse avec une interaction non linéaire potentielle simple, il est possible d'obtenir un très grand nombre d'effets dynamiques : des fluides aux pâtes, des solides aux phénomènes collectifs. L'auteur montre alors qu'il y a 5 combinaisons possibles (IP1, IP2, IP3, IP4 et IP5) pour ces interactions simples et représentées dans la figure suivante :

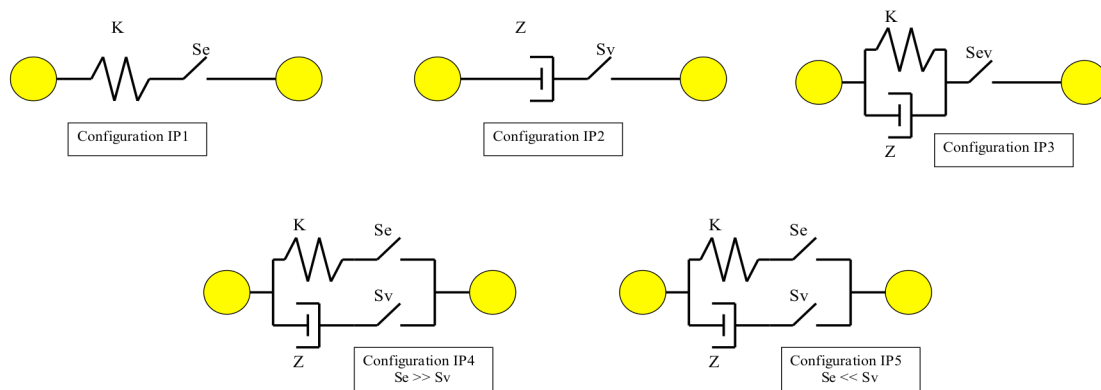


Figure 32 : 5 combinaisons possibles d'interaction non linéaires viscoélastiques simples pour l'animation.

Dans un souci de minimalité du nombre de modules dans notre environnement de conception interactive, nous avons dû faire un choix parmi l'ensemble de ces configurations. Nous avons choisi la configuration IP3 au sein d'un module de butée viscoélastique car il s'agit d'une représentation compacte à partir de laquelle les autres configurations peuvent être obtenues. En effet on obtient IP1 en mettant le paramètre Z à 0, IP2 en mettant le paramètre K à 0 et les configurations IP4 et IP5 en combinant en parallèle IP1 et IP2. En revanche, la minimalité de la représentation modulaire a un coût, celui de la non-minimalité algorithmique dans certains cas. Par exemple, lorsqu'on souhaite utiliser ce module dans sa version purement visqueuse, une multiplication par 0 est réalisée pour rien. Cela n'est pas vraiment problématique dans le cadre d'une simulation hors-ligne, d'autant plus que le calcul le plus coûteux la plupart du temps est celui des tests sur les distances, tests qui ne sont pas plus nombreux avec notre choix. Du point de vue de la représentation fonctionnelle de l'interaction non linéaire visqueuse, nous adjoignons alors aux graphes force-distance des interactions potentielles linéaires par morceaux et présentée dans la partie précédente, une plage découpée par les seuils en distance de ces interactions. Par similitude avec la butée viscoélastique choisie, chaque plage possède son propre paramètre de viscosité sauf lorsque la force de la plage est nulle. La figure suivante montre deux exemples de la représentation choisie de la viscosité non linéaire.

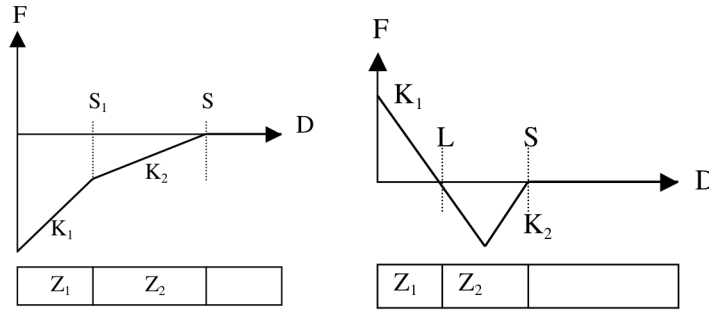


Figure 33 : Représentation fonctionnelle des interactions d'attraction et de cohésion avec adjonction de viscosité non linéaire.

III.3.2. Module d'interaction visqueuse avec seuils en vitesse

La littérature en synthèse d'image nous fournit beaucoup moins d'exemples d'utilisation d'interactions non linéaires visqueuses avec seuils en vitesse qui sont des interactions dont l'intérêt se situe plutôt pour les modèles masses - interactions produisant du son. Nous avons cité [DLC93] pour les modèles d'animation. Un point d'entrée de nos réflexions sur la manière d'offrir de telles interactions pourrait être l'exemple du logiciel GENESIS. Or, un point de différence fondamental du formalisme utilisé par GENESIS est que, dans celui-ci, les modules évoluent dans un espace adimensionnel. Il en résulte que GENESIS propose à l'utilisateur de construire une interaction non linéaire visqueuse dont la fonction représente une valeur de force en fonction de la différence des vitesses scalaires. Cette différence peut être positive ou négative. Dans le cas de MIMESIS, les vitesses sont vectorielles. Ainsi une différence entre deux vitesses ne peut être exprimée qu'en norme et est donc toujours positive. La fonction de l'interaction non linéaire visqueuse de notre module nommé LLMV, donnera donc la force appliquée par ce LIA en fonction de $\Delta V = \|\vec{V}_1 - \vec{V}_2\|$ où V_1 et V_2 sont les vitesses de chacun des deux MAT reliés. Par souci d'homogénéité avec le module d'interaction linéaire par morceaux quelconque, la fonction d'un module LLMV sera une fonction continue par morceaux comme le montre la figure suivante. Cette représentation est très générale car nous ne disposons pas aujourd'hui d'assez d'exemples en animation pour proposer des familles particulières.

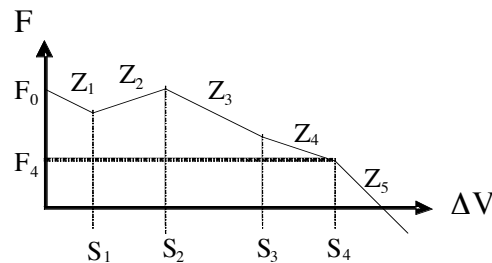


Figure 34 : Schéma de l'interaction visqueuse linéaire par morceaux quelconque avec seuils en vitesse.

III.3.3. Le cas particulier du frottement sec

Nous avons vu l'importance d'introduire dans notre environnement de modélisation l'interaction de frottement sec développée par Jimenez dans [Jim93]. Cette interaction non axiale fait intervenir des forces tangentielles. Il est donc nécessaire de calculer le vecteur tangent à l'axe passant par les deux masses ponctuelles. Or il existe, en 3D, une infinité de vecteurs tangents à une droite. A un échantillon N de la simulation, il nous faut choisir parmi ceux-ci, le vecteur $\overrightarrow{T(N)}$ tangent au mouvement relatif entre les deux masses ponctuelles. Ce vecteur est obtenu par projection du vecteur de la vitesse relative $\overrightarrow{V_r(N)} = \overrightarrow{M1(N)M2(N)} - \overrightarrow{M1(N-1)M2(N-1)}$ entre les deux masses ponctuelles sur le plan tangent à la droite passant par ces deux points comme le montre la figure suivante (représentation du problème dans le plan du mouvement relatif c'est-à-dire le plan porté par les vecteurs $\overrightarrow{M1(N)M2(N)}$ et $\overrightarrow{M1(N-1)M2(N-1)}$).

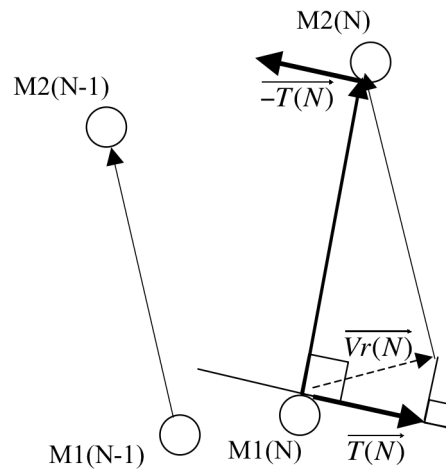


Figure 35 : Construction du vecteur $\overrightarrow{T(N)}$ tangent au mouvement relatif.

La force de frottement sec $\overrightarrow{Ff(N)}$ va s'appliquer sur $M1$ selon $\overrightarrow{T(N)}$ et sur $M2$ selon $-\overrightarrow{T(N)}$. Nous avons vu que le module de frottement proposé par [Jim93] fait intervenir une butée et qu'à l'image de la loi de Coulomb sur le frottement sec, la force de frottement va dépendre de la force exercée par cette butée. Cette force joue en quelque sorte le rôle de la réaction normale de la formulation de Coulomb.

Soit donc $\overrightarrow{Fb(N)}$ la force d'interaction de la butée entre les deux masses. Si la norme de cette force est nulle, nous considérons qu'il n'y a pas « contact ». Aucune force de frottement n'est appliquée.

Dans le cas contraire, il y a une force de frottement. Nous évaluons alors dans un premier temps si cette force de frottement est statique. Cette force de frottement statique est modélisée par une force de type visqueuse portée par $\overrightarrow{T(N)}$ et ayant tendance à faire tendre la vitesse relative tangentielle (qui est aussi le vecteur $\overrightarrow{T(N)}$) vers 0. L'expression de cette force de frottement statique est donc

$\overrightarrow{Ffs}(N) = Z.T(N). \frac{\overrightarrow{T(N)}}{\|\overrightarrow{T(N)}\|}$. Si la norme de cette force est supérieure à $Cs.Fb(N)$ où Cs est un coefficient statique, alors nous sommes bien en régime statique et la force $\overrightarrow{Ffs}(N)$ s'applique. Sinon nous sommes en régime dynamique et la force de frottement est alors une force de frottement dynamique $\overrightarrow{Ffd}(N) = Cd.Fb(N). \frac{\overrightarrow{T(N)}}{\|\overrightarrow{T(N)}\|}$.

Lors de notre implantation de l'algorithme de l'interaction de frottement sec, nous avons eu à nous confronter à la description de l'algorithme faite par [Jim93] ainsi qu'aux sources de celui-ci. Hormis des corrections d'erreurs de formule (notamment dans la projection de la vitesse relative) ainsi que des optimisations apportées à l'algorithme, nous nous sommes aperçus que les sources ne correspondaient pas entièrement avec la description que l'auteur donnait de l'algorithme. En particulier, il présentait l'interaction comme une interaction possédant un automate à états. Or il ne s'agit pas dans les faits d'un automate même si l'interaction peut se trouver dans plusieurs états. En effet, deux critères définissent l'état de l'interaction : le critère de contact et la comparaison d'une force de frottement statique à une force d'interaction de butée. Or dans le cas d'un automate à état, on passe d'un état à un autre en fonction de critères particuliers à l'état, ce qui n'est manifestement pas le cas dans les sources auxquels nous avons eu accès. Une deuxième imprécision théorique résidait dans le fait que l'auteur définissait la force de frottement statique comme une interaction viscoélastique de paramètres K et Z . Or le paramètre K était appliqué à la vitesse tangentielle relative, de même que le paramètre Z ce qui faisait du terme faisant intervenir K un terme du second ordre donc visqueux. Il en résulte qu'il y avait deux paramètres qui intervenaient tous les deux dans un calcul du second ordre. En fait le paramètre K était utilisé pour l'évaluation de la force de frottement statique alors que le paramètre Z était utilisé pour le calcul de la force de frottement statique effectivement appliquée. Mais cela ne correspondait pas à l'analyse préalable, par l'auteur, du concept de frottement sec selon Coulomb.

L'algorithme du module FROS modifié par nos soins est résumé dans la figure 36 :

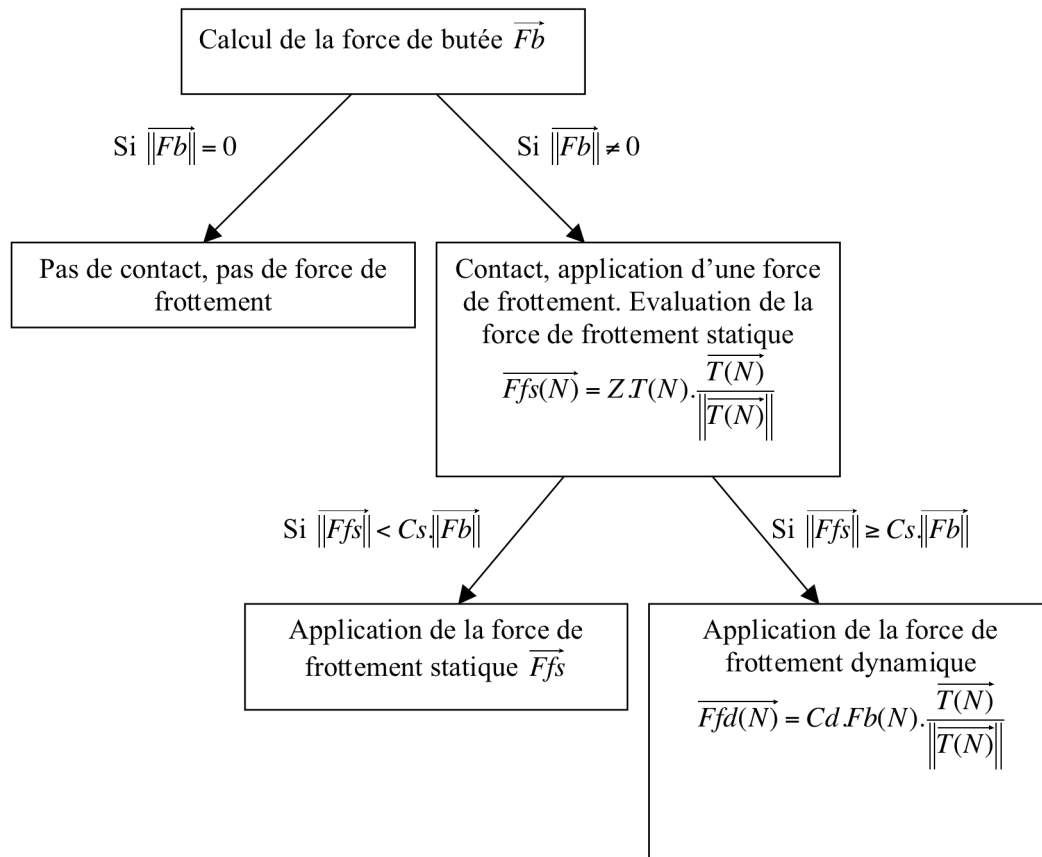


Figure 36 : Schéma de l'algorithme du module FROS.

Pour valider notre nouvel algorithme, le paragraphe II.3.5. du chapitre 5 de validation proposera des modèles présentant des phénomènes d'accrochage (passage de l'état dynamique à l'état statique) et de décrochage (passage de l'état dynamique à l'état statique).

III.4. Manipulation graphique des paramètres

III.4.1. Augmenter la compréhension de la signification des paramètres et des interactions

Les valeurs des paramètres des interactions que nous proposons à l'utilisateur pourraient simplement être saisies dans un champ. Cela est d'ailleurs suffisant pour les interactions simples comme la butée ou encore l'interaction d'élasticité linéaire. Or pour d'autres interactions, le nombre de paramètres augmente. Il est plus difficile pour l'utilisateur de s'y retrouver. Une simple représentation symbolique, par exemple la représentation graphique de la fonction d'interaction avec ses paramètres, peut suffire à l'utilisateur. Mais il peut aussi entrer des valeurs qui modifient le profil de la fonction d'interaction de telle sorte que celle-ci perd sa fonctionnalité originelle, par exemple l'entrée d'une valeur d'un seuil.

Nous pouvons certes interdire la saisie de certaines valeurs, mais c'est dans la manipulation graphique interactive, dans les interactions qu'elle propose et celles qu'elle interdit que le sens de chaque paramètre devient naturellement compréhensible. De plus c'est aussi en manipulant chaque paramètre que l'utilisateur peut modifier la fonction d'interaction tout en conservant le profil global de la courbe. Ce faisant il explore ce profil et s'en fait une représentation mentale catégorisée : je joue sur deux seuils pour une fonction de cohésion, ces deux seuils définissent des zones de répulsion et d'attraction et de liberté, dont je peux augmenter la taille, cette interaction de cohésion est donc une interaction avec une zone de répulsion suivie d'une zone d'attraction puis d'une libération. Il faut dès lors choisir une méthode de manipulation. Cette méthode doit être spécifique pour chaque paramètre (raideurs, seuils, distance au repos). Pour définir une méthode de manipulation cohérente, il est nécessaire de s'assurer que la manipulation graphique d'un paramètre ne modifie pas les autres. Les deux paragraphes suivants présentent des manipulations de raideurs et de paramètres spatiaux pour certaines familles d'interaction que nous proposons à l'utilisateur. Nous terminons sur le cas général des liaisons linéaires par morceaux quelconque.

III.4.2. Manipulation d'une raideur

Manipuler un paramètre de raideur sur le graphe revient à modifier une pente. Or il s'agit de ne pas modifier les autres paramètres, notamment les seuils en distance donc les abscisses des extrémités des segments de droites. Il ne faut pas non plus que le profil général des interactions puisse être modifié. Pour une interaction d'attraction, par exemple, aucun point de la courbe ne doit se trouver dans le domaine des forces positives. Dans le cas contraire, on pourrait permettre à la fonction de l'interaction d'attraction d'avoir une valeur nulle pour une distance donnée, donc d'introduire un point d'équilibre là où le concept d'interaction d'attraction n'en permet pas (figure 37). Pour éviter ce changement de catégorie qui modifie l'interaction d'attraction en interaction de cohésion, il faut interdire un changement de signe du paramètre de raideur. De manière générale, ce critère est suffisant pour toutes les interactions concernées.

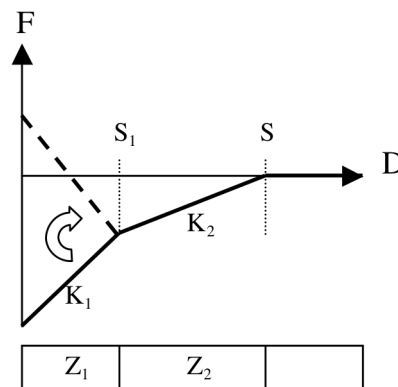


Figure 37 : Modification de raideur de la part de l'utilisateur conduisant à un changement de catégorie de comportement pour l'interaction ATR3.

Deux cas de manipulation des paramètres de raideur se présentent, ayant des incidences différentes sur la modification graphique des fonctions. Dans le premier cas, la modification d'une pente d'un segment de droite ne déplace pas les autres segments. Les paramètres de raideur des interactions de cohésion COH3 et COH4 sont concernés ici. Les points de la courbe dont les abscisses sont les paramètres de l'interaction deviennent des points autour desquels le segment de droite concerné par la manipulation peut pivoter. Les autres parties de courbes sont étendues et non déplacées (figure 38).

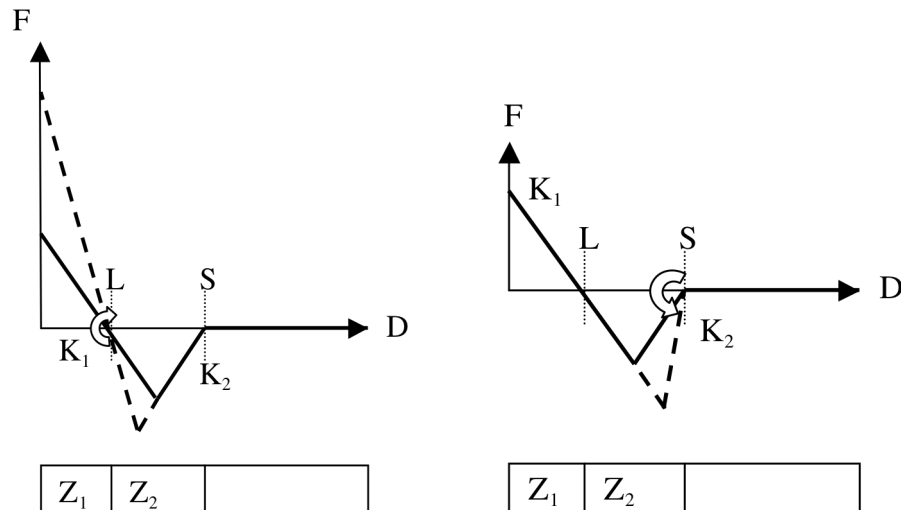


Figure 38 : Manipulation graphique des paramètres de raideur de l'interaction COH3.

Manipulation du paramètre K1 (à gauche) par rotation autour du point d'abscisse L. Manipulation graphique du paramètre K2 (à droite) par rotation autour du point d'abscisse S.

Dans le deuxième cas, celui des interactions de type attraction et répulsion, certains segments de droites ont leurs deux extrémités fixées en abscisses par les paramètres de seuil de distance. Or, la modification de la pente de ces segments de droite a nécessairement une incidence sur au moins une des deux ordonnées des extrémités. Par conséquent, une telle modification amène nécessairement au déplacement d'un autre segment de la courbe. Une contrainte apportée par le profil de ces courbes est que le point d'abscisse S après lequel la force appliquée est nulle doit toujours avoir une ordonnée nulle. Il en résulte que pour ce type d'interaction, c'est nécessairement l'extrémité gauche des segments de courbes qui doit être modifiée (figure 39). Que des segments de courbes se translatent selon les ordonnées alors qu'on ne modifie pas un paramètre les concernant n'est pas très gênant, en tout cas moins gênant que d'autres modifications. En effet, l'utilisateur de ces interactions n'est pas amené à penser en termes de valeurs de force mais selon des valeurs de raideur et de seuil de distance qui ne seront pas modifiées.

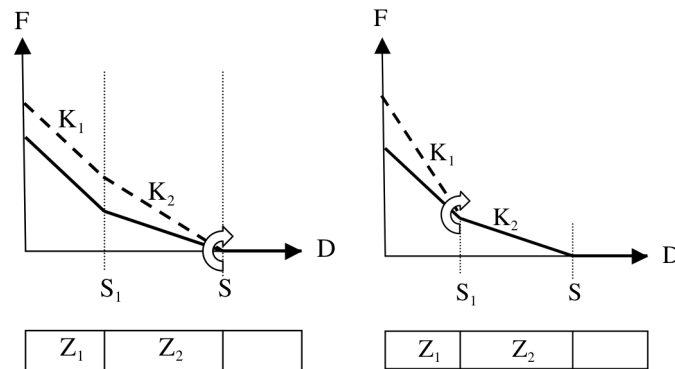


Figure 39 : Manipulation graphique des paramètres de raideur de l'interaction REP3.
 A droite : manipulation du paramètre K_2 de l'interaction REP3. Le segment de droite $[0, S_1]$ se translate en ordonnées. A gauche : manipulation du paramètre K_1 de l'interaction REP3. Seul le segment de droite d'abscisses extrémales 0 et S_1 est concerné.

III.4.3. Manipulation d'un seuil en distance

Manipuler les paramètres de seuil en distance revient à déplacer un point sur l'axe des abscisses du graphe de représentation des interactions. Il faut veiller encore à ce que les profils des fonctions soient respectés. Dans le cas de l'interaction de cohésion COH3, il ne faut pas, par exemple, que disparaisse l'une des deux zones cohésive ou attractive. Dans ce dessein, il est nécessaire que l'ordre des paramètres de seuil en distance soit respecté. De manière plus générale, ce critère est suffisant pour conserver des profils comme le comportement des interactions ne soit pas modifié.

Nous sommes encore face à deux cas distincts. Dans le premier cas, les segments de droites sont translatés selon l'axe des abscisses. Les interactions COH3 et COH4 sont conjointement concernées ici, puisque les paramètres de seuil en distance sont liés à une partie non extrémales des segments de droite (voir figure 40). De plus, le changement des intervalles de définition des segments de droites implique un changement de la taille de ces segments.

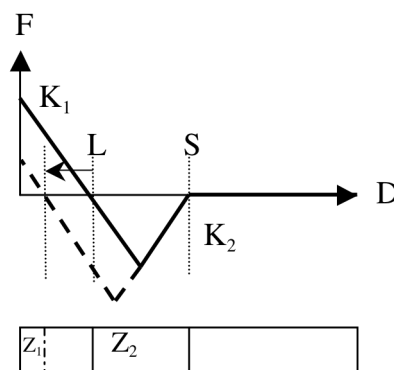


Figure 40 : Manipulation graphique du paramètre L de l'interaction COH3.

Dans le deuxième cas, celui des interactions REP3, REP4, ATR3 et ATR4, le paramètre de seuil en distance est souvent lié à deux extrémités de segments de droite. Pour de tels paramètres, si l'on souhaite que les pentes soient conservées, un des segments doit subir une translation selon l'axe des ordonnées. De plus, la longueur des deux segments est modifiée. Puisqu'une contrainte est que la valeur de la force au point extrémal d'abscisse S soit nul, c'est le segment gauche qui est toujours translaté suivant les ordonnées (voir figure 41).

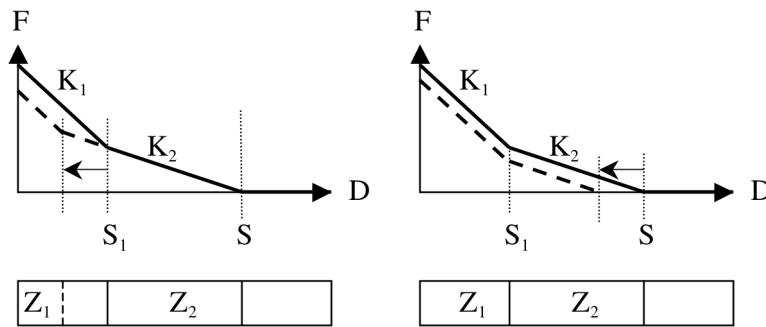


Figure 41 : Manipulation graphique des seuils de distance de l'interaction REP3.
 Manipulation des paramètres S1 (à gauche) et S (à droite) .

III.4.4. Edition d'interactions linéaires par morceaux quelconques

Contrairement aux modules particuliers que nous avons présentés, les modules d'interaction linéaire par morceaux quelconque qui sont bien plus généraux donc en même temps plus enclins à amener le modélisateur débutant vers des impasses de modélisations, ne peuvent être des modules dont les paramètres ont une signification que nous lui assignons au préalable. Le cas de la manipulation de tels modules est en fait bien plus délicat. Nous avons vu qu'une telle interaction au niveau du modelleur CORDIS est définie par la données de triplets de paramètre (K_i , Z_i , S_i) et d'une valeur de force à l'origine. Or nous l'avons déjà dit, il n'est pas aisé de raisonner en termes de valeur numérique de force. Laisser à un utilisateur la possibilité de définir un profil de courbe quelconque linéaire est certes intéressant car on augmente la genericité des fonctions d'interactions possibles mais cela lui demande à l'inverse d'avoir acquis une expérience suffisante de la modélisation pour créer des fonctions d'interaction pertinentes. Nous avons fait le choix que l'information sur la pente et donc la « raideur » correspondant à chaque segment de droite lui soit offerte, mais qu'elle ne soit pas modifiable car il n'y a pas de choix unique possible et générique pour la modifier comme nous l'avons vu dans le paragraphe II.4.2.

En posant $F_i = F(S_i)$, une relation linéaire simple lie K_i , S_i , S_{i-1} , F_i , F_{i-1} :
 $K_i \cdot (F_i - F_{i-1}) = S_i - S_{i-1}$. Nous obtenons alors le profil général suivant :

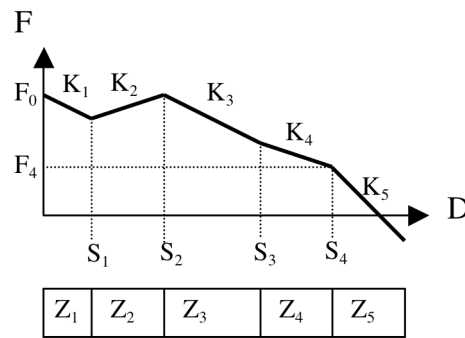


Figure 42 : Représentation fonctionnelle de l'interaction LLM.

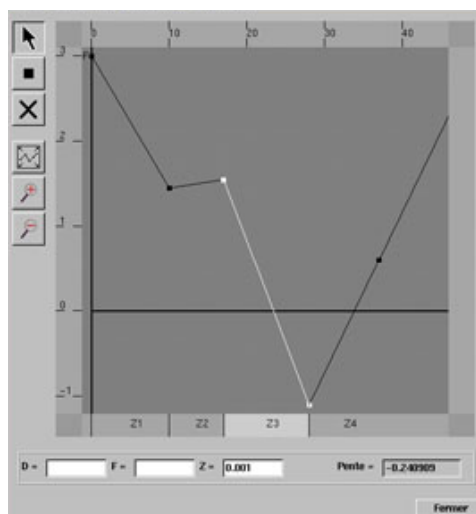


Figure 43 : Editeur d'interactions linéaires par morceaux quelconques.

III.5. Mise en œuvre de la modification de paramètres en fonction des variables de la simulation

La mise en œuvre d'un outil permettant de modifier en ligne les paramètres de la simulation n'est pas neutre du point de vue de la modélisation et de la signification du modèle. Nous avons expliqué que la modification paramétrique mène le système à ne plus être conservatif. C'est le cas par exemple lors de la modification du paramètre de raideur pour les oscillateurs paramétriques [BPV88]. Ce point doit être pris en compte par le modélisateur. Mais un outil qui reviendrait à pouvoir donner n'importe quelle valeur à des paramètres pendant la simulation pourrait amener l'utilisateur à considérer cette modification comme un outil de détermination des paramètres d'une simulation. Or, ce n'est pas parce que l'utilisateur aura changé un paramètre pendant la simulation pour obtenir à un moment un comportement souhaité (par exemple, une déformation souhaitée) qu'en reportant ce paramètre au début d'une nouvelle simulation, il obtiendra le dit comportement dès le début. Une modification paramétrique pendant la simulation implique un apport d'énergie à l'objet simulé. Le moment où la variation intervient doit donc être pris en compte dans le modèle lui-même.

Nous proposons donc d'inclure la modification paramétrique qui s'opèrerait pendant la simulation et en fonction des variables de la simulation, dans le modèle conçu par l'utilisateur. Pour cela, nous proposons de mettre à sa disposition un module de modification paramétrique venant se « brancher » sur une entrée de paramètre des modules. Ce module appelé MCP prend alors en entrée une valeur scalaire qui évolue au cours de la simulation (cf. figure 44). Nous avons considéré que la valeur scalaire faisant le plus de sens dans l'optique d'un modèle CORDIS-ANIMA était une valeur de position fournie par une masse dans un espace scalaire. Le fait de contrôler un paramètre à l'aide de la position d'une masse dans un espace scalaire n'est pas qu'une simple économie de représentation. Il possède l'intérêt d'inclure cette masse elle-même dans le modèle. D'une certaine manière, l'utilisateur de MIMESIS est amené à dessiner alors un contrôleur physique de son modèle. L'introduction d'un tel principe de modélisation a été validée dans [AEL07] par un modèle de striction où le contrôleur est indépendant du modèle qu'il contrôle et par un modèle de motricité où le contrôleur est relié au modèle se mouvant par une interaction entre un espace 1D (celui du contrôleur) et un espace 3D (celui du modèle contrôlé). Ainsi le dernier avantage du contrôle paramétrique par une masse CORDIS-ANIMA est la possibilité de créer des modèles faisant intervenir un contrôle paramétrique bouclé.

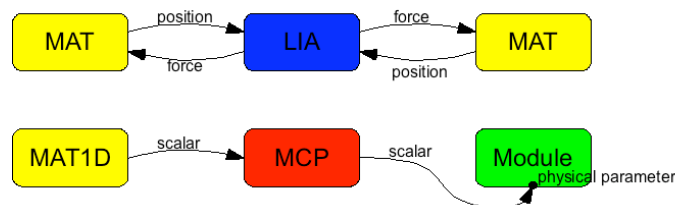


Figure 44 : Module de modification paramétrique en ligne.

IV. Bilan

Nous avons présenté dans ce chapitre une mise en œuvre des interactions non linéaires dans un environnement de conception interactif de modèles CORDIS-ANIMA. Il est nécessaire de disposer de telles interactions pour que l'environnement prétende à une généricité suffisante au regard des comportements dynamiques simulés pour l'animation. Jusqu'ici la base d'interactions non-linéaires proposée et déduite d'une analyse fonctionnelle menant à une typologie des interactions a été suffisante pour obtenir un très grand nombre de modèles en pratique.

La partie III.3 du chapitre 5 présente un certain nombre de modèles élémentaires du mouvement faisant appel à ces fonctionnalités pour les valider.

Nous avons terminé notre présentation sur le contrôle paramétrique en ligne en fonction des variables de la simulation. Cette question a fait l'objet du stage de Master d'Ali Allaoui qui nous avons co-encadré. Nous nous sommes permis à ce titre d'évoquer cette fonctionnalité même si la thèse d'Ali explorera plus en détail cette question. Les possibilités de modélisation qu'une telle modification de paramètres apporte sont, en effet, encore aujourd'hui largement inexplorée mais offre d'intéressantes perspectives.

- Chapitre 4 -

Insérer MIMESIS dans une chaîne de production globale d'animation

I. Du mouvement du point au mouvement de la forme

I.1. L'habillage du modèle physique particulière : d'une topologie physique vers une topologie spatiale

I.1.1. Deux systèmes de représentation différents

Les modèles CORDIS-ANIMA produisent des mouvements de points. Une difficulté majeure inhérente à la production d'images animées à l'aide de modèles masses - interactions réside dans le fait qu'il est difficile de produire des séquences d'images animées par le rendu direct des masses ponctuelles produisant le mouvement puisque ces masses n'ont, par définition, pas de spatialité. D'une manière générale, il est donc nécessaire de développer des méthodes qui étendent la spatialité de ces masses ponctuelles, pour compléter la chaîne de production d'images animées par modèle physique particulière. Il s'agit de passer du mouvement du point au mouvement de la forme spatiale, de la génération du mouvement à l'occupation spatiale de celui-ci, nous dirons pour faire plus court qu'il s'agit de se confronter à un problème de relation entre forme et mouvement. Ce problème n'est pas des moindres. S'il se pose en des termes très pratiques, celui de définir des formes à partir des points en mouvement, il s'agit de faire face à la question de la généricité du lien entre un formalisme issu de la physique comme le formalisme de physique particulière comme CORDIS-ANIMA et un formalisme issu de la géométrie, qui s'intéresse donc aux formes. Or ces deux domaines sont historiquement distincts en science.

La discrétisation de la matière en masses ponctuelles pour représenter le mouvement réclame en général un nombre de points inférieur à celui nécessaire à la visualisation d'un objet. Il est par conséquent nécessaire de créer des points supplémentaires pour donner une forme spatiale au nuage de points dont le mouvement est produit par une simulation physique particulière. Ces nouveaux points constituent ce qu'Habibi appelle dans [Hab97] un habillage de ce nuage de point. Le choix du terme habillage souligne que cette extension spatiale est un processus de dépendance non-dynamique entre le mouvement des masses ponctuelles simulé par un modèle physique particulière et le mouvement des points visibles constituant une forme. Le modèle du mouvement, modèle amont, est l'entrée d'un modèle de forme, modèle aval, qui devient alors une forme en mouvement.

La forme en mouvement obtenue donne une information nouvelle par rapport au nuage de points en mouvement en sortie de la simulation physique particulière. La forme permet de définir une topologie

spatiale, c'est-à-dire de décrire le fait qu'il y a un ou plusieurs objets, que ces objets se mélangent, se recomposent, se dissocient. Cette information n'est pas nécessairement contenue dans le modèle amont de mouvement dont la topologie, topologie physique cette fois. Ces relations topologiques sont différentes des interactions physiques entre masses ponctuelles. Les relations topologiques dans le cadre des modèles physiques particuliers sont en effet des relations fonctionnelles qui sont différentes des relations visibles. Deux masses peuvent participer au mouvement d'un même objet alors qu'elles ne sont pas en interaction dans le modèle physique particulière amont mais deux masses peuvent aussi participer au mouvement de deux objets distincts alors qu'elles sont en interaction dans le modèle physique amont.

Pour illustrer cet argument, prenons un exemple élémentaire mais significatif. Soit le modèle de boules suspendues déjà utilisé dans la partie II du chapitre 2. Les boules modélisées par des masses ponctuelles vont s'entrechoquer grâce à des interactions de butée viscoélastiques les reliant. Chaque boule se balance de plus en étant reliée à deux masses fixes par l'intermédiaire d'une chaîne de masses en interaction viscoélastique simple. Ce modèle est reproduit dans la figure 45, à gauche. Il est envisageable d'habiller le nuage de points en mouvement obtenu par simulation de ce modèle à l'aide de sphères modélisant géométriquement les boules et de splines modélisant le fil qui relie ces boules aux masses fixes comme le montre la figure 45 au centre. Deux masses en interaction de butée, ayant donc un lien dans la topologie physique du réseau masses - interactions, modélisent le mouvement de deux objets s'entrechoquant, n'ayant donc aucun lien topologique d'un point de vue spatial. Mais on peut aussi imaginer appliquer un habillage totalement différent fait de polygones reliant certaines masses entre elles, que ces masses soient celles qui modélisaient le mouvement d'une boule ou participaient au mouvement d'une partie du fil qui la suspend, que ces masses soient en interaction ou qu'elles ne le soit pas dans le modèle physique masses - interactions amont. L'effet visuel est alors radicalement différent. On obtient par exemple un drap qui se balance (figure 45 à droite) sans pour autant avoir modélisé le mouvement de ce drap en plaçant des interactions élastiques partout où il y a un lien visible entre les points représentés par les masses ponctuelles. Contrairement à ce qui aurait pu être fait à l'aide d'un système masses - ressorts, la topologie du réseau masses - interactions n'est pas calquée sur la topologie de la maille habillant le mouvement de ce réseau.

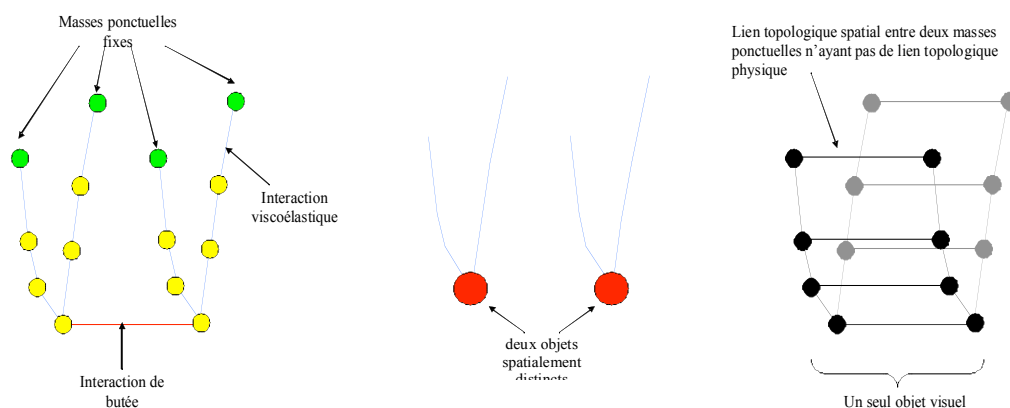


Figure 45 : Deux types d'habillage pour un même modèle particulière.

A gauche : modèle masses - interactions amont de mouvement. Au centre : habillage rendant compte de deux objets visuels. A droite : habillage rendant compte d'un seul objet visuel.

Le chronogramme suivant reproduit le mouvement habillé dans MIMESIS des deux manières différentes précédemment citées.

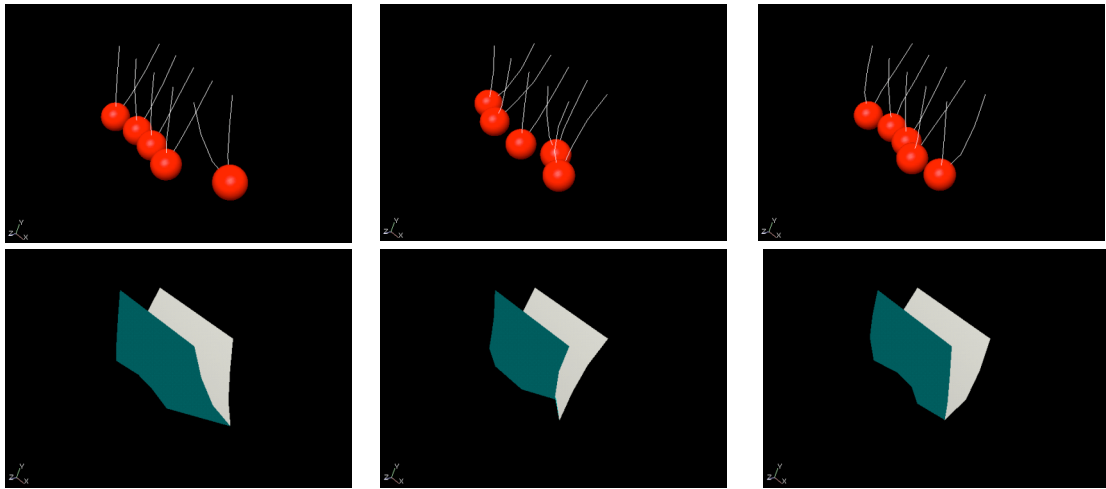


Figure 46 : Chronogrammes avec deux habillages différents pour la même simulation.

En haut : habillage « boules suspendues ». En bas : habillage « drap ».

Nous voyons donc par cet exemple que la relation entre le mouvement des points généré par CORDIS-ANIMA et le mouvement des formes qui habillent ce mouvement, et une relation de *mapping* qui dépend du choix de l'utilisateur. Or un tel choix lui est rarement offert dans les logiciels d'animations « classiques ».

1.1.2. Proposer des outils de modélisation topologique à l'utilisateur prenant en compte les modèles CORDIS-ANIMA en entrée

Dans la pratique de l'animation par modélisation et simulation de réseaux CORDIS-ANIMA au sein d'un environnement de conception interactif tel que nous le proposons, nous avons vu que l'utilisateur est amené à réfléchir en terme de mouvement de masses ponctuelles en interaction avant de pouvoir étendre la spatialité de ces masses pour obtenir un objet visualisable. Le processus de création de formes en mouvement qui va du mouvement modélisé par un système de masses en interaction vers le contrôle d'un modèle de forme est donc une question à laquelle doit se confronter de manière très explicite et pratique tout utilisateur d'un tel environnement de conception. Cela ne signifie pas que la spatialité n'intervienne pas dans la phase de modélisation du mouvement. Au contraire, nous avons vu dans les parties III et IV du chapitre 2 que la spatialité intervenait dans la détermination des paramètres physiques spatiaux et des valeurs initiales des variables d'état de la simulation. En revanche, l'intervention de la spatialité a une place différente dans les logiciels d'animation « classiques » qui font appel à la modélisation physique particulière. En définissant d'abord l'objet par son contour, ses caractéristiques spatiales, et donc en restreignant les possibles modèles de mouvements qui ne sont pas accessibles à l'animateur, la question du lien entre topologie spatiale et topologie physique n'est pas posée. Pourtant, généralement, les architectures logicielles sous-jacentes à ces environnements, disposent d'un moteur physique qui se place en amont de l'animation des objets

visuels dans la chaîne de production de l'image animée. Le *mapping* entre topologie physique et topologie spatiale est donc résolu de manière ad hoc et non générique.

Puisque nous soutenons la thèse qu'il est fécond de laisser à un animateur le processus de génération du mouvement à l'aide d'un formalisme générique et modulaire comme CORDIS-ANIMA, il semble alors essentiel de lui permettre aussi d'avoir accès à un environnement de modélisation topologique permettant de décrire explicitement le *mapping* entre topologie du modèle physique du mouvement et topologie du modèle de la forme. Sans quoi le processus global de production d'images animées est amputé des possibilités d'habillage des modèles CORDIS-ANIMA.

Un tel environnement n'existe pas à notre connaissance et la question reste ouverte aujourd'hui. Nous ne saurions proposer une solution pour un problème aussi nouveau dans un unique chapitre de thèse. Nous ne pouvons pas non-plus poser le problème sans apporter des éléments de réponse. Ces éléments initieront, nous l'espérons, des travaux futurs. Dans la suite de ce chapitre, nous proposons donc une première analyse du problème, en recensant divers cas de liens possibles. Nous rendrons compte alors de la situation actuelle des possibilités de *mapping* avec des logiciels de modélisation et d'animation de formes. Il est en effet possible d'utiliser ces logiciels aujourd'hui pour réaliser des *mappings* simples qui ne sauraient en revanche être suffisants pour créer une chaîne générique de production d'images animées.

I.2. Les *mappings* possibles entre le modèle du mouvement et le modèle de la forme

Cette partie dresse un état de l'art des possibilités d'habillage de modèles physiques particuliers en catégorisant les pratiques selon les données qu'elles mettent en œuvre. Pour chaque catégorie, nous posons alors les problèmes liés au *mapping* entre le modèle physique amont et la topologie spatiale du modèle de forme.

I.2.1. Les points en mouvement issus de la simulation physique contrôlent des descriptions explicites de forme

Du point au point

De nombreuses méthodes ont été développées pour habiller un modèle particulier en utilisant les points en mouvement issus de la simulation physique comme points de contrôle de formes décrites explicitement. Il peut s'agir de placer ces points aux sommets d'un maillage [DSB99][Maya04], de faire de ces points les points de contrôle des courbes et des surfaces paramétriques [Pie91][RNN00], de les constituer en treillis, support de déformation de formes libres [CHP89][Coq90][GMV06] ou entrant dans la définition de squelettes de formes [AM94][BL99], expansions des points par ajout de primitives géométriques volumiques [BBB98], fonctions globales de déformation [PW89][WW90][MT92]. Ce *mapping* direct d'un point vers un point caractéristique de la forme fait passer l'utilisateur d'un nuage de points sans topologie spatiale définie, à des objets dont les contours sont

définis explicitement et qui possèdent donc une topologie spatiale. Ainsi, si l'utilisateur n'a pas de contrôle sur la topologie du modèle d'habillage qu'il a choisit, il risque d'obtenir des résultats non souhaités lorsque la configuration spatiale des points en mouvement opère au cours du temps des modifications dans la topologie de la forme. Comme illustration, prenons par exemple le cas d'un habillage faisant intervenir une triangulation de Delaunay du nuage de points. Une certaine configuration spatiale des points à un moment donné impliquera une certaine configuration pour les relations d'adjacence entre ces points. Mais une configuration spatiale à un autre moment pourra brutalement modifier ces relations d'adjacence menant à des effets caractéristiques de clignotements comme le montre la figure 47.

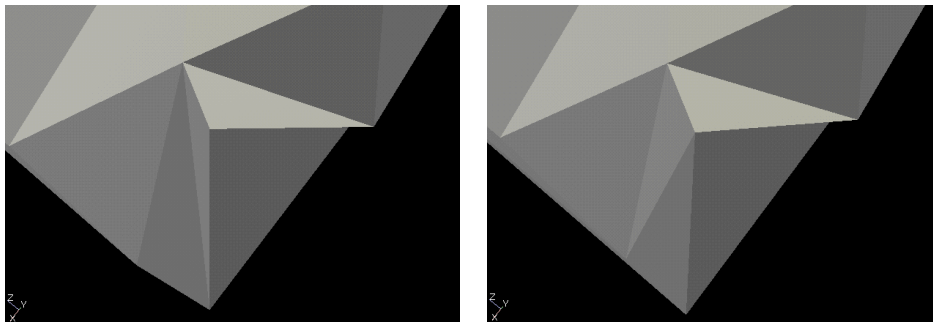


Figure 47 : Clignotement de l'habillage du modèle « boules suspendues » par une triangulation de Delaunay

À l'inverse, si la configuration topologique d'une maille reste inchangée au cours d'un mouvement (même relation d'adjacence entre les points), d'autres effets particuliers peuvent apparaître : polygones trop étirés, trop larges, retournement. La synthèse de formes peu recourir dans ces cas à des opérations de raffinement des maillages comme les surfaces de subdivision augmentant le nombre de sommets et de polygones. Mais dans ce cas, l'utilisateur doit pouvoir s'assurer qu'il contrôle le lien entre les points issus de la simulation et la liste de point de la nouvelle maille.

D'un point de vue général, il est donc nécessaire que l'utilisateur puisse avoir accès à tout instant de la simulation à la relation entre les points issus de la simulation et leur ordre dans la structure décrivant la forme.

Définition de points intermédiaires

Il se peut que l'utilisateur ait envie de définir des points intermédiaires qui apparaissent naturellement lorsqu'on observe la configuration spatiale de l'ensemble des masses du modèle du mouvement. Le mouvement de ces points n'est pas simulé par le modèle mais se déduit des positions des masses par une relation géométrique. Il est donc défini par un algorithme faisant référence à la géométrie algorithmique.

Imaginons deux masses mobiles m_1 et m_2 toutes deux en interaction élastique avec une masse fixe s de telle sorte que la masse m_1 ait un mouvement oscillant horizontal et la masse m_2 le même mouvement mais vertical. Les trois masses m_1 , m_2 et s semblent définir un carré se contractant et se dilatant. Pour autant le quatrième coin n'est pas donné par une masse du modèle. On peut alors avoir

envie de révéler ce carré. Cela peut se faire en définissant un point à partir d'une relation d'intersection entre deux droites l'une verticale et portée par $m1$ et l'autre horizontale et portée par $m2$ (voir figure 48).

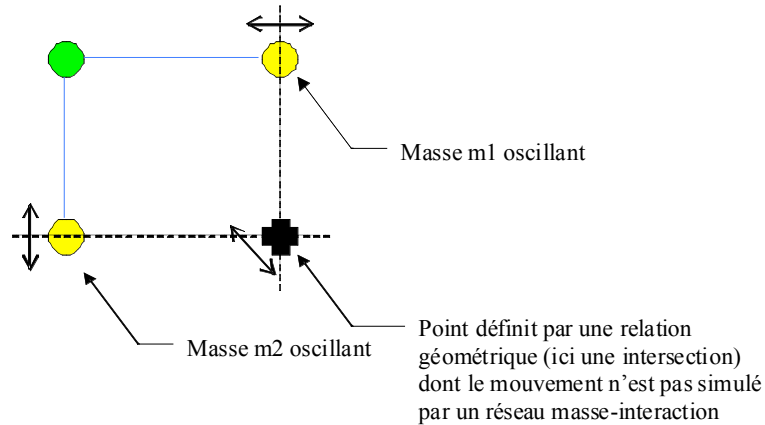


Figure 48 : Définition d'un point intermédiaire par une relation géométrique.

Une relation de *mapping* entre un modèle physique particulière et un modèle de formes dont le déplacement est contrôlé par ces points n'est donc pas nécessairement une relation d'un unique point vers un unique autre. Il faut pouvoir permettre des relations de plusieurs points vers d'autres points n'appartenant pas au nuage dont le mouvement est généré par le modèle physique particulière.

Plusieurs points pour un comportement global de la forme

Les données définissant une forme explicite ne sont pas nécessairement des points. On peut par exemple définir une forme qui ne se déforme pas au cours du temps par sa position mais aussi par son orientation son orientation qui est une donnée vectorielle et non plus ponctuelle.

Le déplacement en translation d'une forme sans déformation dans l'espace peut s'effectuer à partir du repère géométrique relatif à cette forme. Ce déplacement est trivialement et directement contrôlable en contrôlant le centre du repère la position d'un point. Ce procédé est largement suffisant lorsqu'il s'agit de modéliser le déplacement d'objets rigides en translation. Le cas typique est celui de formes définies par des primitives géométriques élémentaires 2D ou 3D. En revanche, en ce qui concerne les déplacements en rotation, une rotation ne peut pas être obtenue par un *mapping* aussi direct. Par principe du formalisme des simulations de type particulière, il faudrait au moins deux points pour opérer une rotation. Cela signifierait que dans le modèle amont deux masses ponctuelles seraient simulées dans une relation d'interaction suffisamment rigide pour être par la suite utilisées comme contrôleurs de deux points d'une forme rigide ou comme contrôleurs de l'orientation d'un des vecteurs du repère.

Une relation de *mapping* entre un modèle physique particulière et des points participant au modèle de la forme n'est donc pas suffisante. Il faut pouvoir permettre des relations entre plusieurs points et des données non ponctuelles définissant une forme.

1.2.2. La forme conditionnée par les paramètres spatiaux des interactions du modèle masses - interactions

Nous avons vu dans le chapitre 2 que les modèles masses - interactions faisaient intervenir des paramètres physiques spatiaux comme les distances au repos et les seuils en distance des interactions. Ces paramètres spatiaux peuvent induire des contraintes sur les paramètres des formes habillant le mouvement des points issus de la simulation physique. Pour illustrer ce fait, prenons le cas de la modélisation d'objets rigides en collision et plus particulièrement celui d'une simple collision raide entre deux masses ponctuelles modélisée par une interaction de butée élastique de seuil égale à 10. Dans le modèle masses - interactions, une force de répulsion sera appliquée sur chacune des masses dès lors que leur distance est inférieure à 10. La manière la plus simple d'habiller ce mouvement pour rendre visuellement cette collision entre deux objets serait de choisir deux sphères dont le mouvement est contrôlé par le mouvement des deux masses. Si l'on souhaite que ces deux objets sphériques semblent entrer en collision, il suffit simplement que la somme des rayons des deux sphères soit égale au seuil de l'interaction de butée élastique, soit 10. Dans le cas de deux masses, il y a une infinité de combinaisons possibles pour le choix de ces deux rayons. La figure 49 illustre cette situation. Cette indétermination peut aisément être levée, en laissant l'utilisateur effectuer le choix qui lui convient, ou bien en ajoutant une relation supplémentaire comme par exemple que le rayon des sphères entourant chaque masse dépende d'un critère de masse volumique associée à chacune, selon la métaphore qu'une masse ponctuelle pourrait correspondre à une condensation de la sphère que l'on re-expanse lors de l'habillage.

Cet exemple montre au travers d'un cas simple comment notre approche, dans laquelle le mouvement est premier, diffère des autres méthodes de « détection de collision » généralement pratiquées à partir d'un formalisme de solides en mouvement pour rendre des collisions entre objets [TKH*04]. Dans ce type de pratique, les objets sont initialement définis géométriquement. Il s'agit alors de détecter au cours de leur mouvement les zones d'interpénétrations potentielles et de prévenir ces interpénétrations en appliquant des forces répulsives. Le mouvement est donc d'une certaine façon conditionné par la forme. Dans le cas l'approche de création d'images animées par modèles CORDIS-ANIMA, le mouvement, déterminé par le modèle physique amont impose au contraire des contraintes à la forme. Le mouvement est premier et c'est donc la forme qui est conditionnée pour partie par le mouvement.

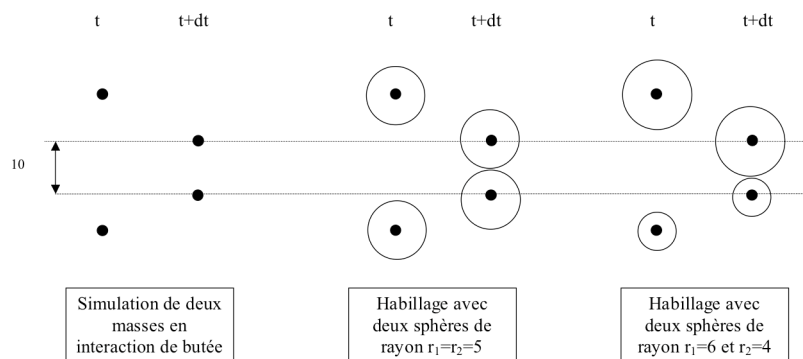


Figure 49 : Le mouvement généré peut imposer des contraintes géométriques sur l'habillage d'objets rigides multiples en interaction.

Remarquons que si nous généralisons ce problème à 3 masses en interaction, l'indétermination est levée puisqu'il s'agit de déterminer 3 rayons liés par 3 équations imposant que la somme de deux des rayons soit égal au seuil en distance de l'interaction reliant masses concernées. Pour N masses, nous nous retrouvons avec $N(N-1)/2$ équations pour N inconnues. On ne peut plus prendre n'importe quelles valeurs pour les seuils en distance des interactions, il faut que ces seuils correspondent à la somme de rayons que l'on a prédéfinie à l'avance. Nous nous retrouvons dans une situation où les choix des paramètres spatiaux des interactions dépendent des caractéristiques des formes des objets modélisés.

Quoiqu'il en soit, un outil complet permettant un *mapping* d'un modèle de mouvement masses - interactions vers un modèle de forme doit prendre en compte en entrée les paramètres physiques spatiaux du modèle amont.

1.2.3. Forme définie par le même point donné à plusieurs instants de la simulation

Jusqu'ici les formes sont redéfinies à chaque instant. Nulle part le temps n'apparaît dans leur construction. Or pour certains mouvements, la forme peut dépendre de l'évolution dans le temps des positions des masses. Il faut donc alors définir des formes à partir des positions des masses de la simulation à des instants multiples. Le choix de ce type de forme peut faire apparaître des « figures » dans le mouvement qui n'apparaîtraient pas en choisissant un habillage ne dépendant que des positions à un instant donné.

Un exemple de forme de ce type est la « trace ». La trace d'un point est une forme composée de points dont les positions sont celle du point au temps t ainsi qu'à un certain nombre de temps antérieurs. Il s'agit d'une des plus simples manières de créer une forme dépendant d'instant multiples de la simulation. Mais cet exemple est suffisant pour montrer toute la richesse et la nécessité de ne pas construire simplement des formes successives ne dépendant que de positions à un instant donné.

La littérature nous offre des exemples. Prenons-en un. Une visualisation de type « trace » est appliquée à un modèle de mouvement de foule [HLT*03]. Les travaux de Heigeas et al. proposent d'habiller un mouvement de foule dans un univers urbain, à l'aide de la trace des masses représentant une personne. Dans leur modèle, plusieurs groupes de personnes avec des buts différents viennent à se croiser. La visualisation trace fait alors clairement apparaître des phénomènes émergents tels que des vorticités lors du croisement des groupes.

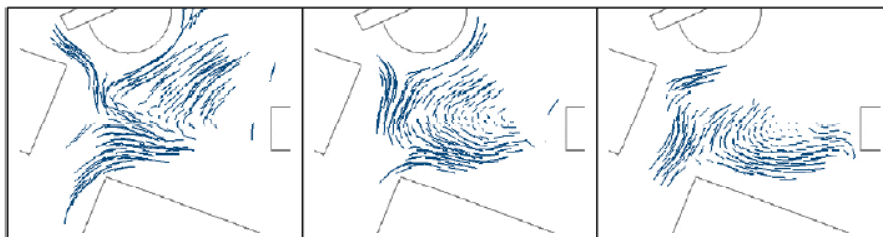


Figure 50 : Visualisation de la trace d'une simulation masses - interactions de mouvement de foule.

Un environnement de modélisation topologique de la forme habillant le mouvement doit pouvoir prendre en compte pour une forme à un moment donné, différentes positions d'un même point à plusieurs instants.

1.2.4. Épaississement du point : les changements topologiques émergents

Les masses ponctuelles dont le mouvement est simulé par un modèle physique ont une occupation ponctuelle de l'espace. Notre but jusqu'ici a été d'étendre cet espace pour créer la forme. Or pour modéliser un certain nombre d'objets en mouvement peu structurés (fumées, pâtes, etc.), il n'est pas nécessaire de faire appel à une structuration topologique. Ce sont les objets aux contours généralement flous. Or il n'est pas non-plus nécessaire d'utiliser un grand nombre de masses ponctuelles pour modéliser le mouvement de ces objets. Il convient alors d'épaissir ces masses qui n'exhibent pas d'organisation topologique. C'est ce que permettent les surfaces implicites. A partir d'une masse est défini un champ qui, combiné aux champs produit par les autres masses forment une surface quand on trace la surface isopotentielle résultante. Bien souvent de telles surfaces sont, dans la littérature, redéfinies à chaque instant de la simulation. L'histoire du mouvement n'y est pas inscrite. Ce sont des surfaces implicites statiques. Il existe en revanche des méthodes qui permettent à la fois d'obtenir cet épaississement du point tout en générant une forme qui peut dépendre de l'histoire du mouvement. Ce sont les surfaces implicites dynamiques.

Surfaces implicites statiques

Une surface implicite statique est l'isosurface d'un champ de potentiel à un moment donné de la simulation. De manière générale, la fonction de champ est donnée comme une fonction décroissante monotone de la distance à un squelette. Un squelette est une primitive géométrique pour laquelle il est possible de définir correctement une fonction de distance. Il peut s'agir tout aussi bien d'une courbe, d'une surface, d'un volume mais aussi d'un simple point. On peut donc considérer qu'une surface implicite est une extension spatiale, un *épaississement* plus ou moins complexe, c'est-à-dire plus ou moins isotrope, d'une forme de base définie par un ou plusieurs points.

Les travaux pionniers de Desbrun et Cani [DC98] proposent par exemple un habillage pour un modèle très déformable de type liquide visqueux modélisé par des systèmes de particules. Foster et Fedkiw utilisent eux aussi des surfaces implicites pour modéliser des surfaces en mouvement, même si la méthode n'est pas directement un habillage de particules mais plutôt de champ de vitesses obtenu par exemple par la résolution numérique des équations de Navier-Stokes [FF01].

Pour une introduction assez complète des surfaces implicites, on pourra se reporter à un des ouvrages de référence sur le sujet, écrit par Bloomenthal et al. [BBB*97] et intitulé *Introduction to Implicit Surfaces*.

Il y a deux choix principaux à faire si l'on souhaite habiller un ensemble de particules à l'aide de surfaces implicites : le choix de la primitive implicite et le choix de la méthode de mélange de ces primitives.

Il existe une littérature très fournie définissant des primitives implicites. On pourra les classer en deux catégories :

- les primitives centrées sur un point. Elles construisent un champ scalaire autour d'un point. Elles peuvent donner naissance à des surfaces simplement sphériques ou ellipsoïdales comme les *blobby molecules* premières primitives introduites par James Blinn en informatique graphique [Bli82], les *metaballs* [NHK85] ou les *soft objects* [WW89] dérivant de fonctions polynomiales mais encore des primitives plus complexes plus complexes comme les superquadriques [Bar81] dérivées des quadriques.
- Les primitives définies à partir d'une courbe ou d'une surface qui joue le rôle de squelette. La notion de primitive implicite à squelette étend la notion de surface implicite dont le champ était défini à partir d'un simple point. Ici, le champ potentiel est obtenu à partir d'une forme plus complexe. Parmi ces primitives on citera les primitives définies à partir d'un champ de distance au squelette [BW90, PT92]. Les surfaces de convolutions introduites par Bloomenthal et al. [BS91] remplace le calcul de distance par une intégration de toutes les distances au squelette. Crespin et al. proposent d'intégrer le concept de primitives d'extrusion au calcul de fonctions potentielles permettant d'obtenir des surfaces implicites [CBS96, CGD97]. Une primitive d'extrusion est obtenue en déplaçant un profil 1D fermé suivant une trajectoire donnée. Les auteurs, suivant ce principe, définissent une fonction potentielle rendant compte de la surface obtenue par une telle méthode d'extrusion.

Il est possible d'obtenir une grande variété de forme à partir des méthodes que nous avons présentées. Pour autant, même en utilisant l'ensemble de ces primitives, on arrive assez rapidement à une limite dans la complexité des formes. Elles ont souvent un caractère plutôt « rond ». Ces primitives peuvent en revanche constituer une base de fonctions combinables entre elles et capables d'être mélangées. D'autant plus que la formulation implicite des courbes se prête d'avantage que d'autres méthodes de description géométrique au *mélange* et même plus, au *mélange lisse*. La communauté de chercheurs travaillant autour des surfaces implicites espère donc pouvoir mettre en œuvre des outils de création de formes utilisables, fondés sur un nombre restreint de primitives de bases et sur des méthodes de mélange intuitives et contrôlables, tout en étant assez génériques pour obtenir une grande variété de formes complexes. L'introduction des primitives implicites par Blinn [Bli82] à travers les *blobby molecules*, est concomitante à l'idée de mélange par union. Une généralisation et une formalisation de ce principe se met peu à peu en place [NHK*85, WPW86, BS95, DG95]. Soit un ensemble de n primitives implicites, chaque primitive i dérive d'une fonction potentielle F_i telle que $F_i = f_i \circ d_i$ où d_i est une fonction potentielle et f_i une fonction de mélange. La fonction potentielle de la forme résultant de l'union des primitives implicites est la somme pondérée des F_i . Il s'agit alors de définir des familles de fonctions de mélange [NHK*85, Gas93, BS95] permettant un mélange plus ou moins bombé [Blo95], plus ou moins tranchant. D'autres méthodes de mélange sont envisageables comme les opérations booléennes s'inspirant des méthodes de CSG [SGS96, BGC98].

Quelle que soit la méthode de mélange appliquée, il apparaît clairement que les surfaces obtenues de la sorte n'ont pas de topologie fixe mais que les changements de topologie émergent (dissociation, recombinaison de l'objet). Il est impossible de contrôler en l'état les modifications topologiques des objets. Il faut donc rajouter une description topologique pour pouvoir permettre ou ne pas permettre le mélange des primitives. Si l'on modélise par exemple une main à l'aide de surfaces implicites, on ne voudra pas voir apparaître de mélange entre les doigts lorsque ceux-ci se rapprochent. Pour pallier ce problème Wyvill et al. [WW89] proposent de regrouper les primitives qui doivent être mélangées. M.-

P. Gascuel propose, quant à elle, un modèle créant des renflements au niveau de la zone de contact entre deux primitives dont on souhaite qu'elles présentent une transition franche [Gas93]. A la suite de cela Guy et Wyvill [GW95] proposent un graphe de mélange au sein duquel les primitives d'un même groupe se mélangent, et celles qui appartiennent à deux groupes différents s'écrasent à la manière de Gascuel. Le problème des surfaces implicite est de cette manière similaire au problème de l'habillage du mouvement généré à partir de modèles masses - interactions même si le nombre de point mis en jeu est plus important. Les surfaces implicites s'adaptent bien dans le cas d'objets de topologie floue mais requièrent des traitements ad-hoc, relevant du choix du modélisateur, lorsqu'il souhaite modéliser des objets ayant une topologie plus définie.

Surfaces implicites dynamiques

L'habillage de fonctions d'évolution par un épaississement du type des surfaces implicites statiques revient donc à définir un champ potentiel dans l'espace. Ce champ va varier au cours du temps, mais la méthode est statique. À aucun moment le calcul du champ potentiel ne fait intervenir le temps. Habibi et al. [HL02] proposent une méthode fondée sur la simulation d'un réseau masses - interactions pour générer ce champ potentiel. Mais, à la différence des réseaux masses - interactions utilisées pour générer un mouvement la topologie spatiale du réseau de masses en interaction est fixée. Les masses composant ce réseau produisent une valeur scalaire correspondant à une déformation de ce que l'on pourrait appeler un « écran » de masse. L'image de l'écran est ici prise de l'écran d'épingle d'Alexeïeff. [Ben01]. Ce graveur de formation met en place au début des années 30, un support de gravure fait d'une multitude d'épingles couissant perpendiculairement à un plan de fixation. Alexeïeff vient alors graver ce support en déplaçant les épingles à l'aide d'objets divers faisant office de marqueurs. Quatre sources de lumière éclairent cet « écran d'épingles » et font apparaître des noirs dans les zones où les épingles ont l'altitude la plus élevée et des blancs dans les zones où elles sont les plus enfoncées. La succession des photos, prises de cet écran dans des configurations d'altitudes modifiées au fur et à mesure par le graveur, crée artificiellement le mouvement lors de leur projection séquencée. Les épingles de Habibi et al. sont, elles, des systèmes masses - interactions fixées en deux ou trois dimensions et évoluant selon une dimension supplémentaire. Ces « épingles dynamiques » vont alors être en interaction avec des masses dont le mouvement est produit par un modèle amont. Suivant la métaphore du graveur, le modèle masses - interactions du mouvement en amont est le marqueur d'un modèle masses - interactions aval à topologie spatiale fixée qui lui fait office de surface gravée.

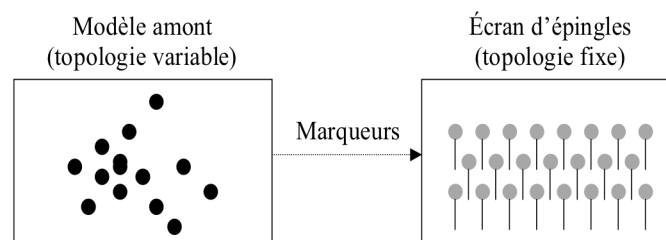


Figure 51 : Du modèle de mouvement simulé par un réseau masses - interactions à son habillage à l'aide de l'écran d'épingles.

Cette méthode a été initialement proposée par Habibi et al.[HL02] à la suite de sa thèse [H97]. De manière très basique, l'écran d'épingle est un réseau CORDIS-ANIMA.

L'écran d'épingles dynamiques est un ensemble de phyxels (contraction de physical elements) ou d'épingles disposées dans l'espace selon un tramage régulier à topologie fixe. Chaque phyxel est caractérisé par une variable scalaire extensive du type d'une position. Si l'écran est 2D, le scalaire code en quelque sorte la hauteur de l'épingle. Plus précisément, chaque phyxel est composé d'une masse fixe et d'une masse mobile en interaction unidirectionnelle.

Les phyxels peuvent aussi interagir entre eux. Ainsi chaque phyxel (plus précisément masse mobile) est relié à ses voisins les plus proches. Cela permet une diffusion spatiale des traces générées, une propagation des phénomènes. Le type d'interaction entre ces phyxels peut être de plusieurs sortes, la première implantée étant une interaction viscoélastique.

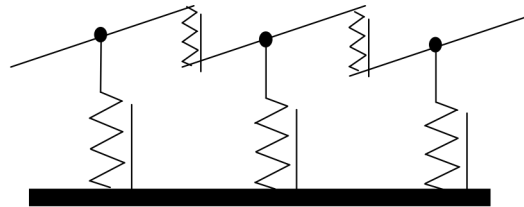


Figure 52 : Interactions entre épingles voisines en 1D.

Une troisième interaction va permettre de créer la trace sur l'écran, c'est-à-dire de venir perturber la position à l'équilibre de l'écran d'épingles : les marqueurs sont en interaction viscoélastique unidirectionnelle (les épingles n'agissent pas sur les marqueurs) seuillée en distance avec chaque masse mobile de chaque phyxel.

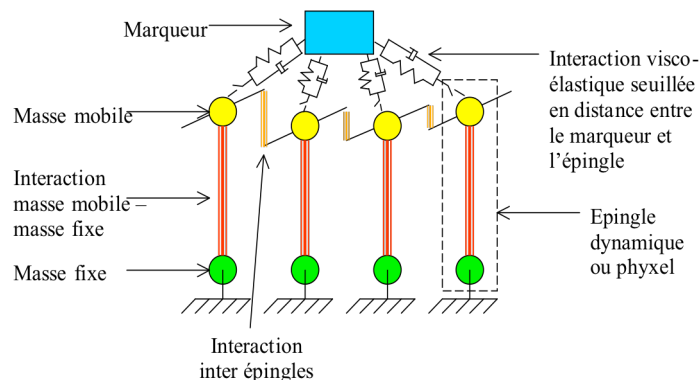


Figure 53 : Schéma de présentation générale du fonctionnement de l'écran d'épingles.

Les surfaces implicites dynamiques permettent donc de modéliser des objets en mouvement en conservant un même formalisme appliqué à des échelles différentes (mouvement global du modèle amont, habillage dynamique raffinant ce mouvement global). Une telle méthode possède donc les avantages liés au formalisme CORDIS-ANIMA en traversant toute la chaîne de production d'images animées de la modélisation du mouvement à la modélisation de la forme. Mais de la même manière que le cas des surfaces dynamiques statiques, un tel procédé ne permet pas de contrôler les changements topologiques qui pourraient arriver dans le cas des objets bien structurés ou dans le cas des objets ne se mélangeant pas. Pour appliquer des méthodes de contrôle de la topologie spatiale des objets, il est nécessaire d'ajouter directement une description topologique au sein du formalisme CORDIS-ANIMA.

I.3. Logiciels de synthèse et d'animation de formes utilisés comme moyen d'habillage

La synthèse de forme est un domaine si vaste qu'il existe un nombre incalculable d'outils pour créer des formes suivant des formalismes divers. Il ne peut être question ici de faire une revue des bibliothèques disponibles gratuitement ou à l'achat, résultats de travaux de thèse, de travaux de laboratoires ou d'industries de la synthèse de forme. Mais il est au contraire intéressant d'étudier les systèmes les plus génériques et interactifs et de poser le problème du *mapping* des sorties de MIMESIS sur ces entrées. C'est en effet un moyen pratique de résoudre la question de l'habillage dans les cas particuliers qui font appel au *mapping* de point à points décrivant une forme. Si cela ne résout pas le problème du *mapping* d'un modèle physique particulière vers un modèle de forme dans le cas général, cela étend considérablement les possibilités de créativité d'images animées par rapport aux habillages élémentaires proposés par MIMESIS.

Nous proposons ici une utilisation détournée de ces logiciels puisqu'il n'est pas question d'utiliser leurs fonctionnalités d'animation mais d'utiliser les mouvements générés par modèle physique comme entrée de ces logiciels à des fins d'habillage de ces mouvements. Or une étude préalable des méthodes d'animation de ces logiciels nous permet de pointer le fait qu'un tel *mapping* n'est pas nécessairement directe. Nous avons eu à faire pendant nos travaux, principalement à deux logiciels MAYA et BLENDER que nous allons comparer maintenant. Puis nous expliquerons comment il est possible pratiquement de les utiliser comme moyen d'habillage de modèles physiques particuliers en utilisant les scripts utilisés par ces logiciels.

I.3.1. L'animation dans les logiciels de synthèse et d'animation de forme

Le logiciel MAYA

MAYA est à ce jour l'un des logiciels dans le domaine de la synthèse 3D qui propose le plus de fonctionnalités pour créer, déformer, contrôler et combiner des primitives géométriques. Les sommets, points de contrôles, points caractéristiques sont accessibles à l'utilisateur qui peut les déplacer

librement dans un espace 3D. Les points d'une forme sont nommés à l'aide d'un code faisant intervenir le nom de la forme, le type du point (point de contrôle, sommet) et son index. Ainsi tous les points de même nature d'une forme sont stockés dans un tableau ce qui en facilite la désignation et l'accession. L'édition de l'évolution de ces points peut se faire à la main, par manipulation directe ainsi qu'en modifiant un graphe des fonctions d'évolution appelé « Graph Editor ». L'animation est un ensemble de positions « clés » pour les objets et les points entre lesquelles les positions intermédiaires sont potentiellement interpolées. Mais une des particularités de MAYA est que l'ensemble des actions sont scriptées. Toute sélection, tout déplacement d'un point de la forme correspond à une ligne de commande dans un langage propre au logiciel, le langage MEL. Il est donc possible de définir des positions « clés » à l'aide de ce langage et ainsi générer une fonction d'évolution.

Finalement, les positions de tous les points sont données en coordonnées absolues. MAYA est donc parfaitement adapté pour recevoir des fonctions d'évolution issues de MIMESIS qui produit lui aussi des signaux dans un système de coordonnées absolu.

Le logiciel BLENDER

De la même manière que MAYA, Blender permet de combiner des primitives géométriques pour créer des formes complexes. Blender propose de la même manière de scripter les actions utilisateur à l'aide du langage de script Python. Mais la grande différence entre les deux logiciels réside dans la méthode d'animation.

Deux techniques existent pour l'animation de formes avec Blender :

- L'animation d'objet : il s'agit d'une animation globale de l'objet (translation, rotation, scaling) sans changement de sa topologie. On détermine une suite de positions « clés » au cours du temps puis on choisit un mode d'interpolation (constant, linéaire, ou de type Bézier). Cette méthode est semblable à celle de MAYA.
- Déformation de l'objet au cours du temps. On peut déformer les objets au cours du temps. La méthode choisie par Blender est relativement différente de celle de MAYA. Elle est nommée *relative vertex key method*. On détermine un ensemble de formes déformées relativement à une configuration de base que l'on appelle (peut-être injustement) « clés ». On définit alors pour chaque clé une influence au cours du temps reportée dans une courbe appelée IPO. Cette influence peut aller de 0 (pas d'influence) à l'infini, sachant qu'une influence de 1 correspond à la déformation définie par la clé. Il est donc possible de composer plusieurs influences de plusieurs clés au même instant (voir figure 54). L'intérêt de ce principe d'animation peut se trouver dans l'animation d'un visage. On peut imaginer modéliser un visage d'expression neutre qui correspondra à une première clé, puis le déformer pour lui donner un aspect de colère. La courbe IPO servira à définir la manière dont on passe du neutre au colérique. Si en plus de cela on définit une nouvelle clé montrant ce même visage mais triste cette fois, en jouant sur la courbe IPO du visage triste, on pourra ajouter un peu de cette expression au visage en colère. C'est donc l'idée du morphing qui est derrière la méthode d'animation de Blender alors que MAYA est plus proche du signal brut de l'animation.

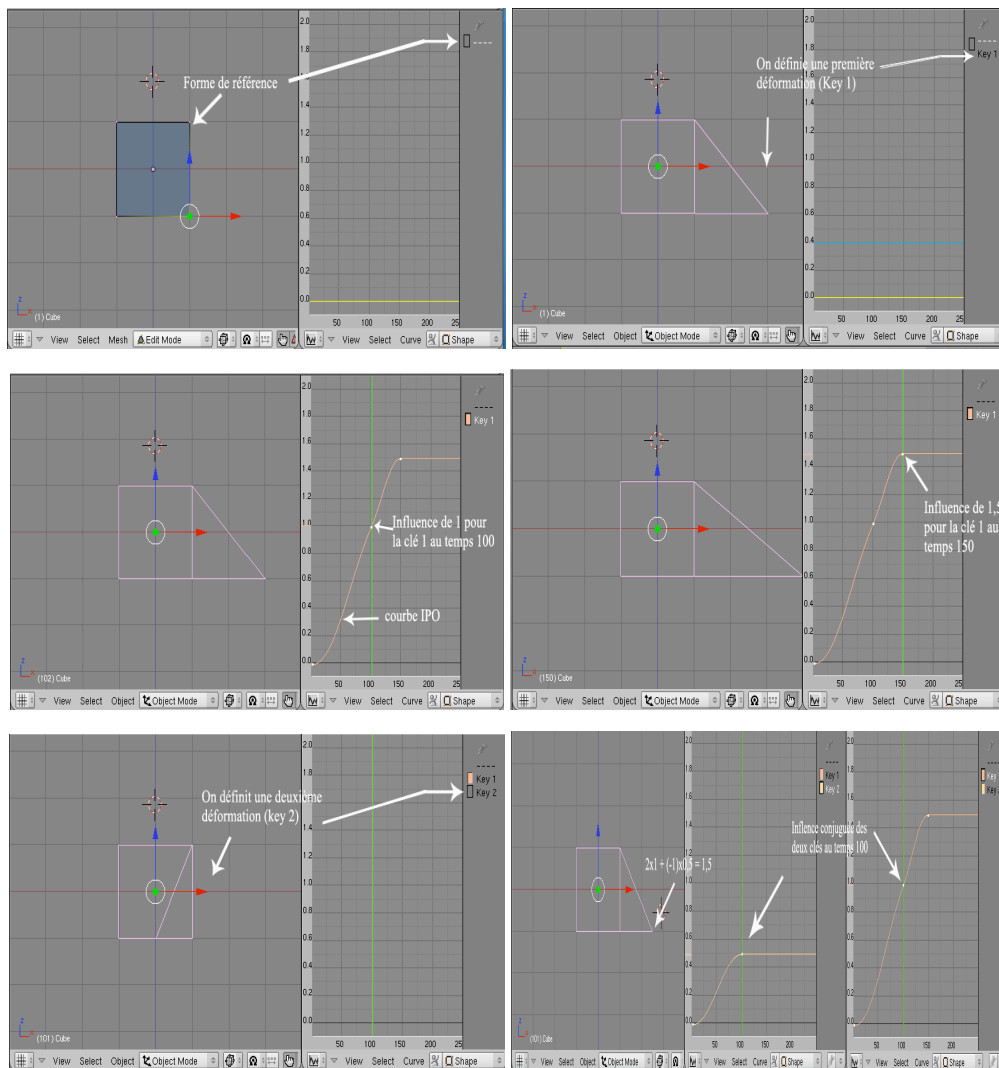


Figure 54 : L'animation dans Blender.

(1) on définit la forme (clé) de base. (2) on définit une déformation : clé 1. (3) la clé 1 à une influence de 1 au temps 100 : courbe IPO a 1 au temps 100. (4) la clé 1 à une influence de 1,5 au temps 150. (5) on définit une deuxième déformation : clé 2. (6) influence conjuguée de la clé 1 (influence de 1) et de la clé 2 (influence de 0,5) au temps 100.

1.3.2. L'entrée par script

Beaucoup de logiciels de modélisation et d'animation de formes 3D proposent aujourd'hui à l'utilisateur d'effectuer des opérations à l'aide de scripts : scripts MEL pour MAYA, Python pour Blender mais aussi Maxscript pour 3D Studio Max ou encore COFFEE pour Cinema 4D. En outre ces scripts permettent de développer des interfaces personnelles à l'intérieur du logiciel. Les scripts offrent aussi des fonctionnalités de lecture de fichiers. Il est alors parfaitement envisageable de réaliser un script permettant de lire des positions de masses ponctuelles évoluant dans le temps au sein d'un fichier, pour les affecter à la fonction d'évolution des points caractéristiques des formes des logiciels 3D. En réalisant un script permettant de le lire et de lier les fonctions d'évolution des points 3D issues

de la simulation de MIMESIS aux fonctions d'évolution des points du logiciel de synthèse et d'animation de formes 3D, l'utilisateur choisit lui-même les correspondances. En créant une correspondance, le script copie la fonction d'évolution choisie en traduisant un ensemble de positions au cours du temps suivant le schéma de représentation de l'animation du logiciel concerné. La figure 55 montre une fenêtre de mapping résultat d'un script MEL implémenté par nos soins.

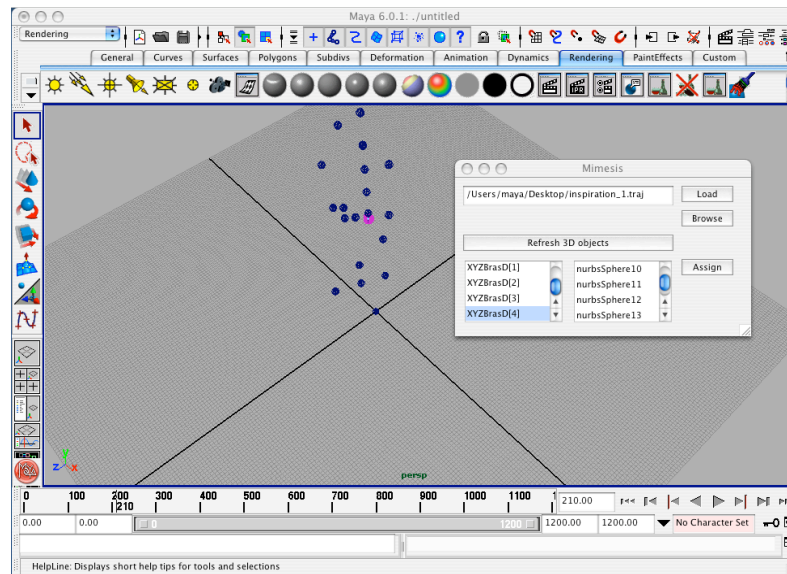


Figure 55 : Fenêtre de mapping dans MAYA implémentée dans le langage de script MEL.

Dans Maya, le *mapping* est direct. À une position dans le fichier à un instant donné pour un point donné correspond une position d'un point de MAYA au même instant. En revanche, le *mapping* dans Blender est moins direct à réaliser techniquement puisque la méthode d'animation sous-jacente diffère. Dans Blender, il convient dans un premier temps de définir les clés de l'animation Blender. Une clé correspond à l'ensemble des positions des points de contrôle de l'objet à un instant donné. Puis pour chacune de ces clés, il faut éditer la courbe IPO (0 partout, 1 au temps où la clé intervient). Il y aura donc autant de courbes IPO que de nombre d'échantillons et donc que de clés. Prenons un exemple simple. Soit un fichier composé de 2 points P1 et P2 dont les fonctions d'évolution sont données pour 3 échantillons. Nous disposons donc des positions des points P1 et P2 aux temps t , $t+dt$ et $t+2dt$. Pour réaliser le *mapping* des fonctions d'évolution de ces deux points sur deux points de Blender il va falloir définir 3 clés représentant la configuration de la forme dans Blender au temps t , $t+dt$ et $t+2dt$. Il va ensuite falloir définir 3 courbes IPO dont la première vaudra 0 partout sauf au temps t où elle vaudra 1. Il en sera de même pour les deux autres courbes en remplaçant t par $t+dt$ et $t+2dt$ (voir la figure 56).

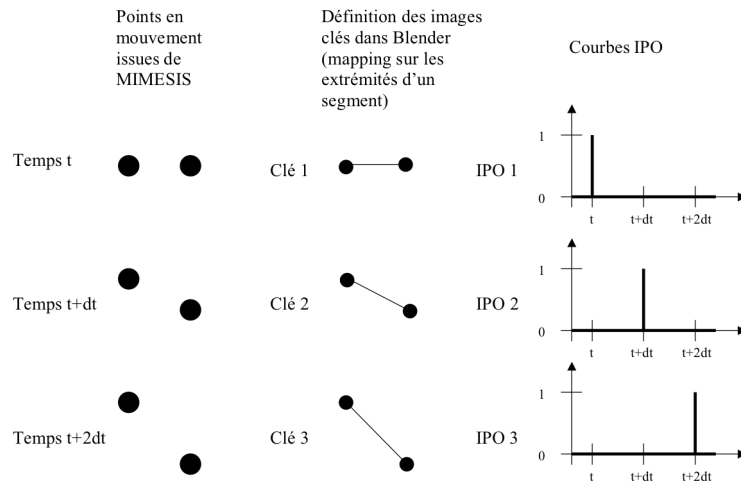


Figure 56 : *contrôle de l'évolution d'une forme dans Blender.*

I.4. Vers de nouvelles pratiques de l'habillage

Si la mise en place de scripts a permis une pratique de la production d'images animées dans un cadre complet de production lors de travaux récents du groupe ACROE-ICA, scripts qu'a utilisé notamment largement Chimin Hsieh dans ses travaux de recherche et de création qui seront présentés dans la partie III.4.4. du chapitre 5, une telle pratique qui détourne les logiciels de synthèse et d'animation de forme reste une pratique ad-hoc qui n'entre pas dans un cadre générique de résolution du problème de la relation entre la forme et le mouvement produit pas un modèle physique particulière comme CORDIS-ANIMA. Pour résoudre d'une manière générique la question de l'habillage des modèles physiques particuliers, le simple *mapping* entre les sorties d'un modèle physique particulière et les entrées des logiciels de modélisation et d'animation de forme est loin d'être suffisant. Ces logiciels n'incluent en effet pas ou incluent à minima les possibilités d'habillage que nous avons cité notamment les surfaces implicites qui sont présentes de manière rudimentaire. Il peut arriver que le principe d'animation qui les porte soit difficilement compatible avec les sorties d'un modèle physique particulière ou au prix de contorsions ad-hoc. Ils ne permettent pas en outre d'accéder à la topologie spatiale des formes décrites et ne permettent donc que des *mappings* simples de type point à point sans être capables par exemple d'en contrôler l'évolution au cours du temps. En rapportant la technique mise en œuvre jusqu'à aujourd'hui en utilisant les logiciels MAYA et BLENDER, nous espérons montrer que celle-ci reste très ad-hoc et que la mise en place d'environnement de modélisation permettant un contrôle souple de la topologie des formes par un modèle particulière de type CORDIS-ANIMA, est le seul moyen de résoudre de manière générique la question de l'habillage des modèles physique particuliers. Notre pratique actuelle de l'utilisation des logiciels de synthèse de forme doit donc être vue comme un pis-aller. Cette analyse, à laquelle nous avons dû nous confronter pendant nos travaux de thèse, a motivé le dépôt d'un projet ANR baptisé DYNAMé, projet accepté, naissant à l'heure de la rédaction de cette thèse et conduit par le groupe ACROE-ICA en collaboration avec le laboratoire XLIM des Universités de Poitiers et de Limoges et le laboratoire LAM de Paris. Ce projet aura pour but d'avancer sur la spécification d'outils capables de permettre une relation entre mouvement généré par modèle physique particulière et mise en œuvre d'une topologie spatiale pour ces modèles dans un cadre plus large, largeur que nous avons tenté d'explicitier dans cette partie.

II. Communication entre systèmes par signaux de geste et de mouvement

Le développement de systèmes de capture du mouvement ou encore des contrôleurs gestuels et des systèmes haptiques pose la question du geste et du mouvement de manière nouvelle. Dans le cadre de l'animation, ces systèmes ont eu une répercussion conséquente sur la pratique des animateurs. Un grand nombre d'animations convaincantes a été obtenu à l'aide de ces dispositifs. Dans le cadre de l'animation de personnages, par exemple, le mouvement d'acteurs à l'aide de systèmes de capture du mouvement pour contrôler le mouvement d'avatars virtuels ou encore le déplacement d'objets virtuels a permis d'obtenir des animations très naturelles. Or, entre toutes ces applications et systèmes circulent des signaux codant les gestes et les mouvements.

Le formalisme CORDIS-ANIMA est en quelque sorte un formalisme doublement compatible avec les concepts de capture du mouvement. D'une part, CORDIS-ANIMA permet de produire des signaux de mouvement similaires à ceux produit par les systèmes de captures du mouvement. D'une autre part, CORDIS-ANIMA peut accepter en entrée ces signaux pour contrôler les objets simulés par lui. La mise à disposition de formats d'entrée et de sortie du type de ces signaux de geste ou de mouvement est d'une importance cruciale pour un environnement de conception qui souhaiterait prétendre à une interopérabilité avec les différents autres systèmes les utilisant. Cela permettrait d'intégrer cet environnement dans une chaîne de production d'animation plus vaste faisant intervenir systèmes de capture du mouvement et tout autre producteurs/consommateurs de ces signaux.

Lorsque nous nous sommes intéressés à la mise en œuvre de ces entrées/sorties, la question du format de codage pour ces signaux s'est posée inévitablement. Or nous nous sommes rendu compte qu'il n'existait véritablement aucun format possédant les propriétés d'être à la fois de bas niveau, générique et minimal. On dira d'un format qu'il est bas niveau lorsqu'il code directement le signal considéré et non des propriétés de ce signal, ni n'en propose une compression. Le format sera générique s'il permet à de nombreux systèmes de communiquer entre eux. Finalement il sera minimal si aucune information particulière à une application donnée n'y est adjointe. Le critère de minimalité à une influence sur celui de généricité car des informations spécifiques peuvent nuire au bout d'un certain temps de développement au fait de conserver une base commune générique. La combinaison de ces trois propriétés est nécessaire au développement d'une réelle interopérabilité entre systèmes dans le cadre d'applications professionnelles. De tels formats existent en effet par exemple pour le son (AIFF,

WAVE) ou encore pour l'image (BMP) et ont permis le développement de communautés capable de travailler à partir d'une base commune bien qu'à côté de cela de nombreux formats plus particuliers existent comme, par exemple, le format MP3 pour le son ou le format GIF pour l'image.

Avant même de permettre à un utilisateur de MIMESIS d'utiliser ces signaux, il nous a paru de la première importance de définir un format de codage bas niveau, générique et minimal dont la spécification serait fondée sur une analyse de leurs propriétés. Ce travail s'est inscrit dans une problématique générale du groupe ACROE-ICA qui souhaitait permettre en premier lieu l'interopérabilité entre les différents systèmes faisant intervenir des signaux de geste et de mouvement. En effet, dans un contexte où cette question semble de plus en plus urgente, question soulevée notamment au sein du réseau d'excellence ENACTIVE INTERFACES, le groupe ACROE-ICA semblait l'un des plus à même de la prendre en main, puisqu'il œuvre à la croisée de domaines aussi variés que l'animation par ordinateur, la musique par ordinateur, la simulation multisensorielle et les systèmes de réalité virtuelle tous producteurs ou consommateurs de tels signaux. Une fois une telle analyse réalisée et un format mis en place nous avons pu nous pencher sur la question de la mise en œuvre de ces signaux dans le cadre plus strict de l'animation à l'aide d'un environnement de conception interactive comme MIMESIS.

II.1. Définitions

Les signaux en entrée et en sortie des réseaux CORDIS-ANIMA ou plus généralement de tout système de simulation capable d'être contrôlé par des signaux de la nature énoncée précédemment sont de même nature. Mais leur utilisation dans le cadre de situations nouvelles apparaît avec l'arrivée des technologies de l'information. La nouveauté de cette situation rend difficile l'établissement d'un vocable pour les désigner. Dans le cadre des phénomènes acoustiques captés ou générés à l'aide de transducteurs on parle de signaux sonores. Dans le cadre d'images en mouvement captée par une caméra ou retransmise sur un écran on parle de signaux vidéo. Mais dans le cadre de mouvements captés pour être utilisés en entrée d'un système afin de le contrôler ou dans le cadre inverse de mouvements produits par un système pour contrôler un autre système ou même rendre ce mouvement par l'intermédiaire d'un transducteur gestuel, de quels signaux parler ?

II.1.2. Mouvement/Action/Geste dans une situation sans technologies de l'information

Un *mouvement* dans une situation naturelle ne faisant pas intervenir de technologies de l'information est un déplacement dans l'espace d'une partie ou de la totalité d'un système (un être humain, un objet...). En revanche, une action est, elle, le résultat de mouvements. Je décris une action lorsque je dis que je tape à l'ordinateur ce texte, je décris un mouvement ou une suite de mouvements lorsque je dis que mon index est allé de cette façon vers la touche 'e' du clavier et a appuyé de cette autre façon sur cette touche, etc.

Pour Mary M. Smyth et Alan M. Wing [SW84], psychologues du comportement, « l'action » diffère du mouvement un peu de la même façon que la perception diffère de la sensation. Lorsqu'on décrit une suite de mouvements, on dit « j'ai pris le crayon », mais on ne dit pas « j'ai bougé ma main suivant cette trajectoire », ou « j'ai contracté mes muscles dans cet ordre ». C'est que, quand nous pensons à l'action, nous la définissons en termes de buts, sans penser aux mouvements particuliers par lesquels elle a été mise en œuvre. L'utilisation du terme *action* implique une description d'une autre nature de ce que nous faisons.

Les auteurs distinguent alors trois niveaux dans l'observation d'un déplacement matériel:

- Niveau 1 : le mouvement dans son *unicité*, sa configuration spatiale et temporelle, c'est-à-dire la manière dont une action est effectuée, par exemple l'ensemble des trajectoires, les muscles mis en jeu...
- Niveau 2 : l'action proprement dite, ou la tâche, qui est la réalisation d'un événement : atteindre une cible, boire un verre, tomber à terre ...
- Niveau 3 : les éléments de ce mouvement qui en font un item *catégorisable*, la structure spatiale et temporelle qui reste constante entre deux mouvements, deux réalisations d'une même action.

Le geste serait alors le regroupement des deux termes «mouvement» et «action», c'est-à-dire un acte, considéré à la fois dans son objectif, descriptible par exemple de manière symbolique, et dans ses réalisations cinématiques en tant que phénomène spatio-temporel. Un geste de la main pourra par exemple être un mouvement de gauche à droite, main en l'air, tout autant qu'une action visant à saluer une personne.

II.1.3. Mouvement/Action/Geste dans une situation faisant intervenir les technologies de l'information

Si l'on considère maintenant une situation dans laquelle le déplacement de la matière est capté, traité, utilisée dans un système ou encore rendue par l'intermédiaire d'un transducteur comme un moteur, le contexte change. Nous nous intéressons en effet maintenant aux signaux qui circulent entre capteurs, effecteurs et systèmes. Deux grandes catégories de signaux peuvent alors être considérées, les signaux de haut niveau proposant une description de l'action réalisée et les signaux de bas niveau codant directement les déplacements et les forces mises en jeu dans le cadre de la communication homme-machine.

Les signaux de haut niveau ne sont pas ceux qui intéressent notre sujet. Ces signaux ayant rapport à des événements ne sont en effet pas ceux qui vont être produit par les capteurs. Ils peuvent être le résultat d'un traitement extrayant les propriétés d'un signal de bas niveau ou à l'inverse servir à produire un signal de bas niveau. On pourra citer par exemple les formats MIDI ou OSC. Dans le cadre de l'informatique musicale, un tel signal pourra par exemple être obtenu en traitant les informations en provenance d'un dispositif gestuel. Prenons le cas d'un clavier MIDI. Le format codera le moment où les touches sont enfoncées et la vitesse avec laquelle ces touches sont enfoncées. Mais le format MIDI ne codera pas l'ensemble des positions de la touche à tous les instants d'un échantillonnage temporel donné. Suivant un processus inverse, le format MIDI pourra servir à

contrôler le déclenchement d'évènements donnant lieu à la mise en oeuvre de signaux de bas niveau comme le signal acoustique d'un instrument virtuel.

Parallèlement à cela circulent entre différents systèmes des signaux bas niveau. Lorsqu'un déplacement de matière est capté pour produire un signal qui servira à contrôler un processus comme la simulation d'un objet physique virtuel, on parlera volontiers d'un signal de geste car le mouvement capté a une action sur un objet virtuel même si cette action n'est pas décrite. En revanche, lorsqu'un signal est produit par ce même processus, par exemple le mouvement d'un objet physique virtuel, on parlera plutôt de signal de mouvement. Mais rien n'empêche d'utiliser par la suite ce signal de mouvement en entrée d'un autre processus afin de contrôler celui-ci. Dans ce cas le signal de mouvement produit devient un signal de geste. On pourra parler par exemple du signal de geste d'un vent simulé appliqué sur une dune simulée.

Il est alors important de remarquer que si on fait abstraction de l'utilisation que fait de ces signaux, en entrée ou en sortie d'un système comme un simulateur du type de CORDIS-ANIMA, ceux-ci possèdent certaines propriétés qui leurs sont propres, les déterminent et les différencient par exemple des signaux acoustiques et des signaux vidéos qui peuvent être produit eux aussi par une simulation.

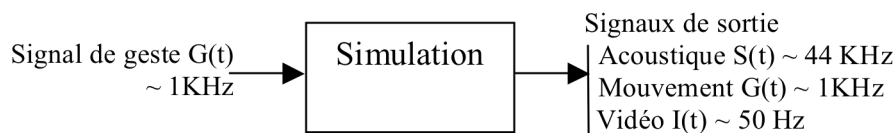


Figure 57 : Signaux mis en jeu en entrée et en sortie d'une simulation physique.

La question de la dénomination de ces types de signaux de bas niveau codant des déplacements de matière est assez délicate dans une situation d'interaction avec un objet virtuel par l'intermédiaire d'un transducteur gestuel rétroactif ou interface haptique. Dans cette situation, les signaux qui transitent entre la simulation et le transducteur, dans les deux sens (signaux produits par les capteurs et signaux fournis aux effecteurs) sont bien de même nature et de la même nature que ceux présentés dans la situation précédente. L'utilisateur, lui, est face à une interface gestuelle avec laquelle il interagit. On peut alors difficilement parler de signal de mouvement en sortie de la simulation puisque ces signaux participent directement à une situation gestuelle où l'utilisateur interagit avec un instrument.

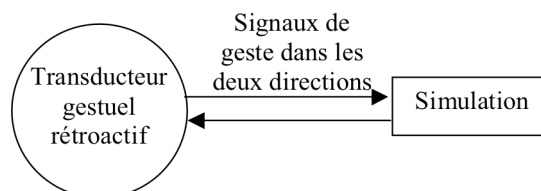


Figure 58 : Signaux mis en jeu dans une situation d'interaction avec une simulation par l'intermédiaire d'un transducteur gestuel rétroactif.

La distinction que nous faisons donc entre signal de mouvement (en sortie d'un processus de génération de geste) et signal de geste (en entrée d'un processus pour le contrôler) est en fait purement une question de sémantique. La difficulté à trouver une dénomination globale nous vient du fait que cette situation est nouvelle et ne trouve pas son équivalent dans des cas où les technologies de l'information n'intervenaient pas. Les propriétés de ces signaux sont en revanche similaires si on omet leur fonction dans l'utilisation qu'on en fait. Nous choisissons donc de les nommer « signaux de geste et de mouvement » à défaut d'un terme général difficile à trouver dans le vocabulaire existant. Nous allons en énoncer plus exactement les différentes propriétés.

II.2. Caractéristiques des signaux de geste et de mouvement

Les signaux de geste et de mouvement ont des propriétés communes spécifiques qui les distinguent donc d'autres signaux temporels comme les signaux acoustiques ou les signaux vidéos, et ce, indépendamment de la façon dont ils ont été produits (capture du mouvement d'un homme ou d'un objet, simulation...), qu'ils soient considérés comme entrée ou comme sortie d'un système évoluant temporellement. Nous énonçons dans ce qui suit les propriétés de ces signaux à savoir leur variabilité morphologique, les échelles quantitatives mises en jeu et le types des variables qu'ils codent.

2.1. *Variabilité morphologique*

Une première spécificité des signaux de geste et de mouvement est leur versatilité morphologique. Les images et les sons sont transmis à l'homme via des transducteurs dont la morphologie peut être fixe : par exemple, un écran avec une taille fixe et une résolution prédéfinie pour l'image, un ou plusieurs haut-parleurs pour le son. Au contraire, les transducteurs gestuels ont une morphologie bien plus variable qui dépend largement de la tâche à accomplir et des objets manipulés. Nous proposons de structurer cette variabilité selon deux dimensionnalités complémentaires : une dimensionnalité géométrique et une dimensionnalité structurelle.

Dimensionnalité géométrique

La dimensionnalité géométrique fait référence à la dimensionnalité de l'espace dans lequel le geste ou le mouvement évoluent. Cette dimensionnalité peut varier, d'un scalaire unique à un ensemble de scalaires pour un espace 1D, 2D, 3D... Par exemple, une seule dimension du geste est prise en compte lors de l'enfoncement de la touche d'un piano ou de la clé d'une clarinette. Ce geste est donc descriptible par un scalaire. Plus généralement, un contrôleur des paramètres d'un son (par exemple la valeur d'une raideur ou l'amplitude d'une enveloppe) fait appel à des technologies où une seule dimension est nécessaire à la description du geste opéré, dimension orientée ou non. De manière différente, dans l'animation 2D, l'espace d'application du geste est souvent un plan. Le crayon ou le stylet évoluent dans un espace 2D, sur une feuille ou une palette graphique par exemple. Ces gestes peuvent être décrits suivant deux axes orientés.

Lors de la capture d'un geste de danseur par exemple, l'espace d'évolution est elle 3D. On peut considérer que cette dimensionnalité augmente jusqu'à un espace 6D quand on manipule un objet capable d'entrer en translation et en rotation. En reprenant le cas du crayon, on peut s'intéresser à la capture de son mouvement descriptible à l'aide de 6 dimensions alors que les deux dimensions sur laquelle la mine évolue peuvent être suffisantes pour décrire le dessin en train de se réaliser et non pas le geste global du crayon tenu par le dessinateur. La figure suivante présente différentes situations dans lesquelles les gestes et les mouvements sont présents et illustre les diverses dimensionnalités qui peuvent être à l'œuvre : 1D non orientée (c et d), 1D orientée (a), 2D (e, f, g) ou encore 3D (h).



Figure 59 : Situations variées où les gestes et les mouvements interviennent dans des dimensionnalités différentes.

Dimensionnalité structurelle

Les signaux de geste et de mouvement présentent un autre type de variabilité qui découle d'une organisation fonctionnelle. Nous appelons cette seconde variabilité, la dimensionnalité structurelle du geste. Nous pouvons essayer d'illustrer cette propriété à l'aide de quelques exemples. Lorsque nous jouons sur un clavier à n touches, nous avons vu que la dimensionnalité utile du geste réalisé ou, de la même manière, le signal produit par chacune des n touches est 1D. Mais rien n'empêche de considérer également le système comme composé de n entités à une touche. On obtient alors un système à n degrés de liberté. Considérer un clavier comme n systèmes 1D ou comme un système n D est donc un choix, une façon de voir, qui dépend de la tâche à effectuer ou du type de critères sur lesquels on met l'accent. Ainsi par exemple, il n'est probablement pas réducteur de considérer un clavier

orthographique comme un n systèmes 1D, alors que dans le cas du piano, si l'on souhaite mettre l'accent sur les composantes de co-articulation des doigts d'une main (lors d'un arpège par exemple), il sera nécessaire de considérer le système comme un système nD . Ainsi, le choix de l'organisation fonctionnelle est un choix faisant intégralement partie d'une activité de modélisation.

On peut retrouver cette variabilité dans le cas d'une corde frottée simulée à l'aide d'un modèle physique. Dans ce cas, les deux dimensions de la déformation (la pression sur la corde et le frottement) sont généralement découplées et le système peut être considéré comme deux systèmes 1D superposés. Donc comme un système à deux degrés de liberté 1D.

On peut aussi considérer le mouvement d'un corps humain. La dimensionnalité géométrique de l'espace est 3 mais la dimensionnalité structurelle est variable. Elle dépend du nombre de degré de liberté considéré. Elle s'élève à plus de 200 quand on considère le corps réel mais elle chute à une dizaine lorsqu'on capture le mouvement à l'aide d'un système de capture.

Pour un système physique de n particules 3D, n peut être très important, de l'ordre de plusieurs milliers. Ce flux de particule peut alors contrôler une scène complexe et de grande étendue spatiale (par exemple un vent qui contrôle un instrument de musique, ou des déformations océaniques). Le « geste » de ce vent dans ce cas peut être vu comme un signal de geste de n degrés de liberté 3D.

Cette analyse et ces exemples montrent que la dimensionnalité géométrique des signaux de geste et de mouvement n'est pas fixe et peut varier d'un scalaire à une plus haute dimension. Mais ils montrent aussi que la dimensionnalité structurelle peut largement varier aussi (de 1 à plusieurs centaines). De plus, si la dimensionnalité géométrique est une donnée objective de la situation, la dimensionnalité structurelle elle peut varier suivant l'analyse que l'on en fait et donc dépend de la spécification de celui qui met en place une scène où le signal de geste et de mouvement entre en jeu.

2.2. Echelles quantitatives

Outre les questions de dimensionnalité, géométrique et structurelle, les signaux de geste et de mouvement présentent un ensemble de propriétés spatiales et temporelles spécifiques et d'ordre quantitatif.

Echelle de temps

Comme le montre la figure, la bande passante en fréquence d'un signal de geste et de mouvement est une propriété qui le distingue d'un signal acoustique ou d'une image. Alors que les signaux visuels varient de quelques Hertz à une centaine et que les signaux acoustiques sont de l'ordre de plusieurs dizaines de milliers de Hertz, les signaux de geste et de mouvement varient eux de quelques Hz à quelques milliers de Hz. Notamment, manipuler un objet avec une interface à retour d'effort nécessite souvent bien plus que la fréquence des signaux visuels. On peut donc placer les signaux de geste et de mouvement entre les signaux acoustiques et les signaux visuels pour ce qui est du critère de la bande passante.

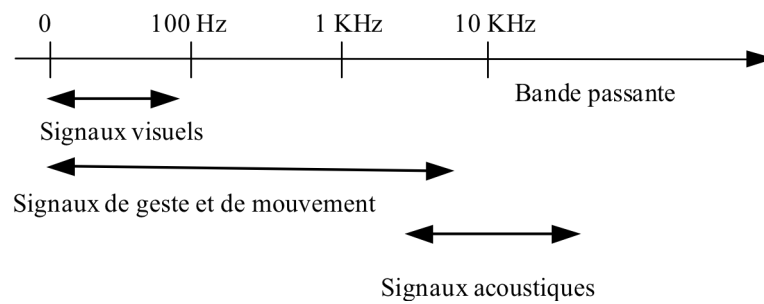


Figure 60 : Comparaison de la bande passante des signaux de geste et de mouvement avec d'autres types de signaux.

Echelle spatiale

Les positions spatiales et l'échelle des positions sont d'autres caractéristiques des signaux de geste et de mouvement. Les signaux acoustiques sont des signaux vibratoires centrés, c'est-à-dire oscillant autour de zéro sans composante continue et correspondent à des déformations de quelques millimètres d'amplitude spatiale. Les signaux de geste et de mouvement, de même que les signaux visuels correspondent à des déformations et à des déplacements qui ne sont pas forcément centrés et dont l'amplitude spatiale peut varier de quelques centimètres à plusieurs mètres. On peut donc dresser une première conclusion. Les propriétés spatiales et temporelles quantitatives des signaux de geste et de mouvement les placent au milieu des autres signaux représentant des phénomènes sensoriels. Lorsqu'on fait intervenir un retour d'effort ou une simulation physique dans une plateforme de simulation multisensorielle, les signaux de geste et de mouvement mis en œuvre sont en quelque sorte similaires aux signaux visuels en ce qui concerne la spatialité mais demande une plus grande plage de fréquence, en revanche, ils nécessitent une fréquence moins importante que le son, mais nécessite une plage spatiale plus importante et pas nécessairement centrée sur l'origine.

2.3. Type de variables

Lorsque les gestes et les mouvements sont produits par un système physique, qu'il soit réel ou virtuel, et sont utilisés comme contrôle d'un modèle physique, les données incriminées peuvent être de deux types différents. Ce peut être des variables extensives, c'est-à-dire des variables qui dérivent des informations spatiales comme les positions, les vitesses, les angles. Mais ce peut être aussi des variables intensives comme les forces ou les couples. Quand les gestes sont utilisés pour manipuler des objets, la corrélation entre variables extensives et variables intensives échangées entre deux corps en interaction doit être considérée soit de manière explicite comme dans des formalismes de type newtoniens, soit de manière implicite comme dans des formalismes faisant intervenir la notion d'énergie. Les lois décrivant ces situations proposent une corrélation entre des variables extensives et des variables intensives. C'est le cas explicitement dans un formalisme tel que le formalisme newtonien. Cela fait encore une différence avec les signaux acoustiques et les signaux visuels qui ne font intervenir que des variables extensives. Les variables extensives autant qu'intensives sont nécessaire à un codage complet du geste lorsque ce geste intervient dans une situation d'interaction.

II.3. Un format de bas niveau, générique et minimal n'existe pas encore

Plusieurs formats existent qui codent des signaux de mouvement ou de geste sous forme de signal [Me00, VanB00]. Nous en présentons les plus connus avant de les considérer du point de vue des propriétés que nous avons énoncées pour les signaux de geste et de mouvement en nous demandant s'ils peuvent prétendre à être à la fois de bas niveau, génériques et minimaux

II.3.1. Revue des formats codant des signaux de geste et de mouvement

Les formats BVH et BVA

Le format BVH (pour Biovision Hierarchical Data) est l'un des plus couramment présent dans les applications faisant intervenir des animations humanoïdes. Il a été initialement développé par Biovision, une entreprise de motion capture. Il est considéré comme une amélioration du format BVA, son prédécesseur. Le but de Biovision était de donner une certaine standardisation des structures humanoïdes en gardant une spécification très simple.

Le format MNM

Ce format permet de renommer les segments d'un fichier BVH. L'utilisateur peut définir un nom associé à un label prédéfini des fichiers BVH.

Les formats ASK/SDL

Ces formats sont une variante du format BVH développé par Biovision. Le format ASK (Alias Skeleton) contient uniquement des informations sur la structure du squelette humanoïde. Les données de mouvement sont codées dans un fichier SDL séparé associé au fichier ASK. Ce fichier peut en outre contenir d'autres informations concernant la scène. Il est à noter que les coordonnées sont données de manière absolue alors que les coordonnées des fichiers BVH étaient relatives à la hiérarchie.

Le format AOA

Adaptative Optics est une entreprise dont l'activité est dédiée à la création de support matériel pour la motion capture. Leur format ASCII décrit simplement les capteurs et leur position à chaque instant d'échantillonnage.

Les formats ASF/AMC

Ces deux formats ont été développés par Acclaim, une entreprise de jeux vidéo. Une fois entrés dans le domaine public, ils ont été utilisés par Oxford Metrics pour les systèmes de motion capture Vicon. Les deux formats vont de pair. L'un code la structure du squelette (le fichier ASF) et l'autre code le

mouvement (le fichier AMC). Une telle séparation a été opérée car une configuration de squelette sert souvent pour plusieurs mouvements.

Le format BRD

Ce format est utilisé par les systèmes de motion capture Flock of Birds de Ascension Technology et développé par Lambsoft. Il stocke des données provenant d'un système de motion capture magnétique directement posé sur le corps de l'acteur.

Les formats HTR and GTR

Le format HTR (Hierarchical Translation Rotation) a été développé pour le logiciel Motion Analysis pour coder le mouvement de structures humanoïdes. Il a été créé car les développeurs n'étaient pas satisfaits du format BVH qui convenait mal à leur application. Une variante de ce format existe. Il s'agit du format GTR (Global Translation Rotation) qui ressemble au format HTR mais donne moins d'informations sur la structure du squelette.

Le format TRC

Le format TRC est un autre format utilisé par Motion Analysis. Il contient uniquement les données brutes venant de leur système de capture du mouvement du corps et du visage.

Le format CSM

Le format CSM est un format utilisé par Character Studio (un *plug-in* d'animation et de *skining* pour 3D studio MAX) qui permet d'importer des données provenant de marqueurs. Ce format est très spécifique au logiciel, les marqueurs ont un nom prédéfini.

Le format C3D du National Institute of Health

Beaucoup d'entreprises de motion capture ont un pied dans la recherche en biomécanique. Leurs systèmes peuvent par exemple être utilisés pour évaluer les performances d'un athlète ou bien les besoins d'une personne handicapée moteur. Les besoins des chercheurs, souvent confrontés à l'utilisation de plusieurs systèmes, a amené à la définition d'un format commun pour coder les données de mouvement. Le format C3D. Ce format a été conçu pour permettre de porter de façon la plus complète possible les informations utiles à la recherche en biomécanique.

Ce format offre la possibilité de stocker des « données analogiques » c'est-à-dire ici des signaux provenant directement de la mesure ainsi que des « données tridimensionnelles » obtenue par traitement des données brutes. Il est de plus possible de stocker des informations sur le matériel utilisé, le type de capteur (marqueurs de position, capteurs de force), sur le processus de mesure (taux d'échantillonnage, date, type d'examen) ainsi que sur le sujet lui-même (nom, âge, paramètres physiques...). Les bibliothèques gérant ce format offrent la possibilité d'ajouter de nouvelles données à celles déjà enregistrées. En outre, c'est le seul format binaire dans la liste des formats que nous avons présentés.

Tous les formats considérés ont été historiquement introduits dans des contextes très spécifiques d'utilisation et d'application. La plupart viennent de la capture de mouvements humains. Bien que certains de ces formats aient été utilisés pour des applications autres que celles d'origine, les

spécifications ont été variablement motivées par une généricité permettant un usage très large des formats. En conséquence, la plupart du temps, la manière de coder les données dépend fortement de l'utilisation que l'on prévoit de faire de ces formats. Nous effectuons dans la suite de cette partie une étude de ces formats, au regard du problème des critères de généricité, de minimalité, et du caractère bas niveau en nous interrogeant sur le fait que toutes les propriétés que nous avons définies pour les signaux de geste et de mouvement soient bien présents.

II.3.1. Comparaison de ces formats selon les critères recherchés

Dépendance vis-à-vis de l'objet en mouvement

Excepté pour le format C3D, tous les formats présentés ici sont dépendant d'une structure hiérarchique. Cette structure hiérarchique est directement héritée de la structure hiérarchique des objets en mouvement dont le signal est l'image. Il est en effet intéressant de décrire par exemple le mouvement d'un corps humain en faisant intervenir une structure arborescente qui ressemblerait au squelette. Les os sont les liens, les articulations, les nœuds [HFP*00]. Dans ce cas, la plupart du temps, les valeurs de positions et de rotations sont données de manière implicite dans un système de coordonnées dépendant de la structure hiérarchique. De telles structures hiérarchiques ont prouvé leur efficacité dans des domaines comme l'animation de personnages et sont en réalité très adaptées au cas des objets articulés comme le corps humain, les robots... Mais de telles structures hiérarchiques ne peuvent rendre compte des signaux de geste et de mouvement dans l'acceptation générale que nous en donnons, notamment dans le cas des objets déformables non articulés. De plus dans beaucoup de situations où l'interaction avec une interface gestuelle intervient comme un joystick ou encore un stylet, la structure hiérarchique n'est absolument pas pertinente. D'ailleurs, on remarque que dès qu'il a été souhaité de définir un format assez générique capable de s'adapter à différentes applications, comme le format C3D par exemple, la structure hiérarchique en est absente.

Dépendance vis-à-vis du contexte

Parmi les formats présentés, certains contiennent des informations additionnelles sur le contexte d'acquisition des données. Il s'agit, par exemple, d'informations sur les paramètres de la capture du mouvement (nombre de capteurs, type de capteur, sensibilité des capteurs, etc.). Le format C3D est à ce titre très représentatif d'un tel ajout de données. Dans le format C3D, chaque signal est associé à une description du processus de mesure utilisé. Un format tel que C3D est donc initialement indiqué pour traiter des questions de mesure et d'expérimentation. Cela ne fait pas un argument rédhibitoire contre l'utilisation d'un tel format dans la communication par signaux de geste et de mouvement. Mais le caractère minimal du format est remis en jeu. Cela indique un risque de blocages dans l'évolution des spécifications et la possibilité de zones inutiles si l'on souhaite s'intéresser au signal de geste et de mouvement en tant que tel dans un cadre plus large que l'acquisition de données pour la recherche en biomécanique.

Dimensionnalité prédéfinie

La plupart des formats imposent une dimensionnalité prédéfinie pour les données : points 3D dans AOA ou CSM, « points 7D » dans BRD. Dans les formats BVH, ASF/AMC et HTR, on déclare la

dimension pour chaque articulation dans un rapport avec les contraintes imposées par la structure hiérarchique sous-jacente. Ne pas déclarer une coordonnée correspond donc à définir une contrainte pour un système mécanique. Nous devons donc remarquer qu'à nouveau le choix d'une structure hiérarchique sous-jacente au signal de geste et de mouvement implique un manque de liberté au niveau de la description de la dimensionnalité. On peut détourner le format C3D pour traiter certaines questions relatives à la dimensionnalité sans toutefois couvrir toutes les possibilités et, répétons-le, en utilisant des structures qui ne sont initialement pas prévues pour répondre à nos questions. Ainsi, si l'on peut stocker des informations sur des positions 3D, il est possible de stocker aussi des informations 1D ou 2D en fixant une ou plusieurs coordonnées à 0. On voit que cette « astuce » n'en est qu'une et que la question de la versatilité de la dimensionnalité d'une scène gestuelle n'est pas au cœur des spécifications de ce format.

Types de variables

La quasi-totalité des formats ne propose comme types de variable que des variables extensives. Or nous avons vu que les variables intensives comme la force peuvent être une composante des signaux de geste et de mouvement. Là encore, le format C3D pourrait nous permettre de coder des variables de type force puisqu'il permet de définir un nombre illimité de données scalaires telles que des rythmes cardiaques, des électrocardiogrammes... Mais le codage de la force n'aurait pas le même statut que le codage des variables extensives. Comment définir une force 3D, une structuration des différents canaux ?

Codage des variables

Hormis le format C3D, tous les autres formats sont des formats ASCII. Cela vient du fait que la bande passante des systèmes de motion capture est souvent assez basse (de 10 à quelques centaines de Hz) et surtout que l'usage de ces fichiers pour l'animation nécessite une bande passante de l'ordre de la fréquence visuelle et non de la fréquence du geste. Or, aujourd'hui, il est de plus en plus admis que les applications faisant intervenir des interfaces gestuelles, notamment à retour d'effort, nécessitent des bandes passantes de l'ordre de plusieurs kilohertz. De plus, la diversification des objets produisant du mouvement (réels ou virtuels) et la décroissance des prix des capteurs, impliquent que le nombre de voies pour le signal augmente de manière significative. Dès lors, la quantité de données des signaux de geste et de mouvement augmente considérablement. Dans une optique où le temps-réel est recherché, le codage ASCII n'a plus sa place et doit faire la place à un codage binaire. Le meilleur candidat pour coder les signaux de geste et de mouvement est donc le format C3D. Mais celui-ci, initialement prévu pour un but autre que le codage le plus générique des données de geste, dans un cadre où toutes les situations n'ont pas été considérées, ne peut être utilisé par exemple en entrée de MIMESIS que de manière détournée. Deux choix s'offrent à nous, soit nous nous contentons d'un format existant, soit nous en spécifions un nouveau en le souhaitant plus minimal et plus générique. Nous avons fait ce deuxième choix. Les spécifications de ce format sont présentées dans la partie suivante.

II.4. Format proposé : le format TMGMS

Le fait de définir un format générique, minimal et bas-niveau même si celui-ci produit des fichiers de taille importante permet de cimenter une communauté. Cela permet un échange de données, une communication facilitée. Ce format, nous l'avons dit n'existait pas. À ce titre, nous allons proposer un format de codage des signaux de geste et de mouvement que nous baptisons TMGMS, pour *gesture and motion signal*, à partir de l'analyse que nous avons faite de la spécificité des signaux de geste et de mouvement.

Le développement du format TMGMS a été supporté par le réseau européen d'excellence *IST-2002-002114 - Enactive Interfaces*. Il a fait l'objet d'une bibliothèque libre diffusée sur Internet (<http://www-acroe.imag.fr/gms>) et d'une soumission auprès des autorités de dépôts des formats de fichier: type MIME (*Multipurpose Internet Mail Extensions*) et auprès de IANA (*Internet Assigned Number Authority*) ainsi que de plusieurs publications dans le contexte de l'animation [LEC*06], des réalités virtuelles [Lev*06], ainsi que de l'informatique musicale [ECL*06].

II.4.1. Taux d'échantillonnage

Dans la version 0.1 que nous proposons, notre format ne supporte qu'une unique fréquence qui doit être suffisamment importante pour supporter un signal de geste et de mouvement. Ce choix pourra être reconsidéré dans des versions futures de notre format pour permettre la gestion de signaux multifréquence qui peut avoir lieu lors de situations d'interactions gestuelles complexes.

II.4.2. Codage de la géométrie et de la structure

Nous proposons une structuration du signal de geste et de mouvement suivant 4 niveaux qui tiennent compte de l'analyse que nous avons faite : la voie, le canal, l'unité et la scène. Ces 4 niveaux représentent une structuration minimale du signal de geste et de mouvement pour tenir compte des variabilités géométriques et structurelles du geste.

La voie

Une voie contient un signal unidimensionnel $a_i(t)$ qui correspond à une voie des interfaces gestuelles (convertisseur analogique-numérique). Elle est au plus près des informations fournies par le capteur.

Le canal

Le canal supporte la dimensionnalité géométrique du signal de geste et de mouvement. C'est un ensemble de voies. Dans la version actuelle de notre format, un canal peut être 1D0 (un scalaire pur), 1Dx (un vecteur suivant l'axe x), 1Dy, 1Dz, 2Dxy, 2Dyz, 2Dzy ou 3Dxyz. Un canal peut coder soit une variable extensive VE (position P, Angle A, vitesse V, accélération G), soit une variable intensive VI (force F, couple T).

L'unité

Si le canal supporte la dimensionnalité géométrique du signal de geste et de mouvement, l'unité supporte elle la variabilité structurelle du signal de geste et de mouvement. Une unité est donc un ensemble de canaux qui ne sont dynamiquement dépendant (considérés comme tel par le concepteur). Une unité diffère largement du concept de hiérarchie utilisé dans la motion capture. Il n'y a qu'un seul niveau pour les unités ce qui représente une structuration minimale du geste. La définition d'une unité permet d'empêcher les décorrélations malencontreuses du signal lors de son utilisation et donc les utilisations exotiques des signaux de geste et de mouvement.

La scène

La scène gestuelle correspond au signal de geste et de mouvement en entier. La scène porte donc des informations générales comme la fréquence d'échantillonnage. Une scène est composée de plusieurs unités qui ne sont pas couplées dynamiquement.

II.4.3. Exemples de structuration

Nous fournissons dans cette partie une scène gestuelle d'une installation interactive imaginée de façon à illustrer la manière de structurer une scène gestuelle.

Un musicien joue d'un petit clavier à 8 touches. Il peut contrôler la spatialité du son à l'aide d'un joystick 3D. Un acteur contrôle lui l'orientation d'une source lumineuse à l'aide d'un capteur de force 2D. Un danseur équipé d'un système de motion capture avec 16 marqueurs est présent ainsi qu'un jongleur manipulant une interface à retour d'effort 6D. Un fluide virtuel dont les mouvements sont simulés et projetés agit sur une corde virtuelle. Ce fluide est composé de N masses 3D.

Nous pouvons modéliser cette situation à l'aide de 7 unités : l'unité « piano », l'unité « joystick », l'unité « source de lumière », deux unités pour le « stick 6D », l'unité « fluide ». L'unité « piano » est composée de 8 canaux monodimensionnels (8 touches). L'unité « joystick » est un canal 3D. L'unité « source de lumière » est un canal 2Dxy supportant des variables intensives. L'unité danseur dispose elle de 16 canaux 3Dxyz. L'unité « stick 6D 1 » comporte un canal 3Dxyz donnant les positions du stick alors que l'unité « stick 6D 2 » porte les informations d'orientation dans un canal 3Dxyz. L'unité « fluide » comporte n canaux 1D.

Nous remarquons que dans l'implémentation actuelle de notre format, lorsque le cas du 6D apparaît, nous sommes obligés de décorréler les positions des orientations. On peut imaginer étendre ce format pour permettre des canaux 6D mais nous n'avons jamais encore rencontré ce cas d'utilisation lors de nos travaux. Pour autant le format peut prévoir une telle extension sans remettre en cause les spécifications précédentes.

Unités	Canaux
Piano	8 canaux unidimensionnels TP (touches de piano) : TP1 [VE(P), 1Dz], ..., TP 8 [VE(P), 1Dz].
Joystick	Un canal 3Dxyz joystick J: J [VE(P), 3Dxyz]
Source de lumière	Un canal 2Dxy source de lumière SL : SL [VI(F), 2Dxy]
danseur	16 canaux 3Dxyz danseurs D : D1[VE (P), 3Dxyz], ..., D16[VE(P), 3Dxyz]
Jongleur 1 et 2	1 canal 3Dxyz positions stick PS : PS [VE (P), 3Dxyz]
	1 canal angle stick AS AS [VE (A), 3Dxyz]
fluide	n canaux 1D masses fluide MF MF[VE (V), 1D0]

Figure 61 : Situation gestuelle multidimensionnelle complexe

II.4.4. Implémentation du format TMGMS

Ce paragraphe est un résumé de l'implémentation courante de notre format TMGMS dans sa version 0.1.

Le format suit la norme IFF (Interchange File Format) qui est un standard du codage binaire en *big endian*. Une description de ce standard peut être trouvée dans [Mor85]. Son équivalent pour le codage *little endian* est le standard RIFF. Ces deux standards sont largement utilisés. On les retrouve notamment à la base de formats tels que AIFF pour le son, AVI pour la vidéo, ILBM ou GIF pour les images. Le standard IFF nécessite que toutes les données soient encapsulées dans des structures appelées « *chunks* ». Il est de plus possible de concaténer plusieurs fichiers de type IFF mais de nature différente dans un seul fichier. On peut donc imaginer concaténer deux fichiers TMGMS à des fréquences différentes comme première réponse à la multifréquence.

Le tableau de la figure 62 présente une description des données dans chaque *chunk*. Le type binaire de ces données peut être CHAR (un octet), USHORT (entier non signé 2 octets), LONG ou ULONG (entier signé ou non sur 4 octets), FLOAT32 ou FLOAT64 (flottant 4 ou 8 octets suivant le standard IEEE 754 sur le codage des flottants). Dans la colonne de gauche, on trouve le nom du *chunk*. Face à lui, on trouve les données qu'il contient avec à chaque fois leur type.

Header	CHAR[4] chunkId = 'FORM'		ULONG fileSize;
	CHAR[4] fileType = 'GSM'		
Version	CHAR[4] chunkId = 'VERS'		ULONG chunkSize
	USHORT versNum		USHORT subVersNum
Scene	CHAR[4] chunkId = 'SCEN'		ULONG chunkSize
	USHORT sceneNameLength		CHAR*sceneName
	ULONG nbFrame	FLOAT64 freq	USHORT dataType
	FLOAT64 scale	ULONG blockSize	
Unit	CHAR[4] chunkId = 'UNIT'		ULONG chunkSize
	USHORT unitNameLength		CHAR* unitName
Channel	CHAR[4] chunkId = 'CHAN'		ULONG chunkSize
	USHORT chanNameLength		CHAR* chanName
	USHORT dimension		USHORT type
Frame	CHAR[4] chunkId = 'FRAM'		ULONG chunkSize
	TypeOfData[nbTrack][nbFrame] frames		

Figure 62 : Implantation du format TMGMS.

Le *chunk* « version » correspond à la description de la version du fichier. Pour l'instant il s'agit de la version 0.1 mais des évolutions futures sont prévues.

Le *chunk* « scène » contient les informations sur la scène gestuelle. *dataType* donne le type de données du signal. La version actuelle supporte les types FLOAT32, FLOAT64 et LONG. *Scale* est un facteur d'échelle à appliquer au signal dans le cas de dépassement des possibilités du codage des nombres. *blockSize* permet l'alignement des échantillons du signal. Cela est souvent nécessaire lors d'applications temps réel.

Les *chunk unit* et *channel* codent les informations sur les unités et les canaux gestuels. L'entier *dimension* code la dimension des canaux alors que *type* donne le type de variable (position, angle, force...).

Le *chunk frame* donne alors le signal lui-même échantillon par échantillon.

II.5. Les sorties des réseaux CORDIS-ANIMA sont des signaux de geste et de mouvement : les modules enregistreurs

Nous avons affirmé en introduction de ce chapitre que les réseaux CORDIS-ANIMA étaient capables à la fois de produire des signaux de geste et de mouvement et d'être contrôlés par ce type de signaux quand ceux-ci sont placés en entrée de ces réseaux.

La production de signaux de geste et de mouvement paraît assez naturelle. En effet, les réseaux CORDIS-ANIMA font intervenir des variables extensives et intensives dont la nature et l'organisation sont compatibles avec les propriétés des signaux de geste et de mouvement que nous avons énoncées. Le format TMGMS est adapté au codage des sorties des réseaux CORDIS-ANIMA.

II.5.1. Enregistreurs de position

La variable de position en sortie d'un module MAT peut être portée par un canal du format TMGMS puisque ce canal porte l'information de dimensionnalité du signal de geste et de mouvement. Dire que la sortie d'une variable de position est un canal d'un signal de geste et de mouvement revient en effet à dire que les coordonnées en sortie d'un module MAT forment un tout dont l'indivisibilité à un sens du point de vue de la dimensionnalité. Il est possible d'ailleurs d'envisager l'introduction de modules enregistreurs de position (EP) possédant un point L qui vient lire le signal de position en sortie d'un module MAT en l'enregistre par exemple dans un fichier TMGMS en renvoyant à ce module une force nulle de manière à ne pas modifier le comportement de la simulation (cf. figure 63). Ce module de position parfaitement compatible avec le formalisme CORDIS-ANIMA, puisqu'il est composé d'un points de connexion du type de ceux des modules CORDIS-ANIMA, ne modifiant pas la simulation a donc un rôle d'observateur de la simulation de la même manière qu'un capteur ne modifie pas le comportement d'un dispositif réel (ou plutôt que son action est négligeable).

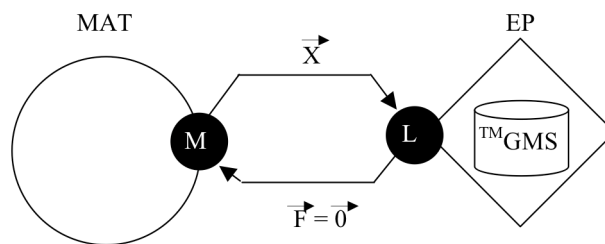


Figure 63 : Connexion d'un enregistreur de position (EP) à un module MAT.

II.5.2. Enregistreurs de force

Si l'on s'intéresse au problème dual, celui du signal de force en sortie des modules LIA, la définition d'un module dont la seule fonction serait d'enregistrer ce signal n'est pas possible. En effet, ce module potentiel serait composé d'un point M que l'on connecterait au point L du LIA. Or ce point L, qui ne peut être connecté qu'à un seul point M, devrait recevoir une information de position qui, étant une variable extensive et non intensive, agirait nécessairement sur le comportement du LIA. Une valeur de position, même nulle, n'est pas une intensité nulle. Enregistrer les signaux de forces qui entre dans les signaux de geste et de mouvement ne pourrait donc se faire, sans rompre avec les spécifications du formalisme CORDIS-ANIMA, qu'à l'intérieur même des LIA qui deviendrait des modules calculant des forces et les enregistrant dans le même algorithme. Si rien n'empêche l'enregistrement des signaux de force, nous voyons qu'au sein du formalisme CORDIS-ANIMA, le traitement de ces deux variables duales n'est pas le même et que par une simple permutation entre points L et points M, la métaphore sous-jacente n'est pas la même voire qu'elle n'est pas forcément possible. Ceci apparaîtra de manière encore plus manifeste dans la suite de cette partie lorsque nous nous intéresserons à la question du contrôle d'un modèle CORDIS-ANIMA par un signal de geste portant une information de force.

II.5.3. L'importance de définir une métaphore

Nous voyons donc que les signaux de sortie des réseaux CORDIS-ANIMA sont de signaux de geste et de mouvement. Les signaux d'entrée étant de même nature que ceux de sortie, il est donc envisageable d'utiliser un signal ayant pour format TMGMS comme signal d'entrée et de contrôle d'un réseau CORDIS-ANIMA. Si, dès lors tout type de modalité de contrôle est possible, est-il possible de donner à n'importe laquelle de ces modalités un sens ? Autrement dit, le fait d'affecter en entrée d'un réseau CORDIS-ANIMA n'importe quel signal de geste et de mouvement est-il justifiable dans un cadre de création et surtout quelle justification peut-on en donner ? Derrière cette question se cache la question de la métaphore utilisée lors d'une telle pratique. Dans le cadre d'un processus de création, la métaphore est en effet une question qu'il est important de soulever au risque de disposer d'outils dont la fonctionnalité est opaque. Plus que pour la sortie, la question de l'entrée de signaux gestuels nécessite une métaphore. Dans les deux parties suivantes, nous allons donc nous interroger sur la pertinence d'utiliser des signaux de geste pour contrôler des modèles CORDIS-ANIMA au regard de la métaphore qui peut être proposée pour ce contrôle.

II.6. Signaux de geste de position en entrée des réseaux CORDIS-ANIMA

II.6.1. De l'interaction bidirectionnelle à « l'interaction unidirectionnelle » dans CORDIS-ANIMA

Le caractère bidirectionnel porté par les modules LIA dans CORDIS-ANIMA est un principe fondateur du formalisme hérité du concept d'interaction newtonien. Sur toute masse ponctuelle ne s'applique une force qu'en ce qu'elle est en interaction avec une autre masse ponctuelle. De même dans le formalisme CORDIS-ANIMA, tout module LIA est connecté à deux modules MAT. Cependant, le caractère bidirectionnel des interactions peut être considéré comme unidirectionnel dans le cas où l'interaction entre deux éléments est telle que l'on peut négliger son influence sur l'un des deux composants comme par exemple quand celui-ci possède une inertie assez importante pour que l'effet de la force ne se fasse pas sentir.

On rencontre très tôt ce genre de situation lors de son parcours scolaire lorsqu'on considère par exemple les problèmes de chute libre de projectiles et que l'on introduit le poids dans les équations. Dans les problèmes de type balistique, jamais on n'écrit en effet l'influence de la force exercée par le projectile sur la Terre quand bien même le principe fondamental de la dynamique de Newton nous dit que l'intensité de cette force est la même que celle exercée sur le projectile par la Terre. C'est que l'inertie de la Terre est bien supérieure à celle du projectile.

Dans le cadre de la simulation de réseaux, et en particulier de réseaux masses - interactions, Habibi [Hab97] montre que si l'on considère deux sous-ensembles de réseaux A et B, couplés uniquement par des liaisons unidirectionnelles toutes dans le même sens (de A vers B ou de B vers A), alors il est possible de simuler de manière disjointe les deux sous-parties du réseau. Plus précisément, si l'ensemble A est connecté avec l'ensemble B par un ensemble d'interactions unidirectionnelles allant de A vers B, alors on peut simuler A avant de simuler B. Dès lors, le résultat de la simulation de A peut-être stocké dans un fichier qui sera utilisé pour la simulation de B. Cette propriété est directement héritée du caractère modulaire des réseaux. La figure 64 donne un exemple de cette possibilité de simulation découpée dans le temps. Les flèches unidirectionnelles symbolisent que toutes les interactions entre les deux parties de réseau peuvent être considérées comme des interactions unidirectionnelles. Les flèches bidirectionnelles symbolisent des interactions normales.

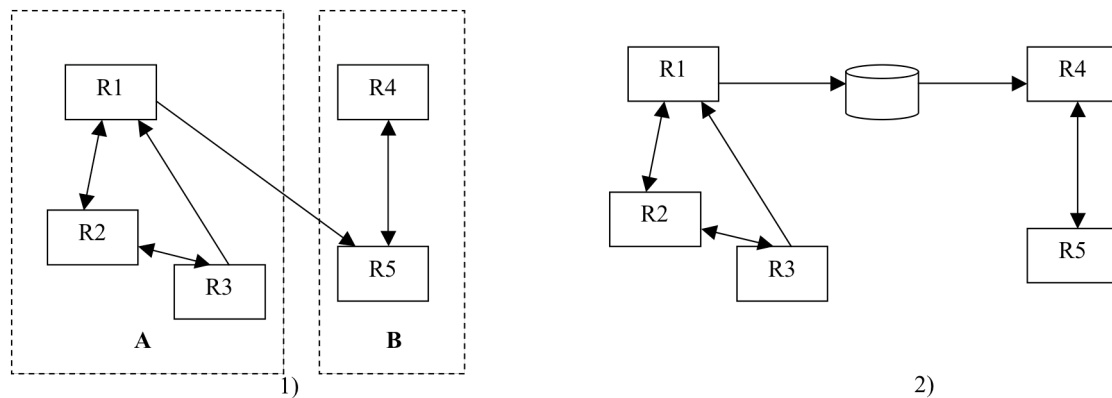


Figure 64 : Découpage temporel d'une simulation.

- 1) Un réseau global composé de deux parties A et B.
- 2) On peut simuler d'abord A et stocker le résultat de la simulation dans un fichier puis utiliser ce fichier pour simuler ensuite B.

Pour illustrer cette considération, nous pouvons prendre l'exemple de la modélisation du mouvement d'un danseur. Dans le cas où le costume entrave d'une façon ou d'une autre les mouvements du danseur, il est a priori nécessaire de modéliser les interactions entre le costume et le corps de celui-ci comme des interactions bidirectionnelles. En revanche, si le costume est léger et ample, il peut être possible de modéliser cette situation à l'aide d'interactions unidirectionnelles du corps vers le costume.

II.6.2. De l'interaction gestuelle avec les modèles CORDIS-ANIMA à l'action gestuelle

Le formalisme CORDIS-ANIMA présentant des points de communication discrets et en nombre fini est nativement compatible avec les transducteurs gestuels rétroactifs (TGR). Ce type de technologie comporte en effet à la fois des capteurs fournissant des signaux en entrée de la simulation et des effecteurs (des moteurs) contrôlés par des signaux provenant de la simulation. Tout se passe comme si l'homme, par l'intermédiaire de l'interface TGR, interagissait avec des « objets », dont le comportement est simulé par l'ordinateur, suivant une modalité bidirectionnelle: l'homme agit sur l'objet de la même façon que l'objet agit sur l'homme. Cette situation est typiquement la situation instrumentale décrite dans [Cad94]. Si dans de nombreux cas, la situation instrumentale est propice à faire émerger des effets, voire du sens qu'on n'aurait pu obtenir autrement que par ce couplage physique entre l'homme et la simulation, il existe un certain nombre de situations où un tel couplage n'est pas nécessaire. C'est le cas lorsque la suppression du retour d'effort n'a pas d'implication sur l'efficacité d'une tâche à effectuer. C'est aussi le cas dans des gestes libres, c'est-à-dire des gestes du corps en mouvement sans interaction avec d'autres objets (geste de la main, du bras), de tels gestes étant alors captés par un système de *motion capture*, le cas idéal étant celui où ces éléments libres du corps ne sont en contact avec le dispositif de capture. Un exemple permettant d'illustrer le cas où le retour de l'information de force n'est plus nécessaire peut être pris à nouveau dans le contexte de la

danse. Cette fois-ci, on ne cherche pas à modéliser les mouvements d'un danseur, mais on les capte. Les signaux ainsi obtenus peuvent être fournis à la simulation d'un costume virtuel qui va être animé en suivant les geste qui ont été captés. Le geste du danseur peut alors tout à fait être enregistré dans un fichier pour servir d'entrée à une simulation aval. Du point de vue du modèle aval CORDIS-ANIMA, celui simulé en dernier, les deux cas présentés, celui où l'on calcule deux simulations à des instants successifs, et le cas où aucun retour d'effort n'est nécessaire à la situation d'interaction entre l'homme et la simulation sont semblables. Dans les deux cas, le modèle reçoit en entrée des signaux dont la nature est la même, qu'ils proviennent d'une simulation amont ou d'un capteur. Ce sont des signaux de geste et de mouvement qui sont utilisés. Plus précisément les signaux enregistrés portent des variables de position puisque ces signaux sont les signaux fournis à l'interaction considérée comme unidirectionnelle et. Pour le réseau CORDIS-ANIMA recevant donc des signaux codant des positions, que peuvent représenter ces signaux. Quelle métaphore est sous-jacente à cette rupture, métaphore compatible avec le formalisme CORDIS-ANIMA ?

II.6.3. Générateur de position et réseaux de Kirchhoff

La théorie des réseaux de Kirchhoff [BN83] nous fournit un élément conceptuel tout à fait applicable aux cas que nous avons développés. Le générateur de tension est un dipôle qui fournit une tension (variable extensive) à un réseau de Kirchhoff, tension dont la valeur ne dépend pas du reste du circuit électrique auquel il est connecté donc est indépendante de l'intensité (variable intensive) qui le traverse. La situation est similaire dans le cas où un signal gestuel contrôle un réseau CORDIS-ANIMA. Les données de position sont indépendantes du système à qui elles sont fournies, la force produite en retour par le système n'étant pas prise en compte. Dans les termes de la théorie des réseaux de Kirchhoff, une telle opération est donc équivalente à l'application d'un générateur de tension. Nous appellerons l'application d'un signal gestuel en position, un *générateur de position* (GP). Ce générateur de position peut être vu comme un module avec un point M similaire à un module MAT dont les positions sont fournies par un signal de geste et qui ne tient pas compte des forces qui lui sont retournées comme le montre la figure 65.

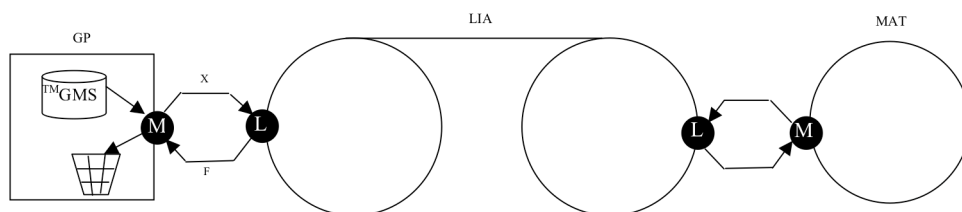


Figure 65 : Module générateur de position dans un réseau CORDIS-ANIMA.

Pour le réseau CORDIS-ANIMA connecté à un générateur de position, tout se passe comme si le mouvement de ce générateur avait été un module MAT dont le mouvement n'est pas influencé par ce réseau. Le mouvement de ce module MAT a donc pu être simulé dans une simulation passée, provenir d'un capteur de position ou encore avoir été construit de manière explicite à l'aide par exemple d'une fonction mathématique.

III. Bilan

Ce chapitre a posé la question de la communication avec d'autres outils pouvant intervenir dans la réalisation d'images animées afin que MIMESIS s'insère dans une chaîne globale de production. Il a mis en lumière que le lien entre les mouvements générés et les modèles de formes n'atteignait pas à ce jour un niveau suffisant de généricité pour que des modèles physiques CORDIS-ANIMA puissent participer de manière conséquente à la production d'images animées. Il nous faut remarquer ici que c'est tout de même le cas de l'ensemble des modèles physiques qui ne sont encore aujourd'hui qu'une part relativement peu importante de l'animation. Aujourd'hui encore l'animation se fait à la main ou à l'aide de mouvements captés dans le cas de l'animation de personnages. Nous prédisons qu'un nouveau souffle pour ces modèles n'existera que lorsque des outils logiciels permettront de prendre en compte de manière souple et accessible à l'utilisateur, la modélisation des liens entre les modèles physiques et la topologie des modèles de formes, mais quand aussi sera pratiqué de manière moins confidentielle le modèle physique masses - interactions. Ce n'est qu'en se confrontant à ce type de modélisation et en acceptant de n'habiller le mouvement qu'une fois le modèle simulé que ce besoin se fait sentir et que nous avons senti qu'il était suscité chez les utilisateurs de MIMESIS. Le chapitre suivant va nous permettre à ce titre de reporter les cas d'utilisation concrète de MIMESIS. Il faudra remarquer qu'encore aujourd'hui, les modèles se contentent d'un habillage sommaire, celui proposé par MIMESIS si ce n'est dans le cadre particulier de véritables créations destinées à être projetées en public comme l'ont pu l'être les travaux de Chimin Hsieh.

Mais la question de l'habillage n'est pas la seule question relative à la communication entre logiciels. Le format TMGMS que nous avons développé avec une équipe de l'ACROE-ICA, permet quand à lui de normaliser les échanges entre les différents outils qui traitent des signaux de geste et de mouvement. La possibilité d'exporter mais aussi d'importer ces signaux dans MIMESIS, permet une ouverture cette fois-ci sur la richesse des mouvements traités par MIMESIS. Nous en sommes encore aux premières utilisations d'un tel format mais les perspectives sont encourageantes car certains utilisateurs ont su l'inclure dans leurs modèles et créer des modèles successifs raffinant chacun le précédent. La notion de normalisation des formats mais aussi des structures de données capables d'accueillir ces formats est une nécessité pour permettre une véritable interopérabilité entre les systèmes. Nous espérons que ces travaux pourront contribuer à faire prendre conscience aux développeurs qu'il est de la première importance de dissocier les différentes phases et les différents formalismes en évitant le piège de modèles tout intégrés qui sont nécessairement ad-hoc. Si de telles normalisations sont aujourd'hui courantes dans le son ou encore dans l'image fixe, la vidéo n'arrive pas encore totalement à résoudre ces problèmes. Il est assez normal d'imaginer donc que dans le cas de la synthèse d'images animées qui est contrainte par des questions techniques encore plus complexes, nous n'en soyons qu'au début de la prise de conscience de telles nécessités.

- Chapitre 5 -

Réalisations et validations

I. Objectifs

Nous avons jusqu'ici montré l'intérêt de disposer d'un environnement de conception interactif pour les réseaux CORDIS-ANIMA et nous avons discuté des principaux verrous à la mise en œuvre d'un tel type d'environnement. Depuis le début de nos travaux de thèse, nous avons été amenés à prendre une part très importante dans le développement de l'environnement MIMESIS qui permet une telle conception. Cet environnement a été plusieurs fois utilisé dans des situations pédagogiques, de recherche et de création. Si nous tenons à dire quelques mots sur l'état actuel où sont arrivés les développements puisque, durant cette thèse, ces développements ont tout de même occupé une part importante de notre activité, si donc l'environnement qui existe actuellement est pour nous un résultat en soit, nous invitons le lecteur, s'il souhaite plus de détails, à se reporter au manuel d'utilisation de MIMESIS [LAE*06]. La validation de notre travail passe bien plus dans le compte-rendu et l'étude des utilisations qui en ont été faite. Notre travail n'a pas été un travail de modélisation mais d'implantation d'un environnement pour faciliter la modélisation. Nous tenons donc, en guise de validation d'un tel environnement, à rapporter les situations nouvelles que l'existence d'un tel environnement a pu susciter.

Dans un premier temps, nous avons implanté nous-mêmes une bibliothèque de modèles déjà existant et nous en avons conçu à cette occasion de nouveaux. Nous étions en quelque sorte alors notre premier sujet d'étude pour l'utilisation de MIMESIS. Cela nous a amené à modifier notamment le comportement de l'interface. Nous ne rendrons pas compte de ces questions qui relèvent de problématiques liées à la conception d'une interface. Beaucoup des questions soulevées sont résolues en effet dans le domaine des interfaces homme-machine et nous avons appliqué ici une analyse et un travail d'ingénieur appliquant ces concepts.

Pendant la durée de nos travaux de thèse, L'environnement MIMESIS est entré dans un cadre de recherche et de création. Dans ce cadre exigeant, nous sommes en mesure de fournir des éléments montrant ce que MIMESIS a pu apporter.

Disposant d'un logiciel stable dont l'utilisation a montré qu'il présentait un nombre de fonctionnalités déjà suffisant pour une activité de modélisation dont la complexité était au moins équivalente à la complexité des modèles des modèles CORDIS-ANIMA du passé, nous avons pu mettre MIMESIS dans les mains d'un panel d'utilisateurs d'origines très diverses, des plus jeunes comme des élèves de seconde, au plus âgés dans le cadre de formations continues, des scientifiques et des non scientifiques

ou en tout cas ni experts en sciences physiques, ni en informatique et ni en mathématiques. La bibliothèque de modèles qui n'a fait que grandir au cours de ces utilisations a toujours pris une part importante de l'activité pédagogique initiale.

Un chapitre de thèse ne suffirait pas à rendre compte précisément de la structure logicielle de MIMESIS, de l'ensemble des modèles réalisés et des situations d'utilisation. Nous nous proposons simplement de donner ici certains éléments significatifs pour attester de la validation de nos travaux.

II. La version actuelle de MIMESIS

Lors de nos travaux de thèse, nous avons participé à la conception et au développement des différentes parties de MIMESIS, environnement de conception de modèles CORDIS-ANIMA, interface de conception interactive de réseau CORDIS-ANIMA. Dans sa version actuelle, cet environnement permet à l'utilisateur de pratiquer la majeure partie des différentes fonctionnalités discutées dans ce manuscrit. Il est composé principalement de trois parties, le modelleur, l'interface graphique, le simulateur.

II.1. Le modelleur

Le modelleur est le principal programme de cet environnement. Il regroupe un noyau fonctionnel permettant de définir les composants du modèle, un langage compilé appelé langage MECA pour la conception de la topologie du réseau ainsi qu'un système de formes pour la visualisation des données de la simulation.

Le noyau fonctionnel propose de gérer la topologie d'un réseau CORDIS-ANIMA, les paramètres physiques et les valeurs initiales des différents modules. Trois familles de modules existent : les modules MAT, les modules LIA et les modules de contrôle paramétrique (MCP). Les modules sont désignés par un type qui permet d'identifier la fonction qu'ils occupent. La famille des modules LIA est la famille ayant le plus de types puisque MIMESIS propose des interactions aux fonctionnalités différentes (cf. chapitre 3): module d'interaction de cohésion (COH3 ou COH4), module d'interaction plastique (PLAST), module d'interaction de frottement sec (FROS), etc.

Le langage utilisé par la version actuelle de MIMESIS est le langage MECA. Ce langage permet de construire la topologie d'un réseau CORDIS-ANIMA. Il permet de déclarer un module ou un ensemble de modules dont tous ont un nom commun suivi d'un numéro d'index. Le langage permet de créer un réseau CORDIS-ANIMA en faisant usage de boucles ou de branchements conditionnels. En outre, il permet de labelliser de manière flexible les modules. Le langage MECA est un langage dont la syntaxe et la grammaire est relativement sommaire. Nous avons, en revanche, participé à la spécification d'un autre langage, le langage LANG basé sur le langage de script Python, qui a un rôle central dans le scriptage de l'ensemble des actions du modélisateur de la version multi-plateforme du logiciel GENESIS dédié à la synthèse de son à l'aide de réseaux CORDIS. Nous avons participé à la

conception de ce langage dans l'optique qu'il devienne un langage commun à GENESIS et aux versions futures de MIMESIS puisque ces deux environnement partagent des concepts communs autour de la modélisation de réseaux CORDIS-ANIMA même si chacun possède leurs spécificités. Il s'est donc agit de définir une partie commune qui n'interdise pas la possibilité d'une extension aux spécificités des deux environnements. Le langage LANG permet de produire les mêmes réseaux que le langage MECA tout en rajoutant certaines fonctionnalités (meilleure gestion de listes de modules, plus de flexibilité dans la labellisation), en plus de permettre des actions sur les paramètres physiques et les valeurs initiales.

MIMESIS possède un ensemble de formes permettant de visualiser le résultat de la simulation. Cet ensemble rudimentaire car comme il est expliqué dans la partie V du chapitre 2, ces formes ne servent qu'à un contrôle visuel des données de la simulation et ne constituent pas un système d'habillage générique pour la production de séquences finales. Les formes implantées appartiennent à l'une de ces classes :

- Primitives linéiques (ligne brisée, ligne brisée fermée, cercle, ...)
- Primitives surfaciques (polygones, ...)
- Primitives volumiques (sphère, cylindre, ellipsoïde, ...)
- Primitives mathématiques (Courbe de Bézier, B-spline, ...)
- Primitives topologiques (Diagramme de Voronoï, Triangulation de Delaunay, ...)

La figure suivante donne des illustrations de formes disponibles dans MIMESIS :

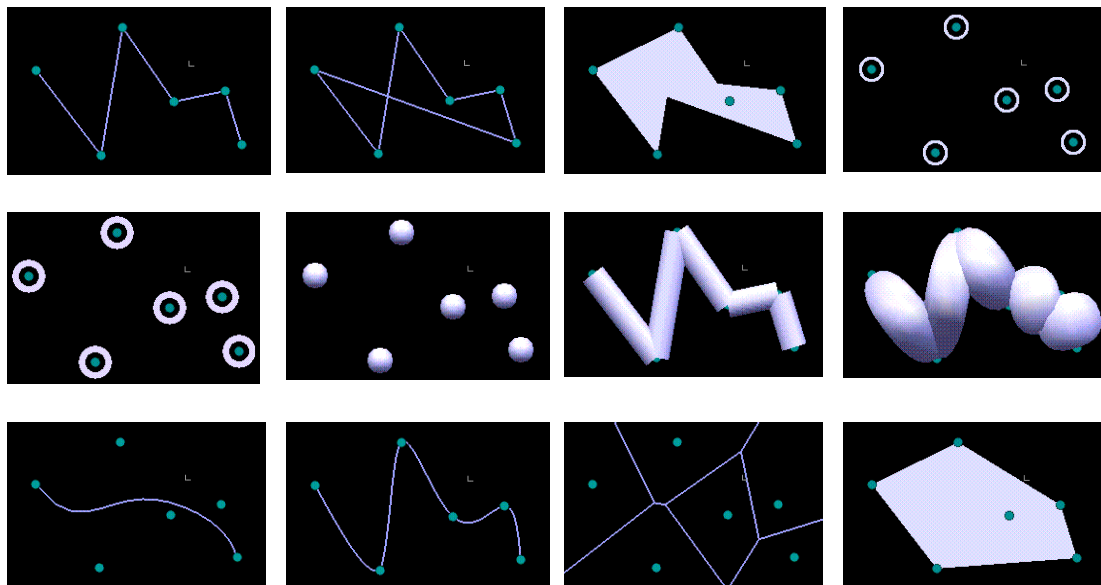


Figure 66 : Différents habillages dans MIMESIS pour une même configuration spatiale de points.
De gauche à droite et de haut en bas : ligne brisée ; ligne brisée fermée, polygones, cercles, couronnes, sphères, cylindres, ellipsoïdes, spline, courbe de Bézier, Diagramme de Voronoï, Triangulation de Delaunay.

II.2. L'interface graphique

L'interface graphique (cf. figure 67) permet d'interagir de manière interactive avec le modèle. Elle est organisée en plusieurs zones :

- La zone de texte pour l'édition de la topologie du réseau (sur la figure 67 à gauche). L'utilisateur saisie le texte du langage MECA qui lui permet de construire le réseau CORDIS-ANIMA en compilant ce texte.
- La fenêtre de modélisation 3D pour le travail interactif sur le modèle (sur la figure 67 au centre, en haut). Cette fenêtre est une fenêtre 3D interactive représentant le réseau CORDIS-ANIMA. Les positions des MAT dans cet espace sont les positions initiales de ces modules. Elle sert aussi de sélection de tout ou partie du réseau pour travailler sur différents paramètres et valeurs initiales des modules. Finalement elle sert aussi à désigner les MAT intervenant dans le contrôle de formes que l'utilisateur peut créer.
- La fenêtre de simulation et de projection 3D (au centre en bas sur la figure 67). Cette fenêtre propose une visualisation de la simulation ou d'une projection de cette simulation lorsqu'elle a été enregistrée. Les données de la simulation sont présentées en étant habillées par les formes définies par l'utilisateur.
- Les onglets (à droite sur la figure 67) permettent de travailler sur la sélection. Il y a 4 onglets : un onglet rendant compte de la structuration du réseau (modules et différents labels), un onglet permettant la saisie des paramètres physique, un onglet pour la saisie des valeurs initiales, un onglet pour l'édition des formes habillant les données de la simulation.
- Plusieurs fenêtres peuvent être ouvertes à l'aide d'un menu. Ces fenêtres contiennent des tableaux de paramètres, des outils de placements des MAT dans la fenêtre de modélisation 3D, des préférences, des outils d'importation et d'exportation de fichiers TMGMS, de fichiers de paramètres.

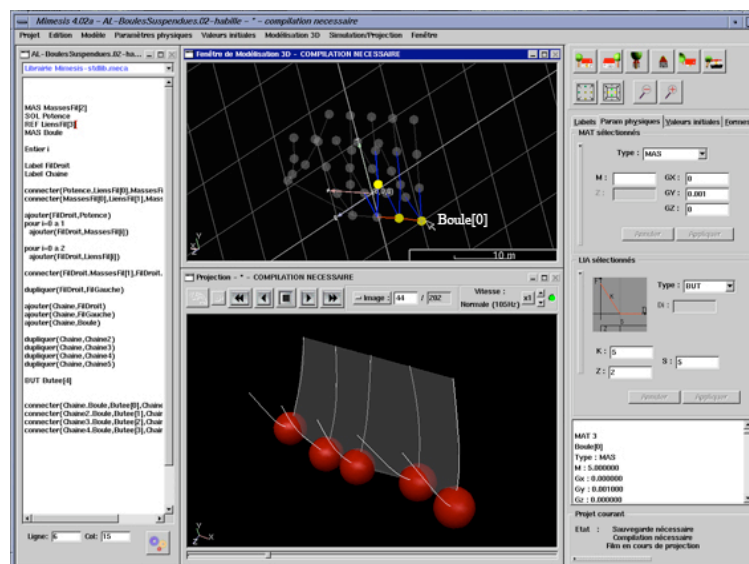


Figure 67 : Aperçu de l'environnement graphique actuel de MIMESIS.

II.3. Le simulateur

L'environnement MIMESIS actuel dispose d'un simulateur embarqué et multi-thread répondant aux analyses de la partie V du chapitre 2.

1.3.1. Les simulateurs dans MIMESIS

MIMESIS possède un noyau fonctionnel auquel peut-être lié deux simulateurs différents.

- Le simulateur temps-réel synchrone TELLURIS permettant une simulation multisensorielle faisant intervenir une interface gestuelle à retour d'effort.
- Le simulateur CORDIS-OFF, asynchrone et hors-ligne, dédié à la simulation hors ligne de mouvement 3D et de sons.

Dans le cas du lien avec CORDIS-OFF, il se peut que le calcul des échantillons de la simulation soit coûteux et/ou que le calcul du rendu de la simulation le soit. Ces différents cas ont été explicités précédemment. Pour parer à toutes ces éventualités, l'environnement MIMESIS propose 4 modes :

- Priorité à la simulation. Le simulateur simule au plus vite sans tenir compte de la phase de visualisation. Des échantillons pourront ne pas être visualisés.
- Affichage de toutes les images. Le simulateur est alors asservi au temps de calcul de la méthode de visualisation.
- Temps synchrone. On peut faire l'hypothèse même lors de l'utilisation de CORDIS-OFF, que le temps réel est atteint dans certains cas. Dans ces cas, un appel synchrone au calcul des échantillons peut permettre un rendu graphique synchrone pendant la simulation et ce même si certains échantillons ne sont pas visualisés si la méthode de visualisation est trop lourde. Mais si l'utilisateur se place dans le cas d'utilisation de CORDIS-OFF, rien ne lui garanti que le temps réel soit assuré et que le temps de calcul des échantillons déborde sur le temps qui lui est imparti.
- Mode production. Rendu graphique hors ligne pour visualisation dans un logiciel de lecture de films.

Les modes 2, 3 et 4 sont des modes valables pour une phase de « projection » où les échantillons de mouvement ne sont plus donnés par le simulateur mais par un fichier dans lesquels ils ont été sauvegardés. Le pendant au mode 1 pour la projection, la priorité à la lecture, a en effet peu de sens puisque dans le cas de la projection l'utilisateur souhaite observer le mouvement.

1.3.2. Sauvegarde sur disque des échantillons produits par la simulation

Le fait qu'il existe une fréquence de simulation et une fréquence visuelle implique que les échantillons simulés ne sont pas nécessairement les échantillons affichés. On affiche généralement moins d'échantillons que ce qui est produit : par exemple, 1 échantillon sur 42 lorsque la fréquence de simulation est de 1050 Hz et la fréquence de représentation visuelle de 25 Hz.

Or, lorsqu'on souhaite sauvegarder sur disque les échantillons produits par la simulation, la question de quels échantillons sauvegarder se pose. On peut faire le choix de sauvegarder uniquement les échantillons utiles à la visualisation, dans le but de rejouer la simulation hors-ligne et appliquer un procédé de visualisation peu coûteux en temps de calcul pour voir le mouvement qu'une simulation lourde empêchait de voir. Dans ce cas, on a perdu les échantillons non visualisés que l'on considère sans intérêt. Il peut être intéressant de conserver tous les échantillons produits par la simulation par exemple dans le cas par exemple où le modèle produit du son en plus du mouvement.

D'un point de vue pratique, l'écriture sur disque des seuls échantillons utiles à la visualisation n'est pas bloquante (écriture de plusieurs milliers de positions à 25Hz). Cela devient moins vrai lorsqu'il faut sauver tous les échantillons produits par la simulation et donc sauver un échantillon à la même vitesse qu'il a été produit (écriture de plusieurs milliers de positions à 1050Hz ou 44100 Hz). Dans ce cas, l'écriture peut ralentir grandement la simulation.

1.3.3. Architecture logicielle des modes de simulation/projection

Du simulateur à l'affichage graphique de la simulation et la sauvegarde du signal produit, nous avons mis en place 4 processus communiquant entre eux :

- Le processus « simulateur » produit les échantillons de la simulation. Il envoie au processus « consommateur » les échantillons utiles.
- Le processus « consommateur », communique avec le simulateur. Il les récupère les échantillons pour les envoyer au processus de visualisation de la simulation et au processus d'écriture sur disque.
- Le processus « écriture sur disque » récupère les échantillons envoyés par le processus « consommateur » et en réalise la sauvegarde sur disque.
- Le processus « affichage » récupère les échantillons envoyés par le processus « consommateur » et affiche le résultat dans une fenêtre 3D en y appliquant un rendu. Il traite le dernier échantillon qu'il a reçu.

Ces quatre processus ont des temps d'exécution différents qui dépendent de la tâche qu'ils ont à accomplir. Il arrive, par exemple, le plus souvent que le temps de sauvegarde sur disque soit plus rapide que le temps de visualisation d'un échantillon. On peut alors souhaiter écrire sur le disque tous les échantillons alors qu'on en saute la visualisation de certains (dans le cas des modes priorité simulation ou temps synchrone). On met alors en places des FIFO de stockage des échantillons entre

les différents processus communiquant lorsqu'on ne souhaite pas perdre d'échantillons mais que leur traitement peut être différé. Ainsi la structure logicielle de MIMESIS définit une FIFO entre le processus « simulateur » et le processus « consommateur » ainsi qu'entre le processus « consommateur » et le processus « écriture sur disque » (voir figure 68).

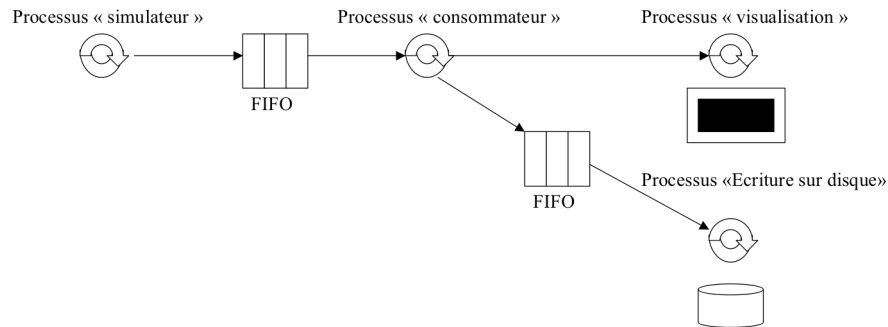


Figure 68 : Architecture globale de la chaîne simulation-affichage-sauvegarde dans MIMESIS.

Les différents modes de simulation sont alors obtenus de la manière suivante :

- Mode « priorité simulation » : Le processus « simulateur » envoie les échantillons utiles au processus « consommateur ». Celui-ci envoie les échantillons qu'il reçoit au processus « Écriture sur disque » et au processus « visualisation ». Comme il n'y a pas de FIFO pour stocker les échantillons entre le processus « consommateur » et le processus « visualisation », il se peut que des échantillons soient perdus par le processus « visualisation » qui peut effectuer un rendu sur un échantillon précédent pendant qu'il en reçoit plusieurs nouveaux qu'il ne traite pas.
- Le processus « consommateur » et « écriture sur disque » peuvent être bloquant pour le simulateur si l'une des FIFO est pleine. Cela n'arrive que lorsque les tâches effectuées par les processus amont sont plus lentes que le calcul d'un échantillon par le simulateur. Mais en pratique la recopie d'un échantillon, opération effectuée par le processus « consommateur » est moins longue que son calcul.
- Mode « Affichage de toutes les images ». Le processus de « visualisation » ne veut pas perdre d'échantillon et tous les afficher au fur et à mesure. Il doit alors bloquer le processus de « consommation » qui ne pourra redémarrer sa tâche qu'à l'envoi d'un signal lorsque l'échantillon en cours de rendu est affiché.
- Mode « Temps synchrone » : le processus « consommateur » reçoit des échantillons provenant du simulateur, mais sa consommation en sortie de la FIFO est synchronisée sur l'horloge interne à la fréquence de représentation visuelle de la simulation. Ainsi deux cas se présentent. Soit le simulateur simule les échantillons utiles à fréquence supérieure à la fréquence de représentation visuelle. Dans ce cas, le processus de simulation est bloqué lorsque la FIFO est pleine. Soit le simulateur simule ces échantillons à une fréquence inférieure et dans ce cas, la synchronisation à la fréquence de représentation visuelle n'est pas possible et la simulation se fait en temps différé.
- Mode « Production » : ce mode est similaire au mode d'affichage de toutes les images mais le processus d'affichage se transforme en un processus d'écriture sur disque des images normalement affichées.

Lorsqu'on se place dans une configuration de « projection » de la simulation, le processus de simulation est remplacé par un processus de lecture des échantillons sur disque.

III. La bibliothèque de modèles

III.1. Pourquoi une bibliothèque ?

Un environnement de conception propose en règle générale de sauver un modèle dans un fichier ou un ensemble de fichiers pour pouvoir revenir dessus plus tard, pour l'échanger avec d'autres utilisateurs. Jusqu'ici les différents modèles conçus au sein du groupe ACROE-ICA l'ont été dans des cadres plus ou moins standardisés en utilisant différentes versions de CORDIS-ANIMA. En outre, la description de la topologie de ces modèles ainsi que les valeurs des paramètres et des variables initiales étaient contenues dans des fichiers de programme mêlés aux diverses librairies CORDIS-ANIMA. En statuant sur une version estampillée à une date donnée comme étant la version officielle de CORDIS-ANIMA, et en réimplantant ces modèles dans l'environnement MIMESIS, il a été donc possible de résoudre ces deux inconvénients. D'une part, tous les modèles (ou une grande partie et au moins un modèle pour chaque type de comportement) se retrouvent au sein d'une même bibliothèque qui est basée sur une unique version de CORDIS-ANIMA. D'autre part, la description de ces modèles est contenue dans un ensemble de fichiers texte normalisés : un fichier pour la description de la topologie, un pour la description des valeurs des paramètres, un pour la description des valeurs des variables initiales... La mise à disposition d'une telle bibliothèque permet alors de faciliter la transmission de ces modèles entre utilisateurs.

De plus, une telle bibliothèque a été une base fondamentale de la pédagogie menant à l'apprentissage de ce type de modélisation. Nous reviendrons sur ce point dans la partie IV de ce chapitre.

Pendant nos travaux de thèse, nous avons augmenté ensuite cette bibliothèque afin de valider par des modèles élémentaires les nouvelles fonctionnalités implantées dans MIMESIS. Nous rendrons compte de certains de ces modèles dans les paragraphes III.3.

De nouveaux modèles, réalisés par d'autres utilisateurs de MIMESIS que nous-mêmes ont eux aussi, complété cette bibliothèque. Nous rendrons compte des travaux les plus aboutis dans la partie III.4 dédiées à l'utilisation de MIMESIS en situation de recherche et de création. Nous concluons la présentation des utilisations de MIMESIS dans la partie IV qui traitera des ateliers pédagogiques pour la modélisation physique à l'aide de MIMESIS.

III.2. Réimplantation des modèles CORDIS-ANIMA du passé

Ce qui suit donne un aperçu de la diversité des comportements dynamiques qui ont pu être obtenus par le passé. Nous n'avons pas réimplanté une copie conforme de ces modèles mais nous avons plutôt proposé un panel de modèles du même type capables d'exhiber par l'exemple les caractéristiques topologiques et paramétriques. Plutôt que de suivre une progression chronologique des travaux, nous proposons ici une revue de familles de modèles suivant un axe allant des modèles aux topologies les plus structurés vers les topologies les moins structurées.

Solides rigides

Ces modèles sont généralement très structurés. Ils font intervenir un faible nombre de masses et d'interactions. Ils se réfèrent notamment aux papiers de Jimenez et Luciani [JL93] traitant d'objets en collision ou encore de Luciani et Godard [LGo97] s'intéressant plus particulièrement à diverses configurations possibles de la topologie des réseaux CORDIS-ANIMA permettant d'obtenir des solides rigides.

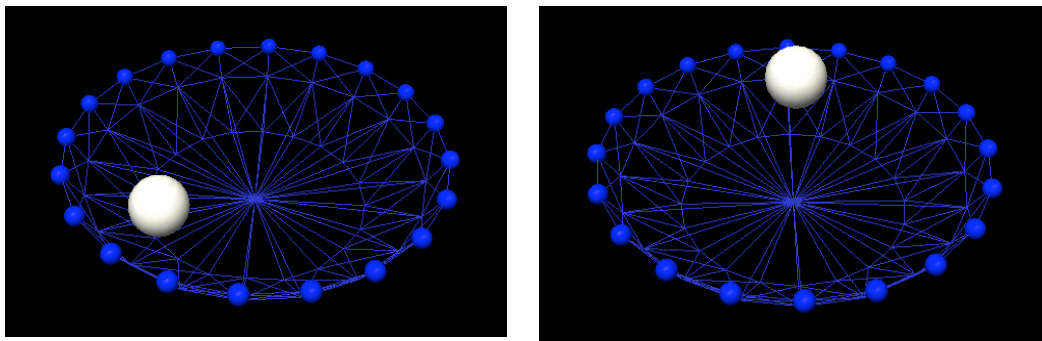


Figure 69 : La roulette et la boule, solides en mouvement.

Systèmes articulés

Les systèmes articulés font eux aussi intervenir un faible nombre de masses et d'interactions. Mais la configuration des interactions permet de laisser au modèle plusieurs degrés de liberté. On pourra se référer aux travaux de Nouiri et al. [NCL94] modélisant des systèmes mécaniques articulés ou encore aux marionnettes de Luciani [Luc93]. L'exemple le plus simple de système articulé est la chaîne composée de n masses disposées en ligne et reliées entre voisines par $n-1$ interactions viscoélastiques.

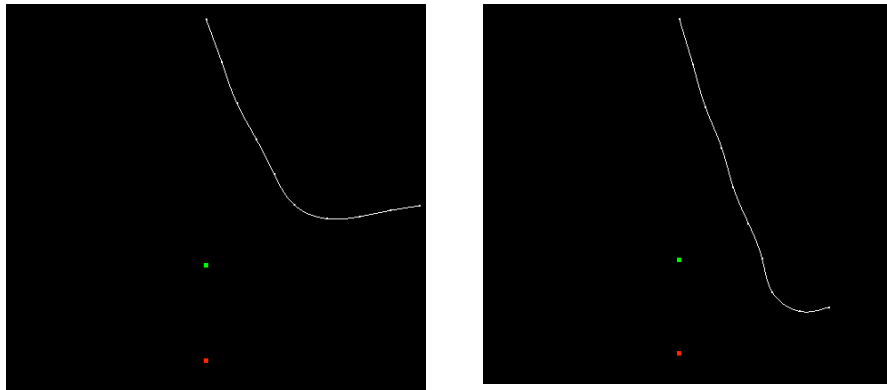


Figure 70 : *La chaîne.*

Solides déformables

Les solides déformables sont eux aussi généralement modélisés par des réseaux très structurés. En revanche les valeurs des paramètres de raideur des interactions viscoélastiques les composant sont inférieures aux valeurs des paramètres de raideur des solides rigides. Des travaux pionniers de modélisation de ce type de solides déformables peuvent être retrouvés dans la bibliographie du groupe ICA-ACROE [LC84, Jim89, drapeau]. La figure suivante illustre un exemple de solide déformable de la bibliothèque de MIMESIS. Il s'agit d'un modèle de tissu flottant.

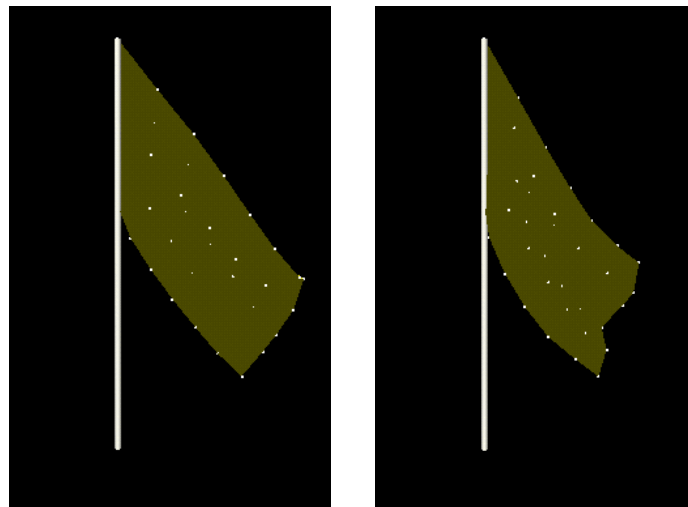


Figure 71 : *Le drapeau.*

Fractures

Les fractures d'objets solides rigides ou déformables peuvent être obtenues à l'aide d'interactions non-linéaires irréversibles comme les interactions de cohésion irréversibles. Il s'agit de changements brutaux des interactions entre les masses ponctuelles qui passent d'une zone de cohésion à une zone libre (cf. chapitre 3). On retrouve de telles fractures notamment dans les travaux de Luciani et Godard [LG97]. Un exemple de modèle de fracture sera présenté dans le paragraphe III.3.2.

Les comportements collectifs

Entre le solide et le fluide existe un ensemble de comportements intermédiaires de la matière. Ce sont les gels, les pâtes [Jun97, GLC03], mais aussi les comportements collectifs que l'on peut retrouver dans les tas des sables [HML94] ou encore les fumées [LVH*95]. On observe de tels comportements collectifs dans les mouvements des foules [LTE06] alors qu'étonnement les personnes la composant ne sont pas des corps inertes mais doués d'une capacité motrice propre soumise à une décision. Ces comportements peu structurés nécessitent généralement que les masses ponctuelles composant le modèle soient toutes en interaction soit pour n masses, $n*(n-1)/2$ interactions. Les figures suivantes présentent des modèles

La figure 72 reproduit des exemples de modèles de comportements intermédiaires de la matière que nous avons obtenus avec MIMESIS : la formation d'un tas de sable et la formation des volutes d'une fumée. La figure 73 reproduit quand à elle un modèle simple de comportement collectif de deux groupes qui se croisent dans un couloir étroit.

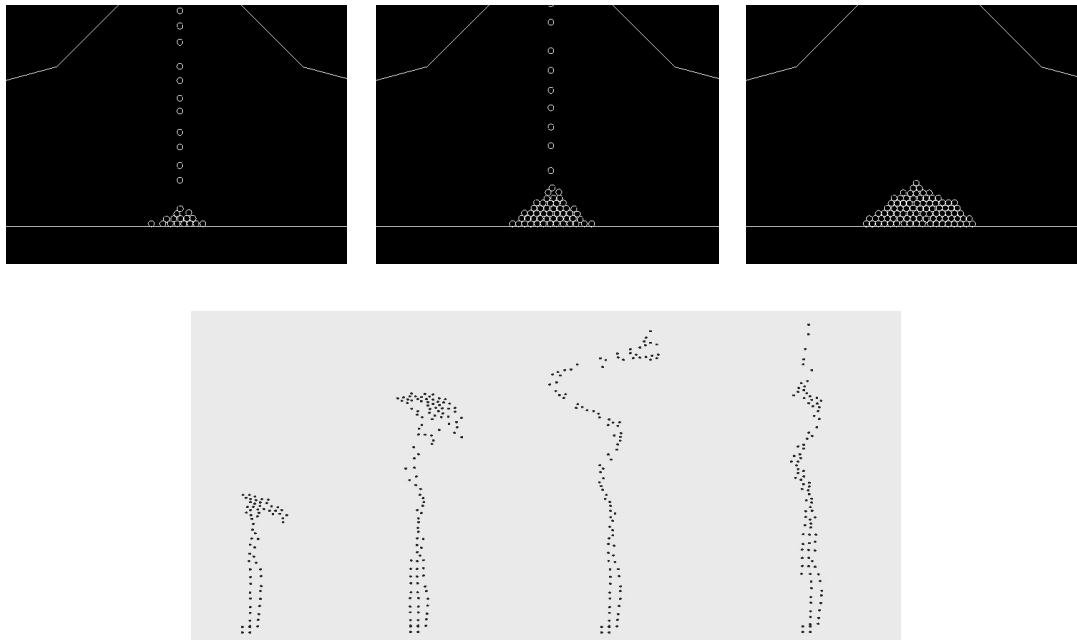


Figure 72 : *Comportements intermédiaires de la matière.*

En haut : formation d'un tas de sable. En bas : formation des volutes d'une fumée

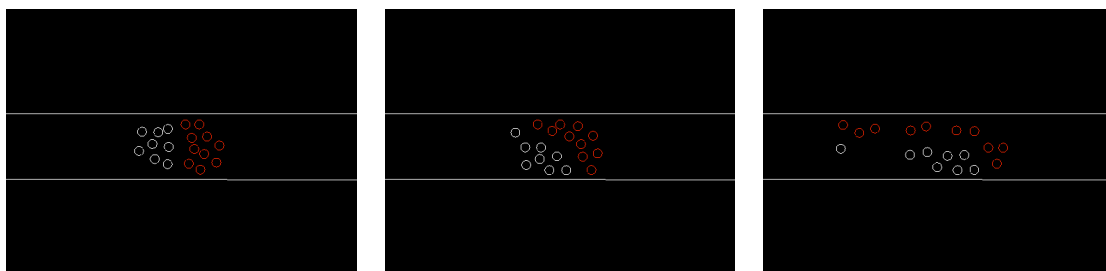


Figure 73 : *Croisement entre deux groupes d'individus dans un couloir étroit.*

III.3. Nouveaux modèles validant l'introduction d'interactions non linéaires dans MIMESIS

Nous avons introduit les modules d'interaction non linéaires précédemment présentés au sein de l'environnement MIMESIS. Ces modules, mis à disposition de l'utilisateur doivent être testés pour que leur utilisation soit validée. D'autant plus que nous avons modifié certains algorithmes comme celui du frottement sec dont nous critiquions le fonctionnement.

Certains des modèles que nous allons présenter dans cette partie sont des modèles que nous avons nous-mêmes mis au point. D'autres sont des modèles qui existaient déjà dans la littérature et que nous avons réimplantés. Le dernier modèle de cette partie a été développé par Ali Allaoui durant son stage de Master 2 que nous avons co-encadré avec Annie Luciani. Certains sont des modèles minimaux avec peu de modules, d'autres sont des modèles à plus grand nombre de masses. Nous tentons donc dans cette partie de proposer une variété de modèles (de modèles élémentaires à des modèles plus complexes), le juste nombre que nous avons choisi car ils nous semblaient balayer un spectre assez large de comportements pour valider notre travail d'implantation d'interaction non linéaires dans un environnement de conception interactive.

Dans cette partie, chaque modèle va être décrit. Lorsque les paramètres seront donnés, ils le seront sans unités. Il faut alors considéré que le système d'unité dans lequel ils sont exprimés est celui de MIMESIS. Toutes les simulations ont été effectuées à la fréquence de 1050 Hz. Elles sont affichées à 25 Hz, c'est-à-dire qu'un échantillon sur 42 est affiché. Pour chaque séquence d'images, nous donnons les numéros des images présentées.

III.3.1. Libération d'une situation de cohésion

Ce modèle très simple est composé de 3 masses ponctuelles m_1 , m_2 et s . La masse ponctuelle m_1 est liée à la masse ponctuelle fixe s par une interaction de cohésion de type Lennard-Jones (figure 74, à gauche). La masse ponctuelle m_2 est, elle, liée à la masse ponctuelle m_1 par une interaction de butée (figure 74, à droite). Initialement la masse ponctuelle m_1 se trouve dans un état cohésif d'équilibre avec la masse ponctuelle fixe s , c'est-à-dire à une distance de s égale à $L=10$. La masse ponctuelle m_2 est lancée avec une vitesse initiale vers m_1 pour rentrer avec elle en interaction de butée. Ces deux masses ponctuelles ont même la même masse $M=1$ et la butée est non visqueuse, de raideur $K=0.1$ et de seuil $S=1$. Ainsi, à l'image du carreau dans le jeu de pétanque, m_2 doit venir buter vers m_1 , lui transmettre toute son énergie cinétique et la remplacer. Pour que m_1 puisse alors se libérer de l'interaction de cohésion, c'est-à-dire dépasser le seuil $S=20$, il faut que l'énergie qui lui est communiquée par m_2 soit suffisante. Pour cela, il faut que la vitesse initiale de m_2 dépasse un certain seuil.

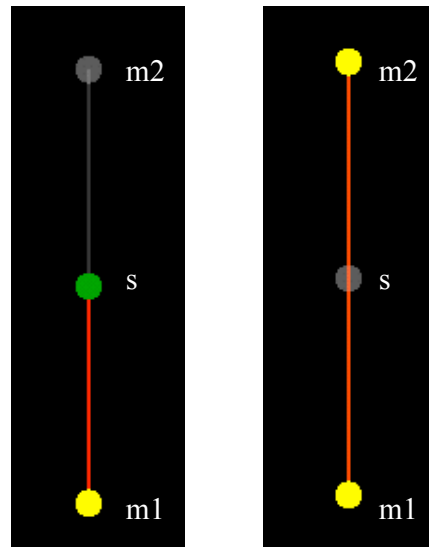


Figure 74 : Le modèle « Libération ».

À gauche, la masse $m1$ est liée à la masse fixe s par une liaison de cohésion. À droite, la masse $m2$ est liée à la masse $m1$ par une interaction de butée.

Deux séquences différentes suivent. Nous visualisons $m1$ à l'aide d'un cercle et $m2$ à l'aide d'un disque. Un point carré indique la limite de cohésion à dépasser pour que $m1$ se libère de l'interaction de cohésion avec s . La première séquence a été réalisée en donnant à $m2$ une vitesse initiale de 47,16 selon l'axe descendant. Cette vitesse n'est pas suffisante pour que $m1$ se libère. On remarque que $m1$ est juste à la limite de se libérer. Ainsi, avec une vitesse juste supérieure (47,17), dans la seconde séquence $m1$ parvient à se libérer.

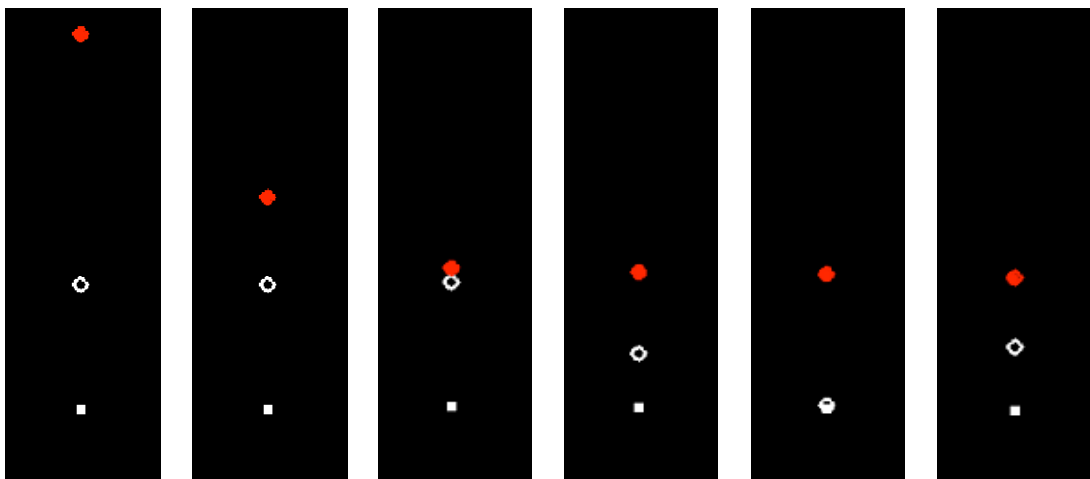


Figure 75 : Interaction de cohésion sans libération.

Non libération de $m1$. Vitesse initiale de $m2$ égale à 47,16. Images 1, 30, 43, 58, 108 et 165.

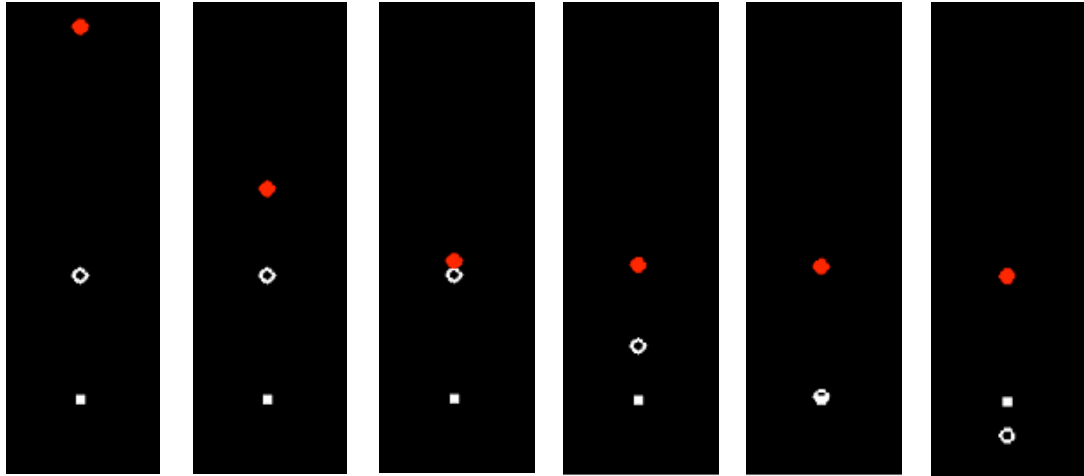


Figure 76 : Interaction de cohésion avec libération.

Libération de m1. Vitesse initiale de m2 égale à 47,17. Images 1, 30, 43, 58, 108 et 165.

III.3.2. Fracture

Le modèle suivant est la fracture d'un mur de double épaisseur de masses ponctuelles. Chaque masse est reliée à ses voisines par une interaction cohésive irréversible permettant de rendre les phénomènes de fracture. Avant toute chose, il convient d'organiser les masses ponctuelles pour qu'elles remplissent un espace ressemblant à un pavé. Nous pourrions choisir de placer les masses par une méthode géométrique, mais la simulation physique permet une autre solution. Nous laissons chuter 100 masses reliées entre elles par des interactions de butées. Toutes les masses ponctuelles sont en interaction de butée avec des masses fixes. Ces interactions permettent de restreindre le mouvement des masses ponctuelles à un espace pavé de manière à ce qu'elles s'organisent et remplissent dans leur chute cet espace. Tout se passe comme si on coulait un mur dans une structure de tasseaux. Les seuils des interactions de butée sont choisis de telle sorte que les masses s'organisent de manière assez homogène. Lorsque la simulation est à l'équilibre, nous obtenons la structure topologique illustrée dans la figure 77 :

Nous partons de la structure obtenue pour réaliser notre modèle en reliant chaque masse ponctuelle à ses voisines proches par des interactions cohésives irréversibles (voir figure 78). Nous donnons aux seuils L_1 de cohésion la valeur de la distance initiale entre les masses ponctuelles qu'elles relient. De cette manière, notre « mur » se trouve dans un état de cohésion à l'équilibre au début de la simulation. Les paramètres des interactions cohésives irréversibles sont $K_1=0.3$, $Z_1=Z_2=Z_3=0.12$, $K_2=K_3=0.001$, $L_3=L_2+0.2=L_1+0.4$. Ainsi, les seuils des interactions dépendent tous du paramètre L_1 et donc de la distance initiale entre les masses.

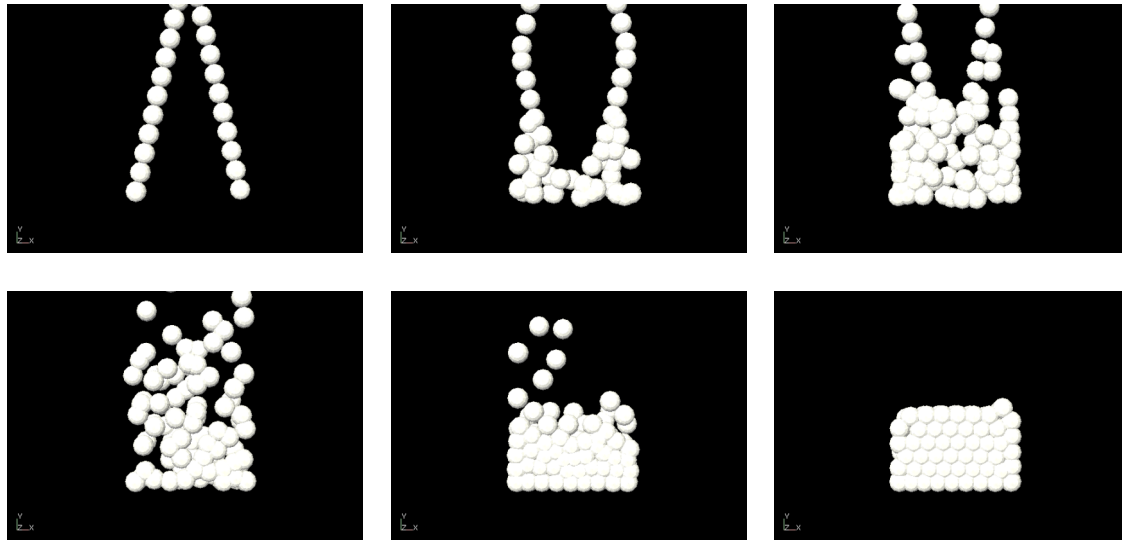


Figure 77 : Auto-organisation de 100 masses dans un espace pavé.

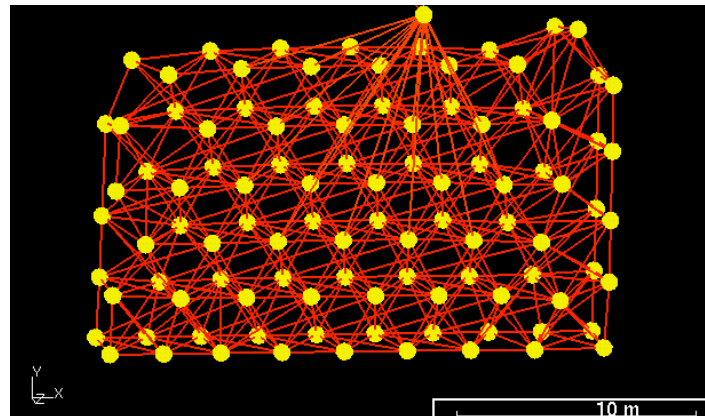


Figure 78 : Topologie du réseau masses - interactions du modèle boule contre mur.

Nous modélisons une boule suspendue qui venir détruire le mur à l'aide d'une masse ponctuelle « Boule » accrochée à une masse fixe S par une interaction viscoélastique et soumise à une force de gravité. La masse ponctuelle Boule est en interaction de butée avec certaines masses du mur, choisies de manière à ce qu'elles soient dans la trajectoire de Boule à son arrivé près du mur. Le choix dépend donc du seuil d'interaction de la butée. Plus ce seuil est petit, moins il y aura de masses ponctuelles concernées. Il n'est en effet pas nécessaire de créer des interactions de butée entre toutes les masses ponctuelles du mur et Boule si Boule n'entre jamais en collision avec elles. Ces interactions de butée ont pour paramètre $K=0.1$ et $Z=0.001$.

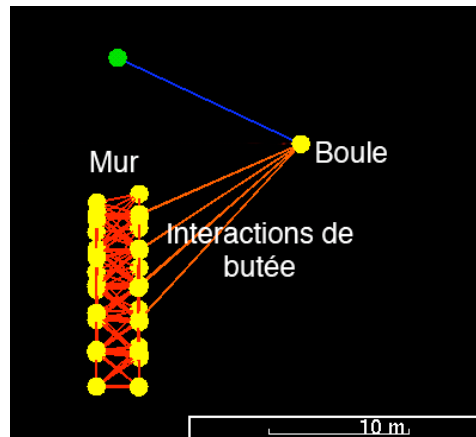


Figure 79 : Autre représentation de la topologie du réseau masses - interactions du modèle boule contre mur.

La figure suivante présente le résultat obtenu par la simulation. La boule vient heurter le mur qui se déforme dans un premier temps et se fracture en de multiples endroits. La boule emporte avec elle toute une partie du mur et laisse debout les parties éloignées de son passage

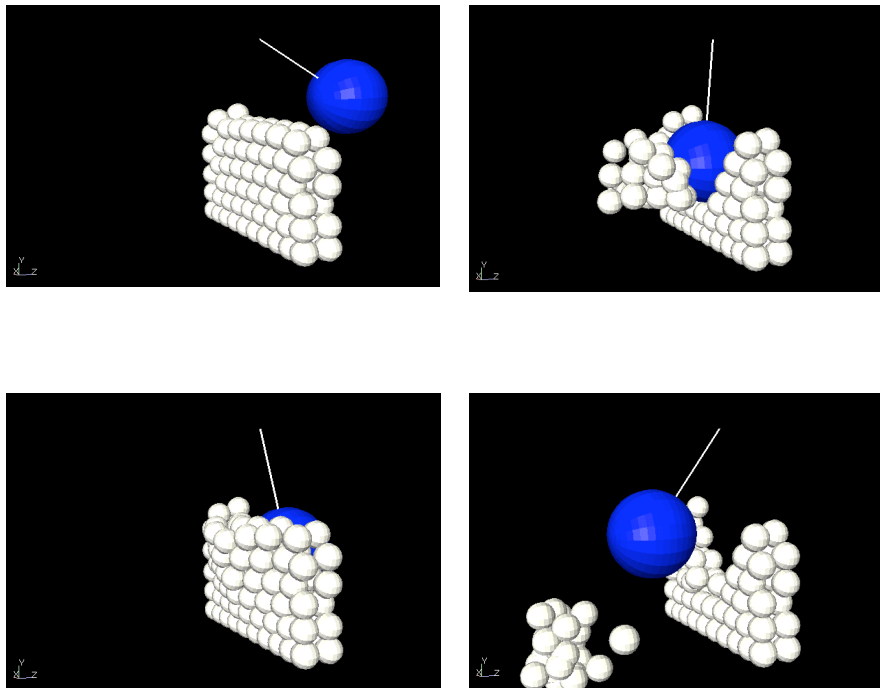


Figure 80 : Simulation d'une fracture par le passage d'une masse ponctuelle en interaction avec de masses ponctuelles formant un mur cohésif capable de se fracturer.

Images 1, 19, 23, 32.

III.3.3. Mouvements d'anticipation

Ce modèle fait suite à la publication de Paris et al. [PPD07]. L'article propose notamment une capture de mouvement faisant intervenir deux personnes suivant deux chemins orthogonaux qui se croisent. Les auteurs mettent en évidence un phénomène d'anticipation mis en jeu par les deux sujets pour éviter d'entre en collision au moment où il se croise. Ils affirment qu'un tel phénomène d'anticipation n'est pas modélisable à l'aide d'un modèle physique. Ils considèrent notamment que les modèles de type masses - interactions sont mal adaptés pour rendre ce type de phénomène. Ce modèle tente de leur donner tort.

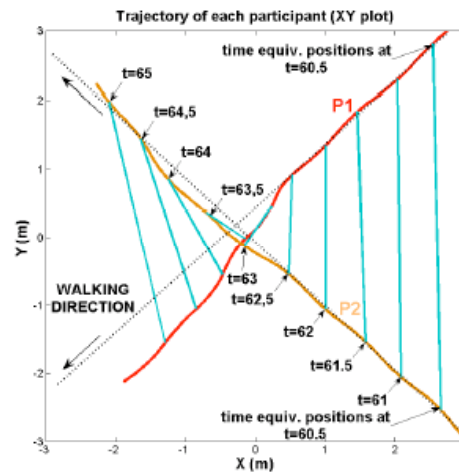


Figure 81 : Anticipation lors d'un croisement de deux sujets.

Figure extraite de [PPD07]

Nous modélisons chacun des deux sujets à l'aide de 4 masses. Une masse ponctuelle représente le corps, deux sont utilisées pour les épaules, et une dernière est placée à l'arrière du personnage. Cette dernière masse ponctuelle est une masse ponctuelle visqueuse dont l'algorithme comporte un terme de viscosité ambiant. L'adjonction de ce modèle empêche le personnage d'opérer des rotations trop brusques comme la tête d'un volant de badminton. Les 4 masses ponctuelles sont reliées entre elles par des interactions viscoélastiques permettant leur cohésion (voir figure 82).

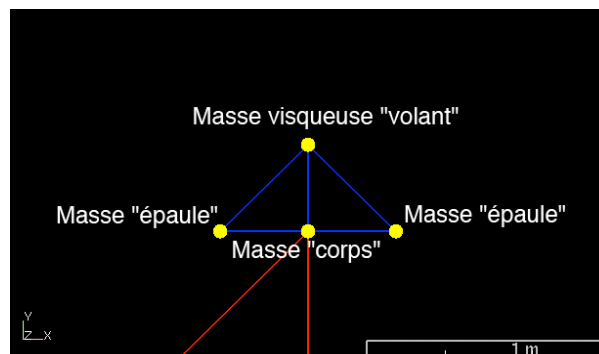


Figure 82 : Modèle du personnage avec masse ponctuelle visqueuse pour éviter les rotations brusques.

Les deux personnages sont en interaction de répulsion à trois paliers (voir figure 83). Le fait d'utiliser une interaction linéaire par morceaux à plusieurs segments permet de modaliser l'interaction en fonction de la distance. A longue distance, nous optons pour une interaction de faible raideur et de faible viscosité ($K=10^{-5}$, $Z=0.001$). Cela modélise la préparation à l'évitement à une distance plus importante que le celle de la potentielle collision. À courte distance, l'interaction est de plus forte raideur et de plus forte viscosité ($K=0.001$, $Z=0.0035$). Les deux personnages doivent en effet éviter à tout prix la collision qu'ils se sont tout de même préparés à éviter avant.

On remarque que la trajectoire du personnage 2 est beaucoup plus rectiligne que la trajectoire du personnage 1. En réalité, la vidéo correspondant à l'article et visible à l'adresse [Par] semble laisser transparaître que le personnage 2 est beaucoup plus décidé à atteindre sa cible que le personnage 1. Nous modélisons ce fait en donnant une masse plus importante au modèle du personnage 2 ($M=1$) par rapport à celle du modèle du personnage 1 ($M=0.7$).

Au final chacun des personnages est soumis à une force constante dans la direction de leur but qui est modélisé par une masse fixe.

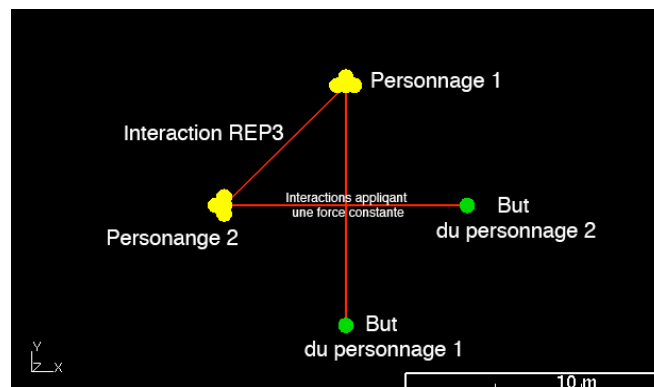


Figure 83 : Modélisation de la situation de croisement avec anticipation.

Le résultat de la simulation est illustré par la figure 84. On repère les deux personnages par un habillage élémentaire montrant la tête et les épaules. Deux lignes droites symbolisent le plus court chemin au but s'il n'y avait eu à anticiper une collision au moment du croisement. La trajectoire du personnage 2 suit bien le profil de trajectoire largement dévié de la droite et montré par la figure 84, alors que le personnage 1 semble bien suivre un chemin plus droit.

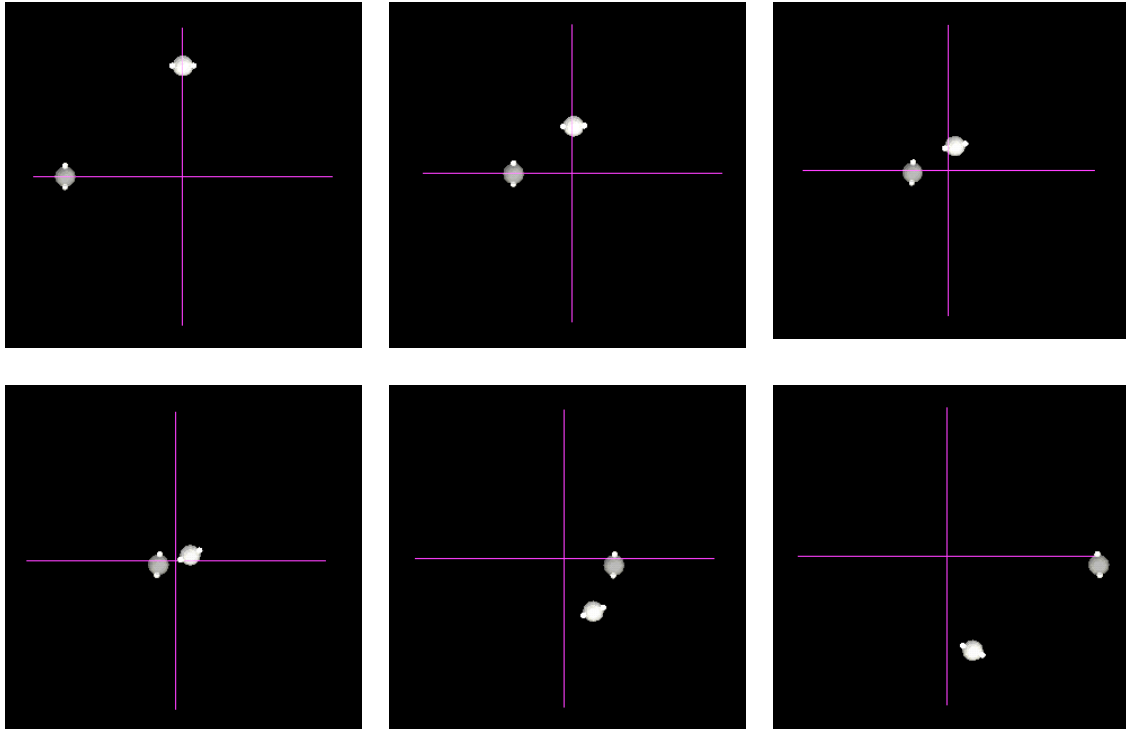


Figure 84 : Simulation du phénomène d'anticipation lors du croisement de deux sujets.
 Images 195, 306, 364, 402, 520 et 660.

III.3.4. Plasticité

Ce modèle est une réimplantation à l'aide de MIMESIS du modèle de sol meuble développé par Chanclou dans [Cha96]. Nous n'entrerons donc pas dans le détail du modèle. Le but est d'obtenir un sol gardant la mémoire du passage d'un véhicule se déplaçant sur roues. De plus, une trace faite par une roue peut être renforcée par le passage d'une roue suivante. Il s'agit précisément d'un phénomène de plasticité qui peut être rendu grâce au module de plasticité parfaite proposé à l'utilisateur de MIMESIS.

Le modèle de sol est composé d'un tapis de masses ponctuelles dont chacune est reliée à une masse ponctuelle fixe différente par une interaction plastique verticale. Les masses ponctuelles fixes sont donc le fond incompressible du sol meuble. De plus, pour éviter les phénomènes de cisaillement dus au passage de véhicule, chaque masse composant le sol est retenue dans le plan horizontal par une autre interaction plastique avec une masse ponctuelle fixe (voir figure 85 en bas à droite).

Le modèle de véhicule est un modèle très simple à 3 roues modélisées par 3 masses ponctuelles (voir figure 85). Chaque roue est en interaction de butée avec les masses composant le sol (voir figure 85 en bas à gauche). Le modèle proposé ici est un modèle en 2D suffisant pour rendre compte de l'action irréversible du passage successif des roues sur le sol meuble.

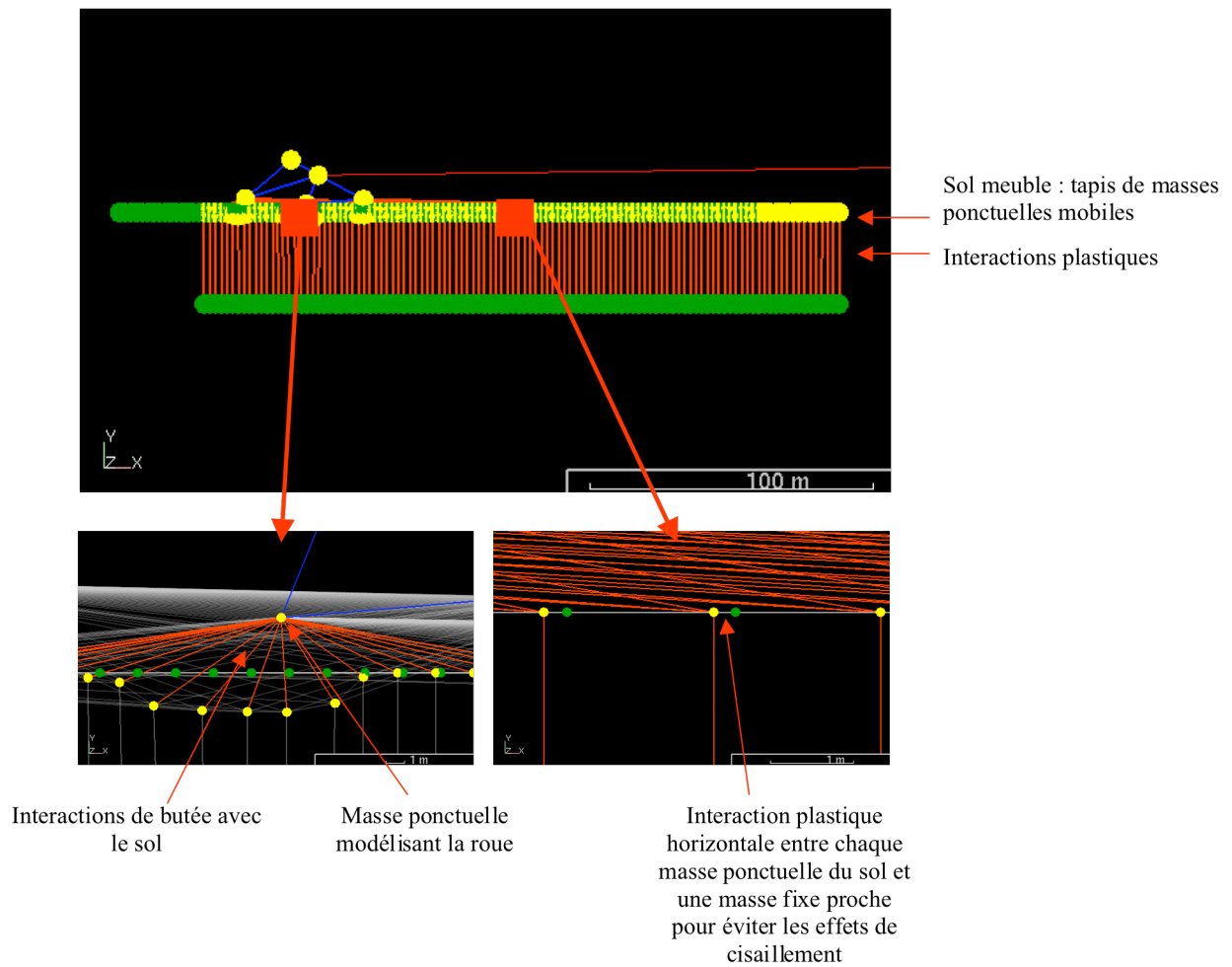


Figure 85 : Modèle 2D de véhicule sur sol meuble.

En haut : le modèle global. En bas à gauche : gros plan sur l'interaction entre une roue et le sol. En bas à droite : gros plan sur l'interaction plastique prévenant les effets de cisaillement.

La figure suivante donne le résultat de la simulation. La trace de la déformation du terrain est laissée en fond avec une couleur moins intense pour permettre de mettre en évidence la formation de traces successives.

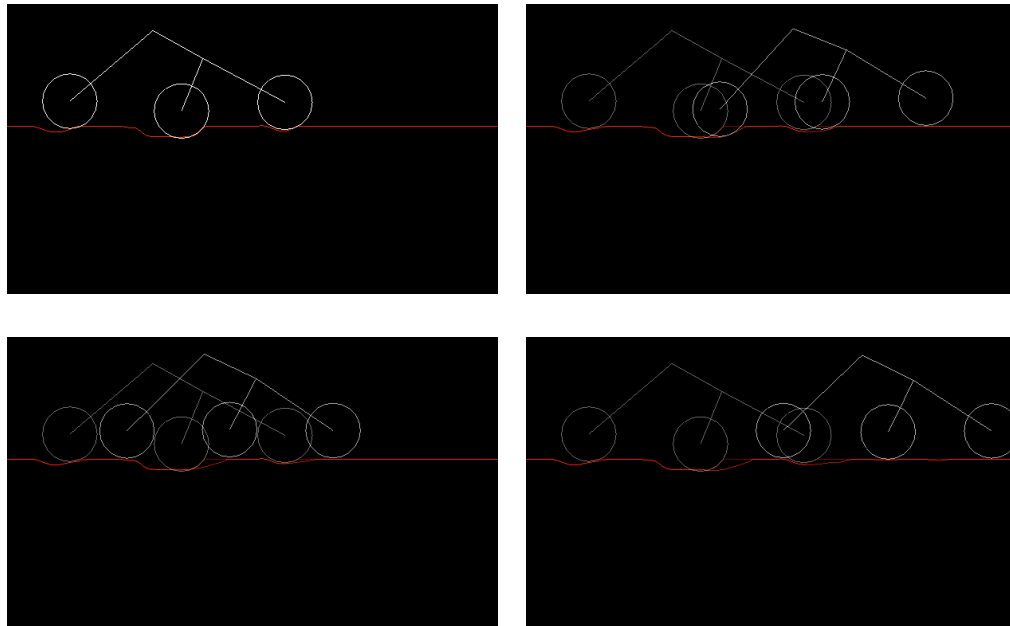


Figure 86 : Simulation d'un véhicule sur sol meuble.

Images 1, 16, 24, 30.

III.3.5. Frottement

Décrochage

Dans les classes de physique du lycée, un problème souvent pratiqué est celui du pavé sur un plan incliné. Il s'agit donc de présenter un pavé posé sur un plan inclinable et qui est retenu de glisser avant que l'inclinaison n'atteigne un certain angle par des forces de frottements. Le frottement passe alors d'un état statique qui retenait le pavé à un état dynamique. Le pavé décroche. Nous remarquons qu'à aucun moment, la forme du pavé n'intervient dans la résolution de ce type de problème. Même si les schémas des livres de physique dessinent un pavé, tout le problème est résolu en un point. Il doit donc être possible de modéliser la situation sous MIMESIS à l'aide du module de frottement que nous y avons introduit. Le modèle que nous proposons est un modèle très simple. Le plan est modélisé par une interaction entre une masse mobile « Pavé » sur laquelle s'applique une force de gravité et une masse mobile « M » reliée par une interaction de frottement sec. Cette interaction comporte une butée viscoélastique à laquelle nous donnons un seuil très grand devant le déplacement attendu du pavé. En effet, la zone d'interaction de la butée est assimilable à un plan dans un voisinage de Pavé, de taille d'autant plus importante que sera importante la distance entre pavé et m (voir figure 87). Nous inclinons le plan en faisant tourner m autour d'une masse fixe « S » initialement placée au même endroit que pavé et reliée à m par une interaction viscoélastique très raide (voir figure 93). Pavé à une masse de 1 et m à une masse de 1000 ce qui empêche Pavé d'avoir une action sur M.

Initialement Pavé et M sont alignés selon l'axe de la force de gravité $G_y=0.01$. M est lancé à une certaine vitesse dans une direction orthogonale à cet axe et commence à tourner autour de s. Le plan s'incline. Or m est en fait une masse visqueuse donc le plan va s'incliner jusqu'à un certain angle.

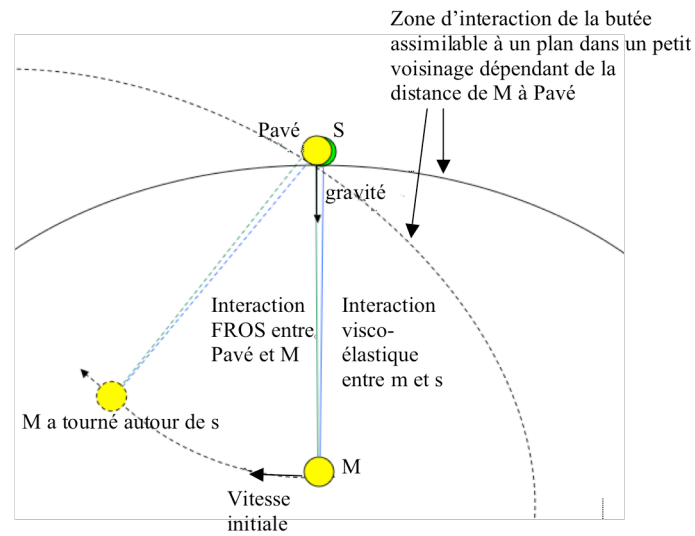


Figure 87 : Modèle du pavé posé sur un plan incliné.

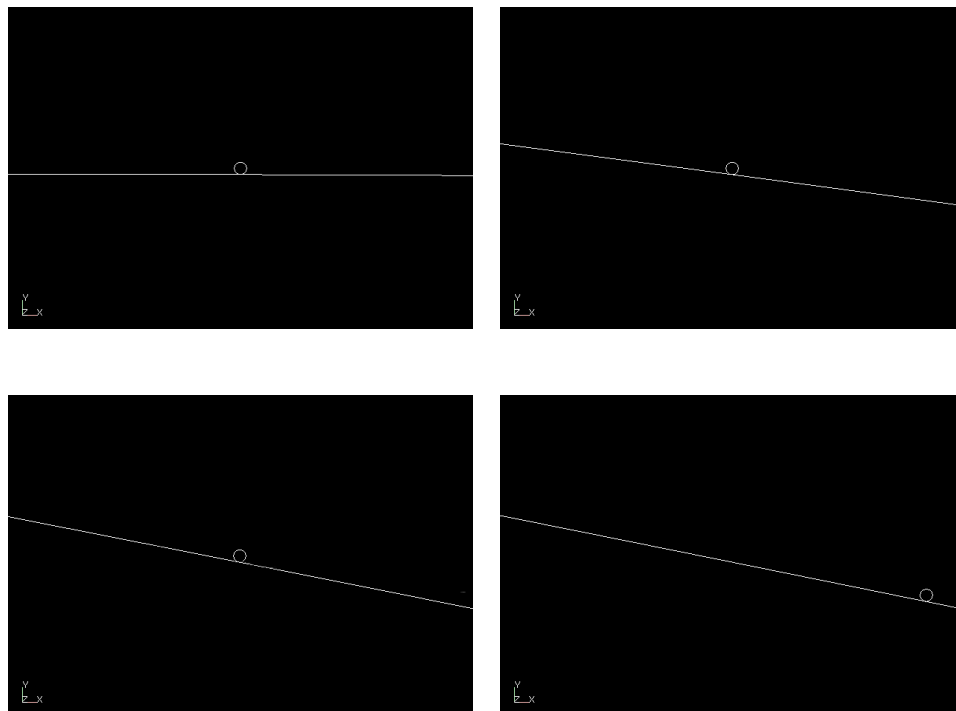


Figure 88 : Simulation de l'inclinaison d'un plan jusqu'à l'inclinaison limite permettant à une masse ponctuelle de commencer à glisser.

Images 1, 106, 121 (instant où la masse se met à glisser), 140.

Nous pouvons alors trouver la vitesse initiale nous permettant d'atteindre juste l'inclinaison nécessaire pour que pavé se mette à glisser par rapport à m . Nous n'avons pas réalisé d'étude poussée de la vitesse initiale à donner à m , mais avons jugé expérimentalement de cette valeur en fonction du coefficient de frottement statique l'interaction de frottement sec. La figure 88 montre le plan en train de s'incliner pour atteindre l'inclinaison extrême pour laquelle pavé commence à glisser. Cette simulation a été obtenue pour des paramètres de l'interaction de frottement sec de $K=1$, $Z=1.5$, $S=10000$ (paramètres de la partie butée viscoélastique de l'interaction de frottement sec), $Z_s=0.1$, $C_s=10$, $C_d=9$ (paramètres de la partie frottement de l'interaction).

Accrochage

Si un véhicule monté sur roue peut avancer sur un sol c'est qu'il y a un frottement. Ce frottement est soit statique, soit dynamique. Dans le cas d'un frottement dynamique, la roue roule avec un glissement, c'est-à-dire que la distance qu'elle parcourt en un tour ne correspond pas à son périmètre. Dans le cas d'un frottement statique, la roue roule sans glissement. La distance parcourue en un tour est exactement égale à son périmètre. On peut passer d'un état dynamique à un état statique, c'est-à-dire accrocher le sol, en diminuant la vitesse de la roue, en augmentant la charge qui pèse sur elle.

Notre modèle est un agglomérat de masses représentant une roue. Toutes les masses sont placées de manière équidistante sur la surface d'une sphère et sont reliées entre elles par une interaction viscoélastique (voir figure 89). Le centre de la roue est attiré par une force résultant d'une interaction avec une masse ponctuelle fixe qui décroît à mesure que la roue se rapproche de cette masse ponctuelle. Pour que la roue entre en rotation, les masses ponctuelles qui la composent sont en interaction frottement sec avec une masse fixe placée très loin dans la direction de la gravité qui s'applique sur la roue. Tout se passait comme si la roue se déplaçait sur un plan. Les paramètres de l'interaction de frottement sec sont les suivants : $K=0.01$, $Z=0.1$, $S=1000$, $Z_s=0.1$, $C_s=1$, $C_d=0.5$. Les masses mobiles sont de masse 1 et soumises à une gravité verticale $G_y=0.001$.

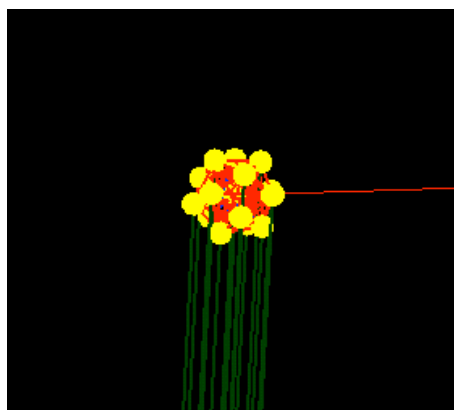


Figure 89 : Modèle de roue en interaction de frottement sec.
Une masse fixe et tirée par une force décroissante avec la distance.

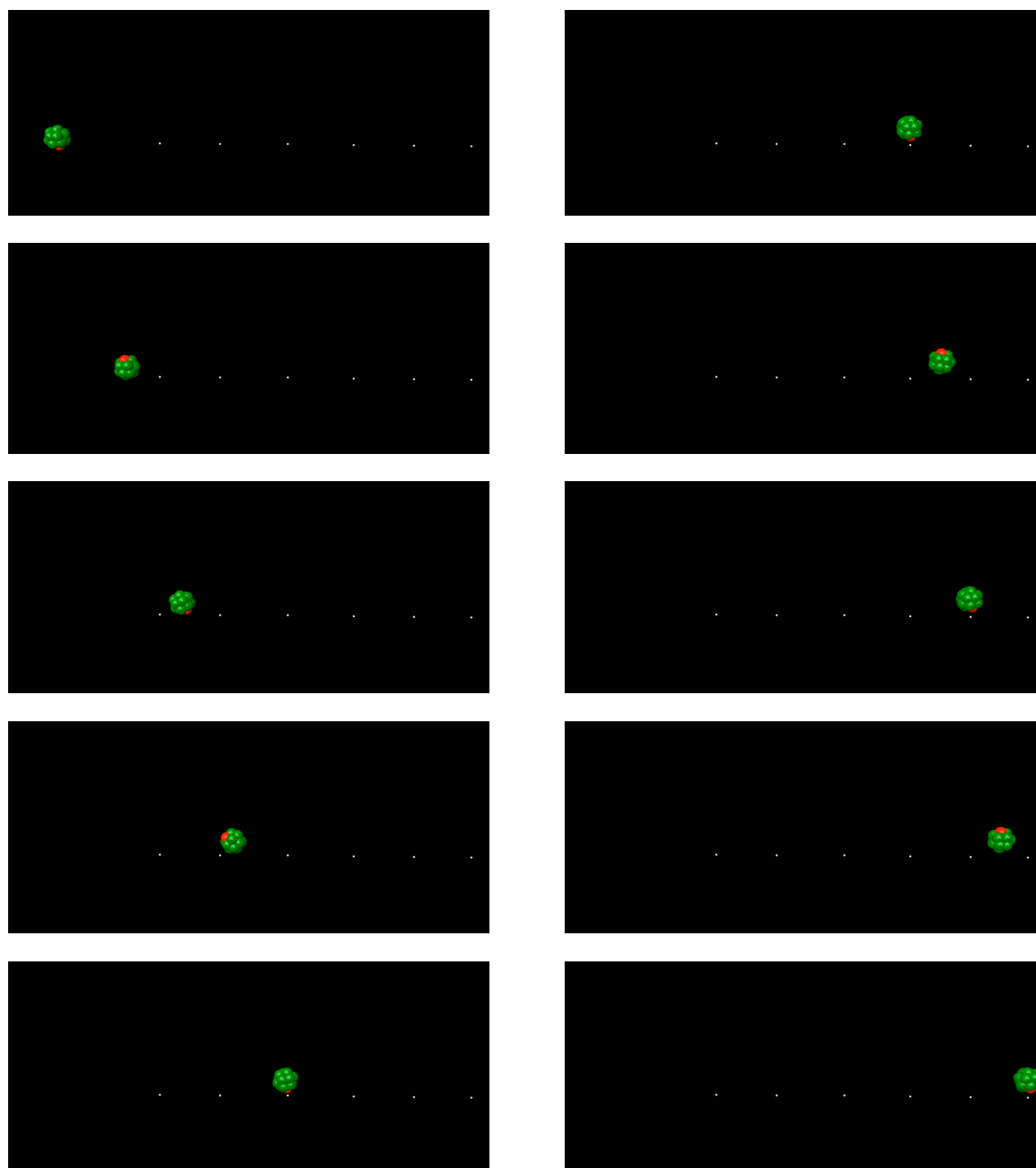


Figure 90 : *Simulation d'une situation d'accrochage.*

La séquence se lit de haut en bas et de gauche à droite. Les images proposées sont les images : 31, 37, 42, 48, 53, 58, 63, 69, 75 et 83 de la simulation.

III.4. Validation de l'outil par des modèles créés par d'autres usagers

Un certain nombre d'usager avec qui nous avons pu collaborer pendant ces années de thèse sont venus incrémenter la bibliothèque de modèles de MIMESIS dans le cadre de travaux de recherche et de création. Nous rendons compte de certains de leurs résultats. Nous tenons à clarifier à chaque fois la place que nous avons pu prendre dans le processus de modélisation en évitant de nous attribuer ce qui ne relève pas de notre travail. Certains de ces travaux ont été l'occasion de valider les nouvelles fonctionnalités de MIMESIS notamment l'utilisation du format TMGMS dans les travaux de Marouane Cherkaoui et de Kevin Sillam.

Dans l'ensemble de ces situations d'utilisation, nous avons pu observer plusieurs choses concernant MIMESIS. MIMESIS a permis d'accéder plus rapidement à la phase de modélisation, de développer des modèles dans un temps plus court, de se poser de nouvelles questions de modélisation, d'avoir une base commune de communication pour s'échanger des modèles et des problèmes.

III.4.1. Modèle de Striction d'Ali Allaoui

Le phénomène de striction ou d'écrouissage advient après qu'une élasticité ait dépassé le seuil de plasticité et que le matériau soit prêt à se rompre. Un certain seuil d'étirement étant atteint, le ressort "casse", la force de rappel devenant plus faible que la force exercée pour étirer le ressort. Un phénomène similaire est celui du goutte-à-goutte du robinet permet d'observer ce phénomène: la goutte est attirée vers le bas par son poids et retenue par une cohésion avec le robinet et l'eau restante. Ce phénomène est d'autant plus facile à observer que le liquide est visqueux comme le miel.

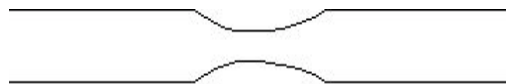


Figure 91 : *Forme caractéristique du phénomène de striction*

Le modèle suivant, a été développé sous MIMESIS par Ali Allaoui au cours de son stage Master 2 que nous avons co-encadré avec Annie Luciani. Pour décrire le modèle qui réalise cet effet de striction, la métaphore de la goutte et du robinet est conservée. Le robinet et la goutte sont deux points matériels en interaction de cohésion de type Lennard-Jones. Le robinet est représenté comme une sphère de rayon égal à la longueur au repos de l'interaction de cohésion. La goutte est initialement dans un état libre au-dessus du robinet. On observe donc qu'elle entre en interaction cohésive avec lui avant d'entrer "en contact", c'est-à-dire avant la collision proprement dite. La figure de l'interaction cohésive (figure 92) montre en effet que la collision se produit au seuil L et l'interaction de cohésion s'installe au seuil S.

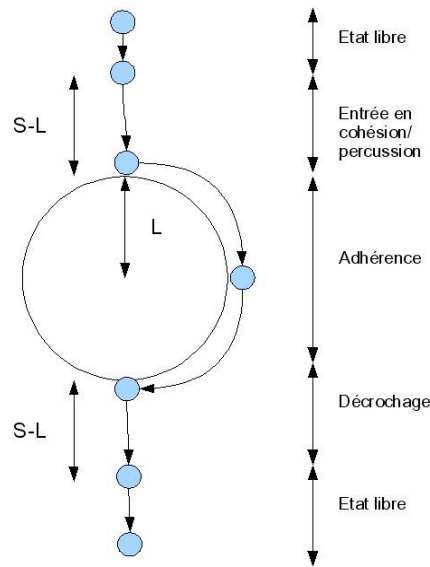


Figure 92 : Les phases successives du phénomène modélisé.

Figure extraite du rapport de stage d'Ali Allaoui

Pour obtenir l'effet de striction, il est possible de diminuer progressivement l'élasticité K_1 .

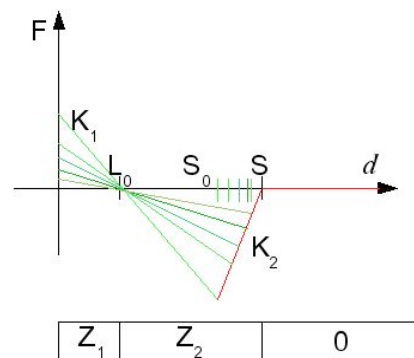


Figure 93 : Modification du paramètre K_1 de l'interaction de cohésion.

Figure extraite du rapport de stage d'Ali Allaoui

Dans le contrôle de K_1 , on remarque que plus K_1 est faible plus la force de rappel maximum F_{rMax} est faible. La force de rappel permet de contrer le poids de la goutte et sa quantité de mouvement, donc si $F_{rmax} < \text{poids la goutte}$ décroche, ce décrochage pouvant arriver avant si la quantité de mouvement n'est pas négligeable. La décroissance de K_1 entraîne donc la goutte vers le bas (la goutte descend comme pour trouver un nouvel état d'équilibre) jusqu'au décrochage. La figure suivante montre le résultat publié dans [AEL07].

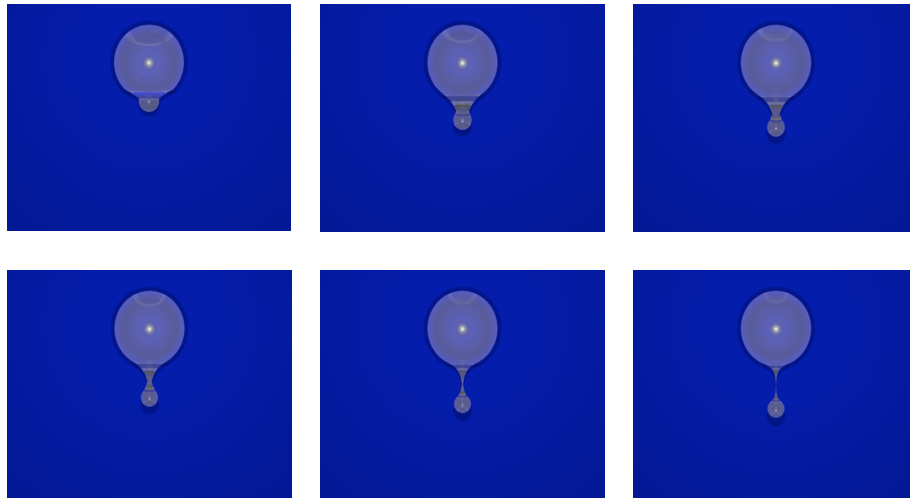


Figure 94 : Modèle de striction habillé par des surfaces implicites et rendus à l'aide de Povray

[AEL07]

III.4.2. Habillage dynamiques de modèles particuliers par Kevin Sillam

Kevin Sillam, dont nous avons co-encadré le stage de Master 2 avec Annie Luciani n'a pas à proprement parlé été centré sur MIMESIS mais plutôt sur la question de l'habillage des modèles en sortie de MIMESIS à l'aide de la méthode des surfaces implicites dynamiques proposée par Habibi et Luciani dans [HL02]. La méthode avait prouvé son intérêt dans de nombreux cas mais au moment où elle avait été développée, la technologie des ordinateurs ne permettait pas de l'inclure dans un environnement de conception interactif car elle faisait intervenir des calculs trop lourds pour l'époque.

Il était donc question de développer une interface pour la conception d'un habillage de modèles physiques particulière selon cette méthode. Notre expertise dans le développement de ce type d'environnement acquise avec le développement de MIMESIS nous a permis de guider Kevin dans les spécifications de ce nouvel environnement qui vient compléter la liste des outils du groupe ACROE-ICA pour la création d'images animées.

Les temps de l'interface ne sont pas exactement les mêmes que dans MIMESIS puisqu'il n'est plus question d'éditer la topologie entière du réseau. Au contraire, une partie de la topologie étant fixée selon une grille, il tient lieu dans un premier temps de calibrer le modèle pour l'adapter spatialement à l'espace du mouvement amont. Une fois cet ajustement spatial réalisé, il convient de définir les interactions à l'intérieur d'un phixel, entre les marqueurs et les phixels et entre les phixels eux-mêmes. La seule opération d'édition topologique est donc de définir la nature des interactions entre des phixels fixes. Une fois la topologie entièrement fixée, l'utilisateur ne peut alors que jouer sur les paramètres physiques du modèle. Au final, une méthode de rendu permet de visualiser le résultat.

L'interface utilisateur de ce logiciel, baptisé « Ecran d'épingle » en l'honneur des travaux d'Alexandre Alexeïff et de ses gravures sur un écran d'épingles mobiles, suit ce processus de création en séparant distinctement ce qui relève de la phase de calibrage, ce qui relève de la phase de modélisation (nature des interactions et paramètres physiques) et ce qui relève du rendu (voir figure 95).

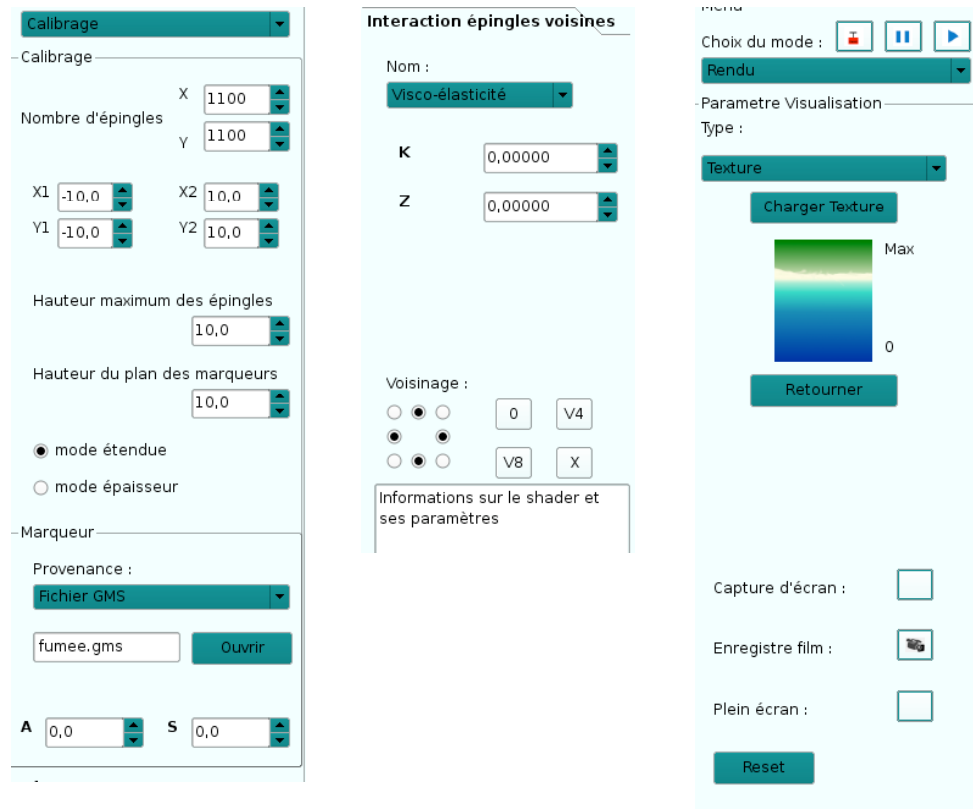


Figure 95 : Les trois temps de l'interface « Ecran d'épingle ».

A gauche l'onglet du mode calibrage, au centre l'onglet du mode modélisation et à droite l'onglet du mode rendu.

De nombreux modèles de comportement de fumée ont été obtenus par le groupe ACROE-ICA à l'aide du formalisme CORDIS-ANIMA [LVH*95]. Ces modèles présentent un petit nombre de particule, de l'ordre de 300, qui, s'il est rendu tel quel, évoque le mouvement des fumées mais n'a pas l'apparence continue d'une fumée (cf. figure 96). Pour résoudre ce problème, Habibi, dans [HL02] propose d'utiliser le signal de mouvement produit par les modèles particuliers de fumée en entrée d'un autre modèle CORDIS-ANIMA basé sur les surfaces implicites dynamiques présentées au paragraphe I.2.4. du chapitre 4. Nous rappelons qu'il s'agit d'un modèle constitué d'un ensemble d'oscillateurs CORDIS-ANIMA quadrillant l'espace dont l'état est modifié par des « marqueurs » qui viennent perturber leurs comportements propres. Ces marqueurs sont en fait des générateurs de positions dont le mouvement est contrôlé par les signaux de mouvement provenant de la simulation du comportement de fumée.

Nous avons réimplanté dans MIMESIS des modèles de fumée similaire à [LVH*95]. Les signaux de positions de ces modèles, enregistrés dans des fichiers au format TMGMS ont été utilisés ensuite par Kevin Sillam en entrée d'un modèle d'habillage par surface implicite dynamique qui a permis de donner une forme plus continue aux fumées en ajoutant en outre des phénomènes de touage, d'effilochement et de micro turbulences qui n'appartenaient pas au modèle amont comme le propose Habibi. Les travaux d'Habibi dont l'habillage des fumées n'est qu'un exemple se sont portés plus généralement sur la question de l'habillage de modèles particuliers dans une vision multi-échelle de différents niveaux de modélisation où chaque modèle peut être simulé avant d'être utilisé en entrée d'un autre qui « habille » le modèle amont.

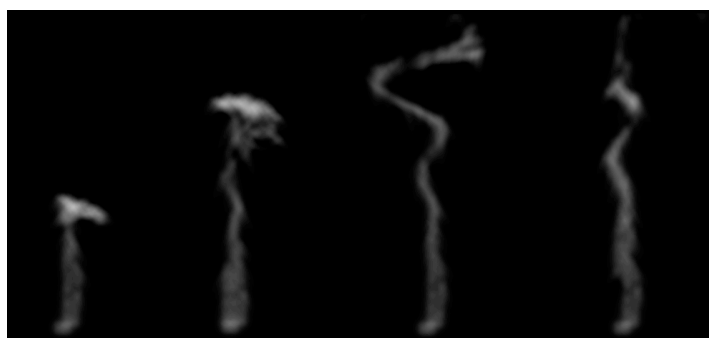


Figure 96 : Modèle de fumée habillé à l'aide du logiciel « Ecran d'épingles ».

III.4.3. Typologie de mouvements de Marouane Cherkaoui

Le sujet du stage de Master de Marouane Cherkaoui était d'initier une réflexion sur une typologie de mouvement de base pouvant amener à la création d'une bibliothèque qui serait plus qu'une simple collection de modèle à la suite notamment de l'approche d'une grammaire des mouvements danser proposée par Chimin Hsieh (cf. partie III.4.4.). Si la tâche est titanesque, l'idée était de partir à minima d'une liste de verbe d'action qui serait évoquée par un modèle de mouvement réalisé avec MIMESIS. En effet, un premier moyen de caractériser des mouvements élémentaires seraient de faire référence à des verbes d'action qui pourraient eux subir les classifications qu'on peut trouver en linguistique par exemple. Pendant son stage, Marouane, qui venait initialement d'une formation en génie civil, a réalisé un ensemble de modèles convaincants et de manière relativement autonome. La figure suivante reproduit certaines images de ces modèles.



Figure 97 : Travaux de modélisation de Marouane Cherkaoui.

A gauche : le modèle « danser dans le vent » et son habillage au centre. A droite : le modèle « se déhancher ».

Lors de son stage de Master 2, Marouane Cherkaoui a utilisé les fonctionnalités de composition de modèles en enregistrant et en les réutilisant dans un modèle amont les signaux de geste et de mouvement. Ce genre de procédé de modélisation a particulièrement été utilisé dans un modèle de frissonnement composé de trois modèles en chaîne. Le modèle est assez compliqué à expliquer mais il est intéressant d'en retenir qu'un modèle de frissonnement minimal composé de 3 MAT est 2 LIA, est ensuite utilisé en entrée d'une structure plus complexe pour obtenir le frissonnement global de cette structure. Le modèle aval simple peut donc être vu comme le moteur minimal du frissonnement. Le modèle amont utilise les signaux produits par le modèle aval par l'intermédiaire des générateurs de position (cf. figure 98). La structure finale a été habillée pour évoquer un personnage en position fœtale qui frissonne. Pour plus de précision sur le modèle, on pourra se rapporter au rapport de stage de Marouane [Mar09].

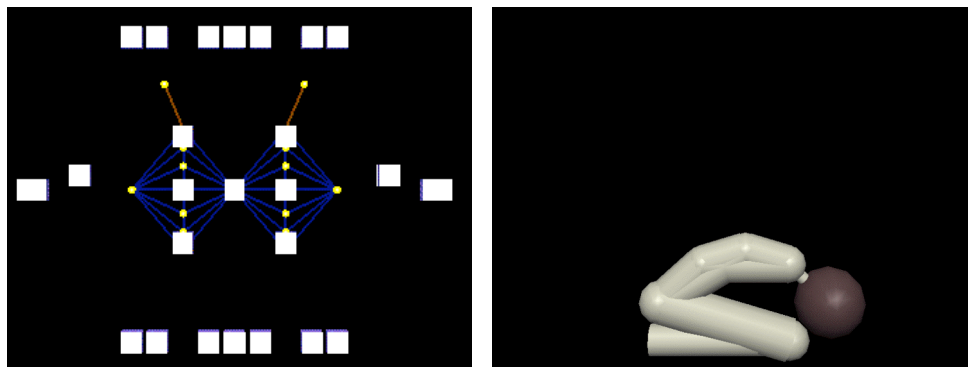


Figure 98 : Modèle de structure frissonnante utilisant des signauxTM GMS.

A gauche, le modèle (les cubes représentent des générateurs de position). A droite le modèle habillé pour évoquer un personnage en position fœtale qui frissonne.

III.4.4. Modélisation de mouvements dansés par Chimin Hsieh

Le cas des travaux de thèse de Chimin Hsieh est important pour nous. Il a en effet effectué l'ensemble de ses travaux de modélisation à l'aide du logiciel MIMESIS. Les habillages ont ensuite été réalisés à l'aide du logiciel MAYA et outils de *mapping* que nous avons développé. À ce titre, Chimin Hsieh a été sans conteste l'utilisateur le plus conséquent de notre environnement. Il est entré en thèse à l'issue du même master que le nôtre, le master Imagerie, Vision et Robotique de l'ENSIMAG, mais son parcours antérieur est bien différent. Il avait en effet effectué au paravent, 5 ans dans une école d'Art à Taïwan. Il ne maîtrisait donc pas réellement les concepts de programmation et ne disposait pas d'une formation aussi avancée que la nôtre en mathématiques et en physique.

Ses travaux étaient motivés par la réalisation de mouvements dansés à l'aide de l'ordinateur. L'approche de la modélisation par réseaux CORDIS-ANIMA l'a tout de suite captivé. Lui-même pratiquant la danse contemporaine, il trouvait une similitude entre la manière de concevoir le mouvement du point de vue des causes au sein du groupe ACROE-ICA et la manière dont on parlait du mouvement dans ses cours de danse. Il s'est donc attelé à réaliser une bibliothèque de verbes de la danse (débouler, valser, marcher...), à l'aide du logiciel MIMESIS. Nous ne pouvons nous empêcher de dire ici notre étonnement constant devant l'ingéniosité des modèles qu'il a développés. Nous le voyions reproduire des mouvements, s'interroger sur les causes de ce mouvement puis se mettre à sa machine et réussir à en reproduire les caractéristiques.

En tant qu'utilisateur assidu de MIMESIS, nous avons dû observer qu'il n'a jamais remis en cause le comportement de l'interface et s'est adapté à celle-ci. C'est certainement la raison pour laquelle nous n'avons pas souhaité nous appesantir dans ce manuscrit sur la description des choix de comportement de l'interface. Il semble en effet que dans une situation d'utilisation de longue durée, l'utilisateur accepte les choix faits par le concepteur même si ces choix ne répondent pas à l'ergonomie la plus intelligente.

En revanche, et à de nombreuses reprises, il a requis de nouvelles fonctionnalités. Dans ces cas, une discussion devait s'engager. Non pas que nous étions hostile à modifier le logiciel pour ses utilisations particulières. Mais il s'agissait de s'assurer que ce qu'il souhaitait rentrait bien dans le cadre et la philosophie de l'animation par réseaux CORDIS-ANIMA. La plupart du temps, il s'est avéré que ses requêtes découlaient d'une mauvaise analyse du problème. Nous pouvons prendre l'exemple d'un cas de modélisation du mouvement de deux danseurs où l'un conduirait l'autre avant que les rôles de s'inversent. Chimin voulait alors pouvoir contrôler des forces en entrée de son modèle. Nous avons vu l'entrée en force pose souvent un problème de modélisation puisque le principe d'action et de réaction sous-jacent au formalisme CORDIS-ANIMA est rompu. Il s'agissait donc de revoir le modèle pour le préciser. En réalité, Chimin cherchait à modéliser des flux d'énergie passant entre les deux danseurs. Il convenait donc de considérer le couple de danseurs comme un unique objet physique au sein duquel circulait une énergie, en choisissant correctement les interactions et les paramètres physiques du modèle. De tels modèles ont pu alors être réalisés à l'aide de MIMESIS, sans rajouter de fonctionnalités, et ont constitué une partie de la pièce Fluente présentée au Forum Monaco Danse en 2006.

Nous avons certaines fois cédés aux requêtes de Chimin, notamment pour lui fournir un programme de conversion de fichiers de mouvement issu de MAYA vers le format TMGMS pris en entrée de MIMESIS. Nous devons remarquer qu'il n'a finalement jamais utilisé notre programme.

Dans une perspective de composition temporelle de mouvement, nous avons fait l'analyse du besoin de pouvoir simuler par partie un modèle. Il s'agit alors de placer des marqueurs temporels au cours de la simulation et de pouvoir modifier les paramètres pour chaque portion de simulation de manière à ce que la modification d'une partie de la simulation soit répercutée sur les simulations suivantes. Le développement d'une telle fonctionnalité demandait de revoir l'intégralité de l'architecture de MIMESIS. Nous avons alors développé un outil permettant de réaliser cette opération à minima : pouvoir exporter un fichier comportant les positions et les vitesses à un instant donné de la simulation et réimporter ce fichier comme valeur initiale d'une nouvelle simulation. Dans ce cas, la fonctionnalité n'était pas pratique mais permettait de réaliser, en prenant beaucoup plus de temps, ce qui n'était pas possible avant.

D'une manière générale, nous avons donc tenu à ne pas nous retrouver dans une situation de producteur/consommateur où Chimin aurait commandé des fonctionnalités et où nous nous serions contentés de répondre au mieux à ses attentes. Cette démarche est d'une importance considérable pour nous. Elle est à la base de notre vision de la pratique des arts numériques où il n'est pas acceptable, pas plus qu'il n'était acceptable dans les autres arts, que l'artiste ne soit que le consommateur d'une technologie. Au contraire, nous avons privilégié une collaboration au sein de laquelle chacun apportait ses compétences, mais devait comprendre ce que l'autre pouvait et voulait faire.

Chimin Hsieh a remporté le premier prix étudiant du forum Monaco Danse 2006 pour ses travaux sur l'étude de la propagation de l'énergie à travers le corps dans sa pièce intitulée *Fluente*. Ce prix lui a ouvert une bourse de 6 mois de résidence à l'Université d'Arizona. Faut noter que c'était la première fois qu'un tel prix était remis à des travaux n'incluant aucun danseur réel au sein de l'œuvre.



Figure 99 : Extraits de *fluente* réalisé par Chimin Hsieh et Annie Luciani.

Les mouvements sont produits par MIMESIS et habillés sous MAYA

IV. Utilisations en situation pédagogique

L'environnement MIMESIS a été mis à la disposition de novices en modélisation masses - interactions à de nombreuses occasions. Cela nous a permis de vérifier à de nombreuses reprises l'hypothèse qu'une interface interactive de conception facilite l'appréhension des concepts de la modélisation masses - interactions pour les novices. Dans un souci de diffusion de cette pratique, nous nous sommes donc attachés à l'organisation d'ateliers pédagogiques pour en susciter l'intérêt et aussi mettre en question les approches traditionnelles de l'animation. Dans le cadre de la pédagogie, MIMESIS a pu tour à tour permettre à des personnes non-expertes en modélisation physique de pouvoir expérimenter un processus nouveau d'animation, de s'extraire de la vision morphologique de l'objet pour se focaliser sur le mouvement ou encore de mettre en perspective les autres environnements d'animation qu'ils pouvaient connaître.

IV.1. Principes de la pédagogie autour de MIMESIS

Au cours de nos travaux, nous avons mis en place une séquence pédagogique destinée aux utilisateurs novices qui souhaitent découvrir la pratique de la modélisation de réseaux CORDIS-ANIMA. Nous avons porté une attention particulière à ne pas forcer ou diriger les apprenants vers un certain type de comportement dynamique à modéliser afin de leur laisser, en fin de parcours, quand le temps le permettait, toute liberté de créer des modèles nouveaux, et de susciter leur créativité. Cette opération requiert de la part du formateur d'éviter de dispenser un apprentissage trop directif comme peut l'être généralement un TP. L'éventail des possibilités de génération de mouvement étant très vaste, beaucoup de possibilités restant à explorer, il ne nous a pas semblé fécond de proposer aux apprenants de résoudre des problèmes déjà résolus, et surtout de ne pas les diriger vers des pistes que le formateur a déjà exploré. Au contraire, il semblait important que l'apprenant s'approprie l'environnement pour tenter de mettre en place un modèle et se confronter au plus vite aux difficultés de la modélisation.

Le formateur est directif dans les premières heures. Ces heures sont dédiées à l'apprentissage du comportement de l'interface. Il change progressivement de rôle pour entrer dans une relation d'interaction, un discours, un échange, une maïeutique, dont le but est de résoudre des problèmes de modélisation que l'apprenant se pose. Le rôle du formateur est alors d'aider l'apprenant à préciser une idée qui lui paraît intéressante. Il ne possède pas nécessairement dans ce cas la solution, mais dispose d'une expérience notable qui lui permet de dépister les impasses. Il sait par exemple qu'il est

nécessaire de s'abstraire d'une vision objet de la modélisation pour aller vers une modélisation du mouvement. Le vocabulaire employé est donc plutôt relatif aux verbes d'action (tomber, frissonner, couler) qu'aux noms communs (tasse, herbe, eau).

Le risque d'une telle approche est néanmoins de laisser l'apprenant face au « complexe de la page blanche », ou alors de partir dans une exploration aléatoire, anarchique et inféconde des réseaux CORDIS-ANIMA. L'exploration des exemples existants dans la bibliothèque a un rôle important à jouer dans ce contexte. L'apprenant est amené à modifier un ou plusieurs modèles de son choix, des modèles possédant un pouvoir évocateur pour lui. Il commence par les modifications les plus superficielles (fréquence de la simulation, habillage) pour entrer progressivement dans les profondeurs du modèle (changement des paramètres puis de la structure topologique). Ce faisant, il acquiert une connaissance du comportement des réseaux CORDIS-ANIMA. Mais l'exploration est aussi une source d'inspiration pour l'apprenant qui est amené à se poser les premières questions de modélisation. C'est seulement après cette phase de modification des modèles existants que l'apprenant peut se confronter à la mise en place d'un modèle personnel. Soit en partant des modèles qu'il a exploré, soit en en construisant un nouveau de toute pièce.

Dans le cas de séquences pédagogiques très courtes permettant seulement une découverte de la modélisation de réseaux CORDIS-ANIMA, nous avons privilégié dans ce cas la résolution de problèmes simples et déjà résolus, pour inclure l'apprenant, encore une fois, le plus rapidement possible dans le processus de modélisation. L'apprenant ne manipule alors généralement pas l'interface. Cette tâche est laissée au formateur qui pose le problème, met en œuvre les idées de l'apprenant, critique les résultats et le mène progressivement sur la piste de la solution.

Les divers exemples que nous allons décrire dans la suite de cette partie n'ont pas tous abouti à un modèle final. Le temps a parfois manqué, tous les apprenants n'ont pas trouvé de l'intérêt à la pratique. Mais nous sommes parvenus dans un certain nombre de cas à des résultats convaincants que nous allons exposer ici brièvement.

IV.2. Mise en situation

IV.2.1. Ateliers classe découverte

Dans le cadre de la sensibilisation des jeunes à la science et aux études d'ingénieur, le groupe ACROE-ICA a participé à la tenue d'ateliers classe découverte organisés par Grenoble INP. Nous avons participé aux 4 séances de 3 heures des 16 et 17 décembre 2008 ainsi qu'aux 4 autres séances des 25 et 26 juin 2009. Le travail autour de MIMESIS tenait généralement une heure à la fin de chaque séance, après que leur ait été présenté la modélisation de phénomènes sonores à l'aide du logiciel GENESIS et la relation instrumentale homme-ordinateur par l'intermédiaire des transducteurs gestuels rétroactifs. En une heure, il s'est agi de résoudre un unique problème. Partant du modèle de tas de sable de la bibliothèque d'exemples, nous avons supprimé les interactions de frottement avec le sol,

essentielle dans la constitution d'un tas. Sans frottement, les grains de sables se répandent sur le sol pour former une couche unique. Les élèves étaient interrogés sur la manière d'obtenir un tas à partir du modèle de départ. Ils n'ont que rarement trouvé immédiatement la solution. La première idée était de placer des obstacles à gauche et à droite de la couche pour éviter que les grains ne se répandent. Suivant cette démarche, un tas se forme pourtant, car la première couche de grains permet au frottement d'émerger. Nous avons vu en effet dans le chapitre 3, paragraphe II.3.1, que le frottement était un phénomène émergeant des interactions entre surfaces non lisses au niveau microscopique. C'est alors que les idées de rugosité et ensuite de frottement ont pu être trouvées par les élèves pour leur permettre de revenir au modèle implanté dans la bibliothèque, modèle qui fait intervenir des modules de frottement sec de type FROS.

IV.2.2. TIPE d'étudiants en classes préparatoires

Dans le cadre des séances de soutien que nous avons dispensées à l'Ecole des Pupilles de l'Air de Saint Ismier, deux étudiants en première année de classe préparatoire nous ont sollicité pour la mise en œuvre de modèles de comportement du fil produit par l'araignée. Ce fil a trois propriétés notables : il est largement extensible, il exhibe un fort amortissement au niveau des rotations sur lui-même et il possède un fort pouvoir adhésif. Partant des données d'observations d'un certain nombre d'article sur les propriétés de ce fil [Den76, VEF06, EFV07] nous avons pu établir avec les deux étudiants, un modèle pour chacun de ces comportements. L'extensibilité a été obtenue à l'aide d'une chaîne de masses en interaction viscoélastique. Les paramètres physique des interactions ont été obtenus par une méthode de détermination heuristique afin d'obtenir les elongations observées sous l'action de contraintes. Le modèle minimal permettant d'obtenir un pendule de torsion avec un fort taux d'amortissement a été obtenu en utilisant simplement 4 masses placées dans un milieu visqueux en interaction viscoélastiques (voir figure 100). L'adhésion a été obtenue grâce à une interaction cohésive de type COH3. C'est d'ailleurs une interaction de type Van der Waals qui était observés par les travaux cités (voir figure 101).

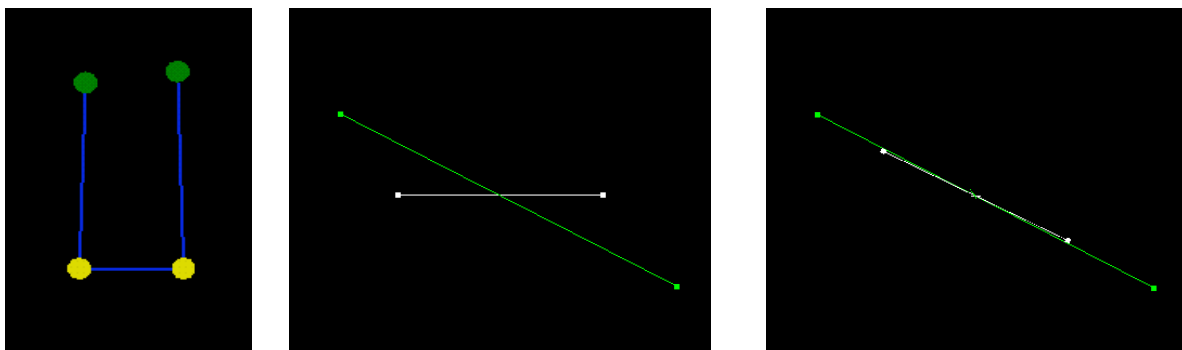


Figure 100 : Modèle de pendule de torsion avec fort amortissement.

A gauche : le modèle. Au centre, le modèle vu de haut en situation de déséquilibre. A droite : retour à l'état d'équilibre avec peu d'oscillations.

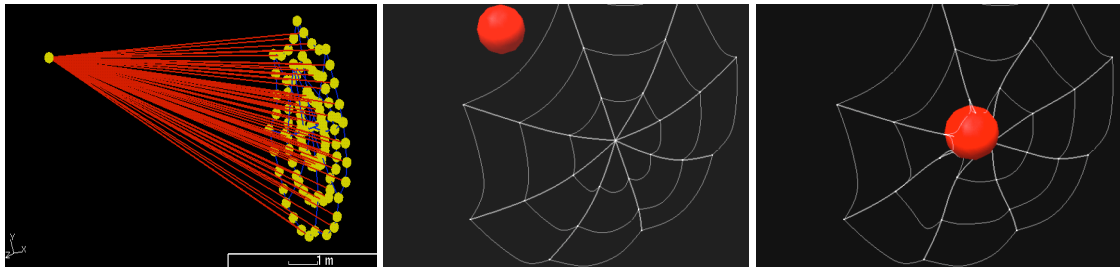


Figure 101 : *Modèle de proie prise dans une toile d'araignée.*

A gauche : le modèle. Au centre et à droite, la proie est prise dans la toile grâce à l'interaction de cohésion de type Van der Waals.

IV.2.3. Ateliers ARCAD

Des ateliers ont été organisés dans le cadre du projet ARCAD financé par la délégation aux arts plastiques. Ce projet a permis une collaboration entre l'Ecole Supérieure de l'Image (ESI) de Poitiers/Angoulême et de l'ACROE. 6 ateliers ont eu lieu en 2005 et 2006 permettant à des étudiants de master de l'ESI de découvrir la modélisation physique à l'aide des réseaux CORDIS-ANIMA. Le but principal n'était pas, dans ce cadre, de produire des modèles mais que les étudiants se consacrer par la pratique à ce type de modélisation afin de les faire participer à des discussions sur les champs de recherche nouveaux et prometteurs pour les arts visuels. Une vingtaine d'étudiants ont pu participer à ces ateliers. De nombreux débats et perspectives ont pu émerger.

Ce manuscrit ne saurait être le lieu du compte-rendu de ces discussions. Il en est fait un rapport dans [LB07]. Il est cependant important de noter que les étudiants confrontés aux logiciels de synthèse de forme et d'animation classiques ont très bien saisi la nouveauté des problématiques posées par l'environnement MIMESIS (prévalence du mouvement, du geste, sur la forme). Pour un certain nombre d'entre eux, l'utilisation de MIMESIS a été l'occasion de mieux formaliser leurs problématiques personnelles en distinguant notamment MIMESIS, des logiciels de conception de forme et de rendu. Citons par exemple Frédéric Voignier, un des étudiants participants aux premiers ateliers: « Il s'agit d'une façon de penser révolutionnaire. Ici on pense d'abord « geste », on pense d'abord « vibration ». Les formes 3D sont de simples renforcements, rajoutées a posteriori ».

IV.2.4. TP du Master Art Science et Technologie

Dans le cadre du master Art, Science et Technologie, nous sommes intervenus dans le cours d'Annie Luciani sur les arts visuels dynamiques. Nous avons participé à l'encadrement des séances de TP sous MIMESIS en 2006, 2007 et 2008. Ces séances donnaient lieu, en fin d'année à un rapport présentant un modèle fini. Ici encore, il ne serait être question de recenser l'ensemble des travaux réalisés à l'issue de ces séances de TP.

Nous pouvons nous intéresser pour exemple aux travaux de Raphaël Loyet sur les réactions en chaîne. Raphaël Loyet propose dans son rapport, un modèle de chutes en série observé en plaçant des boîtiers de CD sur la tranche, côte à côte, et en faisant chuter le premier sur son voisin provoquant, une réaction en chaîne de chute de l'ensemble des boîtiers. Raphaël Loyet remarque qu'après le premier impacte de chute, l'objet n'est pas immobile mais opère un ou plusieurs petits sauts. Ce phénomène est très bien rendu dans la séquence produite par MIMESIS. En revanche, en appliquant une technique de rotoscopie sur des images filmées d'une telle réaction en chaîne (transformation d'une scène filmée en dessin animés), il est difficile de rendre ce comportement fin. Les boîtiers dessinés à partir de la scène filmée, s'écrasent sans rebondir. L'expérience de la modélisation physique a donc pu susciter chez l'étudiant la prise de conscience des possibilités offertes par la modélisation à l'aide de réseaux CORDIS-ANIMA.

IV.2.5. Formation continue

Le laboratoire ICA, dispense des formations dans le cadre des formations continues de Grenoble INP. Nous avons participé à la formation intitulée « Modélisation physique pour l'image » et dispensée à des employés de la société CABRILOG. La société CABRILOG développe des logiciels d'aide à l'apprentissage de la géométrie. Elle prospecte actuellement pour introduire des modèles physiques au sein de ses logiciels afin de rendre les interactions entre les utilisateurs et les objets géométriques plus réalistes et vivants. Il pourrait être par exemple intéressant d'ajouter des phénomènes dynamiques lorsqu'un crayon virtuel se déplaçant contre une règle, rencontre une graduation.

Partant d'une discussion sur les comportements dynamiques intéressant la société, nous avons proposé quelques modèles qu'il s'est agit de modifier en ligne au cours de discussions avec les employés présents. Ils ont pu découvrir à cette occasion des moyens de modéliser des frottements ou encore de créer des attractions entre un objet et un point. Il est apparu que bien souvent nous cherchions un modèle complexe qui ne rendait rien et qu'il fallait revenir à des idées très simples pour modéliser les comportements qu'ils avaient à l'esprit. Il est assez remarquable, que des professionnels de logiciels de géométrie se soient prêtés assez facilement au jeu d'une modélisation qui est très éloignée de leur domaine..

Conclusions et perspectives

Ces cinq années au sein du groupe ACROE-ICA nous ont offert l'opportunité de disposer d'un temps long pour suivre le développement d'un logiciel. Nous avons donc été en mesure de proposer MIMESIS à un nombre significatif d'utilisateurs et d'établir les prémisses d'une pédagogie autour de la modélisation physique par réseaux CORDIS-ANIMA pour l'animation. Il est relativement rare que des travaux de thèses puissent donner lieu pendant leur déroulement à ce type de validation. Il s'agit d'une motivation profonde qui a pu retarder la rédaction de ce manuscrit, rédaction qui arrive bien après la phase la plus soutenue des développements logiciels. Ce n'est certainement pas la seule raison mais c'est une raison essentielle qui démontre par l'exemple la pertinence de la thèse que nous défendons. Encore très récemment, les travaux de modélisation de Marouane Cherkaoui, réalisés dans le temps relativement court d'un stage de master, qui présentent des résultats très convaincants, viennent renforcer la mise en lumière de la nécessité de disposer d'un environnement de conception pour la modélisation de réseaux CORDIS-ANIMA.

Nous ne saurions en revanche affirmer ici que tout peut se faire avec la version en l'état de MIMESIS. Les travaux de thèse d'Ali Allaoui, eux aussi dirigés par Annie Luciani, qui continuent directement notre travail sur les fonctionnalités de MIMESIS à la suite de son stage de Master 2 que nous avons co-encadré, exposeront la nécessité d'augmenter ces fonctionnalités par des outils plus élaborés et ce afin de faciliter encore la gestion de la complexité des modèles et des comportements obtenus. Ali Allaoui travaille en ce moment à l'élaboration d'un langage graphico-textuel où texte et manipulation graphique sont intimement liés dans la construction de modèles faisant intervenir un grand nombre de modules. Il travaille aussi sur la gestion plus élaborée des positions initiales dans des cas types de modélisation ainsi que sur la modification en ligne des paramètres physiques et la collaboration entre espaces physiques de dimensionnalités différentes. Un environnement de conception qui dispose d'une bibliothèque de mouvements est, en outre, un moyen de normalisation des représentations des modèles. L'existence de MIMESIS implique l'existence d'un code graphique pour les modules permettant aux utilisateurs de travailler sur une base commune de représentation des modèles et donc de comprendre plus facilement et plus rapidement ce qui les compose. Il y a derrière la mise en place de tout environnement, la volonté de créer une communauté d'utilisateur capable de communiquer à partir d'un langage commun.

Nous avons aussi, durant nos années de thèse, co-encadré avec Annie Luciani le stage de Master de Kevin Sillam sur la mise en place d'un environnement interactif de visualisation de modèles physiques particuliers par la méthode des surfaces implicites dynamiques développées par le groupe ACROE-ICA. Ce stage de Master l'a amené à continuer ses recherches dans le cadre d'une thèse qui se pose la question de la mise en œuvre de nouveaux éléments physiques pour la visualisation. En effet, les images produites par le groupe ACROE-ICA manquent encore souvent d'un caractère « final ». Cela est certainement dû au fait que les recherches du groupe ont eu peu à cœur d'intégrer les méthodes de modélisation et de rendu de forme développés ailleurs, pour se concentrer sur le problème de l'écriture et la génération du mouvement. Or les travaux de Kevin Sillam montreront que le modèle physique masses - interactions peut aussi servir à la mise en forme et en lumière des modèles de mouvement produits par CORDIS-ANIMA.

Nous n'avons que peu évoqué nos travaux au sein du réseau d'excellence européen Enactive Interfaces. Dans le cadre de ce réseau, nous avons participé à l'organisation de l'évènement multiforme Enactive 07 qui comportait des conférences scientifiques, des expositions artistiques et des concerts ouverts au grand public ainsi qu'une exposition scientifique et technique nommée « Toucher le futur » dont nous avons la responsabilité. Il s'est agi, dans « Toucher le futur », de présenter au public une vingtaine d'expériences sorties directement des laboratoires, et qui avaient la particularité de mettre à contributions leurs sens : la vision, l'audition, le toucher mais aussi le goût. Nous avons, à cette occasion, encadré une quinzaine d'étudiants en thèse dont la mission était de présenter une expérience qu'ils découvraient quelques temps avant au public. Cette activité a été comptabilisée dans les formations non-scientifiques qu'ils pouvaient choisir durant leurs années de thèse. Au sein du réseau Enactive Interfaces, nous avons aussi participé aux études et expériences sur la perception, notamment sur la construction de la notion d'objet dans une expérience de réalité virtuelle. Nous avons eu à prendre part aux discussions sur la crédibilité des objets virtuels. Nous considérons que cette problématique est encore peu explorée dans le cadre de l'animation et plus particulièrement dans le cadre de la perception du mouvement et la construction d'un objet à partir d'une dualité forme, mouvement.

Il est prévu que ces nouvelles pistes soient explorées par le groupe ACROE-ICA dans le cadre du projet ANR DYNAMé, en collaboration avec le laboratoire XLIM de Limoges/Poitiers, et le laboratoire LAM de psychologie expérimentale. Si le mouvement reste une donnée dont on parle encore peu explicitement, même dans les travaux de recherche en animation, c'est certainement qu'il reste encore très mal compris. Le projet DYNAMé sera aussi et surtout l'occasion de poser de manière centrale la question de la collaboration entre topologie physique et topologie spatiale. Cette question nous paraît être une question d'avenir pour l'animation. Les travaux du projet DYNAMé permettront certainement d'apporter de nouvelles pistes dans la résolution de la relation forme/mouvement pour la production d'images animées. Nous considérons que nous avons posé dans cette thèse les prémisses d'une réflexion qui est une problématique en soit et peu conduire à de nombreux sujets de recherche futurs.

Toute thèse s'accompagne de regrets. Notre plus grand regret sera certainement de ne pas avoir évoqué ici l'extraction de formes à partir des données de la simulation. Dans la suite des travaux de thèse de Claire Guilbaud [Gui02], nous avons commencé à tester l'extraction de caractéristiques spatiales à partir des données cinématiques de certains modèles notamment des modèles de pâtes. En effet, Claire Guilbaud met en évidence en visualisant les vitesses et les accélérations de certains modèles que ces données caractérisent certains phénomènes. Lorsque l'on observe certaines simulations, l'œil arrive à détecter des phénomènes qui relèvent de la modification topologique (effondrements internes, fusions de surfaces). Or ces effets ne sont détectables qu'en visualisant le modèle en mouvement. Claire Guilbaud montre que ces effets sont liés à la configuration des vitesses et des accélérations du modèle. Une idée serait donc de créer une forme qui explicite ces phénomènes en extrayant certains critères sur ses données cinématiques. Nous avons réalisé certains modèles mais les résultats sont encore trop peu nombreux pour avoir fait le choix de faire figurer cette problématique liée à la relation forme/mouvement dans notre manuscrit. Pour autant c'est un sujet qui aurait avantageusement complété la partie 1 du chapitre 4.

Nous sortons de cette thèse confortés et renforcés dans l'idée qu'une nouvelle communauté d'animateurs capables de prendre en main à la fois le modèle physique tout en possédant dans le même temps une expertise dans l'observation du mouvement est possible. Pour cela, les efforts de pédagogie autour de MIMESIS ne doivent pas être abandonnés. MIMESIS, en tant que technologie, pose la question du rapport entre art et science. Cette question est une question sociétale. Est-on aujourd'hui prêt à considérer que l'artiste qui se confronte aux nouvelles technologies peu disposer de compétences qui ont été longtemps considérées du seul domaine des scientifiques ? A l'inverse, peut-on admettre que le scientifique, lorsque les résultats de ses travaux produisent des objets sensibles, peut être considéré comme un artiste ? Voici des questions fondamentales posés par la spécialité « Art, Science et Technologie » qui était la spécialité de thèse sous laquelle nous nous sommes inscrits dans les premières années, avant qu'elle ne soit remplacée par l'intitulé actuel pour des raisons administratives de redécoupage.

Nous pensons pour notre part qu'un cap doit être franchi sur cette question. Nous avons la conviction qu'un artiste ne peut être un consommateur de technologies mais qu'au contraire il doit prendre en main cette technologie et être en mesure de posséder la culture scientifique et technique qui lui est afférente. A l'inverse, il nous semble qu'une science qui ne ferait référence qu'à elle-même, qu'aux problèmes qu'elle se pose et ne se confronterait pas à la question de la production de technologies et même d'œuvres d'art, en bref une science qui ne se poserait pas les questions de son apport tangible et sensible à la société, ne serait qu'une science stérile et inféconde. Science sans art, art sans science, science sans technologie, technologie sans science, art sans technologie et technologie sans art, autant de raisons, aussi, de n'être que ruines l'âme.

Bibliographie

- [ACF*07] Allard J., Cotin S., Faure F., Bensoussan P.-J., Poyer F., Duriez C., Delingette H., Grisoni L. **SOFA - an Open Source Framework for Medical Simulation**. Dans *Medecine Meets Virtual Reality* (MMVR'15), pp. 13-18, 2007
- [AG85] Armstrong W., Green M.. **The dynamics of articulated rigid bodies for purpose of animation**. Dans *Proceedings of GRAPHICS' INTERFACE 1985*, pp 407-415.
- [Ala06] Allaoui A. **Introduction de la modification paramétrique dans le logiciel MIMESIS : application aux mouvements évolutifs complexes**. Rapport de Master Art, Science, Technologie. Laboratoire ICA. Juin 2006.
- [AM94] Attali D., Montanvert A. **Semicontinuous skeletons of 2D and 3D shapes**. Dans *Aspects of Visual Form Processing*, pages 32--41. World Scientific, Singapore, Proc. of the 2nd International Workshop on Visual Form, 1994.
- [AM96] Attali D., Montanvert A. **Modeling noise for a better simplification of skeletons**. Dans *Proc. of the International Conference on Image Processing (ICIP)*, Volume III, pages 13--16, Lausanne, Switzerland, septembre 1996.
- [ARS99] Arsch et al. **Les capteurs en instrumentation industrielle**, Dunod, Paris, 1999
- [AS04] Amrani M., Shariat B. **Deformable organs modeling with multi layer particle systems**. Dans *Procs of International Conference on Information Visualisation*, pp 351, ICIV'04, IEEE Computer Society (Washington DC, USA), juillet 2004.
- [BAM81] Bamberger Y. **Mécaniques de l'ingénieur**, T1 (Systèmes de corps rigides) et T2 (milieux déformables) –Ed. Hermann 1981.
- [Bar81] Barr A.H.. **Superquadrics and angle-preserving transformations**. Dans *IEEE Computer Graphics and Applications*, 1(1), pp 11-23, 1981.
- [Bar89] Baraff D. **Analytical methods for dynamic simulation of non-penetrating objects**. Dans *Computer Graphics*, volume 23(3), pp 223-232, SIGGRAPH'89, ACM Press (New York,USA), juillet 1989.

- [Bar91] Baraff D. **Coping with friction for non-penetrating rigid body simulation.** Dans *Computer Graphics*, volume 25(4), pp 31-41, SIGGRAPH'91, ACM Press (New York,USA), juillet 1991.
- [Bar94] Baraff D. **Fast contact force computation for non-penetrating rigid bodies.** Dans *Computer Graphics Annual Conference series*, pp 23-34, SIGGRAPH'94, ACM Press (New York,USA), juillet 1994.
- [BB88] Barzel R., Barr A. H. **A modeling system based on dynamic constraints.** Dans *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, volume 22, pp 179–187, Atlanta, Georgia, SIGGRAPH'88, ACM Press (New York, USA), août 1988.
- [BBB*97] Bloomenthal J., Bajaj C., Blinn J., Cani-Gascuel M.-P., Rockwood A., Wyvill B., Wyvill G. **Introduction to implicit surfaces.** Morgan Kaufmann, 1997.
- [BBB98] Brandel S., Bechmann D., Bertrand Y.. **STIGMA: a 4-dimensional modeller for animation.** Dans *9th Eurographics Workshop on animation and simulation*, Lisbon, Portugal, 1998.
- [BC00] Bourguignon D., Cani M.-P. **Controlling anisotropy in mass-spring systems.** Dans *Computer Animation and Simulation '00*, Springer Computer Science, pp 113–123, Interlaken, Switzerland, EGAnim2000, Springer Verlag, août 2000.
- [BC93] Burdea, Coiffet. **Retours tactiles et retours de force.** Dans *La Réalité Virtuelle*, pages 75-107, Paris, Hermès, 1993
- [BD93] Bechmann, D., Dubreuil, N. **Animation through space and time based on a space deformation model.** Dans *The Journal of Visualization and Computer Animation*, 4(3), juillet–septembre 1993, p 165–184.
- [BGC98] Barthe L., Gaildrat V., Caubet R. **Combining implicit surfaces with soft blending in a CSG tree.** *Proc. of CSG '98*, pp 17-31, 1998.
- [Ben01] Bendazzi G. **Alexeieff: Itinerary of a Master.** Dreamland, Paris, France. ISBN 2-910027-75-9. 2001.
- [BHW94] Breen D. E., House D. H., Wozny M. J. **Predicting the drape of woven cloth using interacting particules.** Dans *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pp 365–372, Orlando, FL, SIGGRAPH'94, ACM Press (New York, USA), juillet 1994.

- [BKO*00] de Berg M., van Kreveld M., Overmars M., Schwarzkopf O. ***Computational Geometry: Algorithms and Applications***. Springer-Verlag, 2nd edition, 2000.
- [BL03] Boulenguez P., Luciani A. **A physical Particle Model of Morphogenesis**. WSCG'03 Plzen (République Tchèque), 2003.
- [Bli82] Blinn J.F.. **A Generalization of algebraic surface drawing**. Dans *ACM Transaction on Graphics*, 1(3), pp 235-256, 1982.
- [Blo99] Bloomenthal J., Lim C. **Skeletal methods of shape manipulation**. Dans *Shape Modeling International* (1999).
- [Blu67] Blum H. **A transformation for extraction new descriptors of shape, models for the perception of speech and visual form**, MIT Press, 1967.
- [BMJ*99] Bonnafous V., Menou E., Jessel J.-P., Caubet R. **LIVE: une plate-forme d'animation utilisant conjointement plusieurs méthodes de controle du mouvement**. Dans *AFIG 99*, Reims, 24/11/99-26/11/99, LERI, IUT Leonard de Vinci BP 1035 51687 Reims, p. 49, novembre 1999.
- [BN83] Boite, Neirynek. ***Théorie des réseaux de Kirchhoff***, presses polytechniques romandes, 1983.
- [Boo78] C. de Boor, ***A Practical Guide to Splines***. New York, Springer-Verlag, 1978.
- [BPV88] Bergé P., Pommeau Y., Vidal C. **L'ordre dans le chaos, vers une approche déterministe de la turbulence**. Herman Editeur des Sciences et des Arts. 1988 (Paris).
- [BS95] Blanc C., Schlick C. **Extended field functions for soft objects**. *Proc. Of Implicit Surfaces'95*, pp 21-32, 1995.
- [BW76] Burtnyk N., Wein M. **Interactive Skeleton Techniques for Enhancing Motion Dynamics in key Frame Animation**. Dans *Communications of the ACM*, 19, 10, pp 564-584, Octobre 1976.
- [BW90] Bloomenthal J., Wyvill B. **Interactive techniques for implicit modeling**. *Computer Graphics (proc. of SIGGRAPH '90)*, 24(2), pp 109-116, 1990 .
- [BW92] Baraff D., Witkin A. **Dynamic simulation of non-penetrating flexible bodies**. Dans *Computer Graphics*, volume 26(2), pages 303-308, SIGGRAPH'92, ACM Press (New York,USA), juillet 1992.

- [BW98] Baraff D., Witkin A. **Large steps in cloth simulation.** Dans *Computer Graphics Proceedings. SIGGRAPH 98*, ACM Press (New York, USA), juillet 1998.
- [Cad94] Cadoz C. **Le geste, canal de communication homme/machine.** Dans *Technique et science informatiques*. Vol. 13, No. 1, pp 31-63, 1994.
- [CBS96] Crespin B., Blanc C, Schlick C. **Implicit sweep objects.** *Eurographics '96*, 15(3), pp 165-174, 1996.
- [CC02] Castagné N., Cadoz C. **GENESIS: A Friendly Musician-Oriented Environment for Mass-Interaction Physical Modeling.** Dans *International Computer Music Conference - ICMC 2002 Göteborg*, pp. 330-337, 2002.
- [CDA99] Cotin S. , Delinguet H., Ayache N. **Real-time elastic deformations of soft tissues for surgery simulation.** Dans *IEEE Transactions On Visualization and Computer Graphics*, 5(1), pages 62-73, janvier-mars 1999.
- [CGD97] Cani-Gascuel M.-P., Desbrun M. **Animation of deformable models using implicit surfaces.** *IEEE Transaction on Visualization and Computer Graphics*, 3(1), pp 39-50, 1997.
- [Cha96] Chancelou, B. **Modélisation physique de véhicules tout-terrain : application à la simulation dynamique de robots mobiles d'intervention en milieu naturel.** Thèse de l'Université Claude Bernard (Lyon 1), Lyon, 190 pages, 1996.
- [Chr08] Christou M. **Physical modeling of movement for a Dancer – Public – Environment interaction.** Rapport de Master Art, Science, Technologie. Laboratoire ICA. Juin 2008.
- [CHC*02] Choi Y.-J., Hong M. , Choi M.-H., Kim M.-H. **Adaptive mass-spring simulation using surface wavelet.** Dans *proc. of International Conference on Virtual Systems and Multimedia*, Gyeong Ju, Korea. VSMM'02, septembre 2002.
- [CLF84] Cadoz C., Luciani A., Florens J.-L. **Gesture, Instrument and Musical Creation. The CORDIS-ANIMA system.** Preprint no. 2086, 75th AES Convention Paris 1984.
- [CLF90] Cadoz, Libowski, Florens. **A modular feedback keyboard design.** Dans *Computer Music Journal*, Vol. 14, No. 2, 1990
- [CLF93] Cadoz C., Luciani A. et Florens J.-L. **CORDIS-ANIMA : a modeling and simulation system for sound and image synthesis.** Dans *Computer Music Journal*, 17(1), pp. 19-29, spring 1993.

- [CLF94] Cadoz C., Luciani A., Florens J.-L. **Physical Models for Music and Animated Image. The use of CORDIS-ANIMA in ESQUISSES** a Music Film by ACROE. ICMC, Aarhus (Danemark) pp. 11-18, 1994.
- [CLH96] Chancelou B., Luciani A., Habibi A. **Physical Models of Loose Soils Dynamically Marked by a Moving Object.** Dans *Proceedings of the Computer Animation*. Geneva, Switzerland, pp 27-35, juin 1996.
- [Coq08] Coquerelle M. **Calcul d'interaction fluide-structure par méthode vortex et application à la synthèse d'images.** Thèse de l'INPG. 2008.
- [Coq90] Coquillard S. **Extended Free-Form Deformation : A sculpturing Tool for 3D Geometric Modeling.** Dans *Computer Graphics (Proc of SIGGRAPH 90)*, vol 24, 187-196.
- [Cor94] Cordier E. **Modèle physique de motricité avec extéroception.** Rapport de DEA, ACROE, 1994.
- [CSH*04] Choi K.-S., Sun H., Heng P.-A., Zou J. **Deformable simulation using force propagation model with finite element optimization.** Dans *Computer & Graphics*, 28(4):559–568, juin 2004.
- [CW00] Cadoz C., Wanderley M. **Gesture – Music.** pp 71-94, Ircam – Centre Pompidou, 2000
- [DC98] Desbrun M., Cani M.-P. **Active implicit surface for animation.** Dans *Graphics Interface*, pp 143-150, 1998.
- [DDB*99] Debunne G., Desbrun M., Barr A., Cani M.-P. **Interactive multiresolution animation of deformable models.** Dans *10th Eurographics Workshop on Computer Animation and Simulation*, pp 133–144, Milano, Italy, EUROGRAPHICS'99, Springer-Verlag Wien New York, septembre 1999.
- [Den76] Denny M. **The physical properties of spider's silk and their role in the design of orb-webs.** *Journal of Experimental Biology* 65. Published by Company of Biologists. pp 483-506. 1976
- [DG95] Desbrun M., Gascuel M.-P. **Animating soft substances with implicit surfaces.** Dans *Proc. SIGGRAPH 95*, pages 287-290, 1995.
- [DG96] Desbrun M. and Gascuel M.-P. **Smoothed particles: a new paradigm for animating highly deformable bodies.** In R. Boulic and G. Hegron, editors, *Computer Animation and Simulation '96*, pp 61–76. EUROGRAPHICS'96, Springer-Verlag, août 1996.

- [DLC93] Delnondedieu, Y., Luciani, A., Cadoz, C. **Physical elementary component for modeling the sensory-motricity. The primary muscle**, 1993/09, vol. 4, pp.1-15, 1993.
- [DM03] Di Giacomo T., Magnenat-Thalmann N. **Bi-layered mass-spring model for fast deformations of flexible linear bodies**. Dans *16th International Conference on Computer Animation and Social Agents*, pp 48–54, New Jersey, CASA'03, IEEE Computer Society, 2003.
- [DSB99] Desbrun M. , Schröder P., Barr A. H. **Interactive animation of structured deformable objects**. Dans *Proceedings of Graphics Interface'99*, pp 1–8, Kingston, Ontario, Canada, Graphics Interface'99, Morgan Kaufmann Publishers Inc. (San Francisco, USA), Juin 1999.
- [EFV07] Emile O., Floch A., Vollrath F. **Time-Resolved Torsional Relaxation of Spider Draglines by an Optical Technique**. *Physical Review Letters*, vol. 98, Issue 16, id. 167402. AVRIL 2007.
- [EWS96] Eberhardt B., Weber A., Strasser W. **A fast, flexible, particle-system model for cloth draping**. Dans *procs of IEEE Computer Graphics and Applications*, pp 52-59, volume 16(5), IEEE Computer Society Press (Los Alamitos, CA, USA), septembre 1996.
- [FCL98] Florens J.-L., Cadoz C., Luciani A. **A real-time workstation for physical model of multi-sensorial and gesturally controlled instrument**. Dans *Proceedings of ICMC'98*. pp. 518-525. 1998.
- [FDF*96] Foley, van Dam, Feiner, and Hughes. **Computer Graphics: Principles and Practice** (2nd Edition). New York: Addison-Wesley. pp 478-530. 1996.
- [FF01] Foster N., Fedkiw R. **Practical animations of liquids**. Dans SIGGRAPH, pp. 23-30.2001
- [FH01] Florens J., Henry C. **Bowed string synthesis with force feedback gesture interaction**. Dans *Proc. of the International Computer Music Conference*. San Francisco, 2001.
- [FM96] FosterN., Metaxas D. **Realistic Animation of Liquids**. Dans *Graphical Models and Image Processing*, 58(5):471–483, 1996.
- [FSJ01] Fedkiw R., Stam J., Jensen H. W. **Visual simulation of smoke**. Dans *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp 15–22, Los Angeles, USA, SIGGRAPH'01, ACM Press (New York, USA), août 2001.
- [Gas93] Gascuel M.-P.. An implicit formulation for precise contact modeling between flexible solids. *Computer Graphics (proc. SIGGRAPH '93)*, pp 313-320, 1993.

- [Gel98] Van Gelden A. **Approximate simulation of elastic membranes by triangulated spring meshes.** In *Journal of Graphics Tools*, volume 3(2), pp 22-41, A. K. Peters Ltd. (Natick, MA, USA), 1998.
- [Ger73] Germain P. **Cours de mécanique des milieux continus.** T.1, Ed. Masson 1973.
- [Gir99] Giraud O. **Architecture logicielle haute performance pour la simulation temps-réel synchrone d'objets physiques multisensoriels : retour d'effort, synthèse de sons, synthèse d'images.** Thèse de doctorat de l'INPG. ACROE-CLIPS. Grenoble. Juillet 1999.
- [GLC03] Guilbaud C., Luciani A., Castagné N. **Physically-based particle simulation and visualization of pastes and gels.** 2003/09, Moscou (2003) pp.7 pages
- [GMT89] Gourret J.-P., Magnenat-Thalmann N., Thalmann D. **The use of finite element theory for simulating object and human body deformations and contacts.** Dans *Eurographics conference proceedings '89*, pp 477-487, Hamburg, EUROGRAPHICS'89, Elsevier Science Publishers B.V. (North Holland), 1989.
- [GMV*06] Gupta R., Montagnol M., Volino P., Magnenat-Thalmann N. **Optimized Framework for Real Time Hair Simulation.** Dans *Proc. of Computer Graphics International (CGI'06)*, LNCS, Springer, pp. 702-710, June 2006.
- [Gre73] Greenspan, D. **Discrete Models. Reading in Applied Mathematics.** Addison-Wesley, 1973.
- [Gre97] Greenspan, D. **Particle Modeling.** Birkhauser Ed., 1997.
- [GSL*05] Guendelman, E., Selle, A., Losasso, F., and Fedkiw, R. **Coupling water and smoke to thin deformable and rigid shells.** Dans *ACM Transactions on Graphics* (July 2005), J. C. Hart, Ed., vol. 24, SIGGRAPH'05, ACM Press (New York, USA), pp. 973-981.
- [Gui02] Guilbaud C. **Modélisation et visualisation de phénomènes naturels simulés par système physique particulaire.** Thèse de l'INPG, Grenoble 2002. 195 pages.
- [GW95] Guy A., Wyvill B. **Controlled blending for implicit surfaces using a graph.** Dans *Proc. of Implicit Surfaces '95*, pp 107-112, 1995.
- [HBG92] House D.H., Breen D.E., Getto P.H. **On the Dynamic Simulation of Physically-Based Particle-System Models.** In Hegron and Thalmann, *Third Eurographics Workshop on Animation and Simulation Proceedings*, Cambridge, UK, September 1992.
- [HL02] Habibi A., Luciani A.. **Dynamic Particle Coating.** Dans *IEEE Transactions on Visualisation and Computer Graphics*, vol. 8, no. 4, Octobre-December 2002, pp 383-394.

- [HL05] Hsieh, C., Luciani, A., **Physically-based particle modeling for dance verbs.** Dans Proceedings of Graphicon'05 20-24, Russie, vol. 1 (2005) pp.85-92, juin 2005.
- [HLT*03] Heïgeas, L., Luciani, A., Thollot, J., and Castagné, N. **A physically-based particle model of emergent crowd behaviors.** Dans International Conference Graphicon 2003, pp 1-9, septembre 2003.
- [HLV96] Habibi, A., Luciani, A., Vapillon, A. **A physically-based model for the simulation of reactive turbulent objects.** Dans Proceedings of WSCG'96, Plzen (Czech Republic), vol. 1, pp.113-122, 1996.
- [HM01] Hadap S., Magnenat-Thalmann N. **Modeling dynamic hair as a continuum.** Dans *Computer Graphics Forum*, volume 20, pp 329–338, Manchester, UK, EUROGRAPHICS'01, Blackwell Publishers, septembre 2001.
- [HML94] Habibi A., Manzotti E., Luciani A. **Modelling, Simulating, and Visualizing Granular Materials.** Dans 5th. *Eurographics Workshop on Animation and Simulation*, Gérard Hégron and Olov Fahlander, Oslo, 1994.
- [HMT94] Huang Z., Magnenat-Thalmann N., Thalmann D. **Interactive human motion control using a closed-form of direct and inverse dynamics.** Dans *Proc. of the 2nd Conference on computer graphics*, Beijing, China, Pacific Graphics' 94, août 1994.
- [HPH96] Hutchinson D., Preston M., Hewitt T. **Adaptive refinement for mass/spring simulations.** Dans *Proceedings of the Eurographics workshop on Computer animation and simulation '96*, pp 31–45, Poitiers, France, EUROGRAPHICS, Springer-Verlag New York, Inc. septembre 1996.
- [Hsi06] Hsieh C. **Grammaire de Mouvements Dansés par Modélisation Physique: Utilisation pour la Composition Chorégraphique.** Thèse INP-G. Filière Art, Science, Technologie. 12 Septembre 2006.
- [Inc96] Incerti E. **Synthèse de sons par modélisation physique de structures vibrantes. Applications pour la création musicale par ordinateur.** Thèse de l'Institut National Polytechnique de Grenoble, Grenoble, 249 pages, 1996.
- [Jim93] Jimenez S. **Modélisation et simulation physique d'objets volumiques déformables complexes.** Thèse de l'Institut National Polytechnique de Grenoble, 225 pages, Grenoble 1993.
- [JL93] Jimenez S., Luciani A. **Animation of interacting objects with collisions and prolonged contacts.** Dans *Modeling in computer graphics---methods and applications*, Springer-

- Verlag, B. Falcidieno and T. L. Kunii, Eds., Proc. of the IFIP WG 5.10 Working Conference, 129—141, 1993.
- [Jun97] Juneidi Z. **Du fluide au sable : recherche d'un modèle rendant compte des états intermédiaires de la matière.** 1997/06/24, ENSIMAG, GRENOBLE, 61 pages, 1997.
- [Kas95] Kass M. **An introduction to physically based modeling, introduction to continuum dynamics for computer graphics.** In *SIGGRAPH Course Notes*, 1995.
- [KNB03] Kacic-Alesic Z., Nordenstam M., Bullock D. **A practical dynamics system.** Dans *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer animation*, pp 7–16 and 361, San Diego, California, Eurographics/SIGGRAPH'03, Eurographics Association (Aire-la-Ville, Switzerland), juillet 2003.
- [LB07] Luciani A., Barthélémy D. **Projet ARCAD, Rapport Final.** Délégation aux Arts Plastiques. Ministère de la Culture et de la Communication. 27 septembre 2007.
- [LC84] Luciani A., Cadoz C. **Modélisation et animation gestuelle d'objets - Le système ANIMA.** CESTA - 1er Colloque Image, pp.183-189, Biarritz 1984.
- [LC86] Luciani A., Cadoz C. **Utilisation de modèles mécaniques et géométriques pour la synthèse sonore et le contrôle d'images animées.** CESTA, Nice 1986
- [LCF98] Luciani A., Cadoz C., Florens J.-L.. **Towards a complete representation by means of computer – the instrumental communication interface concept.** Dans *Cyberworld*, Springer-Verlag, Tokyo, 1998
- [LHM95] Luciani A., Habibi A. and Manzotti E. **A multi-scale physical model of granular materials.** Dans *Proceedings of Graphics Interface'95*, 1995.
- [LG97] Luciani A., Godard A. **Simulation of Physical Object Construction Featuring Irreversible State Changes.** MICAD. 10 pages, février 1997.
- [LGo97] Luciani A., Godard A. **Construction d'objets physiques obtenus par assemblage physique : modèles et animation.** MICAD'97 1997, vol. 12, pp.111-129, 1997.
- [LHV*95] Luciani A., Habibi A., Vapillon A. and Duroc Y. **A Physical Model of Turbulent Fluids,** 6th Eurographics Workshop on Animation and Simulation, Springer Verlag, Maastricht, pp. 16-29, Series, Computer Science, 1995.
- [LJC*91] Luciani A., Jimenez S., Cadoz C., Florens J.-L., Raoult O. **An Unified View of Multitude Behaviour, Flexibility, Plasticity and Fractures Balls, Bubbles and Agglomerates.**

- Modeling in Computer Graphics, Kunii, T.L., Springer Verlag, Tokyo, Proceedings of the IFIP WG 5.10 Working Conference, 1991.
- [LJF*91] Luciani A., Jimenez S., Cadoz C., Florens J.-L., Raoult O.. **Computational physics: a modeler simulator for animated physical objects.** *Proceedings of the European Computer Graphics Conference and Exhibition.* Eurographics'91, pp 425-436, Vienna, Austria, Elsevier Ed, septembre. 1991.
- [LJR*91] Luciani A., Jimenez S., Raoult O., Cadoz C., Florens J.-L. **An unified View of Multitude Behaviour, flexibility, plasticity and fractures balls, bubbles and agglomerates.** Proceedings of the IFIP WG 5.10 Working Conference. Tokyo, avril 1991
- [Luc00] Luciani, A. **From granular avalanches to fluid turbulences through oozing pastes. A mesoscopic physically-based particle model.** Moscou, 8 pages, juin 2000.
- [Luc04] Luciani A. **Why and How are Physically-based model able to model Emergent Crowd Behavior ?** Invited paper. Proc. of 3IA conference. Limoges (France). 2004/05.
- [Luc93] Luciani A. **Ordinateur, images et mouvement. Apparition de l'instrumentalité dans l'art du mouvement visuel.**, IRCAM, Centre George Pompidou, pp.175-189, 1993.
- [Mar09] Cherkaoui M. **Typologie du mouvement et modèle physique Cordis-Anima.** Rapport de stage de master Art, Science, Technologie. Grenoble., juin 2009.
- [Maya04] **Learning maya 6 dynamics.** Alias, 210 King Street East, Toronto, Canada M5A 1J7, ISBN : 1-894893-69-7, mai 2004.
- [MCG03] M. Müller, D. Charypar, and M. Gross. **Particle-based fluid simulation for interactive applications.** Dans *Symposium on Computer Animation, Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer animation*, pp 154–159, Suisse, SIGGRAPH'03, The Eurographics Association 2003 (Aire-la-Ville), Juillet 2003.
- [MDDB01] Meyer M., Debunne G., Desbrun M., Barr A. H. **Interactive animation of cloth-like objects in virtual reality.** Dans *Journal of Visualization and Computer Animation* 12, 1 1–12, 2001.
- [ME98] Meißner M., Eberhardt B. **The art of knitted fabrics, realistic and physically based modelling of knitted patterns.** Dans *Computer Forum 98*, volume 17. EUROGRAPHICS'98, Blackwell Publishers (108 Crowley Road, Oxford OX4 1JF, UK and Main Street, Malden, MA 02148, USA), septembre 1998.
- [Mil88] Miller G. S. P. **The motion dynamics of snakes and worms.** Dans *Computer Graphics*, volume 22, pp 169–178. SIGGRAPH'88, ACM Press (New York, USA), August 1988.

- [MKN*04] Müller M., Keiser R., Nealen A., Pauly M., Gross M., Alexa M. **Point based animation of elastic, plastic and melting objects.** Dans *Eurographics/ACM SIGGRAPH Symposium on Computer Animation (2004)*, pp 141–151. ACM Press (New York, USA), août 2004
- [MM00] Meredith, Maddock. **Motion capture file formats explained.** Département of computer Science, University of Sheffield, 2000
- [Mor85] Morrison. **EA IFF 85 - Standard for interchange format files.** Document de spécification Electronic Arts, janvier 1985
- [MP89] Miller G. S. P., Pearce A. **Globular dynamics : a connected particle system for animating viscous fluids,** Dans *Computers & Graphics*, volume 23(3), pp169–178. Elsevier, 1989.
- [MT95] Magnenat-Thalmann N., Thalmann D. **Finite elements in task-level animation.** Dans *Finite Elements in Analysis and Design*, volume 19(5), pp227-242, Elsevier, mai 1995.
- [MW88] Moore M., Wilhelms J. **Collision detection and response for computer animation.** *Computer Graphics*, vol. 22(4), pages 289-298. SIGGRAPH'88, ACM Press (New York, USA), août 1988
- [NCL94] Nouri J., Cadoz C., Luciani A. **The Physical Modelling of Complex Physical Structures. The Mechanical Clockwork.** Dans *5th. Eurographics Workshop on Animation and Simulation*, Gérard Hégron and Olov Fahlander, Oslo, 1994
- [NFD95] Neveu M., Faudot D., Derdouri B. **Superquadriques-B-déformables pour la reconstruction 3D.** Dans *Technique et Science Informatiques*, 14(10) :1291-1314, Octobre 1995.
- [NFJ02] Nguyen D. Q., Fedkiw R., Jensen H. W. **Physically based modeling and animation of fire.** Dans *Computer Graphics Proceedings, Annual Conference Series, 2002*, pages 721–728, San Antonio, Texas, USA, SIGGRAPH'02, ACM Press (New York, USA), juillet 2002.
- [NT98] Nedel L. P., Thalmann D. **Real time muscle deformations using mass-spring systems.** In *Proceedings of the Computer Graphics International 1998*, pp 156–166, Hanover, Germany, June 1998. CGI'98, IEEE Computer Society (Washington DC, USA).
- [NTB*91] Norton A., Turk G., Bacon B., Gerth J., Sweeney P. **Animation of fracture by physical modeling,** July 1991.
- [OBH02] O'Brien J. F., Bargteil A. W., Hodgins J.-K. **Graphical modeling and animation of ductile fracture.** Dans *Computer Graphics Proceedings, Annual Conference Series, 2002*,

- pp 291–294, San Antonio, Texas, USA, SIGGRAPH’02, ACM Press (New York, USA), Juillet 2002.
- [OH99] O’Brien J. F. Hodgins J. K.. **Graphical modeling and animation of brittle fracture.** Dans *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, Los Angeles, USA, August 1999. ACM SIGGRAPH’99, ACM Press/Addison-Wesley Publishing Co. (New York, NY, USA).
- [Oro98] O'Rourke J. *Computational Geometry in C*. Cambridge University Press, 2nd edition, 1998.
- [PB88] Platt J.C., Barr A.H.. **Constraint methods for flexible models.** *Computer Graphics*, vol. 22(4), pp 279-288, SIGGRAPH’88, ACM Press (New York, USA), août 1988.
- [PGM*98] Peroche B., Ghazanfarpour D., Michelucci D., Roelens M. Informatique graphique : méthodes et modèles. Editions HERMES, Paris, 413 pages, 1998.
- [Pie91] Piegl L. **On NURBS: A Survey.** Dans *IEEE Computer Graphics and Applications*, Vol. 11, No. 1, pp. 55-71, janvier 1991.
- [PL99] Pommereuil G., Luciani A. **Un langage de modélisation des comportements complexes de la matière pour la construction d'objets physiques simulables**, HERMES, Paris (Palais des expositions), pp.75-90, février 1999.
- [PPD07] Paris S., Pettré J., Donikian S. **Pedestrian Reactive Navigation for Crowd Simulation: a Predictive Approach.** Dans *Computer Graphics Forum*, Vol. 26, N° 3, Eurographics’07 Conference Proceedings, p. 665-674, septembre 2007.
- [Pro95] Provot X. **Deformation constraints in a mass-spring model to describe rigid cloth behavior.** Dans *Proceedings of Graphics Interface 1995*, pp 147–154, Quebec City, Canada, Graphic Interface’95, mai 1995.
- [PSE*00] Popovic J., Seitz S. M., Erdmann M., Popovic Z., Witkin A. **Interactive manipulation of rigid body simulations.** Dans *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp 209–217, New Orleans, Louisiana, SIGGRAPH’00, ACM Press (New York, USA), juillet 2000.
- [PT92] Payne B.A. Toga A.W. **Distance field manipulation of surface models.** Dans *IEEE Computer Graphics and Applications*, 12(1), pp 65-71, 1992.
- [PW99] Popovic Z., Witkin A. **Physically based motion transform.** Dans *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, Los Angeles,

- USA. ACM SIGGRAPH'99, ACM Press/Addison-Wesley Publishing Co. (New York, NY, USA) août 1999.
- [Raz86] Razafindrakoto A., **Le système ANIMA : éditeur d'objets producteurs d'images; implantation d'algorithmes de simulation temps réel.** Université scientifique, technologique et médicale de Grenoble et institut national polytechnique de Grenoble, 128 pages, Grenoble, 1986.
- [Ree83] Reeves W. T.. **Particle systems : a technique for modeling a class of fuzzy objects.** Dans *Proceedings of the 10th annual conférence on computer graphics and interactive techniques*, volume 2, pp 91–108, Detroit, Michigan, SIGGRAPH'83, ACM Press (New York, USA), avril 1983.
- [Rey87] Reynolds C. W.. **Computer Animation with scripts and actors.** *Computer Graphics*, volume 16 (3), pp 229–296, SIGGRAPH'82, ACM Press (New York, USA).
- [Rey87] Reynolds C. W. **Flocks, herds, and schools: a distributed behavioral model.** Dans *Computer Graphics*, volume 21, pages 25–34, Anaheim, California,. SIGGRAPH'87, ACM Press (New York, USA), juillet 1987.
- [Rey97] Hugh Reynolds. **An Alternative Inter-Particle Force Model for Coupled System Flexible Body Dynamics.** Dans *8th Eurographics Workshop on Animation and Simulation*, Budapest, Hongrie, septembre 1997.
- [RSG*08] Robinson-Mosher A., Shinar T., Gretarsson J., Su J., Fedkiw R. **Two-way coupling of fluids to rigid and deformable solids and shells.** Dans *ACM SIGGRAPH 2008*, article n°46, 2008.
- [RV82] Requicha A.A.G., Voelcker S.B. **Solid Modeling: A Historical Summary and Contemporary Assessment.** Dans *IEEE_CGA(2)*, No. 2, March 1982, pp. 9-24.
- [SGS96] Sabourdy E., Gaildrat V. et Stolte N. **Constructive Solid Geometry using implicit functions.** *Proc. of CADEX '96*, pp 86-92, 1996.
- [Sha00] Sharp B. **Subdivision Surface Theory.** Gamasutra, April 2000.
- [Sim90] Sims K. **Particle Animation and Rendering Using Data Parallel Computation.** In *Computer Graphics (SIGGRAPH'90 proceedings)*, pp 405-413, ACM Press (New York, USA), août 1990.
- [SP86] Sederberg T. W., Parry S. R. **Free-form deformation of solid Geometric Models.** Dans *Computer Graphics (Proc of SIGGRAPH 90)*, vol 20, 151-160. 1990.

- [Sta02] Stanciulescu B. **Identification de modèles physiques et de contrôleurs en animation.** Rapport de thèse de l'INP-Grenoble. 2002.
- [Sta99] Stam J. **Stable fluids.** Dans *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 121–128, Los Angeles, California. SIGGRAPH'99, ACM Press (New York, USA), août 1999.
- [SW84] Smyth, Wing. **The psychology of Human movement.** Académie Press, 1984
- [TF88] Terzopoulos D., Fleischer K. **Deformable models.** Dans *The Visual Computer*, 4:306–331, 1988.
- [TF'88] Terzopoulos D., Fleischer K. **Modeling inelastic deformation: viscoelasticity, plasticity, fracture.** Dans *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, volume 22, pp 269–278, Atlanta, Georgia, SIGGRAPH'88, ACM Press (New York, USA), août 1988.
- [TKH*04] Teschner M. , Kimmerle S., Heidelberger B., Zachmann G., Raghupathi L., Fuhrmann A., Cani M.-P., Faure F., Magnenat-Thalmann N. , Strasser W., Volino P.. **Collision detection for deformable objects.** Dans *Proceedings of Eurographics 2004, State of the Art*, pages 119–139. EUROGRAPHICS'04, The Eurographics Association 2004, septembre 2004.
- [Ton91] Tonnesen D. **Modeling liquids and solids using thermal particles.** *Graphics Interface'91*, pp 255-262, 1991.
- [Ton92] Tonnesen D. **Spatially coupled particle systems.** Invited chapter in SIGGRAPH'92 Course Notes #16 : Particle System Modeling, Animation, and Physically Based Techniques, ACM, Chicago, Illinois, août 1992.
- [Ton98] Tonnesen D. **Dynamically coupled particle systems for geometric modeling, reconstruction, and animation.** PhD thesis, University of Toronto, Department of Computer Science, Toronto, Canada, novembre 1998.
- [Ton01] Tonnesen D. **Particle systems for artistic expression.** Dans *Proc. of Subtle Technologies Conference 2001*, pp 17–20, Toronto, Canada, Subtle Technologies, mai 2001.
- [TPB*87] Terzopoulos D., Platt J., Barr A., Fleischer K. **Elastically deformable models.** Dans *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, volume 21, pp 205–214, Anaheim, California, SIGGRAPH'87, ACM Press (New York, USA), juillet 1987.

- [TPB*89] Terzopoulos D., Platt J., Barr A., Zeltzer D., Witkin A., Blinn J. **Physically-based modeling: past, present and future.** Dans *ACM SIGGRAPH 89 Panel Proceedings*, volume 23, pp 191–209, Boston, Massachusetts, SIGGRAPH'89, ACM Press (New York, USA), août 1989.
- [TPF*89] Terzopoulos, D., Platt, J. Fleisher K. **Heating and melting deformable models : from goop to glob.** Graphics Interface'89, 219-226, 1989.
- [TPK89] Terzopoulos D., Platt J., Fleisher K. **Heating and melting deformable models : from goop to glob.** In *Graphics Interface '89*, pp 219-226. 1989.
- [TW88] Terzopoulos D., Witkin A. **Physically based model with rigid and deformable components.** Dans *IEEE Computer Graphics and Applications*, pp 41-51, décembre 1988.
- [VEF06] Vollrath F., Emile O., Floch A. **Shape memory in spider draglines.** Dans *Nature*, « Zoology and wildlife conservation ». Macmillan Publishing Ltd. ISSN 0028-0836. 2006
- [WBK01] Witkin A., Baraff D., Kass M. **Physically based modeling.** Dans SIGGRAPH 2001, COURSE NOTES #25, août 2001.
- [Wer93] Wernecke J. **The Inventor Mentor: Programming Object-Oriented 3d Graphics with Open Inventor**, Release 2. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1993.
- [WK88] Witkin A., Kass M. **Spacetime constraints.** Dans *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 159–168, Atlanta, Georgie, USA, SIGGRAPH'88, ACM Press (New York, USA), août 1988.
- [WW89] Wyvill B., Wyvill G. **Field functions for implicit surfaces.** Dans *The Visual Computer*, 5, pp 75-82, 1989.
- [ZB05] Zhu, Y., and Bridson, R. **Animating sand as a fluid.** Dans *ACM Transactions on Graphics*, J. C. Hart, Ed., vol. 24, SIGGRAPH'05, ACM Press (New York, USA), pp. 965–972, Juillet 2005.

Publications personnelles

- [AEL06] Allaoui A., Evrard M., Luciani A. Le contrôle paramétrique, un outil de modélisation pour les mouvements évolutifs complexes et l'autonomie, In Les 19ème journée de l'Association Française d'Informatique Graphique, AFIG06, Bordeaux, LaBRI, pp153-160, novembre 2006.
- [AEL07] Allaoui A., Evrard M., Luciani A. On-line physically-based control of mass-interaction models. Dans Proceedings of Enactive 07. Grenoble. 19-21 novembre 2007.
- [ECL*] Evrard M., Castagné N., Luciani A., Cadoz C. Musical gestures and motion format in VR multisensory context. Dans proceedings of Music and Gesture 2006, Manchester, juillet 2006.
- [EL07] Evrard M., Luciani A. Gesture and motion (encoding of). Dans Enaction and Enactive Interfaces, a Handbook of Terms, A. Luciani and C. Cadoz eds. Enactive System Books, ISBN 978-2-9530856-0-48. pp. 116-118, 2007.
- [EL'07] Evrard M., Luciani A. Physically-based modelling techniques for movement synthesis and animation. Dans Enaction and Enactive Interfaces, a Handbook of Terms, A. Luciani and C. Cadoz eds. Enactive System Books, ISBN 978-2-9530856-0-48. pp. 240-242, 2007.
- [ELC06] Matthieu Evrard, Annie Luciani, Nicolas Castagné, MIMESIS: Interactive Interface for Mass-Interaction Modeling, In Proceedings of CASA 2006, Genève, Nadia Magnenat-Thalmann & al. editors, pp 177-186, juillet 2006. Publié aussi dans GTAS 2006, Forum AFIG, Toulouse, juillet 2006.
- [LAE*06] A. Luciani, A. Arliaud, M. Evrard, N. Castagné. Manuel Utilisateur MIMESIS v4.02a. MIMESIS, un environnement pour la synthèse de mouvement par modèle physique et la réalisation d'images animées. ACROE, 11 décembre 2006.
- [LEC*06] Luciani A., Evrard M., Castagné N., Couroussé D., Florens J.-L., Cadoz C. A Basic Gesture and Motion Format for Virtual Reality Multisensory Applications. Dans Proceedings of the 1st international Conference on Computer Graphics Theory and Applications, ISBN: 972-8865-39-2, Setubal (Portugal), march 2006. Publié aussi dans ENACTIVE Workshop 2006.
- [LTE06] Luciani A., Thill F., Evrard M. Mass-Interaction Model of Emergent Collective Phenomena. Dans Proceedings of CASA 2006, Geneva, Nadia Magnenat-Thalmann & al. editors. pp 197-206, juillet 2006.

- [SEL07] Sillam K., Evrard M., Luciani A. A real-time implementation of the dynamic particle coating method on a GPU architecture. Dans Proceedings of VRIPHYS 07, Dublin, 9 novembre 2007.
- [SEL07] Sillam K., Evrard M., Luciani A. Implantation temps-réel sur GPU d'un écran d'épingles dynamique. Dans les actes du GTAS 2007, Lyon, 25-26 juin 2007.

Sites Internet

- [Sodaplay] www.sodaplay.com,
 [Havok] www.havok.com,
 [Renderware] <http://www.renderware.com>,
 [PhysX] : www.ageia.com
 [IPhys] <http://www.design-simulation.com/>
 [Phun] <http://www.phunland.com/>
 [AOA,BVH,HTR] <http://www.cs.wisc.edu/graphics/Courses/cs-83-1999/Jeff/{AOA,BVH,HTR}.html>
 [ASF] <http://www.darwin3d.com/gamedev/acclaim.zip>
 [BRD,CSM]<http://www.dcs.shef.ac.uk/~mikem/fileformats/{brd,csm}.html>
 [C3D] <http://www.c3d.org>
 [Par] <http://parissebastien.free.fr/>

RESUME

Cette thèse a pour objet la conception d'un environnement pour l'animation à l'aide de réseaux masses – interactions CORDIS-ANIMA. La modélisation physique permet de produire des mouvements complexes que même un animateur confirmé ne serait pas capable d'obtenir avec les méthodes traditionnelles d'animation par positions « clés ». Parmi ces méthodes, les réseaux masses - interactions comme le formalisme CORDIS-ANIMA sont des méthodes génériques qui sont capables de simuler un très large éventail de comportements dynamiques, des comportements de solides et de solides déformables, aux comportements collectifs, effets de turbulence, fractures, etc. De plus, CORDIS-ANIMA a la particularité d'être un formalisme modulaire qui propose des paramètres significatifs dans une approche constructiviste de la modélisation de mouvement. Or, à l'heure actuelle, il n'existe aucun environnement de conception interactif pour ce genre de formalisme.

La généricité et la modularité de CORDIS-ANIMA ayant largement prouvé leur intérêt pour l'animation depuis 25 ans, la conception et l'implantation d'un environnement interactif pour la conception de tels modèle doivent faire face à d'autres problématiques théoriques et pratiques qui seront discutées dans ce manuscrit, dans le but d'inclure dans cet environnement toutes les fonctionnalités requises pour une création interactive de modèles de mouvement et leur insertion dans une chaîne global de production d'images animées.

Cette thèse se terminera par le compte-rendu de situations d'utilisation dans un cadre pédagogique ainsi que de recherche et de création permettant de valider ces travaux

Mots clés : modèle physique, modèle particulière, réseau masses - interactions, CORDIS-ANIMA, simulation, animation, environnement logiciel interactif, habillage de modèles particuliers, mouvement, forme, geste.

SUMMARY

This thesis deals with the design of a computer framework dedicated to animation by the physical mass-interaction CORDIS-ANIMA networks. Physically-based modeling allows producing complex movements that a skilled animator won't be able to obtain with the traditional keyframe methods. Among those methods, mass-interaction network like CORDIS-ANIMA are generic methods that are able to simulate a huge panel of dynamical behavior from rigid or deformable behaviors to collective behaviors, turbulent effects, fractures, etc. Far more, CORDIS-ANIMA is a modular formalism that exhibit parameters easy to understand and proposes a constructive approach of movement modeling. But, surprisingly, no interactive framework exists today to help the conception of such networks.

Genericity and modularity of CORDIS-ANIMA having been still largely proved, the design and the implementation of such a framework have to face with other theoretical and practical problems that are discussed here in order to include every function that are required for an interactive creation of models and the communication inside a global chain of production of animated pictures

This thesis ends on the report of various situation of use in pedagogical, research and creation contexts to validate those works.

Keywords : physically-based modeling, particles models, mass-interaction networks, CORDIS-ANIMA, simulation, animation, interactive framework, coating of particles models, movement, shape, gesture.

Laboratoire ICA
Institut Polytechnique de Grenoble
46 avenue Félix Viallet
38031 GRENOBLE CEDEX