HAL
open science

Représentation géométrique multi-échelles de l'architecture des plantes

Frédéric Boudon

To cite this version:

Frédéric Boudon. Représentation géométrique multi-échelles de l'architecture des plantes. Modélisation et simulation. Université Montpellier II - Sciences et Techniques du Languedoc, 2004. Français. NNT : . tel-00552120

HAL Id: tel-00552120

https://theses.hal.science/tel-00552120

Submitted on 5 Jan 2011

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.

UNIVERSITE MONTPELLIER II
SCIENCES ET TECHNIQUES DU LANGUEDOC

THESE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE MONTPELLIER II

Formation Doctorale	:	Informatique
Ecole Doctorale	:	Information, Structures, Systèmes

par

Frédéric Boudon

Représentation géométrique multi-échelles de l'architecture des plantes

Soutenue publiquement le **27 Février 2004**

devant le jury composé de :

Michel Habib	Président
Marie-Paule Cani	Rapporteur
Hervé Sinoquet	Rapporteur
Christophe Godin	Co-Directeur de thèse
Guy Melançon	Directeur de thèse
Przemyslaw Prusinkiewicz	Co-Directeur de thèse
Daniel Barthélémy	Examinateur invité

Remerciements

Je souhaiterais en premier lieu remercier Michel Habib d'avoir accepté d'être le président de mon jury de thèse.

Un grand merci ensuite à Marie-Paule Cani et Hervé Sinoquet d'avoir accepté d'en être les rapporteurs et d'avoir contribué à l'amélioration de ce document par leurs critiques avisées et constructives.

Je remercie également Daniel Barthélémy qui a bien voulu se joindre à ce jury. Son caractère enjoué, un peu « marseillais » ;-), sa gentillesse en font une personne précieuse. Il est aussi d'une efficacité redoutable. C'est lui en effet qui m'a trouvé mon post doc. Un grand merci pour tout cela.

François Houllier a occupé avant Daniel le poste de directeur du laboratoire AMAP. C'est grâce à son énergie et son soutien que j'ai pu obtenir mon financement de thèse et celui de mon voyage au Canada. Je lui en suis plus que reconnaissant.

Guy Melançon a accepté dès le début d'être le directeur de cette thèse. Nos échanges, bien que peu fréquents, ont été fort constructifs. Il m'a été d'un aide très précieuse pour déjouer les pièges administratifs. Pour tout cela, merci beaucoup.

J'ai eu la chance d'avoir deux encadrants de thèse exceptionnels : Christophe Godin et Przemek Prusinkiewicz. Beaucoup de points communs les relient : gentillesse, rigueur scientifique, débordant d'idées, exigence ... la liste complète de leurs qualités serait trop longue. Si cette thèse est ce qu'elle est, c'est en grande partie grâce à eux : Merci pour tout. Leurs compagnes respectives furent aussi d'un soutien et d'une aide plus que précieux. Elles ont leur part de responsabilité dans l'aboutissement de cette thèse, aussi bien culinaire que scientifique. Un grand merci à Evelyne et à Lynn.

Je remercie ensuite les laboratoires AMAP et Graphic Jungle d'avoir été des environnements chaleureux et inspirants pour tout ce travail. Certaines personnes m'ont particulièrement aidé dans l'élaboration des idées présentées dans cette thèse.

Je tiens en premier lieu à remercier Christophe Pradal, pour son aide précieuse sa gentillesse, son écoute et ses éclairages sur différents sujets. J'associe à ces remerciements Marie René et toute la petite famille Pradal.

Pavol et Frank were wonderfull people that I met in Canada. Numerous debates took place just beyind the university buildings, sometimes in the cold of the snowing winter of Calgary, smoking few cigarettes and chatting a lot about GUI, openGL, design of plants or how goods the steaks at the Keg are. Thank a lot guys for that. Nos destins avec Frank semblent irrémédiablement liés. Il me tarde de voir ce que nous réserve la suite. Bonne route à toi en attendant.

A l'AMAP, je tiens à remercier ceux qui par leurs remarques constructives m'ont aussi fait avancer : Yann, Yves, Philippe, Marc, Eric, Daniel Auclair, Jean Baptiste, etc. Sylvie Beauclair a été une agréable compagne de bureau durant ma troisième année. Un grand merci aussi à Marie-Hélène, Nora, Sylvie, Perinne et tout le personnel administratif de l'AMAP.

Les anciens thésards m'ont prodigué nombres conseils et encouragements (« Ne fais jamais de thèse ! », « Au début, c'est facile ; à la fin, c'est l'horreur ! »). Un grand merci à Pascal et Elise (et toute la famille), Patrick, Christophe et Laetitia, Javier, Tancrède, etc. Je suis

votre exemple et donne moi-même ces conseils aux nouveaux thésards ;-) Merci donc à Carine, Olivier, Pierre, Céline, Céline, Claude-Eric, Jérome qui ont fait semblant d'écouter et d'avoir peur quand j'évoquais ces mêmes conseils. Plus sérieusement, j'ai eu avec ces gens de nombreuses discussions qui m'ont énormément aidé à avancer.

Going in an english speaker foreign country for 8 month is a little bit perturbing for a frog eater like me. Success of thoses trips were due to the friendly environnement of the Graphics Jungle Lab. Thanks and Hi to Radek, Martin, Peter, Richard, Blob, Jennifer, Brendan, Colin, Lars and all the others. A particular thought to my roomates Kelly, Greg and all the others for their kindness and tolerance for my french accent ;-). Merci aussi à Laurent et Vero qui ont bien voulu m'accueillir pour mes premiers jours dans le « grand nord ».

De manière peu originale mais particulièrement sincère, je voudrais remercier mes parents. Ils m'ont soutenu jusqu'au bout, moralement, financièrement, ... Ils étaient là pour tous les moments difficiles. Je pense que peu de personnes mettent autant d'ardeur à tenir leur rôle de parents, sans pour autant étouffer leurs enfants le moindre instant. Ils ont droit à ma reconnaissance éternelle. A ces remerciements, je voudrais associer mon frère et ma soeur. Sans trop exagérer et à quelques détails près, mon frère est le meilleur acteur du monde tandis que ma soeur fut championne du monde de basket ;-) En tout cas, une chose est sûre, je crois en vous ! Finalement, je salue et remercie globalement toute ma famille. Merci aussi à Pierre et Corinne pour le somptueux buffet du pot de thèse.

Je voudrais ensuite saluer et remercier tous les amis de ma vie hors de la thèse (si, si ! ça existe), mais qui d'une certaine manière y ont contribué aussi.

Merci à Sab.

Dans le désordre, merci à Peter, Freddy, Cécile, Laeti, Fanggy, Don Fab, Rafa, Leo, Armelle, Sami, Frizouille, Willy, Toto, Remi, Piouk, Guillaume, Luc, Gillian, Clara, Zef, Claire, Olive des montagnes, Julianne, tout le Lambda Vibes, tout le Saint Sat Crew Faya, au Piss 4 the Piss, tous les Chourmos, le chiffre Zouz, Mathieu, Vally, Yann, Nono, toute la promo IG 2000 de l'Isim, etc. Ils sont plus que onze ;-). Je pourrais, je pense, remplir un annuaire. Citer leurs qualités prendraient au moins une thèse entière (la prochaine ?). Je pense à vous.

J'ai très certainement oublier des gens. J'en suis désolé et frustré d'avance : je pense à vous aussi.

Je voudrais aussi remercier globalement les artistes, dont certaines personnes citées précédemment font parties, et plus particulièrement les dessinateurs de BD ;-), de nombreux groupes de musique et réalisateurs qui finalement ont aussi contribué indirectement à l'inspiration de cette thèse.

Une spéciale dédicace pour Lola et Enzo, la future génération de chourmos. On attend un ou deux ans avant de vous apprendre le foot et la drague ! Longue vie à vous.

Il y a cependant une personne que je ne remercie pas : l'inventeur de la grammaire et de l'orthographe qui m'ont mené la vie dure tout au long de la rédaction de ce document (quel idée ce 's' du pluriel ;-)) !

Finalement, la courageuse Céline, qui quand elle ne met pas au monde des nouveaux nés, s'occupe de corriger mes fautes, me soutenir, et me rendre la vie bien plus agréable. Je lui dois une reconnaissance éternelle.

Avant-Propos

En première approche, la géométrie d'une plante est un objet relativement complexe, combinant les géométries de nombreux différents organes, dépendant de l'échelle d'analyse considérée et très variable dans le temps ainsi que d'une espèce à l'autre. Malgré cette complexité et cette plasticité, la représentation géométrique des plantes se trouve être une préoccupation commune et centrale à tous les modèles dont l'objet est d'étudier l'interaction entre la plante et son environnement. Ils sont nombreux : en foresterie pour maîtriser la croissance des houppiers en fonction de l'environnement lumineux et de l'interaction avec d'autres plantes, en arboriculture pour développer de meilleures stratégies de conduite de vergers, ou étudier les déterminants environnementaux de la qualité des fruits à l'intérieur d'une couronne ; en phytopathologie et en entomologie pour analyser la propagation spatiale et temporelle de maladies ou le déplacement d'insectes à l'intérieur de la canopée ; en infographie pour visualiser des plantes et des écosystèmes de manière réaliste et/ou efficace ; *etc.*

Cette étude commence par l'analyse des différents types de représentation de modèles de plantes existantes. Cette analyse montre comment l'échelle considérée dans chaque approche de modélisation de la géométrie d'une plante conditionne le type de représentations utilisées. Des approches globales considèrent la plante comme un tout et la représentent par un modèle paramétrique plus ou moins complexe figurant le plus souvent son enveloppe. A des échelles plus fines, des approches détaillées utilisent une décomposition spécifique de la plante en modules d'un type particulier (*e.g.* entre-nœuds, unités de croissance, branches, *etc.*). Dans ce cas, la géométrie de la plante est représentée par l'ensemble de tous les modèles géométriques de ses composants.

Plusieurs travaux récents introduisent l'idée d'organiser des modèles de plantes à différentes échelles. En infographie, pour adapter la complexité de la représentation, certaines approches hiérarchisent l'information géométrique à différents niveaux de détails [Neyret, 1998, Meyer and Neyret, 2000, Meyer et al., 2001, Deussen et al., 2002, Soler et al., 2003], permettant un affichage rapide et efficace. Pour différents systèmes de simulation de la croissance des plantes, le problème de modéliser des aspects morphologiques ou physiologiques à différentes échelles de temps nécessite de considérer la structure de la plante à différentes échelles [Remphrey and Prusinkiewicz, 1997]. Dans le cadre de la simulation, pour des raisons de complexité, la génération d'un peuplement, par exemple, prenant en compte l'interaction entre individus ne peut se faire qu'à une échelle macroscopique. Des représentations détaillées de ce peuplement à un instant donné peuvent être calculées à partir des informations macroscopiques en utilisant des algorithmes multi-échelles [Prusinkiewicz et al., 2001]. De la même manière, dans le domaine de l'analyse de l'architecture des plantes, différents niveaux d'organisation doivent être considérés pour une meilleure compréhension du développement de la plante [Godin et al., 1997], *etc.* (voir Figure 0-1).

Afin d'unifier des représentations à différentes échelles, nous développons et formalisons un modèle géométrique multi-échelles général de représentation de l'architecture des plantes, et l'utilisons dans différents contextes. Ce modèle permet l'intégration cohérente, en un seul modèle, de différentes représentations géométriques des plantes à différents niveaux de détail. Ces représentations donnent des points de vue différents et complémentaires de l'organisation de la structure d'une plante. Ce modèle contenant intrinsèquement différents niveaux de détails

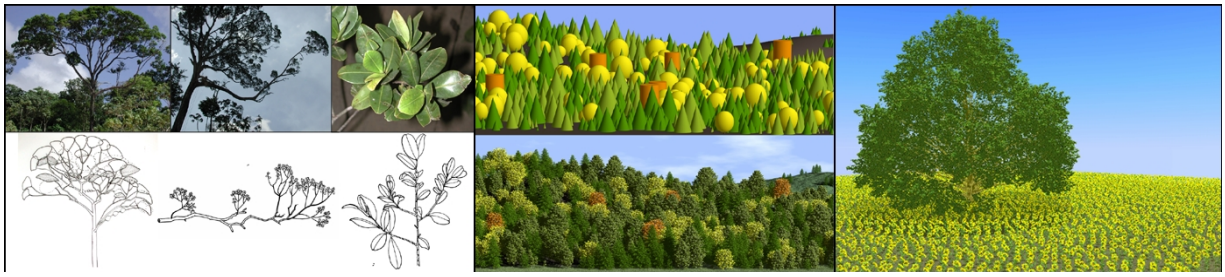


FIG. 0-1 – De gauche à droite, différents domaines de recherche abordant la géométrie des plantes tirent partie de son aspect multi-échelles. Pour la compréhension de la forme, pour les arbres aux couronnes élevées et de grande envergure par exemple, différentes échelles d'analyse sont considérées. Les échelles macroscopiques utilisent des enveloppes. Les représentations détaillées, plus locales, raffinent les descriptions globales (Photos et dessins de Y. Caraglio). Pour la simulation, chaque arbre d'un peuplement est représenté macroscopiquement par quelques grandeurs (*e.g.* diamètres, hauteurs). Le processus de simulation modélise l'interaction entre arbres et avec l'environnement en modifiant ces paramètres macroscopiques. Une représentation détaillée des arbres sera ensuite calculée en fonction des valeurs finales des paramètres en utilisant des algorithmes multi-échelles. Tiré de [Prusinkiewicz et al., 2001]. Finalement, pour le rendu de scènes végétales, l'utilisation de différentes primitives organisées hiérarchiquement en échelles permet d'obtenir un affichage interactif. Tiré de [Deussen et al., 2002]. Aucun formalisme commun n'existe cependant entre ces différentes représentations multi-échelles.

doit permettre de contrôler et d'adapter la complexité de la représentation au cours de certains calculs et une compréhension plus précise de la géométrie complexe des plantes.

Basé sur une première formalisation par Godin *et al.* [Godin et al., 1996], ce modèle s'appuie sur une description topologique (organisationnelle) multi-échelles d'une plante qui est étendue par des informations géométriques. Des modèles géométriques sont associés à toutes les entités définies à toutes les échelles. Une représentation de la plante à une échelle donnée est définie par la juxtaposition de tous les modèles géométriques représentant les entités à cette échelle. Dans ce modèle, une même entité peut être représentée à différents niveaux de détails. Aussi pour exprimer la cohérence de ces représentations, certaines règles peuvent être définies et formalisées sous forme de contraintes sur les paramètres des modèles géométriques utilisés, soit à une échelle donnée (contraintes intra-échelle), soit entre échelles (contraintes inter-échelles). Si une représentation à une échelle n'est pas définie, elle peut être inférée en utilisant les informations géométriques contenues dans une représentation à une autre échelle et les différentes contraintes de cohérence, intra et inter-échelles.

Nous explorons ensuite deux approches d'utilisation de ce modèle : une première approche, dite *montante*, qui permet, à partir d'informations géométriques fines, de calculer des représentations globales correspondant aux formes émergentes de la structure détaillée. Les formes végétales étant particulièrement irrégulières, il est souvent difficile de caractériser ces structures émergentes. Pour cela, un ensemble d'algorithmes d'approximation a été implémenté et utilisé sur différents types de plantes. Des outils de la géométrie fractale permettent de caractériser les résultats obtenus suivant le type d'algorithme et en fonction de l'information géométrique détaillée. Ces premiers outils ouvrent la voie pour l'étude de la géométrie de la plante à différentes échelles, et permettent une compréhension plus précise de l'occupation de l'espace par une plante. Le calcul de forme globale passe bien souvent par une simplification de la géométrie détaillée. Dans une perspective plus infographique, nous nous sommes ensuite intéressés à la simplification et au contrôle de cette simplification dans le cas de la représentation de système

ramifié.

Une deuxième approche, dite *descendante*, utilise des informations géométriques globales pour calculer des représentations fines. Particulièrement adaptée à la simulation, cette approche nous a amenés à développer un modèle multi-échelles de croissance des plantes, permettant une spécification des connaissances biologiques à différents niveaux d'organisation. En particulier, ce modèle permet de contraindre le processus de simulation de la croissance à produire une plante ayant une forme globale contrôlée à différentes échelles (*e.g.* plante entière, branches, rameaux). Ce travail a donné lieu à l'implémentation d'un ensemble logiciel autour des L-systèmes permettant la conception interactive et flexible de la géométrie d'une plante à différentes échelles.

Organisation du document

Ce document se structure en trois parties, chacune respectivement composée de quatre, deux et trois chapitres.

La première partie commence par une revue des différents formalismes utilisés dans la littérature pour représenter l'architecture des plantes. Le premier chapitre décrit les différents modèles géométriques utilisés pour des représentations globales des plantes, des plus simples (des profils) au plus complexes (des enveloppes convexes, asymétriques, ...). Les deux chapitres suivants abordent les représentations détaillées, le premier mettant l'accent sur les représentations topologiques et le second sur les représentations géométriques. Ces deux chapitres s'efforcent de montrer comment l'introduction dans les travaux récents de la notion d'échelle permet de hiérarchiser le modèle, autorisant ainsi une meilleure compréhension et une optimisation de son utilisation.

Le quatrième chapitre présente un modèle géométrique multi-échelles général de l'architecture des plantes et les différentes contraintes de cohérence qui le contrôlent. Grâce à ces contraintes, de nouvelles représentations, à de nouvelles échelles, peuvent être définies. Afin d'illustrer l'intérêt d'utiliser un modèle géométrique multi-échelles dans les applications de modélisation, nous avons développé deux approches montrant des aspects complémentaires de ces structures multi-échelles.

La première approche, présentée dans la deuxième partie, a pour but de caractériser les formes émergentes de la structure particulièrement intriquée des plantes. Au cinquième chapitre, nous présentons différentes méthodes pour calculer des représentations globales de ces structures par des enveloppes. Ces représentations nous permettent notamment de caractériser l'espace occupé par une plante. La géométrie d'une suite d'entre-nœuds peut aussi être représentée globalement par un cylindre généralisé plus ou moins complexe. Cette représentation globale introduisant une simplification de la géométrie détaillée, nous nous intéressons au sixième chapitre, au contrôle de cette simplification dans le cas de la représentation de système ramifié.

La troisième partie présente une deuxième approche d'utilisation du modèle géométrique multi-échelles dans le cadre de la simulation de la croissance des plantes. Ce travail a pour but de simplifier la création de modèles de végétaux en permettant une spécification des informations à différentes échelles. Le développement de cette idée nous a amenés à construire un éditeur interactif multi-échelles de structure végétale. Ce travail étant basé sur les L-systèmes, nous présentons ce formalisme et les extensions existantes pour l'édition interactive dans le septième chapitre. Nous présentons ensuite pour une étude comparative un des systèmes interactifs de modélisation de plante les plus aboutis et les plus détaillés dans la littérature scientifique, XFrog. Le dernier chapitre présente finalement notre système génératif interactif multi-échelles de plantes. L'introduction d'une hiérarchisation en échelles des informations géométriques permet notamment une minimisation du nombre de paramètres des modèles et un contrôle souple à différentes échelles de ces paramètres.

Finalement, cette thèse étant pluridisciplinaire, nous avons essayé de rendre clair et abordable par tous les différents concepts informatiques utilisés. Les lecteurs informaticiens trouveront peut

être certaines définitions un peu trop classiques pour être explicitées dans une thèse. Elles sont cependant souvent utiles pour des lecteurs de sensibilité plus biologique. De plus, l'annexe A contient un glossaire et quelques définitions botaniques basiques sur l'architecture des plantes.

Table des matières

Remerciements **i**

Avant-Propos **iii**

I Formalisation d'un modèle géométrique multi-échelles de l'architecture des plantes **1**

1 Représentations géométriques globales **5**
1.1 Représentations bidimensionnelles 5
1.1.1 Les profils 5
1.1.2 Les imposteurs 6
1.2 Enveloppes simples 7
1.2.1 Les modèles paramétriques 7
1.2.2 Les surfaces quadriques texturées 8
1.3 Enveloppes dédiées à la représentation de houppiers 8
1.3.1 Les enveloppes asymétriques 9
1.3.2 Les enveloppes extrudées 10
1.4 Enveloppes explicites 11
1.4.1 Les polyèdres convexes 11
1.4.2 Reconstruction d'enveloppes à partir de photographies 12
1.5 Conclusion 12

2 Représentations topologiques détaillées **15**
2.1 Modèles topologiques de l'architecture des plantes 15
2.1.1 Les arborescences 16
2.1.2 Les arborescences axiales 17
2.2 Représentations topologiques multi-échelles 18
2.2.1 Approches intégrant différents niveaux d'organisation 19
2.2.2 Analyse multi-échelles de l'architecture des plantes 21
2.2.3 Les arborescences quotientées 21
2.2.4 Les arborescences multi-échelles 22
2.3 Conclusion 24

3 Représentations géométriques détaillées **25**
3.1 Descriptions géométriques 25
3.1.1 Une approche pionnière 26
3.1.2 Les systèmes de particules 26
3.1.3 Description par voxels 27
3.1.4 Descriptions polygonales 30
3.1.5 Représentation géométrique des arborescences axiales 31

3.2 Continuité des représentations géométriques 33
3.2.1 Représentation par des primitives simples 34
3.2.2 Représentation par des modèles paramétrés 35
3.2.3 Les surfaces implicites 37
3.3 Représentations géométriques multi-échelles 40
3.3.1 Représentations fractales 40
3.3.2 Représentations hiérarchiques 43
3.4 Classification croisée 51
3.4.1 Modèles mono-échelle 51
3.4.2 Modèles multi-échelles 52
3.5 Conclusion 53

4 Modèle géométrique multi-échelles 55
4.1 Modèle géométrique d'une plante 55
4.2 Modèles géométriques élémentaires 56
4.2.1 Primitives géométriques 56
4.2.2 Transformations géométriques 57
4.2.3 Modèles géométriques composés 58
4.3 Modèle géométrique associé à un graphe 58
4.3.1 Modèle géométrique d'une plante 58
4.3.2 Exemples de contraintes intra-échelle 59
4.3.3 Utilisation des contraintes intra-échelle 62
4.4 Modèle géométrique associé à un graphe multi-échelles 64
4.4.1 La notion de graphe de scène 64
4.4.2 Les graphes multi-échelles 64
4.4.3 Modèle géométrique multi-échelles 65
4.4.4 Exemple de modèles multi-échelles et de contraintes inter-échelles 66
4.4.5 Utilisation des contraintes inter-échelles 70
4.5 Utilisation d'un modèle géométrique multi-échelles 71
4.5.1 Choix d'une représentation 71
4.5.2 Continuité inter-échelles 72
4.6 Conclusion 73

II Approche montante : Représentation et analyse de structures macroscopiques émergentes dans les architectures végétales 75

5 Caractérisation volumique de structures macroscopiques 77
5.1 Algorithmes d'inférence de géométrie macroscopique 77
5.1.1 Les algorithmes 78
5.1.2 Critères d'évaluation 82
5.2 Etude comparée des différents algorithmes d'inférence 82
5.2.1 Peuplement homogène équien d'eucalyptus 82
5.2.2 Ordre de grandeur des surfaces et volumes 83
5.2.3 Variabilité des surfaces et volumes 85
5.2.4 Corrélation des variations 86
5.3 Spécification d'une échelle macroscopique 88
5.4 Applications 91
5.4.1 Géométrie multi-échelles d'un eucalyptus en croissance 91
5.4.2 Dimension fractale de pêchers 94
5.5 Conclusion 100

6 Représentations géométriques compressées de systèmes ramifiés 101
6.1 Calcul d'une représentation simplifiée . . . 101
6.1.1 Choix d'un modèle . . . 101
6.1.2 Contrôler la complexité de la représentation . . . 102
6.1.3 Les modèles . . . 103
6.2 Calcul optimal : Modélisation par un problème de flot . . . 105
6.3 Une heuristique : ordonnancement des branches . . . 107
6.3.1 La sélection de modèle . . . 107
6.3.2 La procédure de distribution de la ressource K_0 . . . 109
6.4 Résultats . . . 109
6.5 Conclusion . . . 110

III Approche descendante : Design interactif de végétaux 113

7 L-systèmes et design interactif 115
7.1 Les principes généraux des L-systèmes . . . 115
7.1.1 Les D0L-systèmes . . . 115
7.1.2 Les D0L-systèmes parenthésés . . . 116
7.1.3 Les 0L-systèmes paramétriques . . . 117
7.1.4 Le symbole Coupe . . . 118
7.1.5 Les 0L-systèmes stochastiques . . . 118
7.2 Utilisation du contexte . . . 120
7.2.1 Les L-systèmes sensibles au contexte . . . 120
7.2.2 Le transfert rapide d'information . . . 122
7.3 Interaction avec l'environnement . . . 124
7.3.1 Les L-systèmes sensibles à l'environnement . . . 124
7.3.2 Les L-systèmes ouverts . . . 125
7.4 Règles de décomposition et Homomorphisme . . . 127
7.5 Environnement de modélisation . . . 127
7.6 Contrôle global . . . 128
7.6.1 Les topiaires . . . 129
7.6.2 Reconstruction à partir de photographies . . . 129
7.6.3 Utilisation de l'information positionnelle . . . 130
7.7 Modélisation interactive . . . 133
7.7.1 Modélisation continue . . . 133
7.7.2 Interaction avec le modèle : ilsa . . . 133
7.7.3 Interaction avec le modèle : le module prédéfini X . . . 134
7.8 Conclusion . . . 135

8 XFrog, un éditeur graphique interactif de modèles de plante 137
8.1 La méthode de modélisation . . . 137
8.1.1 Les différents types de composants . . . 137
8.1.2 Représentation par un graphe . . . 139
8.1.3 Un exemple . . . 140
8.2 Modélisation de forme naturelle . . . 141
8.2.1 Utilisation des exceptions . . . 141
8.2.2 Modélisation fonctionnelle. . . . 141
8.2.3 Tropismes . . . 142
8.2.4 Les déformations de formes libres . . . 142
8.3 Evaluation de l'interface . . . 143

8.4 Représentations géométriques des plantes 144
8.4.1 Réduction de la complexité du modèle géométrique 144
8.4.2 Représentation non-photoréaliste de plantes 144
8.5 Conclusion 145

9 Génération multi-échelles de modèles de plantes 147
9.1 Structure multi-échelles de la plante 148
9.1.1 Représentation de la plante 148
9.1.2 Les paramètres 149
9.1.3 Simulation multi-échelles 150
9.2 Navigation et édition d'une structure de plante 150
9.2.1 L'explorateur de graphes de décomposition 151
9.2.2 Les éditeurs de paramètres 152
9.2.3 L'explorateur de structures ramifiées 152
9.2.4 L'éditeur de silhouettes 153
9.3 Contraintes multi-échelles 153
9.4 Le processus de modélisation 155
9.4.1 Algorithme génératif 155
9.4.2 Le problème de découplage 157
9.5 Résultats 159
9.6 Conclusion 161

Conclusion 165

A Glossaire sur l'architecture végétale 179

B Symboles interprétés par la tortue 187

Première partie

Formalisation d'un modèle géométrique multi-échelles de l'architecture des plantes

Le petit monde du Golem. Joann Sfar. L'Association.

Introduction

Les représentations de l'architecture des plantes sont utilisées, de manière générale, pour modéliser la structure et le fonctionnement des plantes. Les langages de description et les objectifs à atteindre étant très différents d'une application à l'autre, un nombre important de représentations a été proposé, utilisant différents formalismes avec différentes propriétés. Mais bien que la notion d'architecture de la plante soit fréquemment utilisée dans la littérature, il n'existe pas réellement de définition acceptée par l'ensemble de la communauté scientifique. La signification de ce concept varie suivant le contexte.

D'après Hallé *et al.*, la phrase « architecture des plantes » est fréquemment utilisée pour se référer aux modèles architecturaux des especes de plantes, i.e. la description des modes de croissance d'un individu idéal d'une espèce. Dans ce contexte, l'architecture d'une plante fait référence à un ensemble de règles qui exprime la structure et la croissance de différentes plantes dans un groupe moyen sans condition limitante. Cependant, dans ce même contexte, le terme d'architecture peut également exprimer les processus de croissance d'un individu donné. Dans ce cas, il représente la structure tridimensionnelle d'une plante et comprend à la fois l'organisation des différents composants de la plante et leurs caractères géométriques approximatifs (orientation dans une direction horizontale ou verticale des branches par exemple). Cette seconde définition se rapproche de celle proposée par Ross, pour qui l'architecture d'une plante représente « un ensemble d'attributs définissant la forme, la taille, la géométrie et la structure externe de la plante », ce qui met un accent prépondérant sur la géométrie des individus. Des significations similaires du terme architecture sont adoptées par d'autres axes de recherche : modélisation de la croissance des plantes, mesures des plantes, répartition du carbone dans la plante, ...

Dans une revue des différents formalismes pour représenter l'architecture des plantes, Godin [Godin, 2000] donne une définition générale de l'architecture des plantes comme la description d'un individu basée sur sa décomposition en entités, en spécifiant leurs types biologiques et/ou leurs formes, et/ou leurs positions et leurs orientations dans l'espace et/ou la manière dont ces entités sont reliées physiquement entre elles. Une plante peut donc être décomposée en différents types d'entités comme les axes, les entre-nœuds. L'architecture d'une plante fait référence à la description de ses constituants, de leur géométrie et de leur topologie. Plus précisément, afin d'englober les différents usages du terme architecture dans les applications, nous considérons l'architecture d'une plante comme la description d'un individu contenant au moins l'une des informations suivantes :

- *l'information de décomposition*, c'est-à-dire l'ensemble des entités qui compose la plante ;
- *l'information géométrique* décrivant sa forme ;
- *l'information topologique* décrivant les connections entre ces entités.

Ces différentes informations peuvent donc être combinées pour former une représentation plus ou moins complexe de l'architecture d'une plante.

Dans cette partie, nous reprenons et approfondissons le canevas développé dans [Godin, 2000] pour présenter une revue des représentations de l'architecture des plantes classifiées suivant leur complexité. Au niveau de complexité le plus bas, la plante est considérée comme un tout et est représentée par un modèle unique, généralement une enveloppe. A l'inverse, des approches modulaires reposent sur une décomposition spécifique d'une plante en modules d'un type particulier (*e.g.* entre-nœuds, unités de croissance, axes ou systèmes ramifiés). Un niveau supérieur de complexité structurelle peut être finalement défini lorsque les plantes sont décomposées en une hiérarchie de modules de tailles différentes. Les représentations globales sont présentées au premier chapitre. Le deuxième chapitre passe en revue les différentes représentations topologiques de l'architecture des plantes suivant leurs complexités : les graphes et arborescences et leurs applications sont d'abord présentés, puis les approches hiérarchiques avec les graphes multi-échelles. Suivant la même classification, le troisième chapitre présente les différentes représentations géométriques utilisées dans la littérature : les approches géométriques à une seule

échelle précèdent les approches en niveaux de détail.

Cette présentation des différentes approches modélisant l'architecture des plantes montre ainsi le besoin d'intégrer différents niveaux de représentations à différentes échelles. Cette analyse de la littérature justifie ainsi ce travail de thèse qui a pour but de comprendre et formaliser la géométrie multi-échelles de l'architecture des plantes. Nous présentons au quatrième chapitre notre modèle géométrique multi-échelles général pour modéliser cette architecture.

Chapitre 1

Représentations géométriques globales

Une première approche consiste à représenter une plante comme un tout, sans la décomposer en modules. Une plante est alors représentée globalement par un modèle unique, donnant une représentation compacte et généralement simple. Ces modèles globaux, représentant une plante à une échelle macroscopique, permettent notamment d'étudier l'interaction de la plante et de son environnement. En infographie, résumer la géométrie complexe d'une plante en une image globale permet un affichage particulièrement efficace pour une impression visuelle correcte.

Ce chapitre, inspiré de différentes revues [Robert, 1998, Godin, 2000], recense les différentes représentations globales de la géométrie des plantes décrites dans la littérature. Nous présentons d'abord les représentations en deux dimensions, figurant la plante par un de ses profils. Nous abordons ensuite les modèles 3D, d'abord les plus simples, utilisant des cônes ou des ellipses, puis plus complexes, tels que les enveloppes convexes, ou dédiés à la représentation des houppiers, comme les enveloppes asymétriques ou extrudées. Finalement, nous présentons un modèle d'enveloppe extrudée reconstruit à partir de photographies.

1.1 Représentations bidimensionnelles

1.1.1 Les profils

Il est souvent important dans le cadre d'une démarche de modélisation d'établir une relation entre les dimensions de l'objet réel et les paramètres de son modèle géométrique. Il est alors naturel de chercher des modèles dont les paramètres puissent correspondre à des grandeurs mesurables sur le terrain. Dans le cas de plantes de tailles particulièrement importantes (*e.g.* arbre de forêt tropicale), il n'est possible d'obtenir qu'une estimation de la géométrie de certains de ses profils 2D. Une première méthode consiste à utiliser la projection au sol du houppier des arbres, ou des mesures de certaines dimensions de cette projection dans différentes directions. Smith *et al.* [Smith et al., 1992, Smith, 1994] évaluent à une certaine hauteur quatre rayons moyens latéraux du houppier, perpendiculaires entre eux. Moravie *et al.* [Moravie et al., 1997] s'appuient sur des mesures de la projection au sol du houppier suivant les huit directions N, NE, E, SE, S, SW, W, NW pour construire une représentation 2D octogonale (Figure 1-1). Ces deux types de représentations permettent d'évaluer l'assymétrie de la couronne d'un arbre en fonction de la déformation de sa projection vers une ou plusieurs directions.

Une seconde méthode consiste à reconstruire un profil vertical du houppier à partir de différentes données dendrométriques telles que la hauteur totale ou de la première fourche de l'arbre, l'estimation du diamètre de la couronne à sa plus grande largeur [Mortier, 1990]. Les formes de houppiers utilisées peuvent être diverses : rectangles, polygones, *etc.* (Figure 1-2). Ces représentations simples d'arbres permettent d'étudier la croissance et les interactions des individus d'un

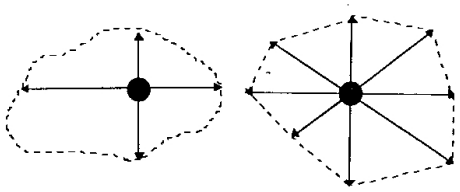


FIG. 1-1 – Principe de la projection au sol du houppier d'un arbre suivant quatre et huit directions. Tiré de [Robert, 1998].

peuplement pour caractériser la dynamique de celui-ci et le statut social de chaque arbre au sein du couvert.

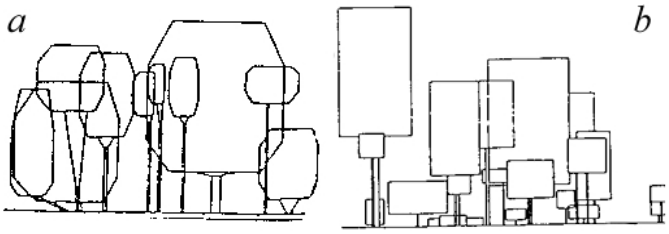


FIG. 1-2 – Représentation de profils verticaux de houppiers selon la conception de : a) Cheviron (1985) b) Brünig *et al.* (1980). Tiré de [Robert, 1998].

1.1.2 Les imposteurs

Un des problèmes majeurs de la synthèse d'images est dû aux capacités limitées des ordinateurs, notamment pour l'affichage de scènes complexes en temps réel. La représentation de forme aussi complexe que celle des plantes a donc fait l'objet de nombreux travaux dans ce domaine qui proposent diverses techniques d'optimisation ou de simplification possibles. Parmi ces derniers, certains proposent de représenter des géométries complexes par de simples images (IBR : Image Based Rendering). Une des premières idées dans ce domaine consiste donc à utiliser une ou deux images de profils d'une plante. Ces images sont appliquées comme textures sur des polygones transparents qui peuvent être ensuite placés dans une scène végétale, donnant l'illusion d'objets complexes éloignés pour un coût en mémoire très faible. Cette technique est cependant dépendante du point de vue sur la scène. En effet, si l'illusion est totale pour un point de vue perpendiculaire au polygone, des artefacts apparaîtront pour des points de vue proches d'une parallèle au plan du polygone. Différentes améliorations ont été proposées pour résoudre ce problème.

Une première approche consiste à utiliser deux polygones texturés perpendiculaires. Pour tous les points de vue à hauteur de l'arbre, au moins un profil est toujours représenté, même si certains artefacts apparaissent encore. Cette technique est encore utilisée dans certains jeux vidéo.

Une deuxième technique, appelée *billboards* [McReynolds, 1997], consiste à toujours orienter le polygone face à la caméra. Cette technique fait cependant l'hypothèse contraignante d'une symétrie radiale de la géométrie de la plante.

Une extension des billboards, les *imposteurs* ou billboards directionnels, consiste à changer l'image texturée sur le polygone suivant l'angle de vue. Une collection d'images de profils de

l'arbre suivant différents points de vue doit être calculée. Ces images sont généralement calculées à partir d'un modèle polygonal complexe tel que ceux présentés dans les chapitres suivants. Elles peuvent être générées à la volée (au moment de l'affichage), ou précalculées. Différents travaux, comme par exemple [Schaufler, 1998], proposent des méthodes pour calculer une image suivant un nouveau point de vue à partir d'images pour des points de vues proches déjà calculés, évitant ainsi des calculs complexes pour la création d'une image à partir d'un modèle détaillé.

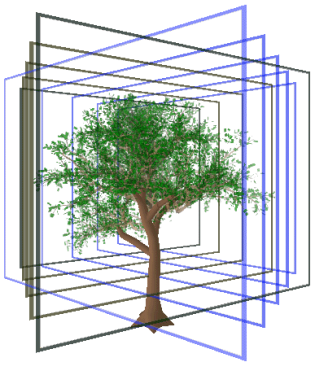


FIG. 1-3 – Représentation en tranches du feuillage d'un arbre. Différents billboards parallèles sont calculés pour représenter des tranches du feuillage [Jakulin, 2000].

L'utilisation de ces techniques nécessite cependant que les objets soient à une distance minimum de la caméra dans la scène. Pour des points de vue proches, des problèmes de parallaxe apparaissent : lors des déplacements, l'utilisation d'une simple image ne permet pas de rendre de manière réaliste les différences de profondeurs de champ entre les différentes parties de l'objet représenté.

Une dernière extension appelée *représentation en tranches* (Slicing) [Jakulin, 2000] permet de résoudre partiellement ce problème. L'ensemble des branches terminales et des feuilles du modèle de plante sont représentées et rendues sous forme d'une série d'imposteurs parallèles, chacun représentant une couche du feuillage de la plante. En découpant un modèle 3D détaillé de plante en couches que l'on projette sur des plans parallèles, on construit différentes images à partir desquelles la série d'imposteurs qui forme une représentation en tranches peut être construite. La plante est alors représentée par différentes représentations en tranches et par la géométrie des axes principaux (Figure 1-3). Cette approche peu coûteuse en mémoire est optimale pour les vues relativement proches où l'on ne distingue pas les détails de chaque feuille.

Finalement, une généralisation des imposteurs par une approche détaillée hiérarchique, permettant des points de vue rapprochés, est présentée à la section 3.3.2.

1.2 Enveloppes simples

1.2.1 Les modèles paramétriques

Des représentations économiques de la plante peuvent être construites en utilisant des modèles paramétriques simples. Des cylindres, des cônes ou des paraboloïdes sont utilisés pour étudier les propriétés mécaniques des plantes [Baker, 1979] ou dans des applications forestières pour modéliser la forme du tronc ou de la couronne [Mitchell, 1975, Courbaud, 1997]. Ces modèles approchent la géométrie d'un houppier par une forme géométrique symétrique simple. Peu de mesures sur le terrain sont nécessaires pour instancier les paramètres de ces modèles, permettant ainsi d'étudier un grand nombre d'individus.

Dans les travaux de Norman et Welles [Norman and Welles, 1983], des ellipses sont utilisées pour modéliser l'interception de la lumière par les couronnes d'arbres. Le volume occupé par un modèle ellipsoïdal est supposé contenir une distribution uniforme de feuilles. Ce qui permet de calculer la loi d'extinction exponentielle d'un rayon lumineux à travers des milieux homogènes [Monsi and Saeki, 1953] :

$$P = \exp(-k\rho_f S)$$

où k est la fraction d'aire foliaire projetée vers la source lumineuse, ρ_f la densité foliaire et S la distance parcourue par le rayon lumineux à travers la canopée. L'équation simple de ces modèles géométriques permet de trouver facilement le point d'entrée et le point de sortie du rayon à travers la canopée et donc de calculer analytiquement la distance S du trajet du rayon à travers la canopée. Norman et Welles étendent ce modèle en représentant la couronne d'un arbre sous la forme d'ellipsoïdes emboîtés (Figure 1-4) ayant différentes densités foliaires et des orientations de feuilles différentes (et donc des surfaces projetées différentes).

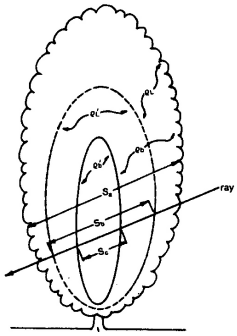


FIG. 1-4 – Représentation géométrique globale d'un houppier par un ensemble d'ellipsoïdes emboités. Ce modèle est utilisé pour calculer l'interception lumineuse de l'arbre [Norman and Welles, 1983].

1.2.2 Les surfaces quadriques texturées

On retrouve aussi l'utilisation de surfaces quadriques[1] pour la représentation de scènes naturelles en infographie dans les travaux de Gardner [Gardner, 1984]. Ces surfaces quadriques bornées par des plans et recouvertes de textures offrent une représentation simple de houppiers, permettant un affichage rapide (Figure 1-5). Cette technique permet d'obtenir un résultat intermédiaire entre des représentations peu coûteuses mais très symboliques telles que celles obtenues avec des imposteurs et des représentations détaillées, assez coûteuses en temps de calcul.

1.3 Enveloppes dédiées à la représentation de houppiers

Afin de prendre en compte l'asymétrie des houppiers et plus généralement de mieux s'adapter à leur plasticité, des modèles géométriques plus complexes, *i.e.* avec plus de paramètres, ont été définis et utilisés par différents auteurs. Pour permettre de représenter des arbres réels, les paramètres de ces modèles sont définis en fonction de mesures réalisables.

[1] surface dont l'équation est de degré 2 (et donc de la forme $ax^2+by^2+cz^2+dyz+ezx+fxy+gx+hy+iz+j=0$)

FIG. 1-5 – Représentation des houppiers d'arbres par des surfaces quadriques texturées [Gardner, 1984].

1.3.1 Les enveloppes asymétriques

Le modèle d'enveloppe proposé par Horn [Horn, 1971] et Koop [Koop, 1989], puis étendu par Cescatti [Cescatti, 1997], est basé sur une forme ellipsoïdale possiblement asymétrique. Le houppier est supposé avoir une base et un sommet clairement identifiés et en son milieu une ligne périphérique dite de plus grande largeur.

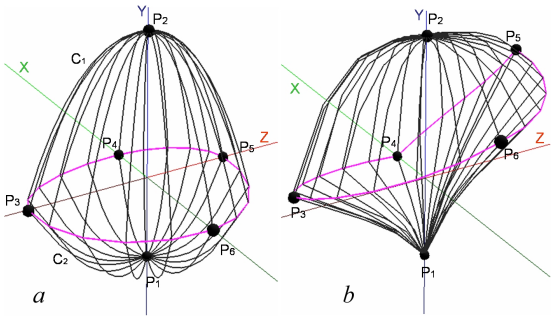


FIG. 1-6 – Modèle de Cescatti [Cescatti, 1997] dédié à la représentation des houppiers. Cette enveloppe est définie par six points de contrôle et deux facteurs de forme. a) les points de contrôle P_3 to P_6 sont dans le plan horizontal xz. b) les hauteurs des points P_3, P_5 and P_6 et les facteurs de forme ont été changés. Tiré de [Boudon et al., 2003].

Cette enveloppe est définie par six points de contrôle et deux coefficients de forme, C_1 et C_2 (Figure 1-6). Les deux premiers points de contrôle, P_1 et P_2, représentent respectivement le sommet et la base de la couronne. Les quatre autres points, P_3 à P_6, permettent de définir une ligne périphérique L à la plus grande largeur de la couronne lorsqu'on la projette sur le plan xz. P_3 et P_5 sont contraints à rester dans le plan xy et P_4 et P_6 dans le plan yz. La projection de la ligne périphérique sur le plan xz est composée de quatre quarts d'ellipse centrée en $(0,0)$. La coordonnée y (la hauteur) des points de la ligne entre deux points de contrôle est une interpolation linéaire entre les hauteurs de ces deux points de contrôle. Ainsi, un point $P_{\theta,i,j}$ de la ligne périphérique, situé à un angle θ entre les points de contrôle P_i et P_j avec i et $j \in [3,6]$ et P_i dans le plan xy et P_j dans le plan yz, est défini de la manière suivante :

$$P_{\theta,i,j} = \left[x_{P_i} \cos\theta, \frac{\theta y_{P_i} + (\frac{\pi}{2} - \theta) y_{P_j}}{\frac{\pi}{2}}, z_{P_j} \sin\theta \right]$$

Finalement, les coefficients de forme décrivent la courbure de la couronne au-dessus et en-

dessous de la ligne périphérique. Les points de la ligne périphérique sont reliés au sommet et à la base par des quarts de super-ellipse respectivement de degré C_1 et C_2. Soit un point P_l de la ligne L et P_1 le point sommet de l'enveloppe, l'équation du quart de super-ellipse supérieur reliant P_l à P_1 est de la forme

$$\frac{(r - r_{P_1})^{C_1}}{(r_{P_l} - r_{P_1})^{C_1}} + \frac{(y - y_{P_l})^{C_1}}{(y_{P_l} - y_{P_l})^{C_1}} = 1$$

avec $r_{P_i} = \sqrt{x_{P_i}^2 + z_{P_i}^2}$. On obtient une équation équivalente pour la super-ellipsoïde inférieure avec P_2 et C_2.

FIG. 1-7 – Quelques exemples d'enveloppes asymétriques montrant la flexibilité de ce modèle.

Des valeurs de coefficients de forme égales à un engendrent des formes de type conique. Des formes concaves seront obtenues pour des valeurs inférieures à un, et des formes convexes pour des valeurs supérieures à un. Une valeur de deux génère des surfaces ellipsoïdales. La possibilité de contrôler séparément ces deux valeurs permet de différencier la forme de la base et du sommet du houppier. On peut ainsi donner une forme concave à la partie basse et une forme convexe à la partie haute du houppier (Figure 1-6.b). En jouant sur ces différents paramètres, une grande variété de formes peut être obtenue (Figure 1-7).

Cescatti propose une méthodologie pour mesurer directement sur le terrain les six points de contrôle, en mesurant la hauteur du houppier à sa base et son sommet, pour les points P_1 et P_2, et le rayon et la hauteur dans du houppier quatre directions orthogonales.

1.3.2 Les enveloppes extrudées

Dans le cadre de la description de l'occupation spatiale de la forêt guyanaise française de Paracou par les arbres, Birnbaum [Birnbaum, 1997] propose un modèle d'enveloppe extrudée construite à partir de deux profils. Chaque houppier est d'abord représenté en deux dimensions par deux profils, le premier vertical et le second horizontal obtenu par la projection au sol de l'arbre (Figure 1-8.A). Le profil vertical est virtuellement découpé en tranches par des plans horizontaux parallèles (Figure 1-8.B) ou par des plans passant par deux points équidistants au sommet de l'arbre (Figure 1-8.C). Pour chacun de ces plans, une représentation horizontale est calculée, ayant la même forme que la projection au sol initiale mais redimensionnée et placée suivant le profil vertical. A partir de cet ensemble de représentations en couche et de la hauteur à l'apex et à la base du houppier, un maillage est calculé.

Ce modèle permet d'étudier les volumes occupés et inoccupés par les couronnes d'arbres d'un couvert végétal. De tels outils ont été définis pour déterminer le type forestier et l'état structural d'une forêt primaire sur de grandes superficies.

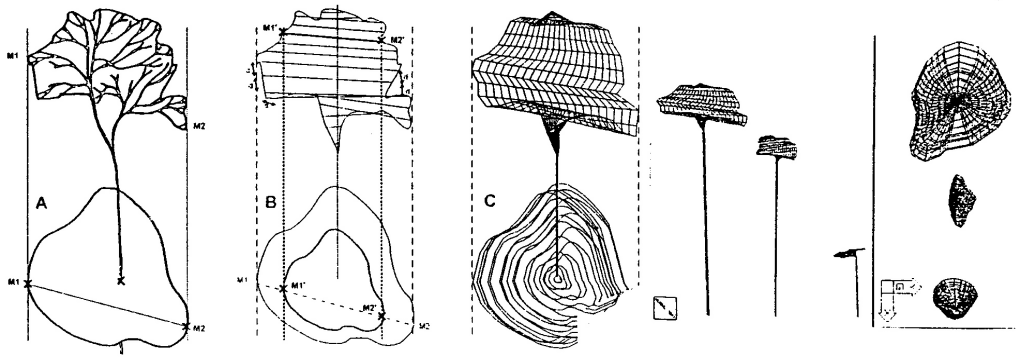


FIG. 1-8 – Enveloppe extrudée construite à partir d'un profil horizontal et d'un profil vertical [Birnbaum, 1997].

1.4 Enveloppes explicites

Les méthodes précédentes approchent la forme globale du houppier par différents modèles paramétriques. Dans cette section, nous présentons des approches utilisant des modèles explicites de représentation du houppier. Ces modèles, plus difficiles à construire à partir de données de terrain, permettent une estimation plus précise du volume occupé par la couronne d'un arbre. Ils permettent donc d'obtenir des résultats plus fiables pour des calculs d'interception lumineuse par exemple.

1.4.1 Les polyèdres convexes

Cluzeau *et al.* [Cluzeau et al., 1995] représentent le houppier d'un système ramifié par son enveloppe convexe.

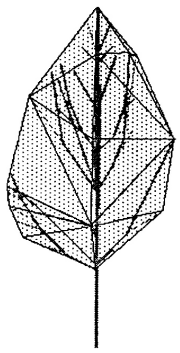


FIG. 1-9 – Enveloppe convexe d'un houppier [Cluzeau et al., 1995].

Cette enveloppe est calculée à partir des points digitalisés sur l'extrémité des unités de croissances des axes du houppier et en utilisant l'algorithme Gift Wrapping [Preparata and Shamos, 1985] (Pour plus de détails sur le calcul d'enveloppe convexe, voir la section 5.1.1). Pour évaluer la validité d'une telle représentation, les auteurs comparent la projection au sol du houppier calculée à partir de cette enveloppe avec celle mesurée sur le terrain et obtiennent des résultats très corrélés. Ils observent que l'utilisation d'une telle représentation semble un compromis intéres-

sant, en terme de complexité et de précision, entre les modèles présentés précédemment et des représentations volumiques détaillées, pour des calculs de rendu ou de pénétration de la lumière à l'échelle d'un peuplement.

1.4.2 Reconstruction d'enveloppes à partir de photographies

Le problème de l'automatisation des méthodes de représentation du houppier d'un arbre a conduit Shlyakhter *et al.* [Shlyakhter et al., 2001] à définir une méthode de reconstruction du houppier à partir de photographies. La position et l'orientation de la caméra dans le repère global de la scène sont supposées connues pour toutes les photos. Chaque image est segmentée de manière semi-automatique (par un utilisateur assisté par ordinateur) pour obtenir le contour du houppier pour les différents points de vue. En projetant les arêtes des contours à partir de la position et dans la direction de la caméra, une forme 3D de type conique, contenant les rayons partant de la caméra qui intersectent l'arbre, est obtenue (Figure 1-10). Chaque cône contient la géométrie de l'arbre. L'intersection des différents cônes donne une approximation de l'enveloppe 3D globale du houppier. La qualité de la reconstruction dépend du nombre de photographies. Cette enveloppe globale est ensuite utilisée pour construire une représentation détaillée de la plante. Elle sert en effet de limite à un modèle de plante écrit en L-systèmes (voir section 7.6.2).

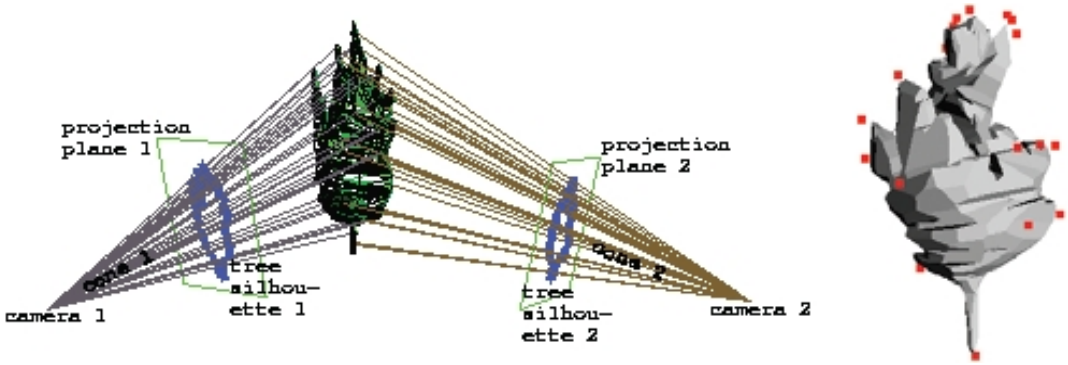


Fig. 1-10 – Reconstruction de l'enveloppe d'un arbre par extrusion [Shlyakhter et al., 2001]. A gauche, pour chaque silhouette obtenue des photos, on calcule le volume contenant la plante. L'intersection de ces différents volumes nous donne une approximation de l'enveloppe de la plante (à droite).

1.5 Conclusion

Ce premier chapitre présente donc les différentes représentations globales de l'architecture des plantes utilisées dans la littérature.

Pour l'infographie, les représentations à base d'images permettent de donner une impression visuelle correcte pour des modèles d'arbres éloignés. Ces méthodes ont été étendues pour les vues proches en décomposant la géométrie d'un arbre en un ensemble d'images plus précises. Elles sont détaillées au chapitre 3.

Pour la modélisation, différents modèles d'enveloppes, plus ou moins sophistiquées, sont proposés pour représenter les houppiers d'arbres. Ces modèles ont été définis en fonction des paramètres mesurables sur le terrain. Des mesures simples, tels que des diamètres et des hauteurs de houppier, permettent de définir de simples profils. Un profil horizontal et un profil vertical ou des mesures de ces deux profils permettent de construire des enveloppes souples tels que des enveloppes extrudées ou des enveloppes asymmétriques. Finalement, l'utilisation d'un plus grand

nombre de profils reconstruit à partir de photographies, permettent de définir des enveloppes plus précises.

Ces modèles sont donc fortement dépendants des moyens d'acquisition des données. L'effort a été porté, ces dernières années, à l'amélioration des techniques d'acquisition pour définir des modèles d'enveloppes plus précis. Aucune étude ne s'est cependant intéréssée à l'évaluation comparative de ces modèles d'enveloppes, et de leur aptitude à représenter une géométrie complexe. Nous présentons, au chapitre 5, une première étude comparative de ces modèles.

Chapitre 2

Représentations topologiques détaillées

Les représentations topologiques décrivent l'organisation des différents organes composant une plante. Ces représentations sont la base d'un grand nombre de modèles structure/fonction pour modéliser des transferts de substances, la croissance, ou pour simplifier les protocoles de mesure de l'architecture des plantes. Bien souvent des *arborescences*, elles décrivent l'organisation de la plante à une échelle donnée, choisie suivant l'application considérée. Cependant, afin de mieux rendre compte de l'organisation générale de la plante, ou pour modéliser des relations entre entités à différentes échelles dans le cadre de la simulation, certains travaux ont été amenés à formaliser des représentations topologiques multi-échelles de l'architecture des plantes.

Ce chapitre reprend et étend la revue de Godin [Godin, 2000] sur les différentes façons dont la topologie a été prise en compte, à une ou plusieurs échelles, dans les différentes applications de modélisation des plantes.

2.1 Modèles topologiques de l'architecture des plantes

Plusieurs modèles de flux hydrauliques dans les plantes ont été construits sur une analogie avec les réseaux électriques [Früh, 1997, Dauzat et al., 2001]. La plante est décomposée en composants auxquels sont associés une conductance hydraulique. Le flux d'eau à travers un composant est supposé proportionnel à la différence d'état hydrique à ses bornes (loi d'Ohm). Les transferts d'eau dans la plante sont donc définis par un "réseau hydraulique" qui s'appuie sur la topologie de la plante : comme pour les réseaux électriques, la loi de Kirchhoff [Chen, 1976] est satisfaite pour chaque composant, *i.e.* le flux d'eau entrant dans un composant est égal à la somme des flux sortants.

La topologie de la plante est aussi utilisée pour aborder les problèmes d'allocation de carbone. Dans la théorie du "pipe model" introduit dans les années soixante par Shinozaki [Shinozaki et al., 1964a], la plante est considérée comme un "ensemble de tuyaux unitaires" (Figure 2-1), chaque tuyau portant une unité de surface foliaire [Valentine, 1985, Nikinmaa, 1992]. Des structures ramifiées complexes peuvent ainsi être représentées en connectant ces tuyaux modélisant des composants de la plante. La structure résultante, illustrée dans la figure 2-1, définit un réseau d'aubier sur lequel la loi de Kirchhoff est satisfaite de la manière suivante : le nombre de tuyaux dans un composant est égal au nombre total de tuyaux des composants connectés au-dessus [Perttunen et al., 1996].

Les représentations topologiques sont aussi utilisées de manière plus abstraite pour simuler la propagation des substances à travers les composants de la plante. Un premier problème ici consiste à simuler la compétition entre les branches pour des ressources limitées à travers le réseau de composants de la plante [Borchert and Honda, 1985, Reffye et al., 1997]. Un deuxième

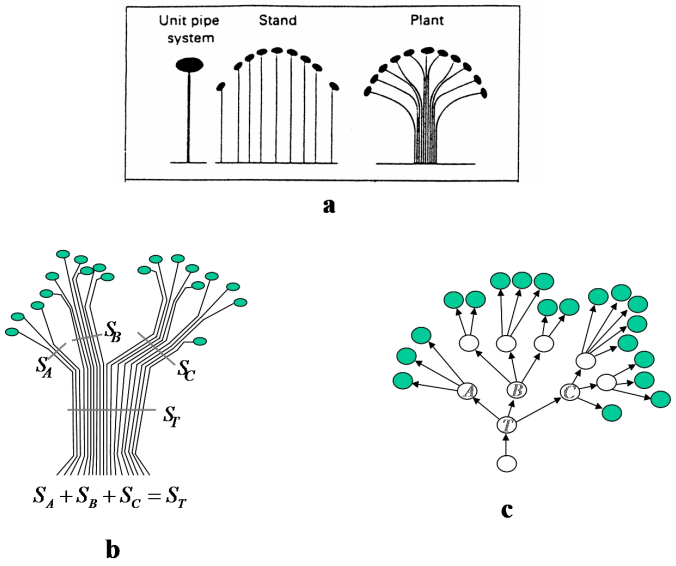


FIG. 2-1 – Description modulaire en utilisant la théorie du "pipe model". a) Représentation classique d'une plante par des tuyaux [Shinozaki et al., 1964b]. b) Représentation d'un système ramifié par des tuyaux : chaque segment d'un arbre est représenté par un ensemble de tuyaux. La loi de Kirchhoff exprime la conservation de flux. c) L'arborescence associée au modèle b). Chaque ensemble de tuyaux est représenté par un sommet et les connections entre les ensembles de tuyaux par des arêtes. Tiré de [Godin, 2000].

problème réside dans l'étude de la propagation de signaux à travers la topologie de la plante. Une telle modélisation peut être utilisée pour représenter le temps de floraison dans les inflorescences ramifiées par exemple [Janssen and Lindenmayer, 1987, Prusinkiewicz and Lindenmayer, 1990].

Les ordinateurs devenant de plus en plus puissants, les programmes de simulation de la croissance des plantes ont utilisé abondamment les représentations topologiques de l'architecture des plantes pour obtenir des représentations 3D réalistes de plantes simulées (voir chapitre 3).

La liste d'applications utilisant des représentations topologiques de l'architecture des plantes présentée ci-dessus n'est naturellement pas exhaustive. Cependant, elle reflète la grande variété des champs où la topologie des plantes a été utilisée pour raffiner leurs représentations. Tous ces modèles de plantes ont en commun une structure sous-jacente d'*arborescence*.

2.1.1 Les arborescences

Soit un ensemble de composants venant de la décomposition d'une plante en modules. Le réseau fait par les composants d'une plante et leur relation d'adjacence peuvent être représentés par une relation binaire définie sur l'ensemble des composants, *i.e.* un *graphe*. Du fait de la nature de la croissance des plantes, ces graphes sont sans cycle : ce sont donc des *arborescences*.

Formellement un graphe, un arbre et une arborescence (au sens algorithmique) se définissent de la manière suivante :

Définition 1 *Un graphe orienté $g = (V, E)$ est le couple constitué d'un ensemble V dont les éléments sont appelés* sommets *(ou* nœuds *ou* vertex*) et d'un ensemble E dont les éléments sont des paires ordonnées de sommets appelés* arcs *(ou* arêtes*).*

Remarquons que tout graphe $g = (V, E)$ représente une relation binaire $\Re$ sur l'ensemble des sommets V : tout couple de E représente les éléments mis en relation par la relation binaire $\Re$.

Définition 2 *Un* arbre *est un graphe simple, connexe, et sans cycle.*

Définition 3 *Une* arborescence *est un arbre orienté, possédant un sommet r qui est relié à tous les autres sommets par un chemin unique dont il est l'origine.*

Le sommet r est appelé la *racine* de l'arborescence. Tous les sommets sans successeur sont appelés les *feuilles* de l'arborescence.

Godin et Caraglio [Godin and Caraglio, 1998] introduisent un formalisme général pour modéliser l'organisation des plantes à une échelle par des arborescences. La figure 2-2 illustre une arborescence dans laquelle chaque nœud (au sens botanique) de la plante est représenté par un sommet dans le graphe et les connections entre les nœuds sont représentées par des arcs. Deux types de connections peuvent être distingués pour marquer l'organisation des composants d'une plante. Un '<' (précède) dénote la connexion entre deux composants qui ont été créés par le même méristème apical. Un '+' (porte) dénote la connexion entre deux composants qui ont été créés par différents méristèmes apicaux. Des informations additionnelles peuvent être associées aux organes de la plante dans la représentation topologique en ajoutant des attributs aux sommets correspondants dans l'arborescence (*arborescence augmentée*). Ces informations peuvent correspondre à la position dans l'espace de l'organe, sa géométrie, ou d'autres caractéristiques de l'organe.

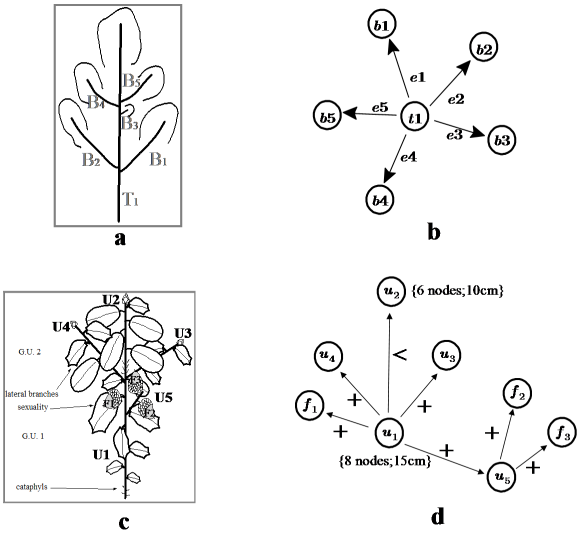


FIG. 2-2 – a) Un arbre, considéré comme un ensemble de branches, et b) la représentation par une arborescence de la topologie de ses branches. c) Un système ramifié d'un chêne décrit en terme d'unités de croissance et d) l'arborescence augmentée correspondante. Tiré de [Godin and Caraglio, 1998].

2.1.2 Les arborescences axiales

Dans le contexte de la simulation de la croissance des plantes avec les L-systèmes (détaillé au chapitre 7), Prusinkiewicz et Lindenmayer [Prusinkiewicz and Lindenmayer, 1990] proposent des arborescences particulières, appelées *arborescences axiales* (Figure 2-3). De la même manière que Godin et Caraglio [Godin and Caraglio, 1998], ils utilisent le fait qu'un nœud va pouvoir porter plusieurs nœuds mais un seul au plus sera créé par le même méristème, pour distinguer un arc particulier à chaque sommet. Cet arc sera dit *direct* ; tous les arcs restants sont dits *latéraux*. Dans ces graphes, la notion d'axe est introduite de la manière suivante.

Définition 4 *Une séquence d'arcs forme un* axe*, si :*

- *le premier arc dans la séquence est un arc latéral ou part de la racine de l'arborescence,*
- *chacun des arcs suivants est un arc direct, et*
- *le dernier arc n'est pas suivi par un arc direct dans l'arborescence.*

Un axe et tous ses descendants constituent une sous-arborescence axiale. Les axes sont ordonnés : l'axe qui part de la racine est d'ordre 0 ; les axes qui partent d'un arc latéral d'un axe parent d'ordre n sont d'ordre $n+1$. L'ordre d'une sous-arborescence axiale est égal à l'ordre de son axe de plus petit ordre. On peut noter que le regroupement des arcs en axes forme un partitionnement de l'arborescence.

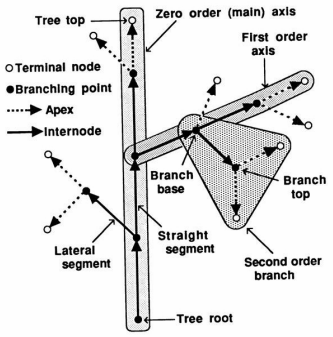


FIG. 2-3 – Une arborescence axiale. Tiré de [Prusinkiewicz et al., 1988].

Pour représenter les arborescences axiales, Lindenmayer [Lindenmayer, 1968, Lindenmayer, 1971] introduit une notation par chaîne de caractères parenthésée. Une arborescence axiale avec des arcs étiquetés avec l'alphabet V est représentée par un *mot* (chaîne de symboles ou lettres) m sur l'alphabet $V_E = V \cup \{[,]\}$:

$$m = x_1[\alpha_1]x_2[\alpha_2]...x_n[\alpha_n]x_{n+1}$$

On suppose que les sous-mots $x_1, x_2, ..., x_{n+1} \in V^*$ ne contiennent pas de crochets, et les sous-mots $\alpha_1, \alpha_2, ..., \alpha_n \in V_E^*$ sont bien formés (*i.e.* $\forall \alpha_i, \alpha_i$ contient autant de symboles [que de symboles]). Les mots $x_1, x_2, ..., x_{n+1}$ représentent l'axe principal (d'ordre 0) de m. Les mots $\alpha_1, \alpha_2, ..., \alpha_n$ représentent les sous-arborescences axiales d'ordre 1 de m. Chaque branche α_i peut se décomposer de la même manière que m, faisant ressortir les axes d'ordre 1 et les sous-arborescences axiales d'ordre 2. Cette décomposition peut se faire récursivement.

Il est souvent plus pratique de caractériser les modules d'une plante (et les relations entre eux) avec plus de détail que ce que permet l'utilisation des lettres seules. Pour ce faire, on peut associer un ou plusieurs paramètres numériques avec les symboles de la chaîne parenthésée. Un symbole (une lettre) avec des paramètres associés est appelé *lettre paramétrique*, et une chaîne de lettres paramétriques un *mot paramétrique*.

Un mécanisme d'interprétation géométrique des chaînes de caractères permet d'obtenir une représentation graphique de l'arborescence et sera détaillé dans la section 3.1.5.

2.2 Représentations topologiques multi-échelles

Pour différents systèmes de simulation de la croissance des plantes, le problème de modéliser des aspects morphologiques ou physiologiques à différentes échelles de temps et d'espace

entraîne le besoin de considérer la structure topologique de la plante à différentes échelles. De la même manière, dans le domaine de l'analyse de l'architecture des plantes, différents niveaux d'organisation doivent être considérés pour une meilleure compréhension du développement de la plante. Le besoin de formaliser de telle description multi-échelles a été défendu par différents auteurs [Godin et al., 1997, Remphrey and Prusinkiewicz, 1997]. Les premières descriptions multi-échelles informelles furent utilisées dans l'analyse architecturale des plantes où l'existence de différents niveaux de détails dans un modèle architectural est rendu avec rigueur dans le dessin. La Figure 2-4 illustre une telle description multi-échelles. Cette image contient des détails à l'échelle de la forêt, des arbres, des systèmes ramifiés, des axes et des segments.

Nous présentons dans cette section différentes applications utilisant des approches multi-échelles pour organiser la topologie de leurs plantes, puis un modèle formel multi-échelles de représentation de l'architecture des plantes.

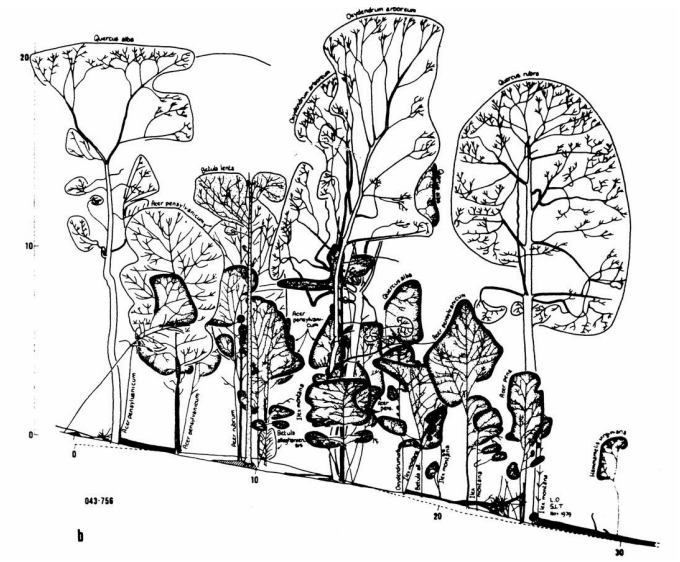


FIG. 2-4 – Description architecturale d'une forêt montrant différents niveaux de détails. Tiré de [Oosterhuis et al., 1982].

2.2.1 Approches intégrant différents niveaux d'organisation

Approches fractales

L'aspect fractal de l'architecture des plantes a conduit plusieurs chercheurs à concevoir des systèmes de simulation de la croissance des plantes qui utilisent des concepts fractals tels que l'auto-similarité, les échelles, les systèmes de fonctions itérées, les dimensions fractales, *etc.* [Smith, 1984, Oppenheimer, 1986, Prusinkiewicz and Hanan, 1989, Chen et al., 1994] (détaillé à la section 3.3.1). Dans ces approches, la représentation topologique des plantes n'est pas fondamentalement différente de celle utilisée dans les approches détaillées. L'aspect fractal de ces approches tient au processus récursif qui génère une structure auto-similaire en quelques étapes [Barnsley, 1988, Prusinkiewicz and Hanan, 1989]. De la même manière que pour les approches détaillées, la topologie de la plante est représentée par une arborescence, en utilisant une seule "unité" de description (par exemple les entre-nœuds). Augmenter le niveau de détails s'exprime par l'ajout de systèmes ramifiés, *i.e.* par l'augmentation de l'ordre de ramification (Figure 2-5).

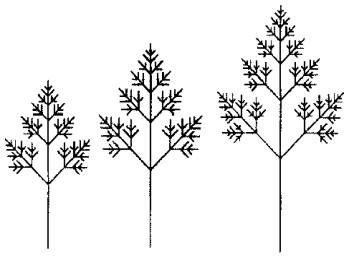


FIG. 2-5 – Séquence de développement d'un système ramifié modélisé par un DOL-système [Prusinkiewicz and Lindenmayer, 1990]. De nouveaux détails apparaissent avec les nouveaux ordres de ramification.

Simulation multi-échelles

Dans les années 1990, d'autres types de représentations multi-échelles des plantes ont été introduits, en rendant explicite l'existence d'entités à différents niveaux de détails. Basé sur le concept d'analyse de l'architecture , le programme de simulation AMAPsim utilise une structure de données en couches où différents types de composants de la plante sont représentés à différentes échelles : les entre-nœuds, les unités de croissance et les axes [Jaeger, 1987, Blaise, 1991] (Figure 2-6). Cette structure de données est utilisée dans la simulation de la croissance des plantes pour adapter la production des méristèmes à la nature (âge, ordre et état physiologique) des unités de croissance et axes les contenant. Une représentation similaire est utilisée dans INCA [LeDizès et al., 1997] où le paradigme des systèmes experts est utilisé pour modéliser le processus de croissance de la plante. Cette croissance est gouvernée par un ensemble de règles (en Prolog) exprimant les conditions de croissance et de différenciation des méristèmes comme une fonction de l'état physiologique des composants de la plante à différentes échelles. En retour, la structure multi-échelles qui représente la plante est mise à jour avec les nouvelles valeurs physiologiques et les nouveaux composants à différentes échelles.

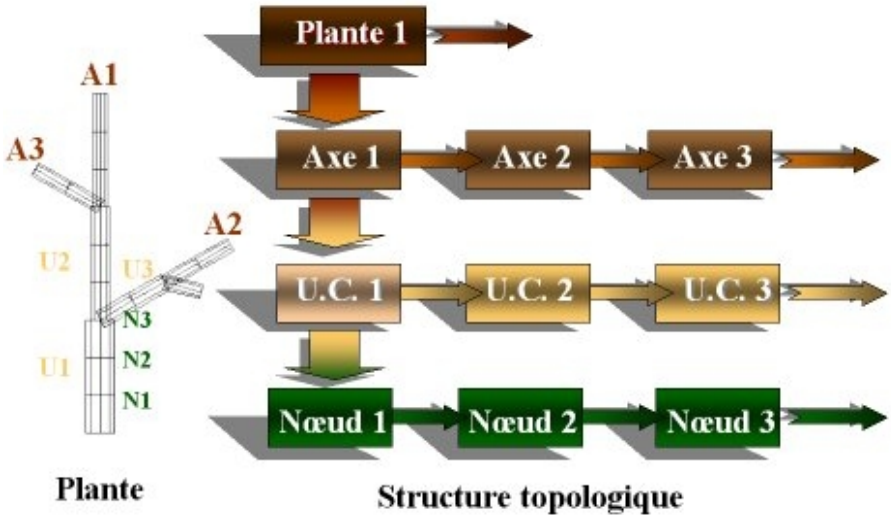


FIG. 2-6 – Organisation des structures représentant l'architecture des plantes dans le logiciel AMAPpara [Blaise et al., 1998].

Approches orientées objets

Les modèles multi-échelles de plantes se retrouvent aussi dans les approches orientées objets. Ici, l'intention est de décrire les propriétés structure-fonction et les relations entre les composants de la manière la plus proche possible de la réalité. L'accent est aussi mis sur la conception d'objets pour représenter les composants de la plante. Dans les approches orientées objets, les objets définissent aussi la manière dont les données qu'ils contiennent peuvent être accédées et utilisées (*i.e.* [Hill, 1996]). Salminen *et al.* [Salminen et al., 1994] utilisent une approche orientée objet pour spécifier la nature des composants de la plante à différentes échelles et déterminer leurs connections structurelles et fonctionnelles. Avec une telle spécification, le programme de simulation construit une représentation de la plante qui contient des composants à différents niveaux de détails, *i.e.* une structure de données multi-échelles. Ici, il est attendu que les propriétés émergentes (telles que la morphologie des plantes, le volume du tronc, l'aspect de la couronne, *etc.*) dérivent de la modélisation locale mais détaillée de l'interaction entre les composants de la plante.

2.2.2 Analyse multi-échelles de l'architecture des plantes

Les approches précédentes ont été conçues pour prendre en compte les interactions entre des composants à différentes échelles pendant la simulation de la croissance de la plante mais ne mettent pas l'accent sur la représentation multi-échelles de la plante elle-même. Godin et Caraglio [Godin and Caraglio, 1998] ont formalisé la notion de structures multi-échelles en croissance et ont étudié leurs propriétés mathématiques. Le modèle de représentation de l'architecture des plantes résultant est appelé *arborescence multi-échelles* (MTG : Multiscale Tree Graph). Comme la définition de la notion d'échelle utilisée dans la géométrie fractale, une échelle dans un MTG est associée au choix d'une unité de décomposition [Godin, 2003]. Dans la figure 2-7, la plante est décomposée à trois échelles différentes correspondant à trois unités de décomposition : les axes, les unités de croissance et les entre-nœuds. A chaque échelle, la structure de la plante est représentée par une arborescence. Toutes ces arborescences sont intégrées dans un MTG en rendant explicite les relations de décomposition entre les composants.

Une plante en croissance est représentée par un MTG variant au cours du temps où toutes les arborescences à chaque échelle sont des arborescences en croissance. En utilisant ce formalisme, Godin et Caraglio montrent que les relations de décomposition entre les composants peuvent varier au cours du temps. Cette propriété peut être utilisée pour modéliser la croissance de structures végétales complexes. Les MTGs sont utilisés dans le programme AMAPmod dédié à la mesure et l'analyse de l'architecture des plantes, comme une structure de données centrale qui organise toutes les informations collectées sur les plantes.

Le travail présenté dans cette thèse étant basé en grande partie sur ce modèle, nous donnerons ici quelques définitions formelles utiles sur les arborescences multi-échelles.

2.2.3 Les arborescences quotientées

Définition 5 *Un graphe quotienté g est un triplet (h, W, π) où :*

- *$h = (V, E)$ est un graphe orienté appelé le support de g.*
- *W est un ensemble de sommets.*
- *π est une application surjective de V sur W, appelée application quotient .*

Pour tout x de V, le sommet $\pi(x)$ est appelé le *complexe* de x et réciproquement, x est un *composant* de $\pi(x)$. $\pi^{-1}(z)$ représente l'ensemble des composants de z de W et si x est un sommet de V, $\Pi(x)$ représente l'ensemble des composants de $\pi(x)$. Le nombre d'éléments de $\Pi(x)$ est noté $|\Pi(x)|$. L'application π induit une partition Π_g sur V : $\Pi_g = \{\pi^{-1}(z) | z \in W\}$. Le graphe quotient $\mathbb{Q}(g)$ associé à g, est le graphe (W, E_π) tel que :

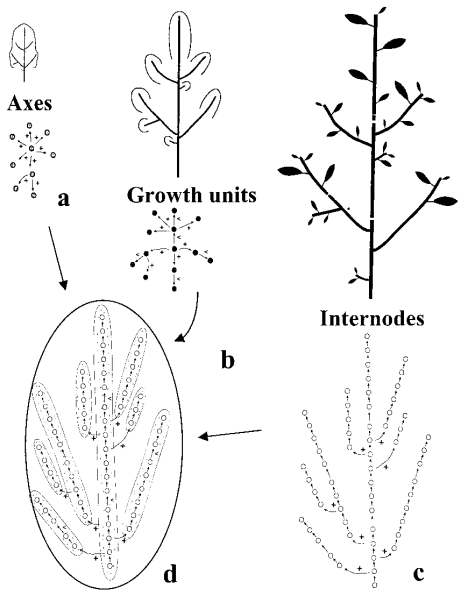


FIG. 2-7 – Une arborescence multi-échelles [Godin and Caraglio, 1998]. La plante peut être représentée par des arborescences à différentes échelles : a) échelle des axes, b) échelle des unités de croissance et c) échelle des entre-nœuds. d) Une arborescence multi-échelles résulte de la superposition de toutes ces représentations.

$$\forall (x, y) \in E, (\pi(x), \pi(y)) \in E_\pi \Leftrightarrow \pi(x) \neq \pi(y)$$

Pour tout graphe quotienté $g = (h, W, \pi)$, si h, le graphe support de g, est connexe, alors $\mathbb{Q}(g)$ est un graphe connexe, et ceci quelle que soit l'application quotient. Néanmoins, la nature même du graphe quotient dépend à la fois du support et de l'application quotient. Par exemple, le graphe quotient d'un graphe quotienté dont le support est une arborescence n'est pas, en général une arborescence. Les graphes quotientés dont, à la fois, le graphe support et le graphe quotient sont des arborescences, sont appelés *arborescences quotientées.*

Une représentation graphique d'une arborescence quotientée est proposée à la figure 2-8.

2.2.4 Les arborescences multi-échelles

Intuitivement, les graphes quotientés peuvent être généralisés par récurrence en appliquant, à partir du graphe initial, une nouvelle opération de quotientement aux graphes quotients successifs. L'application successive des opérations de quotientement conduit rapidement à regrouper entièrement le graphe en un seul sommet pour lequel l'application du quotientement ne modifie plus rien. La structure ainsi obtenue est pyramidale avec, en son sommet, un sommet unique et, à sa base, un graphe.

Définition 6 *La définition suivante formalise ces notions pour définir de manière récursive les* graphes multi-échelles *[Godin and Caraglio, 1998] :*

- *un graphe simple g est un graphe multi-échelles. L'ensemble des sommets du graphe simple g, noté $V(g)$, définit l'ensemble des sommets du graphe multi-échelles.*
- *si h est un graphe multi-échelles, alors $g = (h, W, \pi)$ est un graphe multi-échelles tel que :*
  - *$V(h)$ est un ensemble de sommets ;*

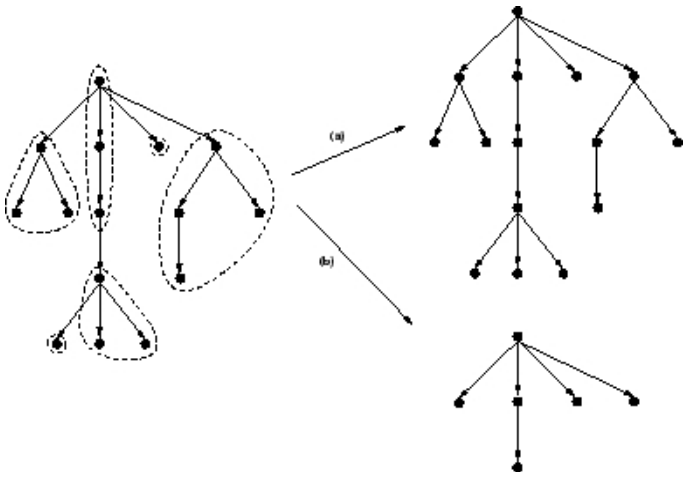


FIG. 2-8 – Une arborescence quotientée (à gauche). a) le graphe support $h = (V, E)$ et b) le graphe quotient du graphe quotienté. Tiré de [Ferraro, 2001].

– *π est une fonction surjective de $V(h)$ sur W (notée π_g en cas de confusion).*

Un graphe simple g est appelé graphe multi-échelles *terminal*, sinon il est dit *récursif*. Pour un graphe multi-échelles $g = (h, V, \pi)$ récursif , h est appelé le *support* de g. Dans ce cas, les trois composants du graphe multi-échelles, (h, W, π), sont similaires aux composants d'un graphe quotienté, excepté le fait que dans le cas général, h est lui-même un graphe multi-échelles, ce qui donne la structure récursive des graphes multi-échelles. De la définition, il découle qu'un graphe multi-échelles g peut être développé récursivement jusqu'à obtenir un graphe simple :

$$\begin{aligned} g = g_1 &= (g_2, V_1, \pi_1) \\ g_2 &= (g_3, V_2, \pi_2) \\ &\ldots \\ g_{m-1} &= (g_m, V_{m-1}, \pi_{m-1}) \end{aligned}$$

La séquence $\{g_1,\ g_2,\ldots,\ g_m\}$, où $g_1,\ g_2,\ldots,\ g_{m-1}$ sont des graphes multi-échelles récursifs et g_m un graphe multi-échelles terminal, est appelé le *développement* de g. Par construction, tout graphe multi-échelles admet un développement fini. Chaque élément du développement d'un graphe multi-échelles correspond à une échelle de décomposition différente. Dans notre définition, les échelles correspondent aux index des graphes du développement de g. Pour deux graphes multi-échelles g et h, si h appartient au développement de g alors h est dit plus *fin* que g.

Nous utilisons ici la définition des graphes multi-échelles donnée dans [Ferraro, 2001], qui est une simplification de la présentation plus générale proposée par Godin et Caraglio [Godin and Caraglio, 1998]. En effet, la définition générale tient compte de l'idée qu'à une certaine échelle, il peut exister des sommets qui ne sont pas décomposés dans un ensemble de sommets plus fins.

La définition récursive complète est un quintuplé qui fait apparaître deux ensembles distincts de sommets, ceux qui sont décomposés et ceux qui ne le sont pas.

La projection $P(g)$ d'un graphe multi-échelles g est un graphe simple défini récursivement par :

- si g est un graphe simple alors $P(g) = g$;
- si $g = (h, V, \pi)$ est un graphe multi-échelles récursif, $P(g) = \mathbb{Q}_{/\pi}(g) = \mathbb{Q}(P(h), V, \pi)$, c'est-à-dire le graphe quotient de la projection de h.

L'opération de projection est idempotente, ce qui justifie à posteriori que l'opérateur P sur les graphes multi-échelles soit appelé une *projection*.

Une *arborescence multi-échelles* (MTG) est un graphe multi-échelles dont le graphe support est une arborescence multi-échelles et la projection est une arborescence.

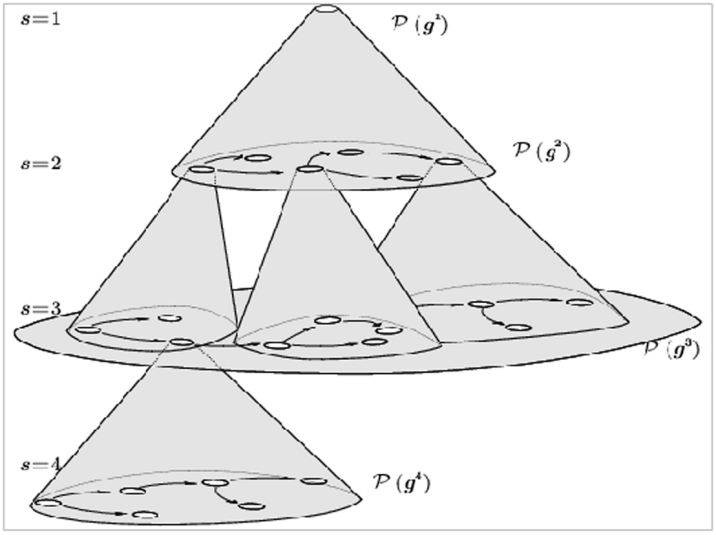


FIG. 2-9 – Une arborescence multi-échelles

Nous donnons avec la figure 2-9 un exemple d'arborescence multi-échelles. Par définition, le graphe g est une arborescence multi-échelles. Nous pouvons remarquer que deux graphes multi-échelles g_1 et g_2 distincts peuvent posséder un support identique. Nous dirons alors que g_1 et g_2 sont deux graphes multi-échelles incomparables. De ce constat, Godin et Caraglio [Godin and Caraglio, 1998] introduisent la notion de graphes multi-échelles généralisés. Un *graphe multi-échelles généralisé* est une séquence $\{g_1, g_2, ..., g_m\}$ de graphes multi-échelles incomparables.

2.3 Conclusion

Dans le cadre de la modélisation de la géométrie de l'architecture des plantes, nous verrons au chapitre suivant comment les arborescences sont utilisées dans la littérature pour organiser des modèles géométriques détaillés. Nous montrerons ensuite dans le quatrième chapitre comment les arborescences multi-échelles peuvent être utilisées pour organiser un modèle géométrique à différentes échelles.

Chapitre 3

Représentations géométriques détaillées

Les représentations géométriques détaillées de plantes sont basées sur une décomposition de la plante en organes de différents types (feuilles, fruits, entre-nœuds) qui sont représentés par différentes primitives géométriques. On trouve dans la littérature une grande variabilité de primitives géométriques pour ces représentations. Certaines approches utilisent des voxels, d'autres des primitives géométriques simples (cylindres) ou plus complexes (cylindres généralisés, surfaces implicites). Pour une optimisation de l'affichage, des imposteurs ou des textures volumiques sont utilisés. Un grand nombre de représentations à base de points, organisées ou obtenues de différentes manières, permettent d'optimiser le rendu en minimisant le nombre de primitives affichées en fonction de la résolution de l'image à calculer.

Ce chapitre propose une classification de ces modèles suivant la manière dont ils organisent leurs primitives géométriques. Une première classe de modèles considère simplement la géométrie de la plante comme un ensemble d'éléments géométriques représentant chacun des composants. Une autre classe de modèles considère aussi l'adjacence entre ces composants pour, par exemple, obtenir des représentations plus réalistes en essayant de représenter au mieux les connections entre organes. Finalement, certaines approches cherchent à tirer parti de l'organisation hiérarchique de la structure d'une plante pour organiser, à différents niveaux de détails, l'information géométrique.

3.1 Descriptions géométriques

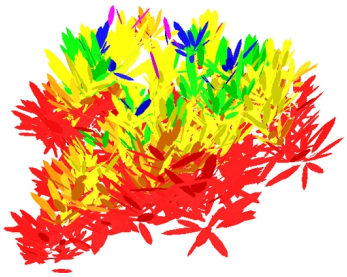


FIG. 3-1 – Sinoquet *et al.* [Sinoquet et al., 1998] considèrent uniquement la géométrie des feuilles pour étudier l'interception lumineuse de ce manguier.

Une première méthode pour décrire une plante consiste à considérer uniquement la forme et l'organisation spatiale des différents organes de cette plante. Les relations entre organes comme

leurs connections ne sont pas prises en compte. Tous les types d'organes ne sont pas non plus pris en compte. On pourra en effet ne considérer par exemple que l'organisation spatiale des feuilles pour étudier par exemple l'interception lumineuse d'une plante [Dauzat, 1993, Chelle, 1997, Sinoquet et al., 1998].

Une deuxième méthode consiste à décrire la géométrie de la plante en utilisant un modèle topologique (généralement une arborescence) pour organiser le modèle géométrique. Cette section présente les différents types de représentations géométriques que l'on trouve dans la littérature, construits sur différentes primitives géométriques et organisés sur des arborescences.

3.1.1 Une approche pionnière

L'un des premiers à s'être intéressé à la représentation des arbres d'un point de vue infographique fut le Japonais Honda [Honda, 1971] qui dès le début des années 70 s'intéressa aux diverses formes adoptées par un arbre en fonction d'un certain nombre de paramètres à caractère géométrique. La structure topologique sous-jacente à tous les arbres dessinés est un arbre binaire parfait. Le modèle géométrique définit :

- des rapports de diminution des longueurs des branches : chaque branche fille droite (respectivement gauche) est un segment de droite dont la longueur est raccourcie par rapport à la longueur de sa branche mère d'un rapport R_1 (respectivement R_2),
- des angles d'embranchement aux différents sommets : une branche mère non verticale (la verticale étant défini comme la direction de la gravité) se ramifie en deux branches filles dans le plan qui la contient et qui est perpendiculaire au plan qu'elle forme avec la verticale (Figure 3-2.a). Les angles de déviation q_1 et q_2 entre les branches filles droite et gauche et leur mère sont fixés. Dans le cas d'une branche mère verticale, le branchement a lieu dans le plan xOz avec des angles q_1 et q_2 par rapport à la verticale. Si l'un des deux angles est nul, une des branches filles reste verticale, et un angle de déviation a en projection dans le plan xOy est alors introduit de proche en proche pour répartir les branches dans l'espace (Figure 3-2.b).

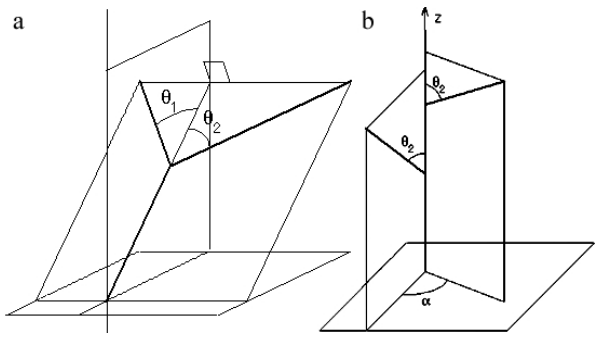


FIG. 3-2 – Géométrie selon Honda : a) Cas général b) Cas particulier. Tiré de [Janey, 1992].

Les structures ainsi obtenues sont représentées en fil de fer par Honda de manière à pouvoir analyser facilement leurs formes.

3.1.2 Les systèmes de particules

Reeves et Blau [Reeves and Blau, 1985] génèrent des modèles d'arbres et d'herbes en utilisant des *systèmes de particules*. Ces systèmes représentent les objets sous forme de nuages de particules élémentaires. Ces particules peuvent changer de forme, et se mouvoir dans le temps.

Dans ce type de système, chaque particule est spécifiée par une forme, une trajectoire, une couleur, une date de naissance et de mort et d'autres attributs qui sont définis par un ensemble de processus stochastiques contraints. Dans le cas des arbres, la forme des particules peut être un segment linéaire pour les branches et ou un petit disque pour une feuille.

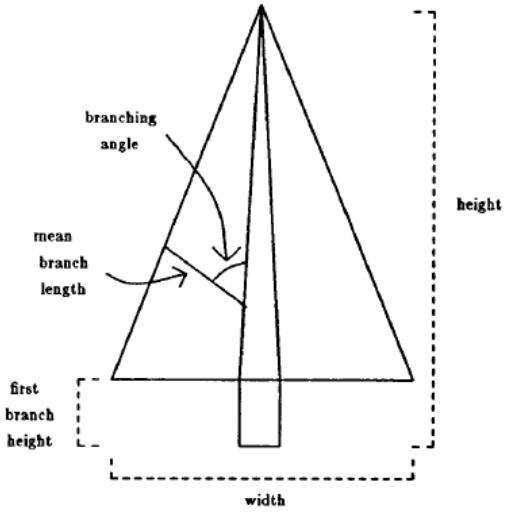


FIG. 3-3 – Géométrie des arbres et les différentes dimensions considérées pour générer les systèmes de particules par Reeves et Blau [Reeves and Blau, 1985]

Les systèmes de particules qui permettent de générer des images de végétaux sont structurés (composés de particules non indépendantes). De nombreuses relations existent entre les particules modélisant un arbre car elles doivent constituer un objet 3D cohérent.

Dans cette approche, la forme générale des feuillus sera considérée comme un ellipsoïde tandis que les arbres à feuillage persistant ressembleront à des cônes (Figure 3-3). Un modèle d'arbre est donc défini par un ensemble de contraintes sur les paramètres le caractérisant. Ces contraintes qui lient les paramètres entre eux définissent des intervalles autour d'une valeur moyenne dans lesquels les paramètres sont choisis avec équiprobabilité. Un algorithme récursif génère les branches à partir de paramètres et de relations permettant de définir le type de branchement, la distance entre deux sous-branches consécutives, l'angle de la branche au tronc, sa longueur et son épaisseur. Chaque branche hérite d'une grande partie des paramètres de ses parentes (éventuellement remis à l'échelle). La récursion est arrêtée quand l'épaisseur de la branche atteint une valeur minimale limite ou quand un certain niveau de profondeur est atteint.

Le nombre et la géométrie des primitives utilisées posent des problèmes de rendu. Afin de minimiser les temps de calcul, Reeves et Blau proposent différentes méthodes, basées notamment sur des algorithmes de tri des particules pour le rendu des primitives et des méthodes statistiques pour les calculs d'ombres. Les résultats obtenus sont tout à fait réalistes (Figure 3-4). Les temps de calcul et d'affichage annoncés sont en contrepartie de plusieurs heures en raison de la quantité importante de calculs.

La modélisation par des particules mobiles se caractérise finalement par une aptitude particulière à modéliser des séquences animées.

3.1.3 Description par voxels

La modularité des plantes peut être exploitée indirectement en subdivisant l'espace qui contient la plante en cellules régulières, appelées *voxels* (abréviation de *volume elements*). L'ensemble des voxels ainsi constitué forme une grille de l'espace contenant la plante. Dans ces

FIG. 3-4 – Résultat de la représentation d'une forêt par un système de particules [Reeves and Blau, 1985].

représentations, les composants de la plante ne sont pas directement considérés. A la place, la plante est représentée par les voxels qui contiennent ou intersectent des composants. Des attributs biologiques peuvent être attachés à chaque voxel pour caractériser leur contenu végétal (densité de feuille, propriétés optiques, *etc.*). La plante est représentée *in fine* par un ensemble de voxels dans un espace 3D. La taille des voxels qui détermine la précision des résultats est variable d'une application à l'autre. Les représentations basées sur les voxels ont été utilisées dans le contexte de la modélisation de l'interception de la lumière [Sinoquet and Bonhomme, 1992], et de la simulation de la croissance des plantes [Greene, 1989, Blaise, 1991].

Dans ce deuxième contexte, on trouve l'utilisation de telles représentations notamment dans le travail de Greene [Greene, 1989] qui réalise la croissance d'une arborescence botanique dans un espace voxel en fonction de contraintes environnementales telles que l'ensoleillement ou la présence d'obstacles physiques. Greene utilise des arbres binaires engendrés par un processus stochastique dont l'un des moyens de contrôle est un retour de la géométrie sur la topologie avec pour conséquence un arrêt de leurs croissances. Ces arbres ne sont pas codés explicitement au cours de la modélisation.

L'intérêt de l'utilisation d'un espace discret dans le cadre de la modélisation 3D, est que certaines manipulations sont grandement facilitées par rapport à des espaces plus classiques (euclidiens). On citera le test d'intersection (en testant si deux objets occupent un même voxel) et le test de proximité (en comptant le nombre de voxels intermédiaires) de deux objets complexes. Les opérations effectuées dans un espace voxel sont forcément approximées en raison du caractère discret inhérent à l'espace, mais les avantages en vitesse et en facilité de travail sont déterminants. Ceci est le cas pour la réalisation d'automates de croissance sensitifs à l'environnement (les obstacles doivent être évités, la proximité de certains objets peut inhiber ou favoriser la croissance, l'évolution est modulée par la lumière).

L'algorithme de croissance utilisé par Greene consiste dans un premier temps, à initialiser l'espace voxel, à définir ensuite des voxels sources, et ensuite, à partir de ces voxels, à générer des éléments géométriques (des branches) selon un ensemble de contraintes. Ces contraintes sont de deux types, soit géométriques pures comme celles sur la longueur et l'épaisseur des arêtes ou les valeurs des angles de branchement (utilisation de variables aléatoires), soit environnementales qui sont à leur tour de deux types :

- les éléments générés ne doivent pas intersecter ou être trop proches d'objets déjà existants dans la scène,
- l'illumination doit être suffisante pour qu'un nœud puisse évoluer.

Des tentatives de génération sont effectuées jusqu'à trouver une position acceptable dans la limite d'un nombre d'essais maximum donné (voir la Figure 3-5).

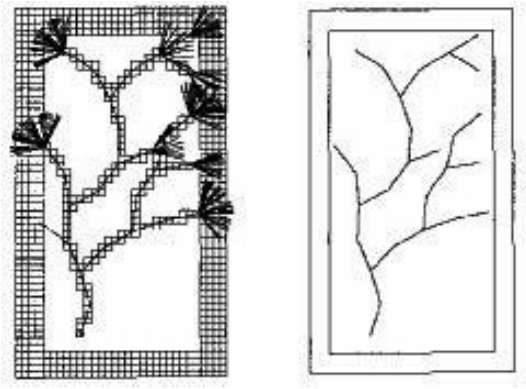


FIG. 3-5 – Retour de la géométrie sur la topologie pour la croissance. A gauche, des tentatives de génération infructueuses forment des éventails et marquent les arrêts dans les ramifications. A droite, structure résultante. Tiré de [Greene, 1989].

Les images présentées par Greene (Figure 3-6) ont été obtenues par des techniques classiques de rendu par facettes. La gestion des couleurs attribuées à ces facettes s'effectue par codage direct au niveau des voxels. La taille de l'espace voxel est comprise entre 150x150x150 et 300x300x300, la scène est donc décrite par 3 à 27 millions de voxels. Logiquement, le temps de construction des arborescences croît proportionnellement avec le nombre de voxels dans l'espace, les temps signalés à l'époque étaient d'environ 30 heures de calcul sur une station Sun4.

Le dessin d'une image peut représenter la gestion de plusieurs centaines de milliers de facettes, mais les résultats obtenus sont très réalistes. La génération de l'arborescence s'accompagne de la génération de feuilles polygonales. La gestion des bâtiments à la base même de la méthode apporte un grand avantage par rapport à toutes les autres techniques pour la réalisation de paysages composites.

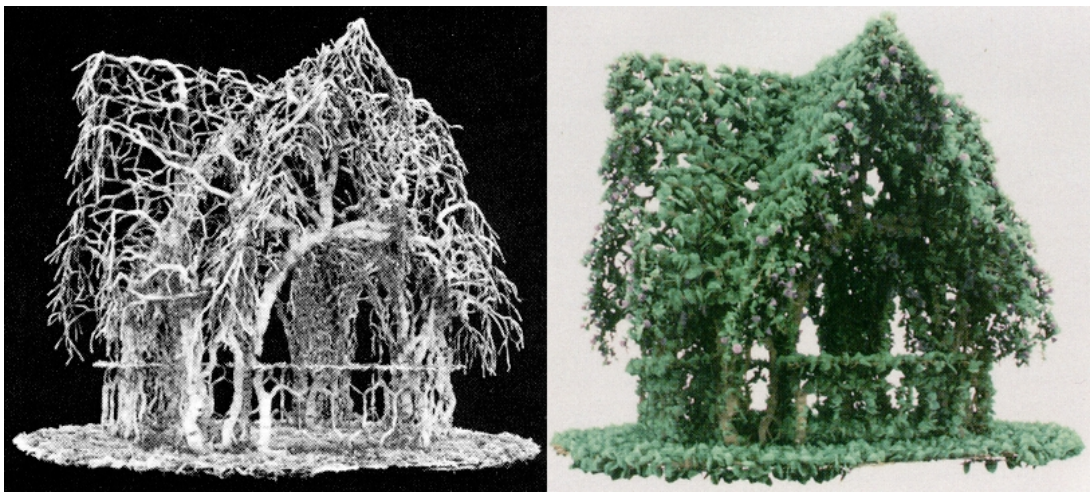


FIG. 3-6 – Modèle "Architecture Organique" présenté à Siggraph'88 [Greene, 1989]. Croissance autour d'une maison, sans les feuilles (gauche) et avec (droite). L'espace voxel est utilisé pour gérer l'interception lumineuse et la collision avec les éléments de l'environement. L'affichage correspond par contre à une représentation géométrique classique, sous forme de polygones texturés.

Le travail de Blaise [Blaise, 1991] est fondé sur le même principe d'utilisation d'un espace voxel pour gérer les évènements extérieurs à la plante. A partir du modèle de croissance de de Reffye *et al.* ([Reffye et al., 1988], voir section 3.1.4), il permet la simulation de contraintes d'ensoleillement ou de contraintes mécaniques (la présence de bâtiments ou la compétition pour la lumière entre branches ou entre plantes peuvent inhiber la croissance de certains composants).

3.1.4 Descriptions polygonales

L'évolution des cartes graphiques favorisant l'affichage d'un nombre toujours plus important de triangles, les descriptions polygonales sont un type de modèle géométrique simple fréquemment utilisé en infographie. De nombreuses approches sont donc fondées sur ce type de description soit pour modéliser l'objet, soit pour finalement l'afficher (les modèles paramétriques sont généralement triangulés pour être affichés). Nous présentons dans cette section une des approches de modélisation de la géométrie des plantes construite sur des symboles polygonaux.

Jaeger [Jaeger, 1987] et de Reffye *et al.* [Reffye et al., 1988] définissent un modèle de croissance de l'architecture végétale générant des structures polygonales réalistes de plantes.

Ils modélisent l'évolution d'un arbre par un processus de croissance discret basé sur l'activité des méristèmes et faisant intervenir la croissance, la ramification et la mort de ces méristèmes. La croissance de l'arbre est rythmée par les tops d'une horloge. Une unité de temps dans ce modèle correspond au temps de mise en place d'une unité de croissance. Cette durée est supposée constante pour tous les axes de même ordre, ce qui implique que la croissance peut être plus ou moins rapide suivant les axes. Deux probabilités sont attribuées à chaque méristème : une probabilité de mort (pour un méristème latéral cette probabilité est la probabilité de non ramification), et une probabilité de pause.

Un modèle unique appelé *axe de référence* contient une suite d'étapes de développement d'un méristème correspondant aux stades successifs que suivra tout méristème de la plante à partir d'un stade donné. Différentes probabilités définissent le passage d'un état au suivant.

Pour une espèce particulière, ces probabilités ne sont fonction que de l'âge, la dimension et l'ordre du méristème considéré. Pour simuler la croissance d'une plante un certain nombre d'autres paramètres doivent encore être définis : les vitesses de croissance des axes (en fonction de l'ordre), le nombre de méristèmes à chaque nœud (en fonction de l'ordre), la probabilité de réitération sur un méristème (fonction des mêmes paramètres que les probabilités précédentes) et finalement l'âge de l'arbre que l'on veut générer.

Le résultat de l'exécution du générateur de plantes est une *ligne élastique* contenant les représentations des différents composants botaniques générés. Une ligne élastique est définie comme une liste de constituants et leurs différents caractères géométriques associés. La forme de chaque constituant est référencée par un identifiant qui caractérise le type du composant (entre-nœud court ou long, feuille, fruit, ...). A chacun de ces identifiants correspond un symbole polygonal dans une bibliothèque de formes, par exemple un cylindre à base hexagonale pour un entre-nœud. Chaque constituant contient aussi les paramètres des transformations nécessaires pour placer, orienter et dimensionner ce symbole (Figure 3-7). Ces paramètres de transformation sont stockés sous la forme d'une matrice 3*4 qui définit les directions principales, secondaires et normales du constituant, les facteurs d'échelle selon ces 3 directions et la position dans le référentiel absolu de la scène. Une transformation non affine de type "taper" est ensuite définie par deux facteurs d'échelle : ces deux valeurs donnent explicitement des facteurs d'échelle pour les plans perpendiculaires à la direction principale à la base et au sommet du symbole. Les facteurs d'échelle pour les plans intermédiaires sont calculés par interpolation linéaire des deux valeurs. Cette transformation permet notamment d'obtenir des troncs de cône à partir d'un cylindre. Elle permet principalement de gérer le recollement des entre-nœuds en faisant correspondre le diamètre sommet d'un entre-nœud avec le diamètre base de l'entre-nœud suivant.

Une ligne élastique est donc un graphe de scène simple où chaque élément est constitué

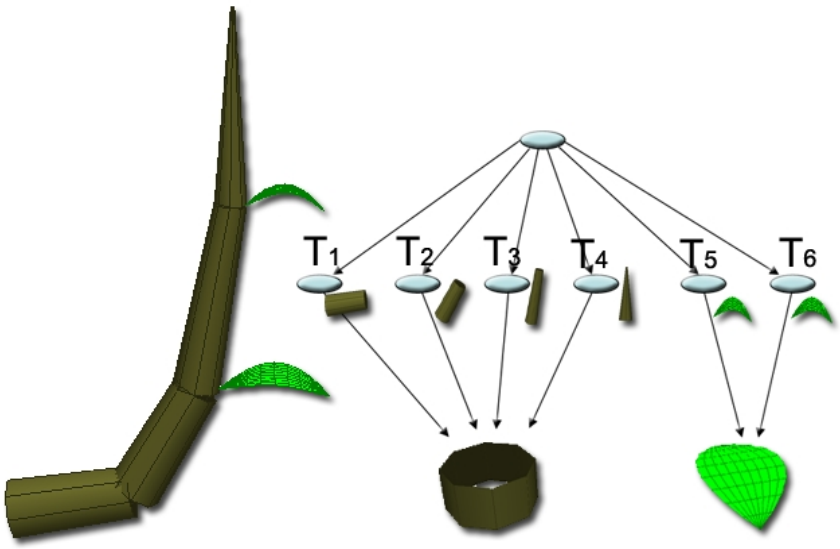


FIG. 3-7 – Une ligne élastique [Jaeger, 1987] : à gauche, une représentation et à droite le graphe de scène correspondant. Le sommet le plus haut du graphe représente la scene entière, les sommets intermédiaires les constituants de la plante. Quatre constituants, correspondant aux entre-nœuds, sont des transformations du même cylindre unitaire et les deux autres des transformations du même symbole de feuille. Les transformations utilisées correspondent à des orientations, des mises à l'échelle et des translations. Pour les quatres premiers constituants, des transformations de type "taper" sont également appliquées.

de transformations et réutilise une forme prédéfinie (Figure 3-7 droite). Différents opérateurs, comme un opérateur de simplification de la géométrie, ont été développés. Cet opérateur fusionne les symboles des constituants de caractères analogues s'ils sont alignés en un seul. Il permet, par exemple, de représenter un tronc rectiligne constitué de plusieurs cylindres par un unique cylindre.

3.1.5 Représentation géométrique des arborescences axiales

Les chaînes parenthésées ont été introduites pour représenter la topologie des structures ramifiées [Prusinkiewicz et al., 1997]. Pour représenter la forme exprimée par des propriétés telles que l'orientation des branches, la longueur des entre-nœuds, une interprétation géométrique doit être assignée aux chaînes. Dans des cas simples, les informations géométriques peuvent être extérieures à la chaîne. Par exemple pour dessiner des structures arborescentes 2D, des règles peuvent définir que les angles de branchement des axes sont de valeur constante dans des directions alternatives, de gauche à droite, et tous les segments sont de même longueur. Les angles de branchement peuvent aussi être fonction de l'ordre de ramification. Malheureusement, des règles si simples, s'appliquant uniformément sur toute la chaîne, ne sont pas suffisamment flexibles pour exprimer toute la variété des formes ramifiées que l'on trouve dans la nature.

Aussi, des techniques plus souples ont été développées, basées sur l'intégration explicite d'informations géométriques dans les chaînes. Certains symboles, avec ou sans paramètres, sont réservés pour représenter des propriétés géométriques de la structure décrite. La correspondance bijective entre les symboles de la chaîne et les modules de la structure décrite est perdue, mais les détails de la structure peuvent être exprimés. L'*interprétation par la tortue*, introduite par Szilard et Quinton [Szilard and Quinton, 1979], puis étendue par Prusinkiewicz [Prusinkiewicz, 1986,

Prusinkiewicz, 1987] et Hanan [Hanan, 1988, Hanan, 1992] au cas des structures ramifiées, est représentative de cette approche.

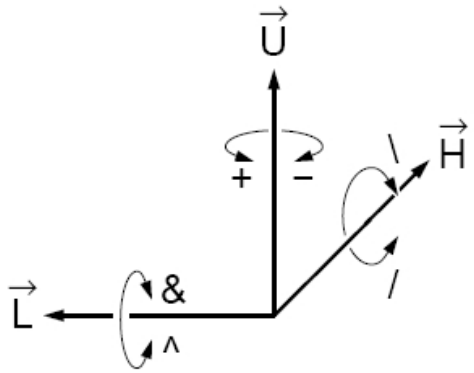


FIG. 3-8 – La tortue et les différentes opérations de rotations en 3D qui la controlent. Tiré de [Prusinkiewicz et al., 1988].

La chaîne à interpréter est scannée séquentiellement de gauche à droite, et ces symboles consécutifs sont interprétés comme des commandes qui manoeuvrent une tortue de style LOGO [Abelson and diSessa, 1984] en trois dimensions. La tortue est représentée par son état, qui consiste en une *position* et une *orientation* dans le système de coordonnées cartésien, ainsi que des attributs supplémentaires comme la couleur courante et l'épaisseur de trait. La position est définie par un vecteur P, et l'orientation par trois vecteurs H, L, et U, indiquant la *direction* (Heading) et la *droite* (Left) et le *haut* (Up). Ces vecteurs sont unitaires, perpendiculaires les uns aux autres, et satisfont l'équation $H \times L = U$. En conséquence, les rotations de la tortue peuvent s'exprimer par l'équation

$$\left[H'L'U'\right] = [HLU]\,R$$

où R est une matrice de rotation 3×3 et H', L' et U' l'orientation de la tortue après la rotation. Plus spécifiquement, les rotations d'un angle α autour des vecteurs U, L, et H peuvent être représentées sous forme matricielle :

$$R_U(\alpha) = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$R_L(\alpha) = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix},$$

$$R_H(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

La tortue est initialement située à l'origine du système de coordonnées cartésien, avec le vecteur de direction H pointant dans la direction positive de l'axe des y, et le vecteur droit L pointant dans la direction négative de l'axe des x. Les actions de la tortue et ses changements d'état sont causés par l'interprétation de symboles spécifiques, chacun pouvant être suivi par des paramètres. Si un ou plusieurs paramètres sont présents, la valeur du premier paramètre affecte l'état de la tortue. Si le symbole n'est suivi par aucun paramètre, des valeurs par défaut spécifiées à l'extérieur du L-système sont utilisées.

On peut noter que, dans cette approche, les positions et orientations des éléments sont définies relativement à celles de leurs prédécesseurs, ce qui rend leurs spécifications généralement plus simples.

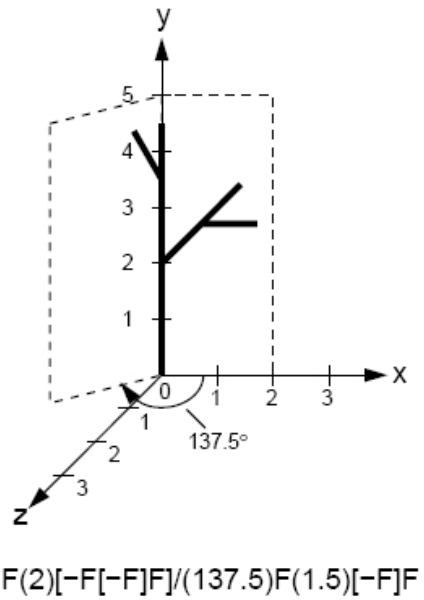


FIG. 3-9 – Exemple d'interprétation par la tortue d'une chaîne de caractères. La valeur par défaut des segments représentant les symboles F sans paramètre est 1, et la valeur par défaut des angles des symboles / et − est 45. Tiré de [Prusinkiewicz et al., 1997].

Un exemple d'interprétation par la tortue d'une chaîne de caractères est donné à la Figure 3-9. Les symboles F sont interprétés par le tracé d'un segment de droite, la longueur de ce segment étant donnée en paramètre ou par défaut égal à 1. Les symboles / et − correspondent respectivement à des rotations de la tortue autour des axes U et H. Lorsqu'elle lit un symbole [, la tortue enregistre son état dans une pile et le restaure avec le symbole]. Nous donnons, dans l'annexe B, l'ensemble des symboles interprétés par la tortue classiquement utilisés [Karwowski, 2002].

La notation par chaînes parenthésées peut être appliquée, par exemple, pour enregistrer la structure de plantes digitalisées [Constable, 1991, McClelland, 1996]. La plupart de ces applications sont reliées à la modélisation et la simulation du développement des plantes avec les L-systèmes (voir chapitre 7).

3.2 Continuité des représentations géométriques

Les modèles présentés précédemment étaient construits comme l'assemblage de modèles géométriques représentant les différents composants d'une plante. Dans leurs définitions orignales, ces modèles ne se posaient pas le problème de la connection géométrique entre composants ad-

jacents. Cette adjacence entre composants, donnée par le modèle topologique sous-jacent, peut en effet être utilisée pour connecter géométriquement les modèles des composants.

Nous présentons ici différentes méthodes, basées sur différentes primitives géométriques, pour connecter géométriquement les représentations des composants adjacents de la structure d'une plante.

3.2.1 Représentation par des primitives simples

Kawaguchi [Kawaguchi, 1982] modélise des formes géométriques du type coraux, cornes, *etc.*, à partir de primitives géométriques simples de forme cylindrique et de règles de collage de ces primitives. L'élément de base de la construction est un prisme constitué de deux bases (supérieure et inférieure) ayant des formes homothétiques à une forme donnée (souvent une ellipse) (Figure 3-10).

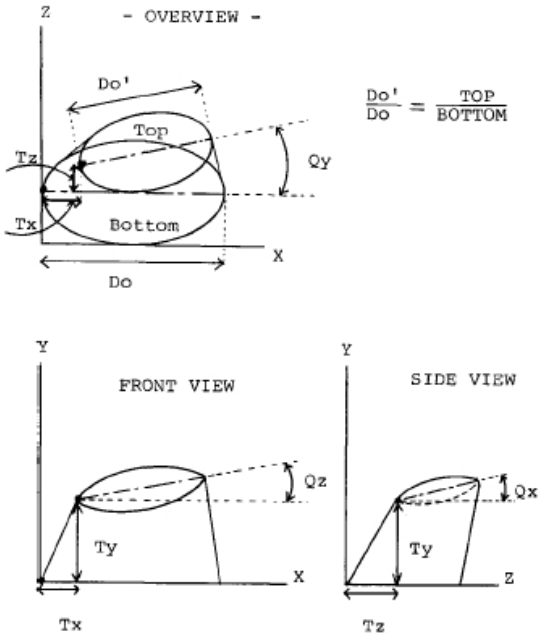


FIG. 3-10 – Géométrie et paramètres d'une primitive élémentaire selon Kawaguchi [Kawaguchi, 1982].

La position et la taille relative de la base supérieure par rapport à la base inférieure sont données par :

- un rapport appelé taux de croissance,
- un vecteur d'inclinaison (q_x, q_y, q_z) donnant en projection dans les plans orthogonaux aux axes, les trois angles entre les deux plans des bases inférieure et supérieure,
- un vecteur (T_x, T_y, T_z) de translation définissant le déplacement de la base supérieure par rapport à la base inférieure.

L'accumulation récursive de ces prismes permet d'obtenir des cornes telles que celles présentées dans [Kawaguchi, 1982] (Figure 3-12).

Pour générer des structures arborescentes, une notion de ramification est introduite, constituée par un prisme ayant une base ellipsoïdale inférieure et deux bases ellipsoïdales supérieures dont les grands-axes sont orthogonaux (Figure 3-11). Des cylindres à section ellipsoïdale sont alors liés comme le montre la Figure 3-11 aux bases de ce prisme. Nous notons donc que dans

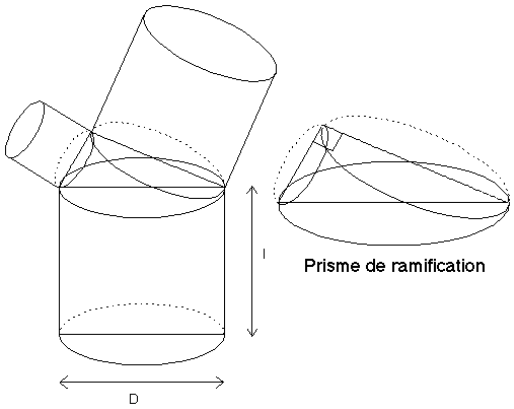


FIG. 3-11 – Cas de la ramification selon Kawagushi. Tiré de [Janey, 1992].

cette méthode la représentation est $C0$ continue[1]. De nombreuses autres méthodes telles que celle de de Reffye (section 3.1.4) ont adapté leurs représentations pour avoir le même type de continuité.

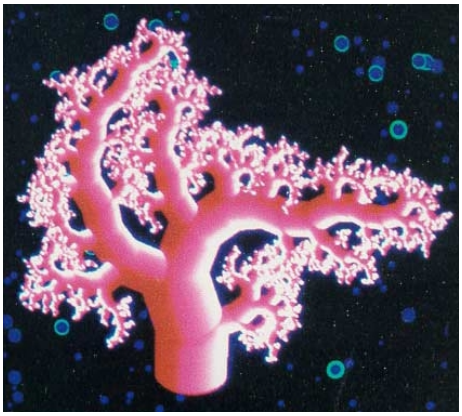


FIG. 3-12 – Un corail construit comme une accumulation de cylindres à base ellipsoïdale [Kawaguchi, 1982].

Le modèle est défini par des rapports constants hauteur sur base et diamètre base sur diamètre sommet pour tous les cylindres. La génération récursive de l'arborescence est stoppée lorsqu'une condition géométrique est vérifiée (par exemple l'épaisseur des cylindres devient inférieure à une valeur limite).

3.2.2 Représentation par des modèles paramétrés

Bloomenthal [Bloomenthal, 1985] propose d'utiliser des B-splines (voir [Piegl and Tiller, 1997] pour plus de détails) pour modéliser la géométrie du bois de structures ramifiées. Le modèle de Bloomenthal part d'un arbre topologique préalablement défini appelé squelette où chaque nœud contient un point 3D. Ce squelette est construit en interpolant chaque suite de n arêtes ($n+1$ nœuds d'embranchement) par une B-spline $C2$ cubique (soit n segments splines consécutifs) de

[1] continuité des modèles, mais discontinuité des tangentes aux points limites entre deux modèles.

façon à obtenir le prolongement par continuité des dérivées première et seconde aux $n-1$ nœuds d'embranchement intermédiaires. Le squelette de l'arbre étant défini par des points interpolés, les points de contrôle conventionnels ne sont pas utilisés pour manipuler les splines. A la place, une branche sera représentée par les coefficients des n polynômes cubiques qui définissent la B-spline. Ces coefficients sont déterminés par les positions et les tangentes aux $n+1$ nœuds.

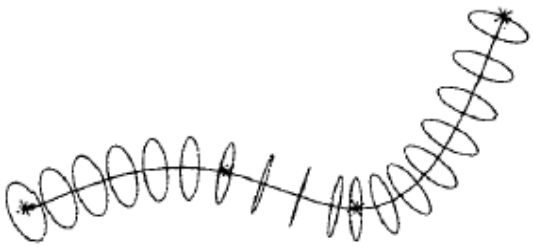


FIG. 3-13 – Représentation d'une branche par un cylindre généralisé [Bloomenthal, 1985]. Trois segments splines interpolent des points donnés (croix) avec une continuité $C2$. plusieurs sections sont évaluées le long de la courbe.

La modélisation volumique de l'arbre est réalisée en considérant chaque branche comme un cylindre généralisé dont l'axe est la courbe B-spline précitée et dont la section est un cercle de rayon variable (Figure 3-13). Pour polygoniser la surface d'un cylindre généralisé, un certain nombre de sections sont évaluées le long de l'axe et reconnectées. La modélisation des embranchements est réalisée par des surfaces du type "selle" obtenues en considérant les trois disques représentant les sections circulaires des branches (la mère et ses deux filles). Les deux sections filles sont reliées par une première spline, appelée "selle", (Figure 3-14 gauche, courbe en noir) en leurs points les plus proches et en fonction des tangentes des squelettes à ces sections. Les points opposés sur les sections filles sont ensuite reliés de la même manière à la section mère (courbes rouges). Une spline (courbe verte) composée de deux segments relie deux points opposés de la section mère à la courbe selle (en son point le plus éloigné du segment reliant ses deux extrémités). Finalement des courbes splines additionnelles sont construites entre les sections filles (passant par la courbe verte) et entre la section mère et les sections filles, comme montrées sur la Figure 3-14 droite.

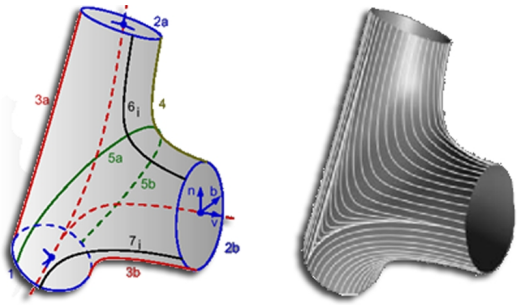


FIG. 3-14 – Modélisation d'un embranchement selon Bloomenthal [Bloomenthal, 1985]. Les points de trois sections sont reliés par interpolation avec des courbes splines.

Les angles de branchement entre branches, leurs longueurs et épaisseurs sont choisies aléatoirement dans des rangs de valeurs définis en fonction du développement de l'arbre.

Dans cette approche le rendu est tout particulièrement étudié. Par exemple les veines de l'écorce sont obtenues par une technique de modulation de la normale ("Bump Mapping") calculée en fonction d'une image aux rayons X du moulage en plâtre d'un morceau d'écorce réel. On crée ainsi un carreau de texture d'écorce qui est ensuite plaqué sur le tronc. Le problème du recollement des bords de deux carreaux de texture connexes est résolu par "mélange" des deux carreaux sur une bande frontière. Les feuilles, elles, sont obtenues par digitalisation de feuilles d'érable réelles et par plaquage de cette image digitalisée sur des surfaces polyédriques. Le nombre de feuilles dessinées sur une branche est déterminé par une loi d'allométrie simple relative à l'épaisseur de la branche. Le positionnement d'une feuille dans l'espace est fonction de la direction de la branche qui la porte. La gestion des feuilles et d'une texture permet ainsi l'obtention d'images en gros plan très réalistes (Figure 3-15).

FIG. 3-15 – Résultat des techniques développées par Bloomenthal [Bloomenthal, 1985] : Un érable.

Dans cette approche, l'utilisation de cylindres généralisés et de surface "selle" permet donc d'obtenir des représentations $C2$ continue. Ces méthodes ont aussi été adaptées à d'autres modèles comme celles des représentations des arborescences axiales.

3.2.3 Les surfaces implicites

Le problème de la représentation des branches et de leurs connections semble finalement trouver une solution simple avec l'utilisation de surfaces implicites. Nous donnons ici quelques définitions, tirées de [Desbrun, 1997], pour illustrer leur utilisation pour modéliser des formes ramifiées.

Une surface implicite est définie comme l'ensemble des points vérifiant une équation :

$$\mathcal{S} = \{P | f(P) = iso\}$$

où iso est une valeur réelle appelée isovaleur, et f une fonction quelconque de $\mathbb{R}^3$ dans $\mathbb{R}$, appelée potentiel. Les points intérieurs à l'objet peuvent être définis par la propriété :

$$f(P) \geq iso$$

L'extérieur possède alors la caractérisation duale : $f(P) < iso$. On dispose ainsi d'une fonction dedans/dehors, facilitant le rendu d'images composées d'objets implicites. Différents types de fonctions potentiel f pour modéliser des formes ramifiées ont été proposés dans la littérature et sont présentés ici.

Potentiel à squelette Bloomenthal et Wyvill [Bloomenthal and Wyvill, 1990] proposent de définir des surfaces implicites engendrées par des squelettes. Ces squelettes peuvent être n'importe quelle primitive géométrique pour laquelle on sait calculer la distance à un point. Chaque squelette crée dans l'espace un potentiel qui décroît avec la distance. On considère alors qu'à chaque point de l'espace est associé un potentiel, résultant de la somme des potentiels créés par chacun des squelettes. Une définition des surfaces définies par un ensemble de squelettes s_i est alors :

$$S = \left\{ P \in \mathbb{R}^3 | \sum_{s_i} f_i(r_i) = iso \right\}$$

où r_i est la distance de P à s_i et f_i est une fonction décroissante.

Un exemple de surface implicite défini par squelette est donné à la Figure 3-16. On remarque que la caractérisation de l'intérieur de l'objet reste identique, et que la construction d'objets même complexes est assez intuitive. Un squelette ponctuel donnera une sphère, un segment donnera un cylindre arrondi aux extrémités, *etc.* Le mélange de squelettes fournira un moyen simple et économique de création d'objets complexes. Une fonction potentiel f_i peut être modélisée par toute fonction décroissante à support compact. Le rayon d'influence d'un squelette ne changera pas toute la forme d'un objet, mais seulement une zone locale, ce qui permet une construction squelette par squelette d'objets complexes.

On appellera e l'épaisseur d'"enrobage" que représente une fonction potentiel, c'est-à-dire le rayon qu'aurait la sphère engendrée par un squelette ponctuel muni de cette fonction ($e = f^{-1}(iso)$).

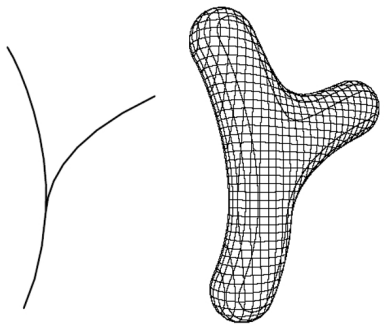


FIG. 3-16 – Un squelette et la surface implicite correspondante. Tiré de [Bloomenthal and Wyvill, 1990].

Potentiel procédural Bloomenthal [Bloomenthal and Wyvill, 1990] fait aussi remarquer que le potentiel peut être défini procéduralement, comme une boîte noire. Ceci peut d'ailleurs simplifier la tâche de l'utilisateur dans quelques cas. Par exemple, si l'on souhaite un embranchement lisse entre deux branches, on peut modifier le calcul de la distance aux squelettes en utilisant la méthode géométrique de la figure 3-17 pour arriver à nos fins. Cette formulation n'est $C1$ que pour des angles d'embranchement inférieurs à $\pi/2$. Une solution plus générale à ce problème spécifique peut être trouvée dans [Ferley et al., 1997].

Potentiel intégral Bloomenthal et Shoemake [Bloomenthal and Shoemake, 1991] ont proposé un type de surfaces implicites encore différent, les surfaces de convolution. A la base de cette modélisation, le problème des bosses (bulges en anglais) : si deux squelettes segments se

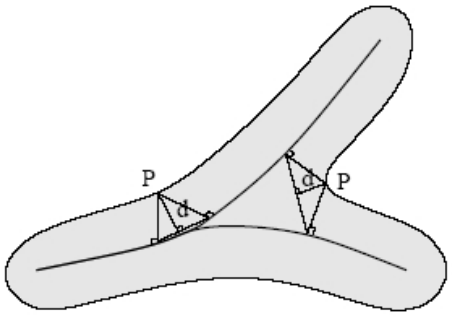


FIG. 3-17 – Construction géométrique pour un embranchement lisse par un surface définie par un fonction potentiel procédurale [Bloomenthal and Wyvill, 1990]. La valeur de la fonction potentiel pour un point P dépend de la distance d. Tiré de [Desbrun, 1997].

coupent en croix, la surface implicite générée par les modèles précédents aura une bosse, autrement dit une épaisseur plus importante à la jonction des deux segments (voir figure 3-18). Pour éliminer ce défaut, il est possible d'écrire le potentiel sous forme de sommes d'intégrales :

$$f(P) = \sum_{s_i} \int_{s_i} e^{-||P-X||^2/2} dX$$

Cette formulation est en fait une convolution du squelette par une gaussienne, permettant de tenir compte des contributions tout le long du squelette. Comme la convolution est un opérateur linéaire, la somme de convolutions est équivalente à la convolution de la somme des squelettes ; il n'y donc pas de bosse à l'intersection de deux squelettes en croix.

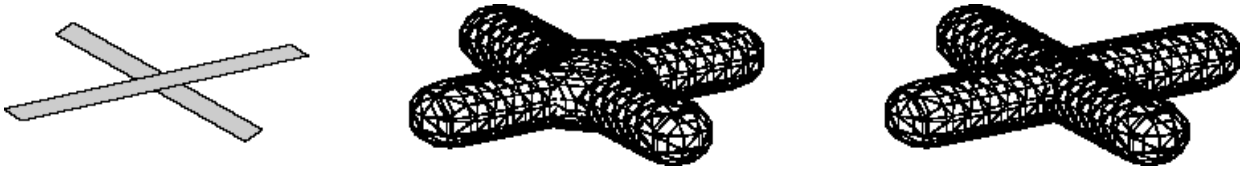


FIG. 3-18 – Un squelette, la forme bosselé qu'il engendre, et sa version convoluée. Tiré de [Desbrun, 1997].

Ce type de surface nécessite plus de calculs, mais peut s'implémenter tout de même de façon efficace, d'autant plus que n'importe quelle fonction de filtre, plus simple que l'exponentielle, peut être utilisée. Cette technique n'est cependant utilisée qu'en dernier recours si l'on ne peut éviter l'utilisation d'une intégrale.

D'autres types de fonctions potentielles existent, pour une revue plus détaillée voir [Desbrun, 1997]. Les surfaces implicites possèdent donc de bonnes propriétés pour la modélisation de formes ramifiées. Grâce aux divers types de potentiels existants, et en particulier aux potentiels engendrés par squelettes, la modélisation est rapide et intuitive. Contrairement aux surfaces paramétriques tels que les B-splines, les surfaces implicites gérent simplement les recollements aux embranchements dans leurs définitions même. Finalement, il est aisé de déterminer si un point est à l'intérieur ou hors d'un objet implicite, ce qui permet d'accélèrer certains calculs tels que la détection de collisions et le rendu par lancer de rayons.

3.3 Représentations géométriques multi-échelles

Finalement, pour maîtriser le coût des calculs d'affichage ou ceux de radiosité sur des structures géométriques complexes telles que celles des plantes, différents travaux organisent leurs modèles géométriques à différentes échelles. Les algorithmes de rendu par exemple (qui calculent l'illumination parvenant à l'oeil) opérant soit par projection des éléments à l'écran (Z-buffer), soit par échantillonnage à l'aide de rayons (ray-tracing), le coût de calcul est croissant avec la quantité d'information (linéairement dans les meilleurs cas). Pour maîtriser ce coût, il est souhaitable d'utiliser une représentation des objets à l'échelle à laquelle on les voit ou manipule. Ce qui conduit donc à définir des représentations multi-échelles.

3.3.1 Représentations fractales

L'autosimilarité que l'on trouve dans certaines structures ramifiées a conduit différents travaux à modéliser la géométrie de l'architecture des plantes comme des objets fractals. Dans ces méthodes, à chaque pas le processus de construction définit des approximations géométriques de l'objet autosimilaire de plus en plus précises et donc à des échelles de plus en plus fines. Nous présentons dans cette section quelques travaux sur la génération de représentations fractales de plantes.

Modèle de Oppenheimer

Les travaux d'Oppenheimer [Oppenheimer, 1986] font un lien entre les travaux de Kawaguchi et les méthodes fractales. La structure topologique sous-jacente est constituée d'arbres binaires ou ternaires résultant d'un processus de génération aléatoire assez élémentaire faisant intervenir le nombre de branchements sur chaque segment de branche principal et permettant d'obtenir des arbres auto-similaires (Figure 3-19).

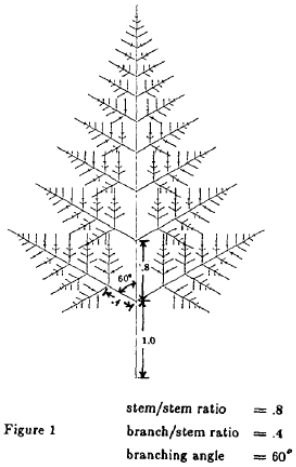


FIG. 3-19 – Topologie et représentation de certains ratios dans le modèle de Oppenheimer [Oppenheimer, 1986].

Oppenheimer construit récursivement les solides 3D en formes de cornes ou spiralées à partir de quatre primitives géométriques simples (cylindre, spirale, hélice et tortillon), de règles de réduction et de collage faisant intervenir les paramètres suivants :

- l'angle entre un segment principal et les arêtes qui s'y connectent,
- le rapport de longueur entre les arêtes principales d'un branchement et les branches qui s'y connectent,

- le rapport de décroissance des arêtes sur un segment,
- la valeur de la rotation hélicoïdale sur les branches (correspondant à la phyllotaxie).

FIG. 3-20 – En perturbant aléatoirement les angles de branchement, Oppenheimer [Oppenheimer, 1986] obtient des modèles visuellement réalistes de systèmes ramifiés.

Pour le rendu, Oppenheimer utilise des vecteurs "anti-aliasés" pour les branches les plus fines, et des prismes polygonaux pour les autres. L'ombrage des facettes est géré. Comme chez Bloomenthal [Bloomenthal, 1985], les veines de l'écorce sont réalisées par modulation de la normale, modulation guidée par un mouvement Brownien fractal en forme de vague (Figure 3-20).

Systèmes de fonctions itérées

Les *systèmes de fonctions itérées* (Iterated Function Systems, IFS) ont été définis pour générer des objets fractals [Barnsley, 1988]. Cette technique a été particulièrement appliquée à la génération et la compression d'images de végétaux [Demko et al., 1985, Barnsley et al., 1988].

On peut construire de manière extrêmement simple des structures fractales, en utilisant un processus récurrent composé d'un initiateur (l'état initial) et d'un générateur (l'opération de récurrence), généralement un IFS. Un IFS est un ensemble fini de fonctions affines strictement contractantes $\{W_n, n \in [1, N]\}$ du plan dans lui même. Un IFS correspond à la transformation W qui à tout compact (ensemble fermé et borné) B du plan associe $W(B)$ tel que :

$$W(B) = \bigcup_{n=1}^{N} W_n(B)$$

L'*attracteur* de l'IFS, correspondant à l'objet fractal, est l'unique point fixe A de W sur l'espace complet des parties compactes du plan muni de la distance de Hausdorff ($A = W(A) = \bigcup_{n=1}^{N} W_n(A)$).

L'IFS peut être *probabilisé* en associant une probabilité p_i à chacune des transformations w_i, $i \in [1, N]$, avec $\sum_{i=1}^{N} p_i = 1$.

Le problème inverse de la détermination d'un IFS ayant pour attracteur une image noir et blanc P donnée (représentant une forme végétale dans notre cas), est résolu par Demko *et al.*

[Demko et al., 1985] en introduisant la notion d'auto-pavage, où l'on recouvre au mieux l'objet A avec un nombre minimum N de copies réduites de lui-même appelées sous-objets. Barnsley *et al.* [Barnsley et al., 1988] proposent un algorithme visuel interactif où l'on réalise à la main sur un écran un pavage du modèle A dont on veut calculer un IFS au moyen de N sous-objets de A. Les N w_i correspondant à ces N sous-objets sont automatiquement calculés. Le théorème de collage permet alors de déterminer l'erreur obtenue par rapport à l'image originale en fonction de la précision de l'ajustement obtenu dans le "collage" (Figure 3-21).

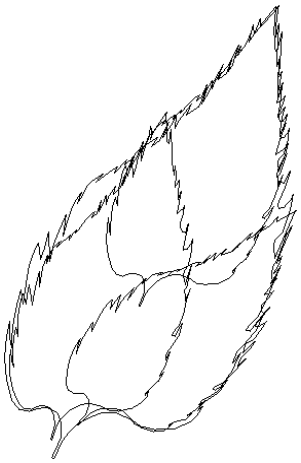


FIG. 3-21 – Théorème du collage utilisé pour reconstituer la géométrie 2D d'une feuille [Barnsley et al., 1988].

Une nouvelle transformation C_B peut aussi être ajoutée à un IFS et permet d'obliger l'attracteur de l'IFS à contenir un ensemble donné appelé *ensemble de condensation* B. Une *hiérarchie d'IFS* est alors définie comme étant une suite ordonnée d'IFS avec ensemble de condensation, chaque ensemble de condensation étant l'attracteur de l'IFS précédent. L'intérêt des hiérarchies d'IFS avec ensembles de condensation est qu'elles permettent de réduire le nombre de fonctions affines nécessaires au rendu d'un paysage complexe, en rendant compte des hiérarchies topologiques entre éléments végétaux (branche $\Longrightarrow$ arbre $\Longrightarrow$ forêt).

FIG. 3-22 – Fougère créée par une IFS [Demko et al., 1985].

Un autre aspect intéressant de la génération d'images par IFS est que le caractère fractal

des images résultantes permet d'obtenir des agrandissements à toute échelle de l'objet modélisé tout en gardant le même code en mémoire et la même précision de rendu. Le calcul itératif de l'image se poursuit jusqu'à l'obtention d'une image suffisamment précise.

Les images obtenues représentent des objets auto-similaires (feuilles d'érable, fougères) comme l'illustre la Figure 3-22. Dans les exemples donnés (images de la côte de Monterey, de la forêt noire, d'un champ de tournesols,... ([Barnsley, 1988]), le nombre d'applications affines des IFS est de 160 à 180. Ce nombre peut être réduit d'un facteur 2 par l'utilisation de hiérarchies d'IFS avec ensemble de condensation construites sur l'organisation hiérarchique des éléments végétaux.

3.3.2 Représentations hiérarchiques

Une technique courante d'optimisation de l'affichage consiste à recourir à des niveaux de détails, revenant à spécifier la géométrie des objets plusieurs fois avec une précision différente, de manière à n'utiliser une représentation complexe que pour des points de vue proches, pour lesquels les détails sont objectivement visibles.

Des limites apparaissent dans ces approches en niveaux de détails. Une première correspond au fait que les détails plus petits qu'un pixel d'image sont supprimés et ne contribuent pas à l'illumination, ce qui ne se vérifie pas dans la réalité [Neyret, 2001] (quelle que soit la taille apparente de ses ondulations, une tôle ondulée reflète globalement la lumière différemment d'une tôle plane). Une deuxième est que la géométrie des objets est modifiée.

Pour une optimisation du rendu, Kajiya [Kajiya, 1985] et Neyret [Neyret, 2001] suggèrent plutôt d'utiliser une hiérarchie de modèles de représentations alternatives en fonction de la distance du point de vue. L'idée est de construire des représentations géométriques minimales selon des critères purement visuels. Pour les objets macroscopiques simples, l'illumination varie dans l'image plus lentement que la géométrie, mais dans ce cas cette géométrie est invisible en elle-même et on ne voit que ce qui "émerge". Dès lors, l'utilisation de représentations alternatives permet de représenter directement les phénomènes visuels émergents, ou du moins quelque chose qui permet de les produire rapidement, et à "oublier" (ou coder séparément à l'échelle utile) la géométrie. De nombreuses voies pour fonder des représentations alternatives aux polygones à différentes échelles existent.

Cette section tente de faire la liste des différentes représentations hiérarchiques utilisées dans le cadre de la représentation des plantes.

Simplification polygonale

Une technique classique pour créer des représentations en niveaux de détails repose sur la simplification automatique de polygones [Popovic and Hoppe, 1997, Hoppe, 1998]. Cette technique repose sur des opérations de division/fusion des sommets d'un maillage triangulaire, permettant d'affiner ou simplifier les triangles. Cependant, comme le souligne Deussen [Deussen et al., 2002], si ces techniques donnent de très bons résultats sur des objets continus tel que les terrains, elles offrent cependant des résultats limités sur la géométrie des plantes.

Une adaptation de ces techniques aux feuillages est proposée par Remolar *et al.* [Remolar et al., 2003] puis amélioré par Zhang et Blaise [Zhang and Blaise, 2003]. L'aspect général d'une plante doit beaucoup à la représentation de ses feuilles, généralement distribuées de manière discontinue dans l'espace. Dans cette méthode, les feuilles sont regroupées deux à deux en fonction de leurs similarités. Pour juger de la similarité de deux feuilles, des critères liés à la position, l'aire, l'âge et la normale sont utilisés. Grâce à ces regroupements, le feuillage est organisé sous la forme d'une hiérarchie binaire. Pour simplifier la représentation, les polygones (des quadrilatères) représentant des feuilles similaires sont fusionnés deux à deux. Pour cela, parmi les huit sommets définissant deux quadrilatères, les quatre sommets qui forment un quadrilatère d'aire maximum sont choisis. La figure 3-23 illustre les résultats obtenus par cette méthode.

FIG. 3-23 – Simplification polygonale de la géométrie des feuilles. Tiré de [Zhang and Blaise, 2003].

Représentation multi-échelles volumique

Marshall *et al.* [Marshall et al., 1997] proposent un système pour le rendu d'un grande collection de plantes paramétrées aléatoirement. Leur système de rendu multi-résolution compile ces modèles de plantes en une hiérarchie de volume basée sur des tétraèdres irréguliers. Ce partitionnement créé un arbre binaire similaire aux arbres BSP, qui peuvent être traversés efficacement. Le modèle de plante utilisé permet de stocker des informations sur la plante à différents niveaux de détails. La génération de la géométrie réelle peut ainsi être reportée jusqu'à ce que cela soit réellement nécessaire. Ce qui permet de réduire de manière importante la consommation en mémoire et le temps d'initialisation, puisque l'arbre binaire ne nécessite pas d'être construit complètement.

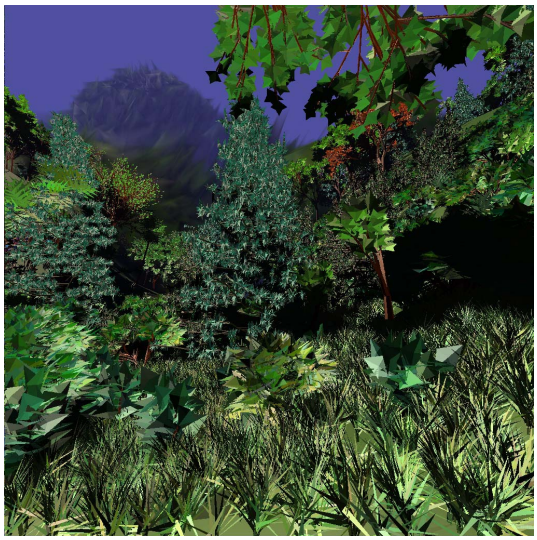


FIG. 3-24 – Rendu d'une scène végétale complexe. La géométrie est stockée dans une hiérarchie de tétraèdres englobants. Tiré de [Marshall et al., 1997].

Le processus de compilation part d'un volume tétraédrique comme première approximation de l'objet, puis le raffine récursivement si nécessaire jusqu'aux polygones individuels. Lorsque l'on calcule l'intersection exacte d'un polygone donné avec un volume tétraédrique, les sous-volumes

résultants ne sont pas forcement des tétraèdres. Marshall *et al.* choisissent une méthode de subdivision d'un tétraèdre en sous-tétraèdres qui dépend du ratio de l'aspect des sous-tétraèdres afin d'éviter des volumes trop effilés et permettre d'obtenir une hiérarchie équilibrée.

Le rendu est fait de manière multi-résolution : pour les objets proches de la caméra, des polygones explicites sont générés, tandis que les objets cachés ou éloignés sont rendus par des microsurfaces approchant le contenu du tétraèdre (Figure 3-24).

Les clusters hiérarchiques

Les méthodes de radiosité ont pour but de calculer l'illumination globale d'une scène. Dans cette méthode, chaque surface a des propriétés radiatives et contribue à l'éclairement de tous les autres éléments de la scène. Le problème de la radiosité se ramène à résoudre un système linéaire qui dépend de facteurs de forme exprimant la contribution d'un élément sur la radiosité d'un autre élément. La résolution de ce système par des méthodes classiques dépend de manière quadratique du nombre d'éléments considérés, et de ce fait est très coûteuse en mémoire et en temps de calcul.

Pour diminuer ce coût, une technique classique en radiosité est l'utilisation de *clusters*, qui vont remplacer un groupe d'objets géométriques par une forme simple ayant des propriétés radiatives équivalentes (calculées par une solution de radiosité locale). Les échanges d'énergie lumineuse entre chaque paire d'éléments de surface sont remplacés et approchés par les échanges d'énergie entre les paires de clusters qui les contiennent. Ce qui entraîne une diminution du nombre de facteurs de forme à calculer. Ce processus de clustering peut être appliqué récursivement sur les clusters eux-mêmes, donnant ainsi une hiérarchie de clusters (Figure 3-26). Cette méthode de clustering hiérarchique permet d'adapter la complexité du système à résoudre. Le choix de la profondeur de la hiérarchie à laquelle on résout le système est un compromis entre précision de la solution et temps de calcul. Une fois que la solution de radiosité est obtenue à un niveau de la hiérarchie, on trouve les solutions aux niveaux plus fins en calculant une solution de radiosité locale à l'intérieur des clusters en fonction de l'illumination qu'ils ont reçue.

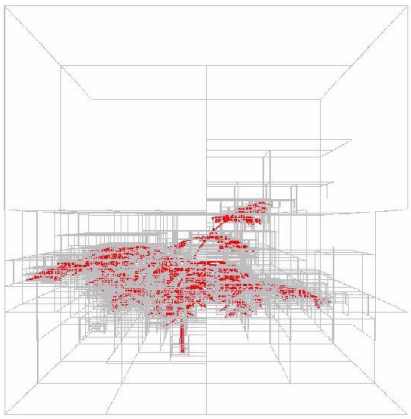


FIG. 3-25 – Espace voxel multi-échelles : l'architecture de la plante est approchée par des voxels (organisés en octree) dont les tailles sont adaptées localement aux irrégularités de la géométrie [Sillion, 1995].

Différentes techniques pour construire automatiquement la hiérarchie de clusters à partir des formes polygonales existent. Une première consiste à décomposer l'espace sous forme d'*octree*. Les octrees sont en fait une extension hiérarchique des grilles de voxels, conçue pour minimiser les besoins en stockage. Cette représentation multi-échelles est obtenue en décomposant l'espace

Euclidien en voxels de tailles variables [Sillion, 1995] (voir figure 3-25). La taille d'un voxel est adaptée à l'irrégularité des formes locales : plus la forme est irrégulière, plus les voxels sont fins.

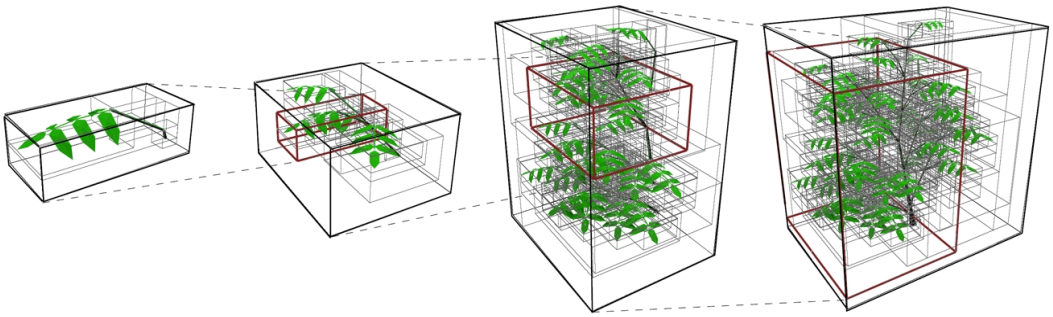


FIG. 3-26 – Représentation des clusters utilisés pour un calcul hiérarchique de la radiosité d'une scène végétale. Ces clusters sont construits comme les boîtes englobantes des modules de la plantes à différentes échelles. Pour limiter le coût en mémoire, ces clusters sont des instances d'un même objet à chaque échelle [Soler et al., 2003].

Si l'objet géométrique est structuré, une deuxième méthode pour construire les clusters consiste à utiliser les boîtes englobantes des modularités de l'objet [Soler et al., 2003] (Figure 3-26). Utilisant le fait que la géométrie d'une plante est une structure répétitive, on peut aussi diminuer le nombre d'objets en instanciant le même cluster en différents endroits de la scène (Figure 3-26). Dans ce cas, le cluster de référence, au lieu d'une géométrie, contient différentes fonctions, simulant l'énergie résultante de l'interaction de la géométrie réelle de l'objet (contenu dans le cluster) avec la lumière provenant de toutes les directions. Ce mécanisme d'instanciation peut se faire à n'importe quel niveau de la hiérarchie.

Les textures bidirectionnelles

Meyer *et al.* [Meyer et al., 2001] ont développé une représentation temps réel à base d'images précalculées (voir section 1.1.2), visant à afficher interactivement des paysages de forêt de manière réaliste, en rendant compte de l'éclairage et des ombres. Le principe consiste à enregistrer, pour de petits objets, l'aspect sous forme d'imagettes pour un échantillonnage représentatif des points de vue et des directions d'éclairage. Comme le montre la figure 3-27, les points de vue sont répartis uniformément sur la sphère englobante de l'objet. Pour chaque point de vue, des images sont calculées pour différentes directions d'éclairage (répartis eux aussi uniformément sur la sphère englobante). Chaque objet est ainsi représenté par un tableau d'images à deux dimensions.

En s'appuyant sur la nature répétitive et récursive des arbres, chaque motif (en l'occurrence une touffe de feuilles ou d'aiguilles) peut être instancié de multiples fois sans coût de stockage supplémentaire, et une hiérarchie de motifs à plusieurs échelles est construite en utilisant comme détails de base les motifs de taille inférieure (Figure 3-27). Lors de l'affichage, les motifs sont restitués sous forme d'imposteurs et choisis suivant les directions du point de vue et de l'éclairage. L'imposteur est en effet calculé pour un point de vue donné en interpolant les neufs images précalculées pour les trois points de vues et suivant les trois directions d'éclairage les plus proches.

Pour pouvoir estimer efficacement les ombres, la visibilité aux divers niveaux hiérarchiques est précalculée au moyen de *cubemaps*, cubes de texture stockant l'accessibilité de la lumière dans toutes les directions. Pour chaque objet, la visibilité de l'objet est échantillonnée pour plusieurs

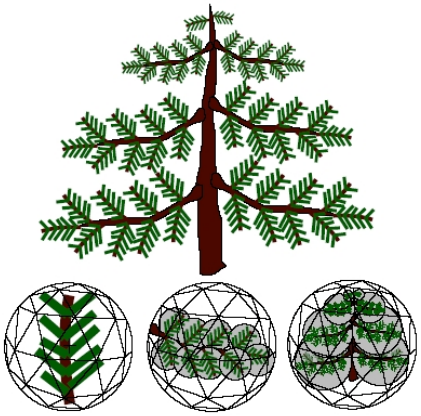


FIG. 3-27 – Hiérarchie de textures bidirectionnelles pour un pin [Meyer et al., 2001]. Les triangles des sphères représentent les différents points de vue pour lesquels une texture bidirectionnelle est calculée.

points de vue (dans ce cas, à partir des huit sommets de la boîte englobante) et stockée dans le cubemap. La visibilité pour un point de vue donné est ensuite estimée en interpolant les images de visibilité des points de vue les plus proches stockés dans le cubemap. De la même manière que les motifs, les différents cubemaps sont finalement organisés sous la forme d'une structure hiérarchique, en utilisant un mécanisme d'instanciation pour minimiser leur nombre.

Les image Z-buffer multicouches

Chen et Williams [Chen and Williams, 1993] puis Max et Ohsaki [Max and Ohsaki, 1995] montrent comment des *images Z-buffer* en projection parallèle précalculées à partir de différentes vues peuvent être reprojetées pour produire des images pour de nouveaux points de vues. Chaque pixel d'une vue calculée est utilisé pour reconstruire un point 3D, en utilisant les coordonnées x et y du pixel, la profondeur du z-buffer et la fonction inverse de la fonction de projection. Le point 3D est ensuite reprojeté suivant le nouveau point de vue.

Dans la suite de cette idée, Max [Max, 1996] propose de calculer des images pour les feuilles et les branches à différents niveaux de la structure hiérarchique d'un arbre, et les combiner de manière adaptative en fonction de la distance du nouveau point de vue. En effet, des parties d'arbres sont remplacées hiérarchiquement par des vues pré-calculées (Figure 3-28). Ces vues contiennent des informations à différentes échelles sur les différentes couches de profondeur des objets représentés dans l'image, leurs couleurs et normales. Un algorithme permet de reconstruire à partir de ces vues particulières de nouvelles vues, avec le niveau de détails requis. L'instanciation de ces vues dans différentes parties de la plante permet de minimiser le nombre de calculs.

Max *et al.* [Max et al., 1999] montrent finalement comment utiliser la carte graphique pour l'algorithme de reconstruction et pour appliquer un calcul d'ombrage sur ces nouvelles vues.

Simplification par points et lignes

Weber et Penn [Weber and Penn, 1995] proposent une représentation en niveaux de détails d'arbres en simplifiant les feuilles par des points et les axes par des lignes. La géométrie d'un élément est dégradée en fonction de la distance de l'objet à la caméra et de la résolution de l'image à calculer. La dégradation est faite par deux mécanismes : le premier est de ne représenter qu'une partie des éléments et le second est d'utiliser des primitives simples telles que des points ou des

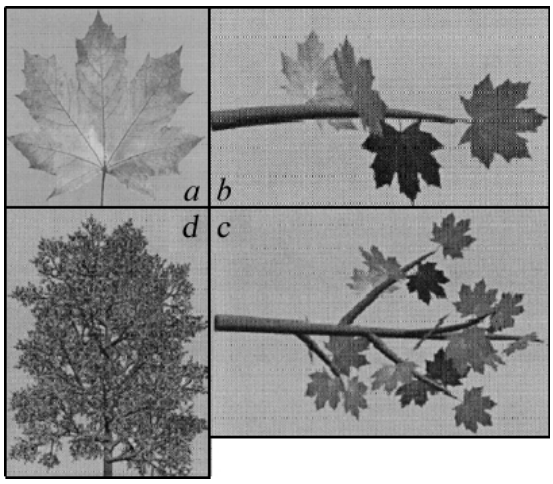


FIG. 3-28 – Utilisation d'images Z-buffer calculées suivant la hiérarchie de la plante pour optimiser l'affichage d'un arbre [Max, 1996]. Des images Z-buffer à différentes échelles : a) échelle des feuilles, b) des rameaux c) des branches et finalement d) de la plante entière.

segments. Les primitives de même type sont regroupées : des groupes de seize pour les entre-nœuds de même ordre de ramification et de quatre pour les feuilles. Des ratios donnent le nombre de primitives à afficher en fonction de la distance. Ensuite, différentes primitives sont utilisées pour représenter le même objet pour des points de vue plus ou moins distants. Ainsi les feuilles sont représentées par des polygones pour les vues proches ou par plus ou moins de points pour les vues éloignées. Dans le cas des branches, suivant leurs importances (déterminées notamment par leur ordre de ramification) et la distance à la caméra, leur représentation est effectuée soit par un ensemble de polygones ou par des lignes composées de plus ou moins de segments (Figure 3-29). Dans les cas les plus éloignés, certains éléments ne seront pas représentés.

FIG. 3-29 – Représentation en niveaux de détails d'un arbre [Weber and Penn, 1995]. Les éléments composants cet arbre sont d'abord représentés par des polygones (à gauche). Puis, pour les vues les plus éloignées, les branches sont représentées par des lignes avec plus ou moins de segments et les feuilles par des points (à droite).

Les surfels

Une autre représentation alternative utilise des points, organisés dans une structure hiérarchique, comme primitive graphique. La première idée générale d'utilisation des points comme

primitives graphiques est présentée par Levoy et Whitted [Levoy and Whitted, 1985]. Shade *et al.* [Shade et al., 1998] échantillonnent des modèles géométriques de plantes en utilisant une procédure de ray-tracing. En utilisant les points résultants, le rendu d'images de différents points de vue est possible. Les résultats obtenus sont de bonne qualité pour des points de vue distants, mais contiennent des trous et ont un aspect granulaire pour les vues proches.

Une méthode générale en niveaux de détails basée sur les points est présentée par Rusinkiewicz et Levoy [Rusinkiewicz and Levoy, 2000]. Une représentation par une *hiérarchie de points* permet une réduction souple de la quantité de géométrie représentée pour les surfaces souples. Pfister *et al.* [Pfister et al., 2000] présentent une méthode de rendu sophistiqué pour les représentations avec des points. Un précalcul est appliqué sur un modèle 3D polygonal pour l'échantillonner sous forme de "surfels", primitives contenant la position, la normale et la couleur en un point d'une surface. Ces surfels sont organisés dans une structure hiérarchique de type octree. Durant le rendu, un algorithme projette les surfels sur le Z-Buffer. Cet algorithme s'arrête dans son parcours en profondeur de la hiérarchie de points lorsque la résolution obtenue est satisfaisante. Un algorithme de lissage permet ensuite de détecter les trous (les pixels où aucun surfel n'a été projeté) dans le Z-Buffer et de les combler. Différentes méthodes de reconstruction d'images telles que le suréchantillonage permettent de favoriser soit la vitesse, soit la qualité. Leur méthode n'utilise cependant pas la carte graphique et reste trop lente pour le rendu interactif de scènes très complexes.

Deussen *et al.* [Deussen et al., 2002] étendent cette technique pour les plantes en représentant la géométrie de la plante à la fois par des points, des triangles et des lignes. Les points semblent particulièrement adaptés pour représenter des objets tels que des feuilles. Pourtant pour des organes tels que les branches fines, les lignes permettent de minimiser tout autant la quantité de données et donnent une meilleure représentation en utilisant des techniques similaires. Ces primitives sont organisées sous forme d'octree. Suivant la distance de la caméra à la plante, l'algorithme de rendu choisit la primitive adéquate. Les triangles sont choisis pour les représentations proches, les points et les lignes sont rendus en utilisant les techniques proposées par Pfister *et al.* pour les points de vues éloignés (Figure 3-30). La réduction de données peut aussi être contrôlée pour favoriser certaines parties de la représentation qui ont un impact visuel plus important.

FIG. 3-30 – Représentation par points d'un pin [Deussen et al., 2002]. De gauche à droite, le modèle polygonal et des représentations avec 13000, 6500, 3250 et 1625 points. Un échantillonage faible est utilisé pour visualiser les modèles de points.

Les textures volumiques

Les *textures volumiques* (Texels) sont une autre catégorie de représentations alternatives qui permettent de représenter des objets complexes, dont les détails sont petits. Les texels sont des

volumes (voxels) contenant une densité et une fonction de réflectance (loi d'illumination locale) de la lumière. Ces volumes servent à recouvrir des surfaces à la manière des textures, constituant une "peau épaisse". Ce modèle a été introduit en 1989 par Kajiya [Kajiya and Timothy, 1989] pour le cas particulier de la représentation de fourrure. Neyret [Neyret, 1996, Neyret, 1998] l'étend par une approche multi-échelles en organisant la texture volumique sous forme d'octree, obtenant ainsi une représentation assez générale et très efficace :

- la faible résolution dans la perception du relief est exploitée par l'usage du volume de voxels (dont la résolution correspond au nombre de plans que l'on peut distinguer en parallaxe),
- la perception moyenne des formes (silhouette, position) est satisfaite par la répartition de la matière au sein de ce volume de voxels,
- la perception fine de l'illumination est assurée par la fonction de réflectance présente dans chaque voxel,

L'efficacité provient d'une part du caractère minimal de la représentation (évitant ainsi le sur-échantillonnage, qui coûte très cher tout en ne pouvant supprimer tout les artefacts d'affichage), et d'autre part de la structuration qu'elle apporte aux données : les voxels étant ordonnés dans le volume et le mappage des volumes étant lui-même ordonné sur la surface, on sait exactement quelles instances de volume et quels voxels parcourir, et dans quel ordre, contrairement à ce qu'il en serait si les détails étaient représentés par des maillages. L'utilisation de texels a notamment été étudiée pour la représentation d'objets végétaux générés par des L-systèmes (Figure 3-31).

Fig. 3-31 – Utilisation de texels pour représenter une forêt [Neyret, 1998].

Meyer et Neyret [Meyer and Neyret, 1998] étendent ces travaux à l'affichage temps-réel (en acceptant de renoncer à une partie du réalisme). Dans ce contexte, un texel est représenté par une série de tranches recouvertes d'une texture semi-transparente (voir section 1.1.2). Un objet 3D est ainsi traduit en tranches. La complexité d'une scène se caractérise alors par le nombre de faces virtuellement apparentes (celles de l'objet figuré) et le nombre de faces réellement tracées (les tranches texturées). Le coût de rendu sur des cartes graphiques haut de gamme étant quasi indépendant de la résolution des textures, une autre propriété de cette représentation est la relation particulièrement économique entre coût et résolution : le nombre de faces à afficher (ici, les tranches) augmente linéairement avec la diminution de taille des arêtes du maillage à figurer, alors que le nombre de faces du maillage (qu'on aurait classiquement utilisé) croît de manière quadratique.

Les shaders hiérarchiques

Meyer et Neyret [Meyer and Neyret, 2000] proposent une méthode de représentation par des modèles de réflectance (ou *shaders*) qui se passe de l'information polygonale explicite des formes. L'idée est d'exploiter au mieux la connaissance à priori disponible pour certaines familles d'objets pour calculer analytiquement une bonne approximation de l'illumination dans la région de l'espace visible à travers un pixel. Ces travaux portent sur les forêts de pins, dans la mesure où la connaissance à priori sur la distribution des aiguilles est très forte. Une hiérarchie de trois shaders intégrant l'illumination (y compris les ombres et la transparence résiduelle) à l'échelle d'une aiguille, d'un cône d'aiguilles, et de toute une pousse est utilisée (Figure 3-32). Leur implémentation tourne environ 8 fois plus vite que Rayshade (un logiciel de rendu par ray-tracing), pour calculer l'image d'une forêt de sapins sans artefacts visuels.

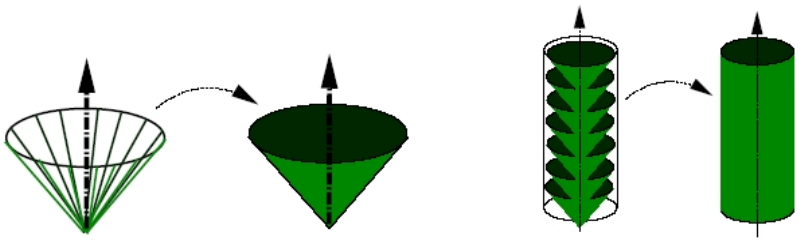


FIG. 3-32 – Représentation à différentes échelles des élements d'un pin [Meyer and Neyret, 2000]. Pour chacune des échelles, un shader donnant la réponse à la lumière de l'ensemble est utilisé lors de l'affichage.

3.4 Classification croisée

Nous donnons ici un récapitulatif des différents modèles géométriques présentés précédemment.

3.4.1 Modèles mono-échelle

Pour chacun des modèles mono-échelle examinés dans ce chapitre, nous caractérisons brièvement son contexte, sa structure topologique sous-jacente, le type de modélisation géométrique et le type de rendu obtenu :

- Le *contexte* du modèle permet d'évaluer le type de résultat souhaité. Pour la modélisation biologique par exemple, le réalisme visuel de l'objet géométrique n'est pas forcément important, à l'inverse des méthodes de simulation et de modélisation géométrique. Nous avons différencier la génération de la simulation par la prise en compte, dans le deuxième cas, du temps dans l'algorithme génératif.
- La *structure topologique* représente l'architecture interne du modèle. Quand elle existe, elle est caractérisée par un objet mathématique (*e.g.* arborescences, *etc.*).
- La *géométrie* donne le type de primitives géométriques utilisés (*e.g.* cylindres, *etc.*).
- Finalement le *rendu* indique si des techniques particulières ont été utilisées et rend compte de la qualité visuelle finale du modèle.

	Contexte	Topologie	Géométrie	Rendu
Sinoquet *et al.*, 1998	Modélisation biologique	Aucune	Collection de polygones	Non réaliste
Honda, 1971	Génération	Arbre binaire	Segments.	Fil de fer
Reves et Blau, 1985	Génération	faible	Système de particules	Impressionniste
Greene, 1989	Simulation	Arborescence	Voxels	Polygones texturés. Réaliste.
de Reffye *et al.*, 1988	Simulation	Arborescence	Polygones	Très réaliste
Prusinkiewicz *et al.*, 1997	Simulation	Arborescence axiale	Tortue LOGO	Cylindres et polygones texturés. Très réaliste.
Kawaguchi, 1982	Génération	Arborescence	Prismes de ramification. Continuité C_0.	Non réaliste
Bloomenthal, 1985	Modélisation géométrique	Arborescence	B-Splines, textures. Continuité C_2.	Très réaliste, même en vue rapprochée.
Bloomenthal *et al.*, 1990 , 1991	Modélisation géométrique	Arborescence	Surfaces implicites. Continuité C_2.	Très réaliste, même en vue rapprochée. Discrétisation complexe.

3.4.2 Modèles multi-échelles

Nous reprenons dans cette section le même canevas de classification pour les méthodes multi-échelles. La topologie étant généralement une arborescence, nous indiquons, dans ce cas, si elle est construite sur une décomposition de la plante fonction de son organisation ou de la proximité spatiale de ses éléments. Les méthodes, introduites ici, étant généralement dédiées à la visualisation réaliste et efficace, le rendu est donc caractérisé plus en détails dans un tableau spécifique.

	Contexte	Topologie	Géométrie
Oppenheimer, 1986	Génération	Fractale	Cylindres, spirales, hélices et tortillons
Demko *et al.*, 1985, Barnsley *et al.*, 1988	Génération 2D	Fractale	Pixels
Zhang et Blaise, 2003	Rendu	Décomposition spatiale	Polygones
Marshall *et al.*, 1997	Rendu	Décomposition spatiale	Tétraèdres, et polygones
Sillion, 1995	Radiosité	Décomposition spatiale	Clusters
Soler *et al.*, 2003	Radiosité	Décomposition structurelle	Clusters
Meyer *et al.*, 2001	Rendu	Décomposition structurelle	Textures bidirectionnelles
Max *et al.*, 1999	Rendu	Décomposition structurelle	Images Z-buffer

Weber et Penn, 1995	Rendu	Décomposition structurelle	Points et lignes
Deussen *et al.*, 2002	Rendu	Décomposition spatiale	Surfels (Points)
Neyret, 1998	Rendu	Décomposition spatiale	Textures volumiques
Meyer et Neyret, 2000	Rendu	Décomposition structurelle	Shaders

Nous nous intéressons, dans le tableau suivant, au rendu des méthodes multi-échelles. Les quatre premières colonnes indiquent si les méthodes proposées supportent certaines propriétés :

- *Instanciation* : la méthode supporte explicitement une réduction des données via l'instanciation.
- *Temps réel* : la méthode supporte un affichage en temps réel.
- *Eclairage dynamique* : la méthode supporte un éclairage dynamique (si non, l'éclairage est précalculé).
- *Pré-calcul* : la méthode nécessite des pré-traitements.

Les deux dernières colonnes quantifient la qualité et les performances du résultat (les carrés pleins indiquent les valeurs sur une échelle de trois). Ces indications ont été déduites de la lecture des articles et sont bien sûr subjectives. Certaines méthodes peuvent très certainement être améliorer pour obtenir de meilleures résultats et de nouvelles propriétés. Cette quantification donnent cependant un aperçu des avantages/inconvénients de chaque méthodes.

	Instanciation	Temps réel	Eclairage dynamique	Pré-calcul	Qualité	Performance
Zhang et Blaise, 2003		√	√	√	■□□	■■■
Marshall *et al.*, 1997		√	√	√	■■□	■■■
Sillion, 1995	√		√	√	■■■	□□□
Soler *et al.*, 2003	√		√	√	■■■	□□□
Meyer *et al.*, 2001	√	√	√	√	■□□	■■■
Max *et al.*, 1999	√	√	√	√	■□□	■■□
Weber et Penn, 1995	√	√			■□□	■■■
Deussen *et al.*, 2002	√	√		√	■■□	■□□
Neyret, 1998			√	√	■■□	■□□
Meyer et Neyret, 2000	√	√			■■□	■□□

3.5 Conclusion

Nous avons étudié, dans ces trois premiers chapitres, les différentes approches pour modéliser et visualiser l'architecture des plantes. A une échelle globale, la représentation de la plante se fait par son enveloppe ou une image de son profil. A des échelles plus fines, l'organisation de la plante est représentée par une arborescence sur laquelle peut être construite une représentation géométrique. A chacun des composants, un modèle géométrique est associé. L'adjacence entre composants, donnée par l'arborescence, peut être utilisée pour connecter les modèles géométriques des composants [Kawaguchi, 1982, Bloomenthal, 1985, Bloomenthal and Wyvill, 1990]. Dans ce cas,

la description topologique induit des contraintes sur la géométrie. D'autre types de contraintes entre modèles géométriques existent, dépendant des primitives géométriques utilisées et du type de plante. Pour représenter de manière cohérente un objet végétal, les systèmes de particules par exemple sont contrôlés par des processus stochastiques contraints [Reeves and Blau, 1985]. Certains systèmes génératifs utilisent ces contraintes pour faciliter la spécification des paramètres du modèle. Les angles de branchement ou les longueurs des axes [Honda, 1971, Kawaguchi, 1982] peuvent être définis relativement à ceux des axes d'ordre inférieur.

Différents travaux introduisent l'idée d'organiser les modèles de plantes à différentes échelles. Dans le domaine de l'analyse de l'architecture des plantes, un modèle topologique multi-échelles a été développé pour une meilleure compréhension du développement de la plante [Godin and Caraglio, 1998]. En infographie, certaines approches utilisent des hiérarchies de modèles pour adapter la complexité de la représentation au cours des calculs [Neyret, 1998, Meyer and Neyret, 2000, Meyer et al., 2001, Deussen et al., 2002, Soler et al., 2003].

Le modèle multi-échelles de Godin et Caraglio [Godin and Caraglio, 1998] semble très général puisqu'il prend en compte les relations entre entités à une même échelle (relations d'adjacence) et à différentes échelles (relations de décomposition). Il ne considère cependant pas la géométrie de la plante. D'un autre coté, les approches géométriques hiérarchiques considèrent plusieurs échelles mais ne considèrent pas les relations entre entités à une même échelle. Ces constatations nous ont amené à définir un modèle géométrique multi-échelles général, en étendant les arborescences multi-échelles à la géométrie. Ce modèle est présenté au chapitre suivant.

Chapitre 4

Modèle géométrique multi-échelles

4.1 Modèle géométrique d'une plante

Suivant le niveau de détail auquel on considère une plante, sa géométrie peut être représentée par différents types de modèles géométriques, de complexité variable, avec différentes propriétés géométriques. Comme nous venons de le voir aux chapitres précédents, deux catégories d'approches de modélisation de la géométrie d'une plante peuvent être distinguées : les approches globales et détaillées [Godin, 2000]. Dans le cas des approches globales, la géométrie de la plante est représentée comme un tout par un modèle géométrique plus ou moins complexe (ellipsoïde, cône, *etc.*) figurant le plus souvent son enveloppe ou par une image d'un de ses profils (voir chapitre 1). Les outils informatiques disponibles depuis une dizaine d'années ont favorisé l'émergence d'approches détaillées. Dans ce cas, la géométrie de la plante est représentée par l'ensemble de tous les modèles géométriques des composants (modules) de la plante (par exemple de l'ensemble des tiges et des feuilles). Les approches détaillées permettent de prendre en compte de façon précise l'organisation topologique et géométrique des plantes dans les modèles de fonctionnement ou de croissance (voir chapitre 2 et 3).

Le problème de la complexité d'un modèle de plante se pose lors de son utilisation. Dans le cadre de la synthèse d'images par exemple, un modèle de plante très détaillé qui ne contribue qu'à une toute petite partie de l'image finale consomme beaucoup de ressources pour un résultat minime (pour lequel apparaissent parfois certains artefacts). Ce constat a suscité la définition de plusieurs méthodes d'affichage en niveaux de détails qui simplifient la géométrie de la plante en la dégradant en fonction du point de vue. Cependant, ces méthodes ne donnent pas des résultats concluants sur la géométrie des plantes. Cette géométrie, particulièrement intriquée, ne se résume pas simplement par des techniques de simplification polygonale classique. Différents auteurs [Kajiya, 1985, Neyret, 2001] suggèrent d'aborder ce problème en introduisant des représentations à différentes échelles. Sur des critères purement visuels, des hiérarchies de représentations à différentes échelles sont construites [Neyret, 1998, Meyer and Neyret, 2000, Meyer et al., 2001, Deussen et al., 2002, Soler et al., 2003]. La représentation d'une plante peut donc être ainsi raffinée ou simplifiée en changeant de niveau dans la hiérarchie. Ces modèles sont construits sur la décomposition des entités d'une plante en sous-entités.

Dans le cadre de la simulation de la croissance des plantes, le problème de modéliser des aspects morphologiques ou physiologiques à différentes échelles de temps nécessite aussi de considérer la structure de la plante à différentes échelles [Remphrey and Prusinkiewicz, 1997]. L'utilisation de fonctions 2D et de l'information positionnelle [Prusinkiewicz et al., 2001] (qui sera détaillée à la section 7.6.3), ou d'opérateur de déformation de forme libre [Lintermann and Deussen, 1999] (section 8.2.4) permet un contrôle global sur un modèle végétal détaillé grâce à des informations liées à des caractéristiques globales d'une plante ou d'une partie de la plante.

Pour l'analyse de l'architecture des plantes, le problème de l'organisation en échelles d'un modèle de plante apparaît aussi. Pour une meilleure compréhension de la structure des plantes,

Godin et Caraglio [Godin and Caraglio, 1998] ont formalisé un modèle topologique arborescent multi-échelles rendant compte des différents niveaux d'organisation dans l'architecture des plantes (voir section 2.2.4). Contrairement aux approches multi-échelles précédentes, ce modèle prend en compte aussi bien l'organisation à chaque échelle de la plante que les différentes relations de décomposition entre niveaux d'organisation. Des premières extensions de cette approche topologique multi-échelles à la géométrie ont été proposées [Godin et al., 1996, Godin et al., 1999a].

En nous inspirant de ces premiers travaux, nous avons formalisé un modèle géométrique multi-échelles général de l'architecture des plantes. Dans ce modèle, des informations géométriques sont associées aux différents composants à toutes les échelles d'un graphe multi-échelles. Un tel modèle contient des représentations à plusieurs échelles d'un même objet.

Comme nous le montrons à la section 3.2, la connaissance de la topologie d'une plante à une seule échelle introduit des relations sur les modèles géométriques. Pour des représentations réalistes, l'adjacence entre composants peut en effet être traduite par une continuité des modèles géométriques. De la même manière, les relations de décomposition introduisent des contraintes entre représentations à différentes échelles. Nous formalisons ces différentes contraintes sous la forme de contraintes *intra* et *inter-échelles* (nous généralisons en fait ces contraintes introduites dans [Godin et al., 1999a] à tous types de primitives géométriques). Nous montrons ensuite comment ces contraintes sont portées par le graphe multi-échelles et peuvent être utilisées pour calculer de nouvelles représentations.

Nous commençons par introduire les différents modèles et attributs géométriques que nous utilisons dans la construction de notre modèle multi-échelles, puis de quelle manière ils sont organisés sur un graphe à une seule échelle et comment nous l'étendons au cas multi-échelles. Ce modèle et son utilisation seront illustrés par différents exemples, le plus souvent tirés de la littérature. Des utilisations originales de ce modèle seront présentées aux chapitres suivants.

4.2 Modèles géométriques élémentaires

La forme d'un objet est modélisée par une primitive géométrique (ou modèle géométrique élémentaire). Le choix d'une primitive particulière est fonction, explicitement ou non, du domaine d'application et de ses exigences en terme de précision et d'efficacité calculatoire.

4.2.1 Primitives géométriques

Deux grands types de modèles géométriques élémentaires peuvent être distingués :

- les *modèles paramétriques*, définissant une courbe, une surface ou un volume par une ou plusieurs équations dépendant globalement d'un ou plusieurs paramètres. Ces équations permettent de générer directement les points de la surface. On y trouve des formes géométriques tel que les surfaces de Bézier, NURBS, *etc.* On trouve, dans cette famille, les *modèles explicites*, qui listent explicitement l'ensemble des surfaces (par un ensemble de polygones, généralement des triangles), ou des points (*e.g.* les Surfels ou QSplat, [Pfister et al., 2000]) de l'espace appartenant au modèle.
- les *modèles implicites*, définissant eux aussi une surface ou un volume par un certain nombre d'équations. Ces équations, dans ce cas, permettent de tester si un point est à l'interieur ou à l'exterieur de la surface (ou du volume). On y trouve des formes géométriques tel que les surfaces implicites, *etc.*

Formellement, soit $\mathcal{R} = (L, \overrightarrow{i}, \overrightarrow{j}, \overrightarrow{k})$ un système de coordonnées. Une surface peut être définie dans $\mathcal{R}$ par un modèle géométrique $\sigma(\lambda)$, où σ est un ensemble d'équations paramétrées par un paramètre λ. Une valeur spécifique de λ permet de spécifier une surface particulière de la famille. Par exemple, $\sigma(\lambda) = \{(x, y, z) \mid \left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 + \left(\frac{z}{c}\right)^2 = 1\}$ est l'équation paramétrique d'une famille d'ellipsoïdes d'origine L paramétrée par $\lambda = (a, b, c)$ définissant les dimensions de l'ellipsoïde.

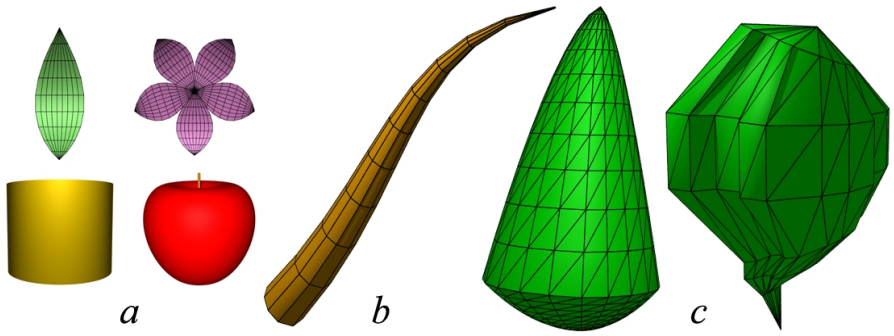


FIG. 4-1 – Exemples de modèles géométriques élémentaires : a) des symboles permettant de représenter des entités fines d'une plante (cylindre, surface de Bézier, surface de révolution, *etc.*) b) un cylindre généralisé représentant la géométrie d'une branche c) deux types d'enveloppe : enveloppe asymétrique [Cescatti, 1997] (voir section 1.3.1) et enveloppe extrudée [Birnbaum, 1997](voir section 1.3.2).

La Figure 4-1 donne des exemples de modèles géométriques élémentaires de la librairie Plant Graphic Library (PGL, anciennement GEOM [Boudon et al., 2001a, Boudon et al., 2001b]) que nous avons implémentée et utilisée pendant cette thèse.

4.2.2 Transformations géométriques

La géométrie d'un objet est définie dans un repère global $\mathcal{R}_0 = (L_0, \overrightarrow{i_0}, \overrightarrow{j_0}, \overrightarrow{k_0})$ ayant pour origine L_0 et pour vecteurs unitaires $\overrightarrow{i_0}, \overrightarrow{j_0}, \overrightarrow{k_0}$. Soit m un modèle géométrique élémentaire, défini dans le repère local $\mathcal{R}_{local} = (L, \overrightarrow{i}, \overrightarrow{j}, \overrightarrow{k})$. Placer un objet dans une scène revient à définir une transformation géométrique généralement composée d'une *translation* L, une *orientation* O et une *mise à l'échelle* S. La translation est donnée par les coordonnées de l'origine L dans $\mathcal{R}_0$, soit le vecteur $[l_1, l_2, l_3]$. L'*orientation* O est donnée par les coordonnées des vecteurs $\overrightarrow{i}, \overrightarrow{j}, \overrightarrow{k}$ exprimées dans $\mathcal{R}_0$. La mise à l'échelle est représentée par un vecteur $S = [s_1, s_2, s_3]$ donnant les dimensions de l'objet suivant les trois axes du repère local. Ces différentes transformations s'expriment aussi sous la forme de matrices 4×4.

La composée T de ces transformations s'expriment sous forme matricielle :

$$T = L + O * S = \begin{bmatrix} i_1 * s_1 & j_1 & k_1 & l_1 \\ i_2 & j_2 * s_2 & k_2 & l_2 \\ i_3 & j_3 & k_3 * s_3 & l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

En coordonnées homogènes, un point quelconque $P = [x_1, x_2, x_3, 1]$ de la surface initiale, est transformé en $P' = [x'_1, x'_2, x'_3, 1]$ de la surface transformée :

$$P' = L \ + \ O * S * P = TP$$

Notons $\mathcal{T}_{\mathcal{R}_{local},\mathcal{R}_0}$ cette transformation (L, O, S). Le modèle géométrique $\sigma(\lambda)$ dans le repère $\mathcal{R}_0$ s'écrit :

$$\sigma(\lambda)_{/\mathcal{R}_0} = \mathcal{T}_{\mathcal{R}_{local},\mathcal{R}_0}(\sigma(\lambda)_{/\mathcal{R}_{local}})$$

La géométrie d'un objet est donc définie par une famille σ de modèles géométriques, la valeur des paramètres λ du modèle, et les paramètres des transformations $\mathcal{T}_{\mathcal{R}_{local},\mathcal{R}_0}$ nécessaires pour placer, orienter et dimensionner cet objet dans l'espace.

4.2.3 Modèles géométriques composés

Un objet peut avoir une forme complexe de sorte qu'il ne puisse pas être représenté facilement par un unique modèle géométrique élémentaire. Dans ce cas, pour représenter sa géométrie, l'objet est décomposé en sous-parties de géométries plus simples. La géométrie de l'objet entier est le résultat de l'agglomération des formes géométriques des sous-parties [Foley et al., 1995].

Plus formellement, soit V un ensemble de constituants, $\mathcal{R}_0 = (L_0, \overrightarrow{i_0}, \overrightarrow{j_0}, \overrightarrow{k_0})$ un référentiel global et Σ un ensemble de modèles géométriques élémentaires. Soit une application ω qui associe à chaque élément x de V un modèle géométrique $\omega(x)$ correspondant à un modèle $\sigma_x(\lambda_x)$ de Σ et une série de transformation (L_x, O_x, S_x) dans $\mathcal{R}_0$:

$$\omega(x) = \mathcal{T}_{\mathcal{R}_x,\mathcal{R}_0}(\sigma_x(\lambda_x))$$

Le *modèle géométrique composé* $m(V)$ associé à V, est définit comme une fonction des modèles géométriques $\omega(x)$:

$$m(V) = f(\{\omega(x)\}_{x\in V}) = f(\{\mathcal{T}_{\mathcal{R}_x,\mathcal{R}_0}(\sigma_x(\lambda_x)\}_{x\in V})$$

La fonction f peut être en général relativement complexe, réaliser des fusions de formes, des raccords de surface (blending), *etc.* Une fonction f simple est par exemple l'union des modèles :

$$m(V) = \cup_{x\in V}\omega(x)$$

4.3 Modèle géométrique associé à un graphe

4.3.1 Modèle géométrique d'une plante

Dans le cas des représentations mono-échelle de plantes, les modèles géométriques détaillés sont construits à partir de l'arborescence représentant l'organisation de la plante. A chacun des sommets du graphe sont attachés des transformations et un modèle géométrique nécessaires pour le représenter. Une plante est en effet bien souvent considérée comme un ensemble d'entités aux formes peu variées (entre-nœuds, feuilles, fruits) agencées les unes par rapport aux autres. Pour cela, les structurations utilisées reposent sur un graphe arborescent dont les arêtes vont représenter l'adjacence des constituants.

Dans le graphe arborescent représentant l'organisation topologique d'une plante, deux types de relations entre les entités sont généralement considérés (voir section 2.1.1). Une entité e_1 peut suivre une entité e_2 (cette relation est notée '<' : les deux entités ont été produites par un même méristème et appartiennent donc au même axe) ou la porter (cette relation est notée '+' : e_2 a été produite par un méristème axilaire de e_1).

Un modèle géométrique construit sur un graphe représentant l'organisation topologique d'une plante peut donc être formalisé de la manière suivante :

Définition 7 *Soit $g = (V, E)$ un graphe. Soit $\mathcal{R}_0$ un référentiel lié à g. On appelle modèle géométrique du graphe g, $m(g) = (\Sigma, \omega)$ tel que :*

- *Σ est un ensemble de modèles géométriques (élémentaires ou composés),*
- *ω est une application associant à tout sommet x de V un modèle géométrique élémentaire $\sigma_x(\lambda_x)$ de Σ placé dans $\mathcal{R}_0$ par la transformation $\mathcal{T}_{\mathcal{R}_x,\mathcal{R}_0} = (L_x, O_x, S_x)$:*

$$\omega(x) = \mathcal{T}_{\mathcal{R}_x,\mathcal{R}_0}(\sigma_x(\lambda_x))$$

L'utilisation d'un graphe basé sur l'organisation topologique d'une plante comme support d'un modèle géométrique revient à donner une interprétation géométrique d'une représentation topologique. Pour calculer les modèles géométriques et leurs paramètres représentant chacune des

entités du graphe, un certain nombre de règles générales, déduites de connaissances botaniques par exemple, peuvent être utilisées. Elles peuvent être globales, par exemple, toutes les entités sont représentées par la même famille de modèle, ou dépendre d'informations contenues dans le graphe, par exemple la famille du modèle géométrique est fonction du type de l'entité à représenter (un entre-nœud, une feuille, *etc.*). Ces règles générales ne permettent pas de calculer toute l'information géométrique. Les règles de construction du modèle géométrique doivent en effet tenir compte de l'organisation topologique des entités. La topologie va induire un certain nombre de contraintes sur les modèles géométriques à instancier : par exemple, pour obtenir un modèle cohérent, deux entités topologiquement connectées devront être représentées par des modèles qui ont une frontière commune. De plus, le type d'adjacence peut permettre de raffiner cette contrainte. Par exemple, deux entités reliées par une connection de type "suit" ('<') devront être représentées par deux modèles géométriques qui ont une continuité d'un certain type (C_1 par exemple) à cette frontière.

Définition 8 *Soit $g = (V, E)$ un graphe et $m(g) = (\Sigma, \omega, C)$ un modèle géométrique associé à ce graphe. La cohérence du modèle est exprimée par C, un ensemble de contraintes (équations ou inéquations) indexé sur E avec $C = \{c_e\}_{e \in E}$, tel que si $e = (x, y)$, $e \in E$, alors l'équation c_e établit une relation entre $\omega(x)$ et $\omega(y)$.*

L'ensemble de contraintes C est appelé *contraintes (géométriques) intra-échelle* [Godin et al., 1999a]. Dans le cas des graphes représentant des plantes, où les arcs représentent l'adjacence entre les constituants d'une plante, l'ensemble de contraintes C permet d'exprimer des conditions de cohérence entre les paramètres géométriques des modèles de deux entités contiguës, du fait de leur adjacence. Formellement, cette relation se traduit par un ensemble d'équations (et/ou d'inéquations) entre les paramètres des deux modèles $\lambda_{\omega(x)} = (\sigma_x, \lambda_x, L_x, O_x, S_x)$ et $\lambda_{\omega(y)} = (\sigma_y, \lambda_y, L_y, O_y, S_x)$: si $\exists e \in E, \quad e = (x, y)$

$$c_e(\lambda_{\omega(x)}, \lambda_{\omega(y)}) = 0$$

Nous dirons que le modèle $m(g)$ est un modèle géométrique *valide* si les paramètres des modèles géométriques de ses constituants vérifient les contraintes.

4.3.2 Exemples de contraintes intra-échelle

S'il existe une connection topologique e entre deux entités x et y, une contrainte exprimant une connection minimale de leurs modèles géométriques peut être :

$$c_e : e = (x, y) \in E \Rightarrow \omega(x) \cap \omega(y) \neq \emptyset \tag{4.1}$$

Des contraintes plus fortes imposant par exemple une continuité des modèles peuvent être définies. Nous illustrons ici avec trois exemples, différents types de contraintes intra-échelle suivant les différents types de relations et différents types de modèles géométriques.

Eléments portés

Dans le cas d'une entité y représentant une feuille ou un fruit et une entité x représentant l'entre-nœud qui la porte, $(x, y) \in E_+$ (*i.e.* x porte y), la contrainte 4.1 suffit pour exprimer la connection physique entre ces deux types d'entités. Cette contrainte peut aussi, intégrer des informations liées à l'orientation relative de ces deux entités. Dans la figure 4-2, on observe que les feuilles sont insérées avec un angle de $\pm 90°$ par rapport aux entre-nœuds qui les portent, ce qui correspond à une disposition alternée distique sur la branche (la branche a une tendance dite plagiotropique).

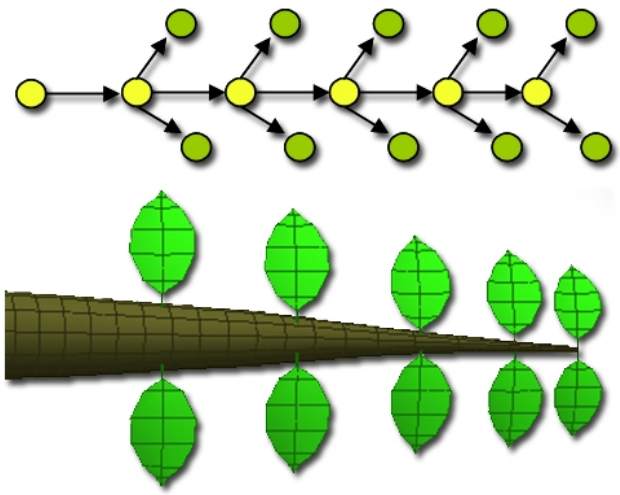


FIG. 4-2 – Une branche composée de cinq entre-nœuds (représentés par des sommets jaunes dans le graphe) portant chacun deux feuilles (représentées par des sommets verts dans le graphe). Les relations entre les orientations des modèles géométriques des différentes entités peuvent être traduites sous forme de contraintes intra-échelle portées par le graphe topologique. Pour les feuilles, ces contraintes s'expriment en fonction de leurs angles d'insertion et de la phyllotaxie de la branche.

La disposition des feuilles sur un axe peut être exprimée par des contraintes intra-échelle de la forme suivante :

- c_e : si $\exists e \in E_+, \quad e = (x, y)$, alors $O_y = O_x.R_\alpha$ où R_α correspond à une matrice de rotation d'un angle α par rapport à l'axe $\overrightarrow{k}$ de O_x.

Plus généralement, des contraintes liant l'orientation, la position et les dimensions de l'entité porteuse et l'entité portée peuvent être définies. Dans le cas de l'orientation, ces contraintes utilisent des critères liés à l'angle d'insertion et la phyllotaxie d'une branche. De nombreuses études botaniques s'efforcent de déterminer ces caractéristiques pour les différents types de plantes existantes.

On peut noter dans ce premier exemple, que les contraintes sont induites par la topologie et par la connaissance botanique associée au graphe ou aux entités du graphes. En effet, les contraintes minimales de connection, déduites de la structure topologique, ne permettent généralement pas de déterminer une valeur à tous les paramètres des modèles. Afin d'obtenir des modèles botaniquement réalistes, les contraintes sont étendues par des critères liés à la compréhension biologique du modèle. Dans ce cas, les deux modèles géométriques utilisés sont de type différent et aucune condition n'est mise sur les paramètres même des modèles.

Continuité des éléments

Ce deuxième exemple, illustré par la Figure 4-3, porte sur la géométrie d'une branche composée de quatre segments. Trois types de contraintes de continuité ont été utilisées lors de la création des modèles géométriques. Dans la première représentation, des troncs de cônes sont utilisés pour représenter les différents entre-nœuds. Les contraintes utilisées spécifient que :

- c_e : si $\exists e \in E_<, \quad e = (x, y)$ (*i.e.* x précède y), alors le diamètre et le centre de la section base du tronc de cône représentant y devront être respectivement égaux au diamètre et au centre de la section sommet du tronc de cône représentant x.

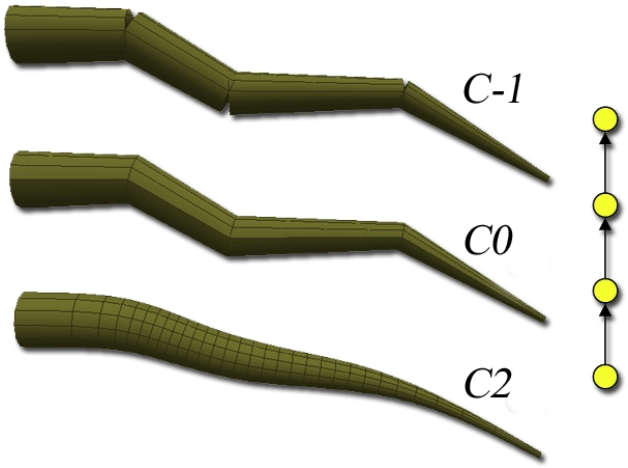


FIG. 4-3 – Une branche composée de quatre entre-nœuds dont la géométrie a été calculée avec différentes contraintes de continuité entre les modèles représentant les entre-nœuds. Dans la première représentation, la surface totale de la branche est discontinue, notamment au niveau des connections ; dans la deuxième, la représentation est C_0-continue et dans la troisième la représentation est C_2-continue.

La géométrie globale de la branche est, dans ce cas, discontinue aux jointures. Si la surface obtenue n'est pas mathématiquement continue, la résolution de telles contraintes est très simple et peu coûteuse en temps de calcul.

Dans la deuxième représentation, les contraintes utilisées spécifient que :

- c_e : si $\exists e \in E_<, \quad e = (x, y)$, alors la section base du modèle représentant x doit être la même que la section sommet du modèle représentant y.

La géométrie globale de la branche est, dans ce cas, C_0-continue. Les tangentes de la surface globale ne sont pas continues aux jointures, donnant une impression de fracture.

Finalement dans la troisième représentation, des contraintes de continuité forte sont appliquées. La géométrie des différents entre-nœuds est construite avec des surfaces B-Splines que l'on contraint pour être continues entre-elles aux jointures (continuité des tangentes et des dérivées secondes). La géométrie totale de la branche est C_2-continue et le résultat est de meilleure qualité visuelle.

On peut noter avec ce deuxième exemple que les paramètres des modèles géométriques élémentaires sont contraints. Le choix de la famille du modèle géométrique est déterminant. Un modèle flexible tel qu'une surface B-Splines permet de résoudre des contraintes complexes de continuité. Un tronc de cône, au contraire, avec son nombre limité de paramètres ne peut pas permettre de résoudre de telles contraintes. Il est aussi plus facile de résoudre des contraintes de continuité entre deux modèles de même type. Dans le cas des surfaces B-Splines, la résolution des contraintes intra-échelle reste cependant complexe à mettre en place. Nous verrons à la section 4.4.4 comment l'utilisation d'autres échelles permet de simplifier le problème et d'obtenir des surfaces de qualité similaire.

Finalement, comme pour le premier exemple, un certain nombre de contraintes liées à la cohérence botanique du modèle peuvent être ajoutées pour déterminer une solution réaliste au calcul de la géométrie des différents composants. Par exemple l'orientation relative entre deux entités peut être formalisée sous forme de contraintes en prenant en compte l'affaissement de la branche sous le poids des fruits et des autres organes, ou plus simplement en étudiant la forme générale des axes d'une plante particulière. De la même manière, les longueurs des modèles représentant ces entre-nœuds peuvent être définies relativement les uns par rapport aux autres.

Par exemple la longueur d'une entité peut être exprimée comme un ratio de la longueur de l'entité précédente.

Branchement

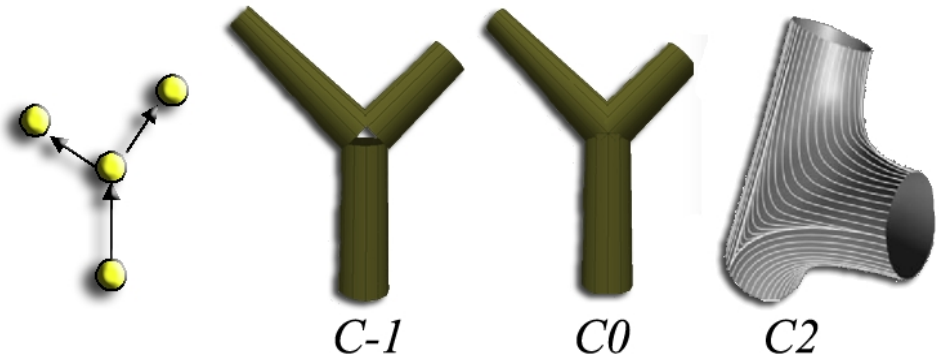


FIG. 4-4 – Exemple de continuité dans la géométrie d'un branchement. Un entre-nœud porte à sa terminaison deux entre-nœuds. Différents types de contraintes de continuité peuvent être utilisés. Dans le premier cas, la géométrie du branchement est mathématiquement discontinue, dans le deuxième cas la surface est C_0 continue. Dans le dernier cas tiré de [Bloomenthal, 1985], l'utilisation de splines interpolées permet d'obtenir un résultat C_2 continue.

Dans ce troisième exemple (Figure 4-4), nous illustrons différents types de contraintes de continuité utilisés pour déterminer la géométrie d'un branchement. Un entre-nœud porte à son extrémité deux entre-nœuds qui initient de nouvelles branches. La première représentation satisfait des contraintes simples de continuité. Des troncs de cônes sont utilisés pour représenter les différents entre-nœuds. Le diamètre sommet de l'entre-nœud porteur est égal aux diamètres base des entre-nœuds portés. Des trous apparaissent dans la représentation. Dans la deuxième représentation, les sections bases des deux entre-nœuds portés et la section sommet de l'entre-nœuds portés sont connectées par un prisme. Le résultat est une représentation C_0 continue. Finalement, la troisième représentation utilise des splines interpolées pour satisfaire des contraintes fortes de continuité. Cette dernière illustration est tirée de [Bloomenthal, 1985]. Le résultat est C_2 continu.

Conclusion

Ces différents exemples illustrent les principales contraintes intra-échelle imposées sur les représentations de plantes utilisées dans la littérature. Elles affectent les paramètres des différentes transformations et des modèles géométriques associés à chacune des entités : une certaine continuité est exigée sur les modèles successifs, le positionnement et l'orientation du modèle d'une entité sont généralement spécifiés relativement au modèle de l'entité précédente et les paramètres des modèles sont contraints pour donner des modèles géométriques successifs continus.

4.3.3 Utilisation des contraintes intra-échelle

Nous avons montré dans la section précédente comment une représentation topologique de l'organisation d'une plante peut être utilisée comme support pour exprimer différentes contraintes sur une représentation géométrique. Ces contraintes attachées aux différents arcs du graphe

contrôlent les différents modèles géométriques attachés aux sommets du graphe. Elles sont définies à partir de certains critères mathématiques ou botaniques. L'ensemble de ces contraintes forment donc un réseau arborescent. Ce réseau de contraintes peut donc être utilisé pour différents objectifs.

Calcul d'une représentation

Une première possibilité est de calculer une représentation géométrique à partir d'une représentation topologique. Différentes approches existent dans la littérature. Dans le contexte de la simulation des plantes, les L-systèmes contruisent au fur et à mesure le graphe par des règles de réécriture en résolvant ces contraintes. Cette construction utilise la propriété arborescente de la structure topologique d'une plante pour calculer la géométrie en parcourant et résolvant les différentes contraintes intra-échelle de la racine vers les feuilles de la structure. En effet, dans ce cas, une entité a sera construite après l'entité b qui la porte et pourra donc être par exemple positionnée simplement en fonction de la position de b.

Des cas plus complexes de résolution de ce réseau de contraintes existent notamment lors de la reconstruction géométrique d'une plante digitalisée. Godin *et al.* [Godin et al., 1999a] proposent une méthode pour digitaliser une plante qui consiste à mesurer certaines positions et dimensions des organes d'une plante et les insérer sur une structure topologique représentant l'organisation de la plante. Il n'est pas possible de mesurer un à un tous les organes qui sont beaucoup trop nombreux. Le modèle géométrique a donc un certain nombre de valeurs de paramètres fixés grâce aux mesures et un certain nombre de valeurs indéterminées. Les informations géométriques manquantes peuvent être calculées en utilisant le réseau de contraintes [Godin et al., 1996, Godin et al., 1999b]. Dans ce cas, le parcours du réseau ne se fera pas nécessairement de la racine vers les feuilles. En effet, si par exemple pour une ensemble de n entre-nœuds, seul le premier et le dernier entre-nœuds ont été mesurés, alors le calcul de la géométrie des $n-2$ entre-nœuds intermédiaires dépendra de deux entre-nœuds mesurés. Dans ce cas, les contraintes sont parcourues dans les deux sens (de la racine vers les feuilles et des feuilles vers la racine). Cette dernière démarche aborde en fait le problème plus général de la satisfaction de contraintes sur un réseau arborescent.

Validation d'une représentation

Le réseau de contraintes peut aussi être utilisé pour vérifier la validité d'un modèle déjà calculé. Cette validation peut être utile notamment dans le cas de la digitalisation, pour détecter des mesures incohérentes causées par des problèmes techniques ou des erreurs de manipulation. Pour détecter des points aberrants, on peut utiliser la redondance de l'information mesurée et les contraintes intra-échelle qui lient les différentes mesures. Par exemple, les longueurs mesurées des entre-nœuds peuvent être comparées à la distance des positions sommets et bases digitalisées.

Cohérence lors de modifications

Ce réseau permet aussi de maintenir la représentation valide, en cas de modification, par exemple dans le cas d'une modélisation de la croissance ou la dynamique des éléments du graphe. En effet, si la valeur d'un paramètre géométrique est changée, cette modification sera répercutée sur un certain nombre de modèles géométriques de la représentation du fait des contraintes intra-échelle. Si par exemple, la longueur d'un entre-nœud est changée, les positions de tous les entre-nœuds successeurs peuvent être modifiées. Dans ce cas, la propagation de la modification affecte un grand nombre d'entités. Le nombre de modifications dépend bien sûr des contraintes intra-échelle et des paramètres fixés. Un deuxième exemple de propagation est le cas où l'on modifie la longueur d'un entre-nœud a qui est suivi par un entre-nœud b dont on a mesuré et donc fixé la position au sommet. Dans ce deuxième exemple, pour rendre valide le modèle, on

peut simplement diminuer la longueur de l'entre-nœud b et la propagation de la modification grâce au réseau de contraintes n'affectera qu'une seule entité.

4.4 Modèle géométrique associé à un graphe multi-échelles

4.4.1 La notion de graphe de scène

En infographie, l'ensemble des modèles géométriques élémentaires représentant les différents éléments d'un objet complexe sont organisés dans une structure de graphe appelé *graphe de scène*. Les différents modèles géométriques élémentaires sont attachés aux sommets du graphe. Cette structure permet de décrire différents types d'organisation.

Pour optimiser l'affichage par exemple, la structure peut hiérarchiser les objets suivant leurs proximités spatiales et permettre ainsi d'obtenir rapidement les objets à afficher suivant la position et la direction de la vue (généralement symbolisée par un objet caméra). Une autre possibilité, couramment utilisée, est d'utiliser le fait que les modèles géométriques possèdent souvent une structure hiérarchique produite par un processus de construction ascendant : les composants sont utilisés comme blocs de construction pour créer des entités de plus haut niveau, qui sont alors utilisés comme nouveaux blocs d'encore plus haut niveau, et ainsi de suite. Le graphe résultant d'une telle construction est un graphe orienté acyclique [Foley et al., 1995]. Les feuilles du graphe contiennent les modèles géométriques élémentaires, et les sommets internes les différentes transformations et les propriétés de matériaux communes au groupe d'objets dont ils représentent la composition. Un exemple de graphe de scène est donné à la Figure 4-5.

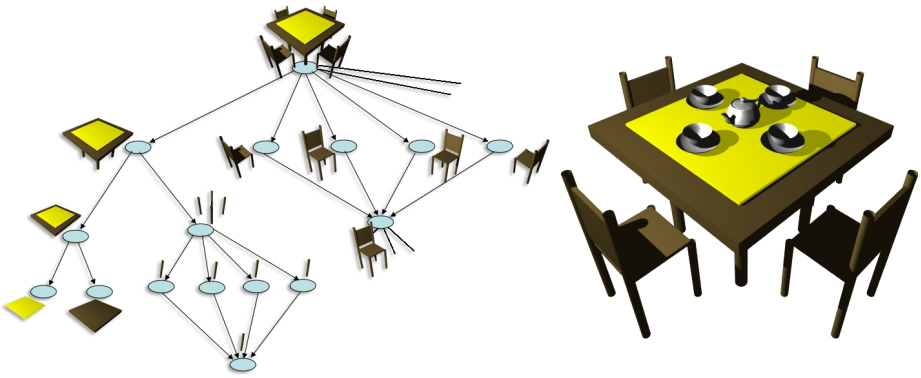


FIG. 4-5 – A gauche, un graphe de scène et, à droite, la scène résultante. Les feuilles du graphe contiennent des primitives géométriques et les sommets internes des transformations. Un même objet peut être placé dans différents endroits de la scène, donnant ainsi des cycles dans le graphe.

4.4.2 Les graphes multi-échelles

Les graphes de scène organisent leurs modèles géométriques élémentaires de manière hiérarchique. La sémantique utilisée dans ce cas pour les arcs du graphe est liée à la notion de décomposition. Les objets à représenter sont décomposés récursivement jusqu'a obtenir des géométries élémentaires qui sont alors hiérarchisées suivant cette décomposition. Les sommets internes contiennent généralement des propriétés globales des géométries élémentaires qui les composent.

A l'inverse, le modèle géométrique présenté précédemment part d'une décomposition existante et organise la géométrie des modules suivant leur organisation topologique. Il forme donc un graphe de scène avec un seul niveau de représentation qui contient des relations topolo-

giques, traduit géométriquement par des contraintes intra-échelle, entre les différents éléments du graphe.

Nous allons voir maintenant comment un graphe multi-échelles peut être utilisé pour construire un modèle géométrique basé sur ces deux types de relations à la fois, permettant ainsi d'allier les bonnes propriétés des deux approches. Les graphes multi-échelles [Godin, 2003] permettent de formaliser la structure d'un objet à différentes échelles. Un objet est décomposé en modules organisés sous la forme d'un graphe. Dans le cas des plantes, ce graphe est une arborescence. Cet objet peut être décomposé à plusieurs niveaux d'organisation et donc plusieurs représentations topologiques à différentes échelles peuvent être obtenues. Une plante peut être par exemple décrite à l'échelle des entre-nœuds, des branches, ou des systèmes ramifiés par exemple. Une représentation à une échelle fine peut ainsi être quotientée pour grouper ses éléments, créant ainsi des sommets macroscopiques et obtenir une représentation topologique à une échelle plus globale. Les graphes multi-échelles utilisent le fait que ces modularités et donc ces représentations à différentes échelles peuvent s'emboîter. En appliquant récursivement des quotientements, on obtient donc des représentations à différentes échelles.

Un graphe multi-échelles consiste principalement en un graphe pour représenter les modularités fines de l'objet modélisé et une séquence de fonctions de quotientement, permettant de calculer les graphes aux échelles plus macroscopiques. Le quotientement d'une arborescence donnant une nouvelle arborescence, les arborescences multi-échelles peuvent donc se définir de la même manière [Godin and Caraglio, 1998]. Une définition plus détaillée des graphes multi-échelles a été donnée à la section 2.2.4.

Les fonctions de quotientement donnent en fait les relations de décomposition entre modules à différentes échelles. Les graphes multi-échelles contiennent donc deux types de relation, celles liées à l'organisation à chaque échelle des composants d'un objet et celles liées à la décomposition de ces composants en sous-composants. Il contient donc un graphe de décomposition et une séquence de graphes représentant l'organisation de l'objet à différentes échelles. On peut noter qu'un graphe multi-échelles ne représente qu'une des séquences de décomposition possibles. Toutes les séquences de décomposition possibles forment en fait un treillis appelé *treillis des décompositions* [Godin, 2003]. Dans ce treillis, certaines échelles de représentation ne s'emboîtent pas mais se chevauchent. La définition d'un graphe multi-échelles revient donc à choisir un chemin particulier dans ce treillis en fonction d'une sémantique de décomposition particulière.

4.4.3 Modèle géométrique multi-échelles

Un modèle géométrique multi-échelles est construit sur un graphe multi-échelles en associant à tous les sommets de l'échelle la plus fine un modèle géométrique et en associant aux sommets macroscopiques des informations géométriques globales sur les groupes qu'ils représentent, permettant ainsi de hiérarchiser cette information.

Comme pour le modèle précédent, les modèles géométriques définis aux différentes échelles doivent respecter des contraintes intra-échelle pour être valides. De plus, de nouvelles contraintes, dites contraintes inter-échelles, apparaissent et sont liées aux relations de décomposition. En effet, si un sommet se décompose en un ensemble de composants, une certaine cohérence entre les informations géométriques aux deux niveaux sera recherchée.

Formellement, un modèle géométrique multi-échelles se défini de la manière suivante :

Définition 9 *Soit $g = (h, V, \pi)$ un graphe multi-échelles emboîté. Soit $\mathcal{R}_0$ un référentiel lié à g. On appelle modèle géométrique du graphe multi-échelles emboîté g, le quintuplet $m(g) = (m(h), \Sigma, \omega, C, Q)$ où*

- *$m(h)$ est (récursivement) un modèle géométrique du graphe multi-échelles emboîté h;*
- *(Σ, ω, C) est un modèle géométrique de la projection de $g : P(g) = (V, E)$.*

- *Q est un ensemble de contraintes (équations ou inéquations) indexées sur V, $Q = \{q_a\}_{a \in V}$. L'équation q_a établit une relation entre les modèles géométriques des composants de a,*

$$\{\omega(x) | \quad \pi(x) = a\},$$

et le modèle géométrique $\omega(a)$.

L'ensemble de contraintes Q est appelé *contraintes (géométriques) inter-échelles*. Nous dirons que le modèle $m(g)$ est un modèle géométrique *valide* si les conditions suivantes sont simultanément vérifiées :

- si le modèle géométrique de g à l'échelle macroscopique, (Σ, ω, C), est valide,
- si le modèle géométrique de h est lui-même valide,
- si les paramètres des modèles géométriques de ses constituants vérifient l'ensemble de contraintes inter-échelles Q : $\forall a \in V$,

$$q_a(\lambda_{\omega(a)}, \{\lambda_{\omega(x)}\}_{x \in \Pi(a)}) = 0$$

Définition 10 *Nous dirons que les contraintes C et Q sont* consistantes *pour g s'il existe au moins un modèle géométrique valide de la forme $m(g) = (m(h), \Sigma, \omega, C, Q)$.*

Modèles explicites et modèles différentiels

L'information géométrique associée au graphe peut être de nature diverse. Comme pour les graphes de scène, cela peut être par exemple une transformation ou un matériel qui s'applique à tous les composants du sommet. Cela peut aussi être une caractéristique globale permettant de calculer la géométrie des composants, telle qu'une longueur globale. Dans ce cas, le modèle géométrique multi-échelles défini sera dit *différentiel*. A chaque échelle les informations contenues dans les sommets raffinent les informations globales présentes dans leurs complexes en définissant une différence ou un delta local. Le modèle géométrique d'une entité est donc "la somme" de toutes les informations géométriques contenues dans les sommets du chemin du sommet le plus macroscopique vers le sommet de l'entité considérée.

Les macro-constituants du graphe représentent des modules globaux de l'objet modélisé. On peut aussi chercher à les représenter par des modèles géométriques élémentaires pour leur donner une représentation simplifiée et globale. Le modèle géométrique multi-échelles est alors dit *explicite*. On remarque que l'information géométrique aux différentes échelles est, dans ce cas, redondante puisqu'elle représente de manière détaillée et/ou simplifiée le même objet.

Dans cette deuxième approche, la décomposition guide le type de simplification obtenue, en donnant les différents groupes de géométrie à simplifier en un seul modèle. Toutes les applications n'utiliseront pas les mêmes types de décomposition. En infographie, des critères visuels ou de proximité spatiale peuvent être utilisés pour regrouper les entités. Pour l'analyse de l'architecture, les entités peuvent être regroupées suivant leur caractères botaniques (*e.g.* tous les entre-nœuds appartenant à la même unité de croissance). Si les différents types de décomposition ne sont pas pertinents pour toutes les applications, le principe d'un tel modèle reste cependant très général. Un seul type de décomposition n'est pas forcément suffisant et le choix même de cette décomposition n'est pas forcement simple à déterminer. Pour cela, il faut parfois explorer les différentes possibilités offertes par le treillis des décompositions. Une méthode pour simplifier cette exploration sur les arborescences multi-échelles est donnée à la section 5.3.

4.4.4 Exemple de modèles multi-échelles et de contraintes inter-échelles

Les contraintes inter-échelles proviennent du fait que deux modèles géométriques à deux niveaux de détails différents représentent un même objet réel. Si une entité y, représentant par exemple un système ramifié, est décomposée en un ensemble d'entités x_i, représentant les

branches contenues dans ce système ramifié, alors le modèle géométrique de y, $m(y)$ et l'ensemble des modèles géométriques des entités x_i, $\cup_i m(x_i)$, doivent avoir une certaine cohérence. Les contraintes inter-échelles vont en fait fixer les degrés de liberté et donc les différences possibles entre ces modèles. Le modèle géométrique $m(y)$ étant une représentation simplifiée de $\cup_i m(x_i)$, les contraintes expriment en fait les critères utilisés pour complexifier $m(y)$ et obtenir $\cup_i m(x_i)$ et inversement pour simplifier les $\cup_i m(x_i)$ et obtenir $m(y)$.

Nous illustrons ici, par différents exemples de géométrie multi-échelles, différents types de contraintes inter-échelles que l'on peut trouver dans la littérature dans différents types de modèles.

Représentations multi-échelles linéiques

Ce premier exemple porte sur la géométrie d'une branche à deux échelles. Une branche y est composée de quatre segments x_1, x_2, x_3 et x_4 comme l'indique le graphe multi-échelles de la Figure 4-6.c. Une représentation, à l'échelle des segments, est donnée par la Figure 4-6.d avec un modèle élémentaire par segment. Les Figures 4-6 a et b illustrent deux exemples de géométries globales à l'échelle de la branche. Dans la Figure 4-6.a, le modèle géométrique élémentaire choisi pour représenter la branche est un tronc de cône. Les contraintes inter-échelles liant ces deux représentations sont relatives aux positions et aux diamètres à la base et au sommet de la branche. Dans ce premier exemple, les contraintes inter-échelles sont appliquées :

- $c_1 : DiametreBase(m(y)) = DiametreBase(m(x_1))$
- $c_2 : PositionBase(m(y)) = PositionBase(m(x_1))$
- $c_3 : DiametreSommet(m(y)) = DiametreSommet(m(x_4))$
- $c_4 : PositionSommet(m(y)) = PositionSommet(m(x_4))$

La Figure 4-6.a constitue donc une représentation simplifiée de la branche ayant une certaine cohérence avec la représentation à l'échelle plus détaillée des entre-nœuds.

La Figure 4-6.b utilise une autre famille de modèles géométriques : les cylindres généralisés. Avec ce modèle, la géométrie des segments est mieux approchée. La contrainte résolue est donc d'être au plus prêt des points base et sommet des entre-nœuds, ce qui définit assez simplement l'axe du cylindre généralisé, et d'avoir les mêmes sections.

- $c_1 : \forall i \in [1,4]\,, \|m(y), PositionBase(m(x_i)\| < \varepsilon$
- $c_2 : \forall i \in [1,4]\,, \|m(y), PositionSommet(m(x_i)\| < \varepsilon$

L'axe du cylindre généralisé, dans ce cas, est une approximation par une courbe NURBS des points des segments, donnant ainsi un lissage des données et permettant une surface continue C_2.

La géométrie à l'échelle de la branche est calculée comme une simplification par rapport à celle de l'échelle des segments. Cette simplification permet, dans le cas de la Figure 4-6.b, d'obtenir une représentation plus lisse et donc de meilleure qualité visuelle. La complexité du modèle 4-6.a est inférieure à la complexité du modèle 4-6.d ce qui peut permettre d'optimiser certains calculs, notamment d'affichage. Le modèle de la Figure 4-6.b reste de complexité supérieure dans ce cas, mais dans des cas où la géométrie d'une branche est définie par plus de points, l'utilisation d'une approximation par une NURBS permet de simplifier (tout en lissant) les données.

Ce premier exemple portait sur des objets géométriques linéiques dont la topologie reste une simple séquence. Une des difficultés dans la représentation multi-échelles d'objets tels que les plantes réside dans la représentation globale d'objets dont la structure est arborescente et dont la géométrie est particulièrement irrégulière.

Représentations multi-échelles surfaciques

Cette section aborde le problème de la représentation multi-échelles de surfaces. Pour le calcul de l'interception de la lumière par une plante, l'orientation et la dimension de la surface

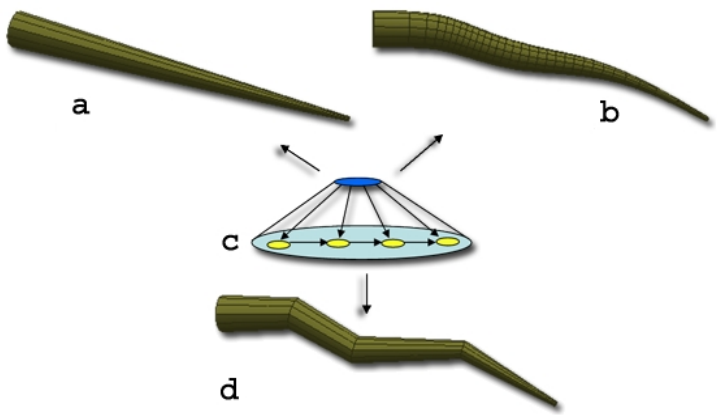


FIG. 4-6 – Représentation multi-échelles d'une branche composée de quatre segments. (c) Une première représentation à l'échelle des segments (d) et deux représentations possibles à l'échelle de la branche : la première (a) avec un simple tronc de cône et la deuxième (b) avec un cylindre généralisé.

des feuilles sont particulièrement importantes. Dans ce deuxième exemple, deux représentations à deux échelles différentes d'un système ramifié simple (Figure 4-7.a) sont liées par des contraintes inter-échelles sur ces deux paramètres.

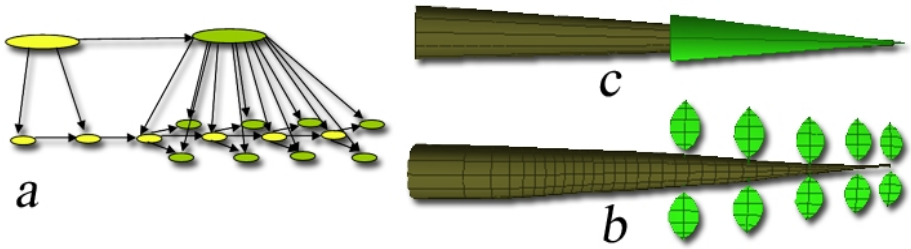


FIG. 4-7 – (a) Représentation multi-échelles d'un système ramifié simple. (b) Une première représentation à l'échelle des entre-nœuds et (c) une deuxième à l'échelle de la branche. Dans ce modèle, les contraintes inter-échelles portent sur l'orientation et la dimension de la surface des feuilles.

La représentation de la Figure 4-7.b, à l'échelle des entre-nœuds, et celle de la Figure 4-7.c, à l'échelle de la branche, ont en moyenne la même orientation et la même surface de feuilles. Dans la deuxième représentation, les feuilles ne sont représentées que par un seul modèle géométrique. On retrouve aussi l'utilisation de telles contraintes dans le travail de Zhang et Blaise [Zhang and Blaise, 2003] (voir section 3.3.2) pour la représentation multi-résolution du feuillage.

Dans le contexte de la simplification de surface, Popovic et Hoppe [Popovic and Hoppe, 1997] proposent une méthode pour automatiser la création de surfaces multi-résolutions. Les points d'un maillage représentant une surface sont fusionnés deux à deux pour créer de nouveaux points, représentant plus globalement une partie du maillage (Figure 4-8). La fusion de deux points x_1 et x_2 du maillage (notés $\{a\}$ et $\{b\}$ sur la figure 4-8 gauche) en un point macroscopique y (noté $\{a\}$ sur la figure 4-8 droite) résoud en fait une contrainte inter-échelles qui est que y doit

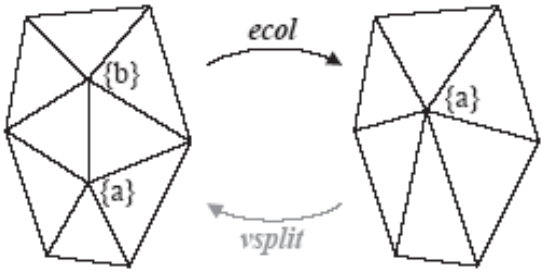


FIG. 4-8 – Création automatique d'une surface multi-résolution en fusionnant deux à deux les points du maillage. Tiré de [Popovic and Hoppe, 1997].

correspondre à l'isobarycentre de x_1 et x_2 :

$$c_1 : m(y) = \frac{m(x_1) + m(x_2)}{2}$$

Ces points sont ensuite stockés dans une structure hiérarchique. Cette structure est déterminée par l'ordre de fusion des points. Cet ordre est calculé comme un problème d'optimisation pour minimiser la déformation de la surface que l'on peut de nouveau simplifier en fusionnant deux de ces points. Chaque fusion de deux points a un coût lié à la déformation de la surface qu'elle impose. La fusion de moindre coût est choisie, créant ainsi une nouvelle structure plus macroscopique. Lors de la création de cette structure multi-échelles, différents chemins dans le treillis des décompositions sont examinés et celui minimisant les déformations successives est choisi.

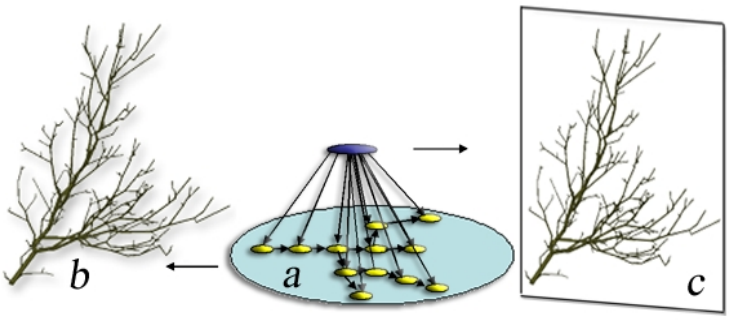


FIG. 4-9 – (a) Représentation multi-échelles d'un système ramifié. (b) Une représentation fine à l'échelle des entre-nœuds et (c) une première représentation globale. La représentation globale utilise un imposteur qui permet d'optimiser l'affichage.

Finalement, la géométrie détaillée d'un système ramifié peut être simplifiée par un imposteur (Figure 4-9). On trouve l'utilisation d'une telle technique notamment dans [Meyer et al., 2001]. Dans ce cas, la contrainte inter-échelles utilisée spécifie que la projection des deux représentations suivant un plan particulier doit être la même.

Représentations multi-échelles volumiques

La géométrie d'un système ramifié peut aussi être représentée macroscopiquement par son enveloppe, dans ce cas convexe (Figure 4-10). Ce type de représentation est utile notamment

dans l'étude d'un couvert forestier ou le volume occupé par un houppier permet de déterminer par exemple la quantité de lumière perçue par les différents étages de végétation. Le volume du système ramifié est donc approché au mieux à une échelle globale. Dans ce cas, la contrainte inter-échelles spécifie que la représentation fine doit être contenue dans l'enveloppe globale et que celle ci doit avoir une surface minimale.

FIG. 4-10 – Représentation macroscopique du système ramifié de la Figure 4-9 par une enveloppe convexe, qui donne une représentation volumique macroscopique du système.

4.4.5 Utilisation des contraintes inter-échelles

De manière similaire aux contraintes intra-échelle, les contraintes inter-échelles peuvent être utilisées pour différents buts : l'instanciation d'informations géométriques manquantes, la validation d'un modèle géométrique multi-échelles, et le maintien de la cohérence du modèle lors de modifications.

La première utilisation, particulièrement importante, permet en effet de calculer de nouveaux modèles géométriques simplifiés ou détaillés à partir de modèles existant à d'autres échelles. Les contraintes permettent de spécifier le type d'instanciation désiré. Ce réseau de contraintes intra et inter-échelles peut être exploré algorithmiquement en utilisant différentes stratégies pour calculer de nouvelles représentations. Les deux plus classiques correspondent à des approches descendantes et montantes. Les approches montantes partent des feuilles de l'arbre de décomposition et résolvent les contraintes en remontant vers la racine de l'arbre. Cela revient à utiliser des informations géométriques détaillées en se servant des contraintes pour calculer des informations géométriques globales. Les approches descendantes utilisent les informations géométriques globales associées aux sommets macroscopiques pour inférer des modèles géométriques détaillés. Les différents exemples présentés précédemment donnent de premières illustrations d'utilisation des contraintes pour calculer des modèles géométriques globaux par une approche montante. Une illustration d'approche descendante se trouve dans le travail de Prusinkiewicz [Prusinkiewicz et al., 2001] (qui sera détaillé à la section 7.6.3), où la géométrie des feuilles d'un système ramifié est contrôlée par une fonction globale attachée au système ramifié.

L'utilisation de ces contraintes multi-échelles pour instancier de nouveaux modèles géométriques de plantes à différentes échelles a fait l'objet du travail de recherche de cette thèse et est présentée plus en détails dans les deux parties suivantes.

4.5 Utilisation d'un modèle géométrique multi-échelles

4.5.1 Choix d'une représentation

A partir d'un modèle multi-échelles géométrique valide construit sur un graphe multi-échelles g, il est possible de construire différentes représentations géométriques de g. Ces représentations peuvent utiliser des modèles à une échelle donnée ou bien faire intervenir des modèles à plusieurs échelles : elles correspondent à la notion de *coupe géométrique* dans un modèle géométrique multi-échelles.

Formellement, soit $g = (h, V, \pi)$ un graphe multi-échelles emboîté et $m(g) = (m(h), \Sigma, \omega, P, Q)$ un modèle géométrique valide de g.

Définition 11 *Une coupe C de x est une antichaîne maximale dans l'arborescence de décomposition de x, Figure 4-11.*

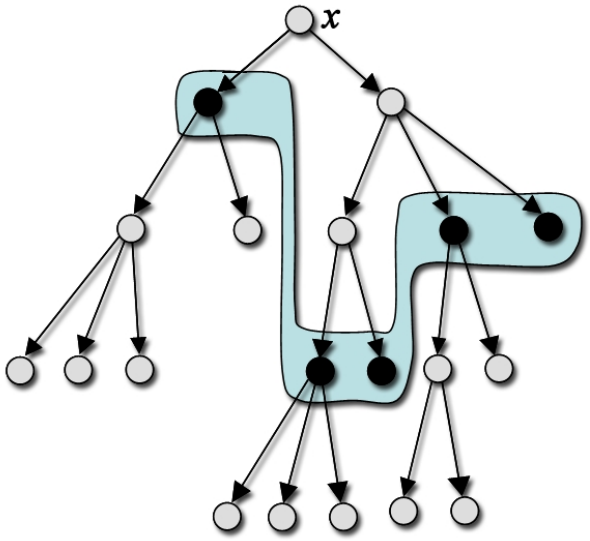


FIG. 4-11 – Coupe dans le graphe de décomposition de x. La coupe C est constituée de l'ensemble des sommets noirs.

Définition 12 *Soit W un ensemble de sommet de g. Un sous-ensemble C de W est une coupe de W si tout élément de C est dans la décomposition de W et si $\forall x \in W$, l'ensemble des éléments de C qui sont dans la décomposition de x, C_x, est une coupe de x.*

Définition 13 *Soit C une coupe de x. La* coupe géométrique *de x induite par C, noté $m_C(x)$, est définie par le modèle géométrique composé de x à partir de la décomposition de x dans la coupe :*

$$m_C(x) = m(C) = \cup_{y \in C} \omega(y)$$

Cette définition s'étend à un ensemble W de sommets : si C est une coupe de W :

$$m_C(W) = m(C)$$

Le choix d'une représentation (et donc d'une coupe dans l'arborescence de décomposition) peut se poser très globalement comme un problème d'optimisation sous contraintes. En effet,

le choix d'une représentation détermine la complexité de la représentation et peut donc être optimisé suivant le type de complexité voulue pour un calcul particulier. Un coût k_i est associé à chaque sommet v_i du graphe suivant le modèle géométrique qu'il contient. Le coût total d'une représentation sera égal à la somme des coûts des sommets de la coupe sélectionnée pour cette représentation. Le problème revient donc à choisir une représentation dont le coût total est maximal mais inférieur à la complexité maximale K souhaitée :

$$\max(\Sigma_{v_i \in C} k_i) \text{ avec } \Sigma_{v_i \in C} k_i < K$$

La résolution d'un tel problème sera illustrée dans le chapitre 6 en permettant de calculer une solution optimale au problème de la compression globale de modèles géométriques détaillés. Elle est cependant coûteuse en temps de calcul et ne convient donc pas pour des applications temps réels.

Une deuxième approche pour déterminer la représentation à utiliser est généralement préférée dans ces cas là. Le graphe de décomposition est parcouru de la racine vers les feuilles et à chaque sommet la complexité des modèles géométriques est examinée. Si cette dernière correspond à la complexité locale voulue alors le modèle est sélectionné, sinon les sommets fils sont examinés. Dans le cas de la visualisation des surfels [Pfister et al., 2000] par exemple, un surfel est sélectionné si sa projection sur l'écran ne recouvre au plus qu'un seul pixel. Si ce n'est pas le cas, les surfels aux échelles inférieures sont alors examinés.

4.5.2 Continuité inter-échelles

Certaines applications, comme par exemple la visualisation, posent le problème de la continuité entre les représentations à différentes échelles. Le modèle multi-échelles que nous proposons dans ce chapitre contient un nombre fini et discret d'échelles. Si deux représentations sont calculées à des échelles éloignées, le passage de l'une à l'autre est parfois trop discontinu et donne l'impression de deux choses différentes. Certains travaux ont étudié comment faire des transitions souples entre des modèles à des niveaux de détails différents.

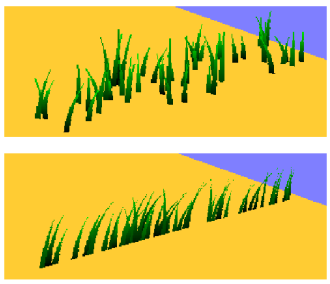


FIG. 4-12 – Transition entre deux représentations à deux niveaux de détails différents. La première (en haut) est construite par un ensemble de polygones, la deuxième grâce à une texture. Tiré de [Perbet and Cani, 2001]

Ces transitions sont fortement dépendantes des types de modèles géométriques utilisés dans les deux représentations. Afin de faciliter cette transition, les approches multi-échelles que l'on trouve dans la littérature utilisent généralement le même type de modèles à toutes les échelles (voir section 3.3). On trouve aussi dans le travail de Perbet et Cani [Perbet and Cani, 2001] un certain nombre de techniques pour faire des transitions continues entre plusieurs types de représentations.

4.6 Conclusion

Nous avons défini dans ce chapitre un modèle géométrique multi-échelles construit sur un graphe multi-échelles. Des représentations à différentes échelles peuvent ainsi être assemblées de manière cohérente, cette cohérence étant exprimée par des contraintes sur le graphe sous-jacent. Ce formalisme n'est lié à aucune primitive géométrique particulière. Il unifie donc les différentes représentations existantes. Des représentations volumiques, surfaciques ou linéiques peuvent être assemblées pour former des représentations d'un même objet à différentes échelles.

Ce modèle offre ainsi un cadre général qui permet d'étendre les modèles multi-échelles utilisés en infographie, pour l'optimisation du rendu par exemple [Neyret, 1998, Meyer and Neyret, 2000, Meyer et al., 2001], à la prise en compte des relations intra-échelle, permettant par exemple d'exprimer la continuité entre composants à une même échelle [Bloomenthal, 1985]. Il permet aussi d'étendre les approches classiques de représentation à base d'arborescence, dans le cadre de la modélisation ou la simulation de la croissance des plantes, tel que les L-systèmes, aux multi-échelles.

Nous abordons des applications de ce modèle liées à ces différents domaines dans les deux parties suivantes.

Deuxième partie

Approche montante : Représentation et analyse de structures macroscopiques émergentes dans les architectures végétales

Chapitre 5

Caractérisation volumique de structures macroscopiques

Le modèle géométrique multi-échelles présenté dans le chapitre précédent permet de combiner de manière cohérente des représentations détaillées et des représentations globales d'une même plante. Les raisons pour relier ces différentes représentations sont multiples. Une première raison est due à la complexité importante des modèles détaillés, que l'on cherche alors à simplifier pour une utilisation optimale dans différents calculs. Une deuxième raison, plus théorique, est que ce lien nous permet d'analyser la forme émergente d'une structure détaillée. Cette forme, résultant de la perception subjective de l'observateur, représente le comportement global de la géométrie de la plante qui émerge de processus de construction purement locaux. Il est nécessaire pour caractériser ces formes de développer des outils conceptuels permettant d'effectuer de tels changements d'échelles. Dans le cadre d'une approche montante, nous explorons, dans ce chapitre, les possibilités offertes par les représentations volumiques.

Nous commençons donc par présenter un ensemble d'algorithmes qui permettent d'inférer une représentation volumique globale à partir de la géométrie détaillée d'un système ramifié. Certains sont assez classiquement utilisés en infographie, comme par exemple les enveloppes convexes, les boîtes et les sphères englobantes. Nous en avons aussi défini de nouveaux pour calculer des modèles d'enveloppes introduits plus spécifiquement pour la modélisation de houppiers d'arbres, comme les enveloppes asymétriques ou extrudées.

Par une analyse statistique, nous caractérisons les différences de comportement de ces différents modèles. Nous définissons dans une deuxième partie une méthode algorithmique souple pour spécifier différentes échelles macroscopiques sur la structure d'une plante. Nous élargissons ainsi notre étude aux différents niveaux d'organisations possibles que l'on peut trouver sur une plante.

Finalement, nous abordons de manière plus détaillée les relations existant entre des représentations à différentes échelles d'une plante. A travers deux applications particulières, nous montrons en quoi l'étude comparée de géométries à différentes échelles permet de caractériser le processus de croissance d'une plante ainsi que sa stratégie d'occupation de l'espace.

5.1 Algorithmes d'inférence de géométrie macroscopique

Différents types d'algorithmes pour inférer des modèles géométriques globaux à partir d'informations géométriques détaillées ont été implémentés. Ces algorithmes nous permettent de calculer une représentation volumique globale d'un ensemble de composants d'une plante. Ils sont donc dédiés à la représentation macroscopique de systèmes ramifiés.

5.1.1 Les algorithmes

Ces algorithmes permettent d'ajuster une enveloppe, délimitant une représentation volumique globale, à l'ensemble de points contenus dans une représentation détaillée.

Enveloppes

Cette première famille d'algorithmes permet de calculer une représentation globale de type enveloppe à partir du nuage de points formé par les modèles géométriques élémentaires.

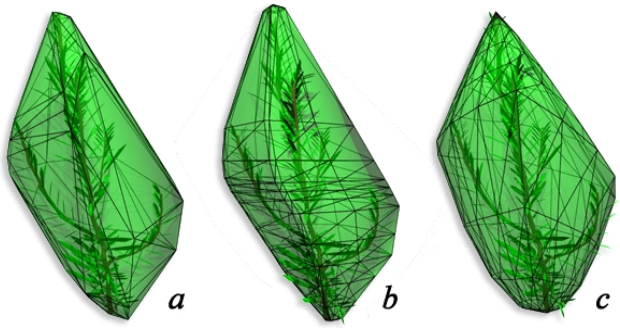


FIG. 5-1 – (a) Enveloppe convexe, (b) enveloppe extrudée et (c) enveloppe asymétrique d'un système ramifié.

Le premier algorithme calcule l'enveloppe convexe (Figure 5-1.a) du nuage de points (et donc des modèles géométriques détaillés). Nous utilisons dans ce cas l'algorithme QuickHull proposé par Barber *et al.* [Barber et al., 1996] de complexité $O(n \log h)$ où n est le nombre de points en entrée et h le nombre de points de l'enveloppe convexe. Intuitivement, cet algorithme commence par trouver un polyèdre composé de $d+1$ points en dimension d (soit un tétraèdre en dimension 3) appartenant à l'enveloppe convexe (en prenant les points qui ont des coordonnées maximales ou minimales sur les axes du repère). Chacune des faces du polyèdre est orientée en direction de l'extérieur de l'enveloppe. En effet, le plan contenant une face partitionne l'espace en deux, dont une partie totalement extérieure à l'enveloppe (l'autre partie de l'espace contenant le polyèdre). Pour chacune des faces du polyèdre, l'algorithme examine si la partie extérieure contient des points du nuage. S'il existe, le point extérieur le plus éloigné de la face est alors inclut au polyèdre. L'algorithme continue ce processus récursivement jusqu'à ce qu'il ne reste plus de point à l'extérieur.

Le deuxième algorithme (Figure 5-1.b) calcule l'enveloppe extrudée [Birnbaum, 1997] (voir section 1.3.2) du nuage de points. Pour cela, les profils convexes 2D des projections horizontale et verticale du nuage de points sont calculées. Ces deux projections convexes donnent les deux profils nécessaires à la construction de ce type d'enveloppe. On peut noter que si les deux projections sont convexes et englobent toutes les projections des points sur les plans horizontal et vertical, cela ne garantie pas que l'enveloppe englobe tous les points. Deux profils ne permettent pas de rendre compte de toute l'irrégularité de la forme. On peut en effet imaginer une forme avec un profil fin suivant un point de vue particulier et très large suivant un point de vue perpendiculaire.

Nous avons finalement implémenté un algorithme (Figure 5-1.c) qui nous permet de calculer l'enveloppe asymétrique [Cescatti, 1997] (voir section 1.3.1) correspondant au nuage de points. Pour cela, six points extrêmes P_i suivant les directions x, $-x$, y, $-y$, z et $-z$ sont choisis dans le nuage de points. Les facteurs de forme sont ensuite calculés pour que l'enveloppe englobe au mieux tous les points : la valeur minimale qui englobe tous les points est choisie. Pour chaque

point la valeur minimale du facteur de forme qui permet à l'enveloppe de l'englober est calculée et la valeur maximale sur tous les points est sélectionnée.

Pour calculer ces valeurs, on détermine, pour chaque point P, dans quelle partie de l'enveloppe il se trouve (en le comparant aux points extrêmes choisis). On peut ainsi déterminer la ligne horizontale de l'enveloppe qui correspond au point. Cette ligne correspond à un quart de superellipse d'équation

$$(\frac{x-x_i}{a})^n + (\frac{y-y_i}{b})^n = 1$$

Les valeurs de a, b, x_i, y_i, x et y sont déterminées par les coordonnées des P_i et de P. Le degré n de l'équation, correspondant au facteur de forme de l'enveloppe (haut ou bas suivant la position du point), est alors déterminé par une recherche récursive :

Les P_i choisis étant les points extrêmes du nuage de points, nous avons $\frac{x-x_i}{a} < 1$ et $\frac{y-y_i}{b} < 1$. De plus si P appartient au quart de superellipse, alors $x - x_i \geq 0$ et $y - y_i \geq 0$. Posons $X = \frac{x-x_i}{a}$ et $Y = \frac{y-y_i}{b}$ avec $0 \leq X < 1$ et $0 \leq Y < 1$ et $f(n) = X^n + Y^n - 1$. Dans ce cas, $f(n)$ est une fonction strictement décroissante, puisque si $0 \leq X < 1$ et $0 \leq Y < 1$ alors $\forall n$, $X^{n+1} < X^n$ et $Y^{n+1} < Y^n$.

Le problème à résoudre revient donc à trouver n_0 tel que $f(n_0) = X^{n_0} + Y^{n_0} - 1 = 0$. Différentes méthodes classiques existent pour résoudre ce genre de problème, la plus célèbre étant la méthode de Newton-Raphson [Press et al., 1996]. Nous avons implémenté une méthode équivalente dite de la position fausse dans laquelle la solution est approchée par deux valeurs limites qui sont récursivement raffinées :

Posons $n_1 = 0$. Ce qui implique $X^{n1} = 1$ et $Y^{n1} = 1$. Nous obtenons $f(n_1) = X^{n_1} + Y^{n_1} - 1 = 1 \geq 0$.

Posons $n_2 = \max(\frac{\log 0.5}{\log X}, \frac{\log 0.5}{\log Y})$. Ce qui implique $X^{n_2} \leq 0.5$ et $Y^{n_2} \leq 0.5$. Nous obtenons $f(n_2) = X^{n_2} + Y^{n_2} - 1 \leq 0$.

Et donc $f(n_2) \leq f(n_0) \leq f(n_1)$ et $n_1 \leq n_0 \leq n_2$.

Puis récursivement, nous posons

$$n = n_1 - \frac{f(n_1)(n_2 - n_1)}{f(n_2) - f(n_1)}$$

Cette valeur nous donne la valeur de $f(n) = 0$ si f était une fonction linéaire sur $[n_1, n_2]$. Plus l'intervalle est petit plus cette hypothèse est vraie, ce qui nous permet de converger rapidement. Ensuite, si $f(n) \geq 0$ alors $n_1 = n$, sinon $n_2 = n$ jusqu'à ce que $f(n) = 0$.

Ces enveloppes délimitent globalement l'espace occupé par la géométrie détaillée. Elles sont utilisées dans la littérature pour représenter la géométrie globale d'un système ramifié (voir chapitre 1). On peut noter que les deux derniers algorithmes sont construits avec des techniques similaires à celles utilisées pour les mesures sur le terrain.

Axes d'inertie

Cet algorithme calcule les axes d'inertie d'un nuage de points. Il permet d'approcher les points par une distribution gaussienne. Le centre du modèle est l'isobarycentre du nuage de points. Les axes sont déterminés par les vecteurs propres de la matrice de covariance dont les longueurs sont égales aux valeurs propres de cette matrice. Différents modèles géométriques sont possibles pour représenter ces axes d'inertie : classiquement l'ellipsoïde est utilisé (Figure 5-2.a), mais les cylindres (Figure 5-2.b) et les boîtes orientées (Figure 5-2.c) permettent aussi de représenter cette information.

Les axes d'inertie permettent ainsi de caractériser l'orientation générale d'une forme telle que celle d'un système ramifié [Sinoquet et al., 1991, Christophe, 2002]. Cette orientation peut par exemple être corrélée à des facteurs environnementaux tels que le vent ou la lumière pour étudier l'influence de ces paramètres sur la forme générale de la plante.

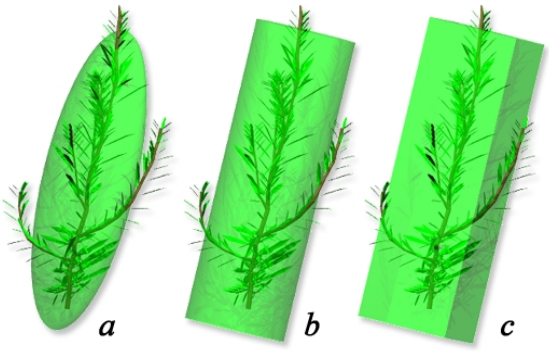


FIG. 5-2 – Représentation des axes d'inertie d'un système ramifié par (a) un ellipsoïde, (b) un cylindre et (c) une boite orientée.

Volumes englobants

Cette famille d'algorithmes permet de calculer différents types de modèles géométriques englobants le nuage de points et de volume minimale. Cette famille contient les algorithmes pour calculer les formes suivantes : les boîtes englobantes alignées sur les axes (Figure 5-3.a), les boîtes englobantes orientées (Figure 5-3.b), les ellipsoïdes englobants (Figure 5-3.c), les cylindres englobants (Figure 5-3.d) et les sphères englobantes (Figure 5-3.e).

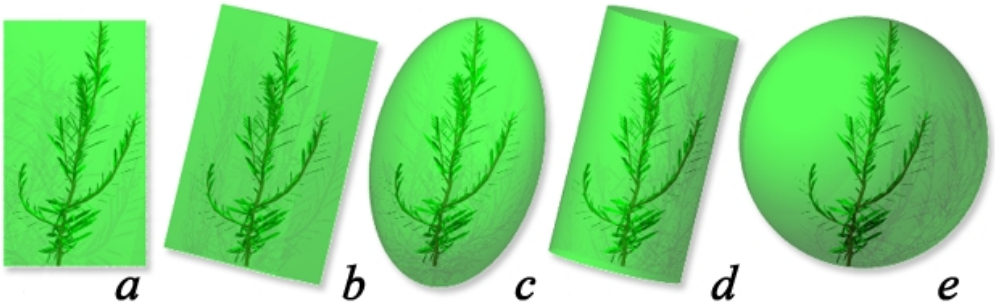


FIG. 5-3 – Différents modèles géométriques englobants de surface minimale : (a) boite alignée sur les axes x, y et z, (b) boite orientée, (c) ellipsoïde, (d) cylindre et (e) sphère.

Le premier algorithme se fait par une simple recherche linéaire des coordonnées minimales et maximales de l'ensemble de points. Pour le deuxième, nous avons utilisé l'algorithme en $O(n \log n)$ proposé par [Gottschalk et al., 1996] qui utilise les axes d'inertie de l'enveloppe convexe du nuage de points et calcule les coordonnées maximales sur cette base pour inférer une bonne approximation de la boîte englobante orientée de surface minimale. L'algorithme exact [O'Rourke, 1985] existe mais est en $O(n^3)$ pour un nuage de points en dimension 3.

Le dernier algorithme utilise l'algorithme linéaire proposé par [Welzl, 1991] qui recherche les $d+1$ points en dimension d (soit 4 en dimension 3) du nuage de points définissant la sphère englobante de surface minimale. De la même manière, l'ellipsoïde englobant peut être calculé en trouvant les $(d+3)d/2$ points la définissant. Nous n'avons pas utilisé ce dernier algorithme car il nécessite une implémentation complexe [Eberly, 2001] qui n'était pas nécessaire pour notre étude.

Les deux algorithmes pour les ellipsoïdes et les cylindres sont aussi des approximations

construites à partir des axes d'inertie de l'enveloppe convexe. Les valeurs propres des axes d'inertie donnent une première estimation des dimensions de l'ellipsoïde. L'ellipsoïde est ensuite redimensionné globalement pour contenir tous les points. Pour les cylindres, le cercle englobant [Welzl, 1991] de la projection des points sur les deux axes secondaires d'inertie nous donne la section et les coordonnées maximale et minimale suivant l'axe d'inertie principal définissent la hauteur.

La Figure 5-3 illustre l'utilisation de cette famille d'algorithmes sur un système ramifié simple.

Volumes moyens

Cette dernière famille d'algorithmes consiste à trouver une représentation globale moyenne. Tous les modèles géométriques en sortie sont centrés sur l'isobarycentre du nuage de points. Les dimensions des modèles sont définies par la distance moyenne des points au centre.

Le premier algorithme (Figure 5-4.a) calcule la distance moyenne du nuage de points au centre indépendamment suivant les axes x, y et z. Le résultat est une boîte alignée sur ces axes. Le second calcule les dimensions moyennes par rapport aux axes d'inertie du nuage de points et représentent ces dimensions par une boîte orientée comme l'illustre la Figure 5-4.b. Le troisième (Figure 5-4.c) calcule les dimensions cylindriques moyennes du nuage de points suivant les axes d'inertie. Le quatrième infère une sphère suivant la distance quadratique moyenne des points au centre du nuage de points (Figure 5-4.d). Le premier et le dernier algorithme sont linéaires tandis que les deux autres sont en $O(n \log n)$ puisqu'ils nécessitent le calcul des axes d'inertie du nuage de points.

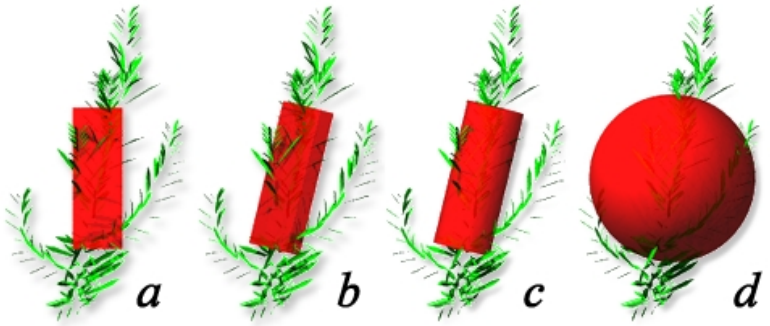


Fig. 5-4 – Différents volumes moyens d'un ensemble de géométries élémentaires, ici les feuilles d'un système ramifié. Le modèle a est aligné sur les axes x, y et z. Les modèles b et c sont alignés sur les axes d'inertie des géométries élémentaires. Les dimensions des modèles a et b représentent les dimensions moyennes suivants trois axes, le modèle c représente les dimensions cylindriques moyennes, et le modèle d la distance quadratique moyenne des points des géométries élémentaires au centre.

Ces algorithmes peuvent être utiles pour, par exemple, représenter en moyenne l'ensemble des feuilles d'un système ramifié, comme le montre la Figure 5-4. Contrairement aux modèles précédents, ces modèles ne permettent pas d'estimer le volume global occupé par le modèle détaillé. Ce type de modèles n'est généralement pas utilisé dans les études botaniques. Nous les avons introduits afin d'étudier si le fait d'avoir un modèle englobant est un critère important pour la représentation macroscopique d'un objet composé. Le calcul d'une moyenne étant une méthode classique de simplification, nous avons donc choisi d'évaluer ce type de représentation pour le comparer aux modèles précédents.

5.1.2 Critères d'évaluation

Ces algorithmes permettent tous de représenter globalement la géométrie d'un groupe d'entités. Différentes manières pour les évaluer et les comparer sont possibles. Une première évaluation est d'estimer la qualité de leur ajustement. Cette évaluation est basée sur les critères utilisés pour construire la représentation et dépend donc du type d'algorithme utilisé :

- La qualité d'ajustement d'un volume moyen est mesurée par la distance des points à la surface. On utilisera différentes distances : distance quadratique basée sur la norme quadratique pour la sphère moyenne, norme infinie pour la boîte alignée moyenne, *etc.*
- Pour les volumes englobants, le volume calculé doit être minimal. On peut donc comparer les volumes des différents modèles pour estimer leurs qualités, *etc.*

La qualité de l'algorithme dépend aussi de sa stabilité. On peut donc estimer leur sensibilité à certains critères, comme par exemple aux transformations :

- Tous les algorithmes sont insensibles aux translations (on retrouve le même résultat à une translation près).
- Ils ne sont pas non plus sensibles aux changements de dimensions (on retrouve le même résultat à un facteur de dilatation près), hormis les problèmes de précision de calcul en nombres flottants.
- La plupart des algorithmes sont insensibles aux rotations. Les boîtes alignées, moyennes et englobantes, ainsi que les enveloppes extrudées et asymétriques sont cependant dépendantes du repère dans lequel on exprime le nuage de points et ne donnent donc pas les mêmes résultats si l'on applique une rotation au nuage de points.

La stabilité des algorithmes s'expriment aussi par la variabilité des résultats lorsqu'on applique de petites variations au nuage de points en entrée.

Finalement, les différents modèles géométriques globaux peuvent être comparés suivant des critères liés à l'application qui les utilise. Pour la botanique par exemple, la surface ou le volume caractérisant globalement la plante est variable suivant le type d'algorithme utilisé. Une première étude de sensibilité sur ces deux derniers thèmes est présentée dans la section suivante.

5.2 Etude comparée des différents algorithmes d'inférence

Cette première étude a pour but de donner des premières méthodes et résultats pour l'évaluation des algorithmes et modèles présentés précédemment. Plusieurs problématiques liés à la représentation à deux échelles sont ici abordées. La première porte sur la relation entre les représentations détaillées et globales, la deuxième sur le choix du modèle global. Ces deux représentations ont-elles des ordres de grandeurs communes ? Suivent elles les mêmes lois de variation ? Les différents types de représentations macroscopiques sont elles équivalentes ? Pour évaluer cela, différents critères et méthodes sont utilisés. Nous examinons, sur un peuplement homogène équien[1] simulé, les ordres de grandeurs et les variations de différentes dimensions (volume et surface) des différentes représentations macroscopiques. Nous étudions ensuite si ces dimensions et variations des différents types de représentations sont corrélées entre elles et avec celles des représentations détaillées.

5.2.1 Peuplement homogène équien d'eucalyptus

Cette évaluation est basée sur un peuplement homogène d'eucalyptus (Figure 5-5) équien composé de 201 individus construits à partir d'un même algorithme[2] dont on a fait varier la graine aléatoire qui contrôle les tirages aléatoires utilisés pour la génération des différents arbres. Les différentes plantes sont donc fortement similaires.

[1] de même âge

[2] un axe de référence dans ce cas

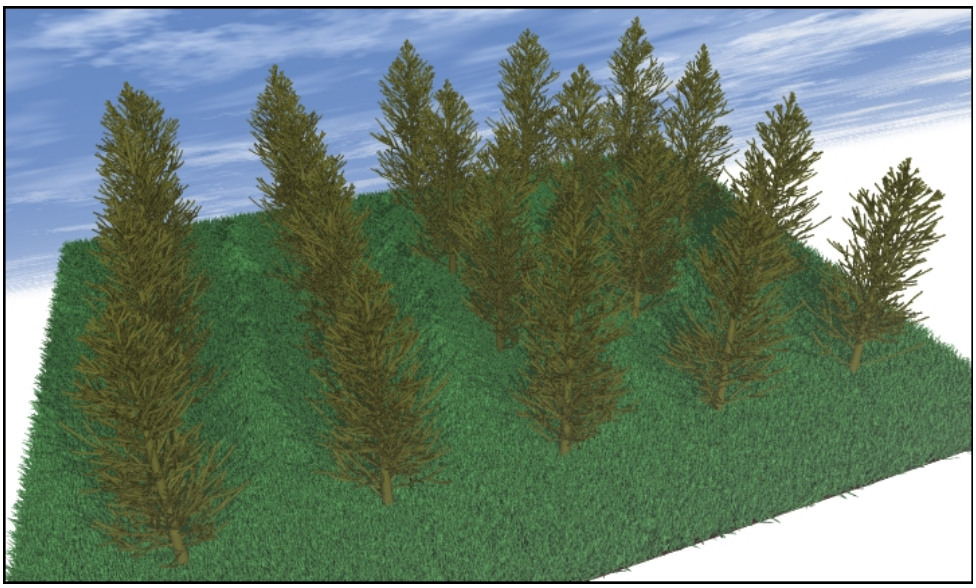


FIG. 5-5 – Peuplement homogène d'eucalyptus simulés avec le même algorithme mais des graines aléatoires différentes. Données de T. Coudurier et Y. Caraglio [Coudurier et al., 1994] construites avec le simulateur AMAPsim [Barczi et al., 1997] par Y. Caraglio. La géométrie a été reconstruite avec AMAPmod [Godin et al., 2002] et rendue avec POV-Ray [POV-Ray, 2003].

Nous avons donc testé les différents algorithmes pour calculer plusieurs représentations globales possibles du houppier de chaque arbre (Figure 5-6). La Figure 5-6 présente une première comparaison des résultats obtenus. Les modèles sont comparés suivant deux caractéristiques générales communes, leurs surfaces et leurs volumes. Nous avons calculé la surface (resp. volume) des représentations détaillées (noté Micro sur le graphique) comme la somme des surfaces (resp. volumes) de chacun des modèles géométriques élémentaires utilisés[3]. Ceci nous permet donc de comparer ces grandeurs à différentes échelles.

5.2.2 Ordre de grandeur des surfaces et volumes

Dans la famille des modèles moyens, on observe une grande similarité entre les résultats obtenus pour les cylindres, les boîtes alignées et orientées. Les valeurs de surfaces et de volumes en moyennes sont les plus faibles et les écarts en valeur absolue très faibles. On remarque que sur ce peuplement, ces deux derniers algorithmes donnent des résultats quasiment identiques, ce qui s'explique par le fait que l'orientation générale des arbres est relativement proche du repère initial. L'algorithme de calcul des sphères moyennes semble donner des modèles avec des dimensions plus importantes et plus sensibles à la variabilité du peuplement initial que les algorithmes précédents.

L'algorithme de calcul des axes d'inertie donne en moyenne des résultats assez similaires aux sphères moyennes pour les critères étudiés.

La famille des modèles englobants semble plus sensible à la variabilité de la géométrie détaillée du peuplement que les familles précédentes, notamment pour les algorithmes des ellipsoïdes, et des sphères. Les sphères englobantes ont les volumes les plus grands. Les ellipsoïdes ont les plus

[3] Dans cet exemple où nous ne considerons que le bois des eucalyptus, le volume des représentations détaillées correspond au volume de bois des arbres. Si l'on considère également les feuilles, cette notion est plus floue car le volume de feuilles (généralement considérées comme des surfaces sans épaisseur) est difficile à apprécier.

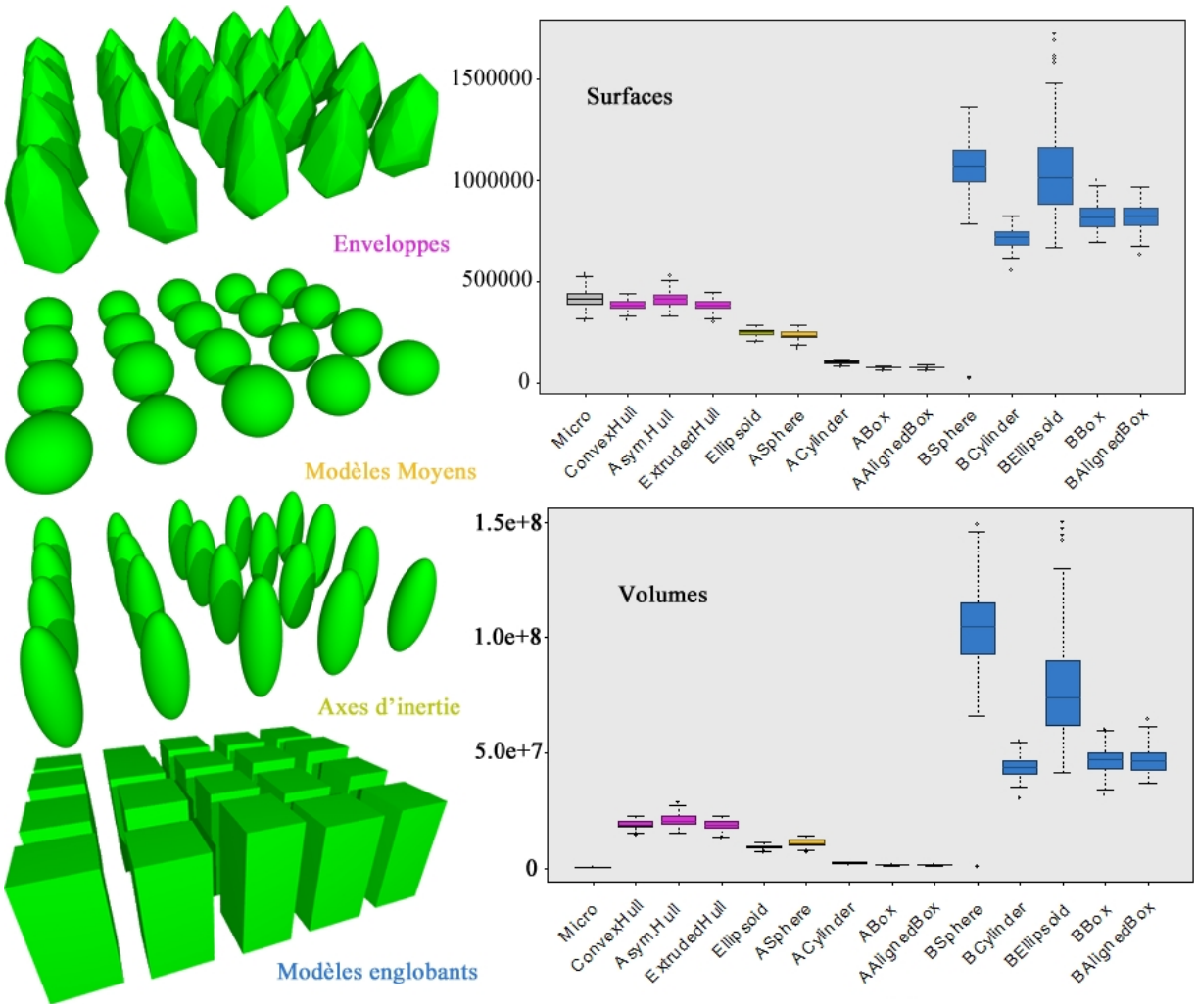


FIG. 5-6 – Les différents algorithmes d'inférence ont été utilisés pour calculer des représentations globales des houppiers d'un peuplement homogène. Les surfaces et les volumes des modèles obtenus sont comparés. Pour chaque type de modèle, l'ensemble des valeurs calculées sur le peuplement est représenté par une boite à moustache construite avec la moyenne, les valeurs maximale et minimale et les quartilles supérieur et inférieur. Les modèles sont classés par catégories auxquelles sont associées une couleur. Les modèles moyens sont coloriés en jaune, les modèles englobants en bleu, les enveloppes en pourpre et les ellipsoïdes représentant les axes d'inertie en vert. Les surfaces et volumes des modèles détaillés (Micro) sont représentés en gris.

grands écarts. Les boîtes englobantes alignées et orientées et les cylindres englobants donnent des résultats similaires, inférieurs aux deux algorithmes précédents.

Les modèles d'enveloppes donnent des résultats assez similaires. Dans cette catégorie, les enveloppes convexes ont les écarts les plus petits en valeur absolue.

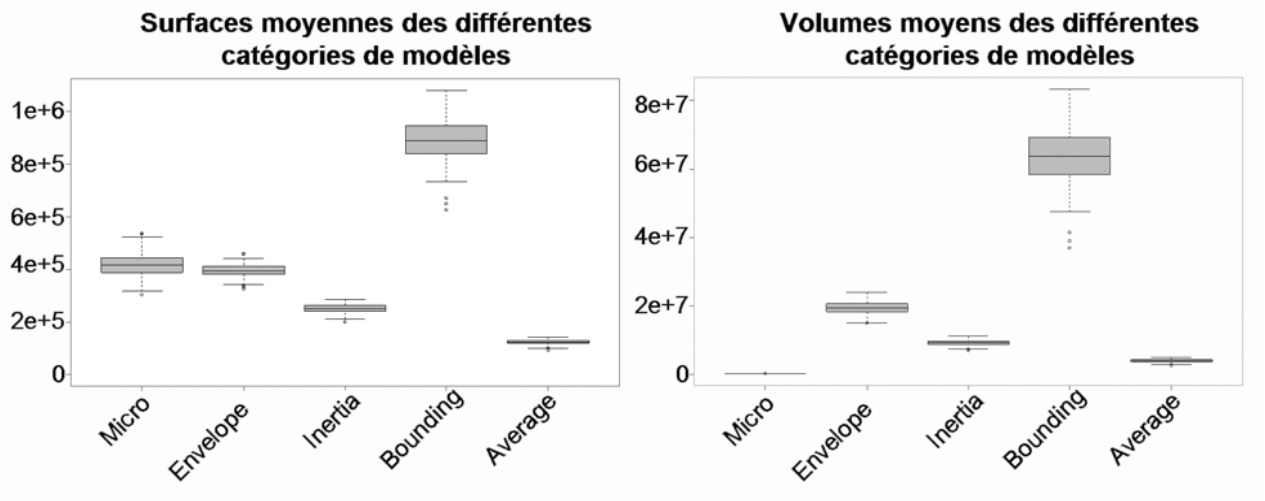


FIG. 5-7 – Comparaison des surfaces et des volumes obtenus pour les différentes catégories de modèles volumiques. Pour chaque catégorie, la moyenne des surfaces et des volumes des différents modèles est représentée. Les surfaces et volumes des modèles des représentations détaillés sont aussi donnés.

Les différentes catégories sont comparées entre elles et aux surfaces des modèles détaillés d'eucalyptus dans la figure 5-7. On observe que les surfaces et volumes moyens ainsi que l'écart en valeur absolue des différentes familles de représentations macroscopiques semblent différentes de celles des représentations détaillées. En moyenne, ces deux dimensions ordonnent les catégories de modèles de la manière suivante : modèles englobants, modèles d'enveloppe, représentation des axes d'inertie et modèles moyens. On trouve en moyenne un facteur 2 (entre 3 et 1.5) entre chaque famille consécutive. Le même facteur est observé pour les écarts en valeur absolue.

Pour cet exemple, les surfaces des modèles détaillés semblent être légèrement supérieures en moyenne et pour leur écart aux modèles d'enveloppes. Les volumes de ces modèles, par contre, sont en moyenne largement inférieurs à ceux de toutes les représentations macroscopiques, environs 30 fois plus petits que les modèles moyens et 500 fois plus petits que les modèles englobants.

Il est à noter que la relation entre les surfaces et les volumes des modèles détaillés et celles des modèles globaux n'est ici présentée que pour donner un ordre de grandeur. Cette relation n'est généralement pas linéaire. On considère en effet qu'un arbre a une géométrie avec des propriétés fractales. Un changement d'échelle introduit de nouveaux détails (qui ne sont pas mesurés à une échelle plus globale). On peut, dans ce cas, utiliser des représentations globales pour déterminer certaines caractéristiques telle que la dimension fractale de cette géométrie. Cette relation et son utilisation pour calculer la dimension fractale de la géométrie d'une plante sont présentés comme une application de ce travail dans la section 5.4.2.

5.2.3 Variabilité des surfaces et volumes

Nous étudions ici les ordres de grandeurs relatifs des variations des surfaces et des volumes des différents modèles macroscopiques présentés précédemment.

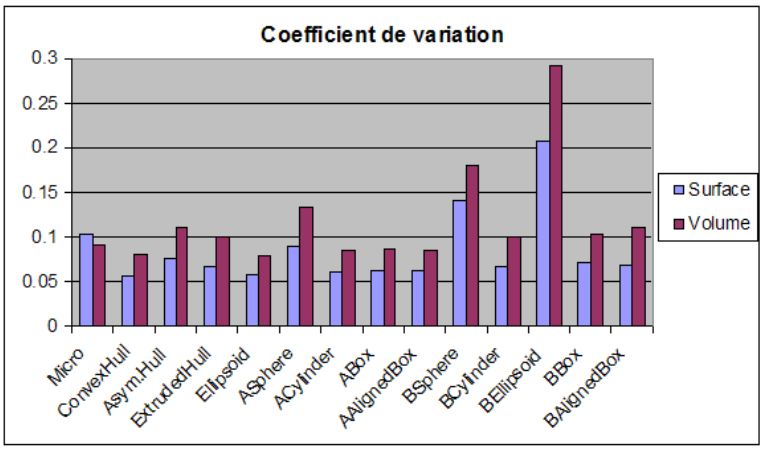


FIG. 5-8 – Coefficient de variation des surfaces et des volumes des modèles résultants des algorithmes d'inférence et des modèles détaillés.

La Figure 5-8 donne le coefficient de variation c_X des surfaces et des volumes pour chacun des modèles macroscopiques et des représentations détaillées.

$$c_X = \frac{\sigma_X}{m_X}$$

avec m_X la moyenne des valeurs et σ_X l'écart type

$$\sigma_X = \sqrt{\frac{n\sum x^2 - (\sum x)^2}{n(n-1)}}$$

Ce coefficient est une grandeur sans dimension qui traduit la dispersion relative des valeurs autour de la valeur moyenne. Il permet de comparer la dispersion de données qui ont des ordres de grandeur différents.

On observe sur la Figure 5-8 que les représentations détaillées ont des variations de 0.10 et 0.09 respectivement pour les surfaces et les volumes. La plupart des modèles (les enveloppes, les axes d'inertie, les cylindres moyens et englobants, les boîtes orientées et alignées moyennes et englobantes) ont des variations inférieures ou équivalentes aux modèles détaillés : entre 0.06 et 0.07 pour les surfaces, et entre 0.08 et 0.11 pour les volumes.

Les sphères englobantes et moyennes, ainsi que les ellipsoïdes englobants ont, pour les deux critères, des variations plus importantes (respectivement 0.09 et 0.13 ; 0.14 et 0.18 ; et 0.21 et 0.29). Ceci peut s'expliquer par le fait que ces algorithmes sont certainement plus sensibles aux points extrêmes des représentations détaillées. Et donc pour des représentations détaillées de surfaces et volumes quasi identiques, si l'une d'elle a des points extrêmes beaucoup plus éloignés, elle aura une représentation globale avec des dimensions plus importantes.

Comme ils résument la géométrie détaillée, on peut donc noter que la plupart des représentations globales sont moins variables que les représentations détaillées. Le modèle des enveloppes convexes semblent être celui qui simplifie le plus les variations de la géométrie détaillée.

5.2.4 Corrélation des variations

Nous évaluons ici si les différentes représentations des eucalyptus du peuplement observent des variations similaires. Pour cela, pour deux types de représentations, un coefficient de corrélation des valeurs des surfaces et volumes obtenus est calculé.

Le coefficient de corrélation entre deux séries de données est défini par :

$$\rho_{X,Y} = \frac{Cov(X,Y)}{c_X c_Y}$$

avec $-1 \leq \rho_{X,Y} \leq 1$ et $Cov(X,Y)$ la covariance des deux séries (la moyenne des produits des déviations) définie par

$$Cov(X,Y) = \frac{1}{n}\sum(x - m_X)(y - m_Y)$$

Les Figures 5-9 et 5-10 donnent respectivement les matrices de corrélation entre les surfaces et les volumes des différents modèles. Pour chacun des couples de modèles, un coefficient de corrélation est calculé. Les valeurs sont coloriées suivant l'intensité de la corrélation. Ces corrélations sont calculées sur une échantillon de 201 arbres et ont donc un intervalle de confiance de 0.14 (les séries de points sont supposées suivre une loi normale et il y a 95% de chance que les valeurs réelles de corrélation soit compris entre ±0.14 de la valeur obtenue).

	Micro	Convex Hull	Asym. Hull	Extruded Hull	Ellipsoid	A Sphere	A Cylinder	A Box	A Aligned Box	B Sphere	B Cylinder	B Ellipsoid	B Box	B Aligned Box
Micro	1.00	0.59	0.38	0.53	0.36	0.22	0.36	0.34	0.35	0.39	0.39	0.11	0.43	0.35
ConvexHull	0.59	1.00	0.68	0.86	0.77	0.60	0.68	0.68	0.68	0.64	0.77	0.29	0.78	0.76
Asym.Hull	0.38	0.68	1.00	0.50	0.49	0.35	0.42	0.42	0.40	0.42	0.56	0.35	0.58	0.66
ExtrudedHull	0.53	0.86	0.50	1.00	0.64	0.52	0.57	0.57	0.56	0.59	0.62	0.18	0.68	0.53
Ellipsoid	0.36	0.77	0.49	0.64	1.00	0.89	0.97	0.97	0.96	0.53	0.50	0.15	0.57	0.60
ASphere	0.22	0.60	0.35	0.52	0.89	1.00	0.86	0.89	0.89	0.51	0.37	0.12	0.45	0.43
ACylinder	0.36	0.68	0.42	0.57	0.97	0.86	1.00	0.99	0.99	0.46	0.42	0.10	0.50	0.51
ABox	0.34	0.68	0.42	0.57	0.97	0.89	0.99	1.00	0.98	0.48	0.42	0.10	0.49	0.51
AAlignedBox	0.35	0.68	0.40	0.56	0.96	0.89	0.99	0.98	1.00	0.47	0.41	0.11	0.51	0.48
BSphere	0.39	0.64	0.42	0.59	0.53	0.51	0.46	0.48	0.47	1.00	0.49	0.31	0.50	0.50
BCylinder	0.39	0.77	0.56	0.62	0.50	0.37	0.42	0.42	0.41	0.49	1.00	0.45	0.80	0.77
BEllipsoid	0.11	0.29	0.35	0.18	0.15	0.12	0.10	0.10	0.11	0.31	0.45	1.00	0.42	0.41
BBox	0.43	0.78	0.58	0.68	0.57	0.45	0.50	0.49	0.51	0.50	0.80	0.42	1.00	0.73
BAlignedBox	0.35	0.76	0.66	0.53	0.60	0.43	0.51	0.51	0.48	0.50	0.77	0.41	0.73	1.00

1.00
0.9 - 0.99
0.8 - 0.89
0.7 - 0.79
0.6 - 0.69
0.5 - 0.59
0.15- 0.49
< 0.15

FIG. 5-9 – Coefficient de corrélation des surfaces des différents types de représentations du peuplement. Ce coefficient nous permet d'estimer si les représentations, sur ce peuplement, observent les mêmes types de variation.

On observe tout d'abord que les deux graphiques donnent les mêmes types de résultats avec des amplitudes légèrement moindres pour les volumes.

Les représentations détaillées ont leurs meilleures corrélations avec les enveloppes convexes, et les enveloppes extrudées. On remarque les très bonnes corrélations entre les modèles moyens et les ellipsoïdes qui semblent avoir des amplitudes de variations semblables. Les ellipsoïdes englobants ont les corrélations les plus faibles avec tous les autres modèles, ce qui semble indiquer que ce modèle (ou du moins notre implémentation) donne les résultats les moins représentatifs de la géométrie des arbres du peuplement. Cet algorithme semble plus sensible aux variations dues aux points extrêmes qui sont dans ce cas plus fortes que celles dues à l'ensemble des géométries détaillées. Cela était déjà souligné par la très forte variabilité des résultats de ce modèle. Certains coefficients de corrélation de ce modèle sont en dessous de 0.14, et il est donc possible qu'il n'existe pas de relation.

Il semble logique que les modèles géométriques globaux aient des variations différentes de celles des modèles géométriques détaillés. Les enveloppes et les modèles englobants, par exemple,

	Micro	Convex Hull	Asym. Hull	Extruded Hull	Ellipsoid	A Sphere	A Cylinder	A Box	A Aligned Box	B Sphere	B Cylinder	B Ellipsoid	B Box	B Aligned Box
Micro	1.00	0.65	0.36	0.54	0.43	0.25	0.41	0.41	0.42	0.47	0.37	0.12	0.42	0.34
ConvexHull	0.65	1.00	0.63	0.81	0.78	0.55	0.71	0.71	0.70	0.65	0.69	0.23	0.73	0.71
Asym.Hull	0.36	0.63	1.00	0.40	0.47	0.29	0.40	0.40	0.38	0.40	0.53	0.36	0.54	0.69
ExtrudedHull	0.54	0.81	0.40	1.00	0.60	0.44	0.54	0.54	0.53	0.56	0.48	0.09	0.58	0.39
Ellipsoid	0.43	0.78	0.47	0.60	1.00	0.75	0.98	0.97	0.95	0.52	0.46	0.11	0.53	0.57
ASphere	0.25	0.55	0.29	0.44	0.75	1.00	0.77	0.76	0.77	0.57	0.32	0.08	0.39	0.38
ACylinder	0.41	0.71	0.40	0.54	0.98	0.77	1.00	0.98	0.98	0.46	0.38	0.08	0.47	0.49
ABox	0.41	0.71	0.40	0.54	0.97	0.76	0.98	1.00	0.97	0.48	0.38	0.08	0.44	0.49
AAlignedBox	0.42	0.70	0.38	0.53	0.95	0.77	0.98	0.97	1.00	0.46	0.37	0.09	0.48	0.45
BSphere	0.47	0.65	0.40	0.56	0.52	0.57	0.46	0.48	0.46	1.00	0.49	0.31	0.50	0.50
BCylinder	0.37	0.69	0.53	0.48	0.46	0.32	0.38	0.38	0.37	0.49	1.00	0.44	0.75	0.72
BEllipsoid	0.12	0.23	0.36	0.09	0.11	0.08	0.08	0.08	0.09	0.31	0.44	1.00	0.39	0.38
BBox	0.42	0.73	0.54	0.58	0.53	0.39	0.47	0.44	0.48	0.50	0.75	0.39	1.00	0.67
BAlignedBox	0.34	0.71	0.69	0.39	0.57	0.38	0.49	0.49	0.45	0.50	0.72	0.38	0.67	1.00

FIG. 5-10 – Coefficient de corrélation des volumes des différents types de représentations du peuplement.

représentent la frontière des houppiers des plantes. Ils ne sont donc pas sensibles aux variations de géométrie à l'intérieur du houppier.

Les enveloppes convexes finalement ont les meilleures corrélations avec les autres modèles. Ce modèle semble être une représentation "moyenne" de tous les autres.

De cette première étude, nous pouvons conclure qu'une représentation globale n'a pas forcement des dimensions d'ordre de grandeur comparable avec celle d'une représentation détaillée, et qu'elle ne suit pas forcement le même type de variation. Des méthodes telles que l'analyse statistique de la moyenne et de la variance nous permettent de quantifier cette différence. Le choix du modèle de représentation global semble important, les dimensions et variations pouvant être très différentes. Des applications possibles de l'étude des différences de variations et de dimensions d'une représentation détaillée et d'une représentation globale portent sur la caractérisation de la stratégie d'occupation de l'espace par une plante et sont présentées dans la section 5.4.

5.3 Spécification d'une échelle macroscopique

Le calcul de représentation globale dépend de la spécification des groupes d'entités microscopiques formant des entités macroscopiques. Les exemples précédents ne portent que sur des groupes formés par toutes les entités d'un individu, à l'échelle la plus macroscopique (*i.e.* des plantes ou des systèmes ramifiés entiers). Pourtant, des échelles intermédiaires d'organisation existent. Les graphes quotientés (et donc, en généralisant, les graphes multi-échelles) sont un formalisme topologique pour exprimer le regroupement d'entités microscopiques en entités macroscopiques.

Ce regroupement revient en fait à définir une relation d'équivalence sur les entités, les classes d'équivalence donnant les groupes. Cela se formalise par l'application d'une fonction de quotientement sur un graphe. Pour un graphe quelconque et un quotientement quelconque, spécifier une nouvelle échelle revient à donner cette fonction de quotientement, *i.e.* à associer à chaque sommet microscopique un sommet macroscopique. Dans le cas général, cette tache peut être complexe. Dans le cas des plantes, les structures topologiques aux différentes échelles sont des arborescences, ce qui permet d'utiliser un type particulier de graphes quotientés : les arborescences quotientées.

Définition 14 *Les graphes quotients dont le support et la projection sont des arborescences sont appelés des* arborescences quotientées.

Dans ce cas, les fonctions de quotientement ont la propriété d'être contractante [Godin, 2003] : les classes d'équivalences des sommets ayant même complexe $\pi(x)$ forment des parties connexes de l'arborescence support.

Proposition 15 *Soit $g = (h, W, \pi)$ un graphe quotient. Si le support $h = (V, E)$ de g est une arborescence,*

$$P(g) \text{ est une arborescence} \Leftrightarrow \pi \text{ est une fonction contractante}$$

Remarque : A l'inverse si la projection de g est une arborescence, et que π est une fonction contractante, le support de g n'est pas nécessairement une arborescence.

Si π est contractante, il existe alors un isomorphisme entre les sommets du support de g qui sont les racines des sous-arborescences formant les groupes et les sommets de la projection de g. Nous pouvons alors utiliser cette propriété pour spécifier une fonction de quotientement en donnant simplement les racines de chaque groupe de composants sur le graphe support.

Définition 16 *Soit $g = (h, W, \pi)$ un graphe arborescent quotienté. Soit la* fonction caractéristique *χ_g de g, de V sur $\{0, 1\}$, définie par :*

$$\chi_g(x) = \begin{cases} 1 \text{ si } x = root(\mathcal{G}(\pi^{-1}(x))) \\ 0 \text{ sinon} \end{cases}$$

avec $\mathcal{G}(X)$ représentant la sous-arborescence engendrée par un ensemble de sommets X du graphe support.

Proposition 17 *Soit $h = (V, E)$ un graphe arborescent simple. Soit ρ un prédicat sur V, tel que $\rho(root(h)) = 1$. Alors il existe (à un isomorphisme près) un graphe arborescent quotienté g, de support h, tel que $\chi_g = \rho$.*

En effet, considérons le graphe quotienté $g = (h, W, \pi)$ tel que :

- W est un ensemble de sommets de taille $\left|\rho^{-1}(1)\right|$. On suppose que les sommets de W sont indexés par les éléments de $\rho^{-1}(1)$: $v_x, v_y, ...$
- $\pi : V \to W$ définie par : $x \to \begin{cases} \pi(x) = v_x & \text{si } \rho(x) = 1 \\ \pi(x) = Father(v_x) & \text{sinon } (i.e.\ \rho(x) = 0) \end{cases}$

Par construction, π est surjective. On vérifie en outre que $\forall z \in W$, $\mathcal{G}(\pi^{-1}(z))$ est connexe.

Godin [Godin, 2003] se sert de ces propriétés pour montrer que si tout graphe arborescent étiqueté peut se représenter par une expression bien parenthésée alors tout graphe arborescent quotienté étiqueté peut aussi se représenter par une expression bien parenthésée. Il généralise ces résultats aux graphes arborescents multi-échelles.

Dans notre cas, nous utilisons ces propriétés pour construire simplement les fonctions de quotientement. En effet, comme le montre la proposition précédente, donner un graphe arborescent $h = (E, V)$ (Figure 5-11.a) et un prédicat ρ sur V, tel que $\rho(root(h)) = 1$ (Figure 5-11.b), nous permet de reconstruire la fonction de quotientement π (Figure 5-11.c) et donc le graphe arborescent quotienté $g = (h, W, \pi)$ (Figure 5-11.d). Ce prédicat peut être vu comme une fonction de coloration des sommets racines. Il permet d'associer chacun des sommets de l'arborescence initiale à un sommet de la projection. Un sommet racine est associé à un nouveau sommet macroscopique. Un sommet non racine a le même complexe que son plus proche ancêtre racine. Comme nous travaillons sur des arborescences, cet ancêtre le plus proche est unique pour tous les sommets qui ne sont pas des racines.

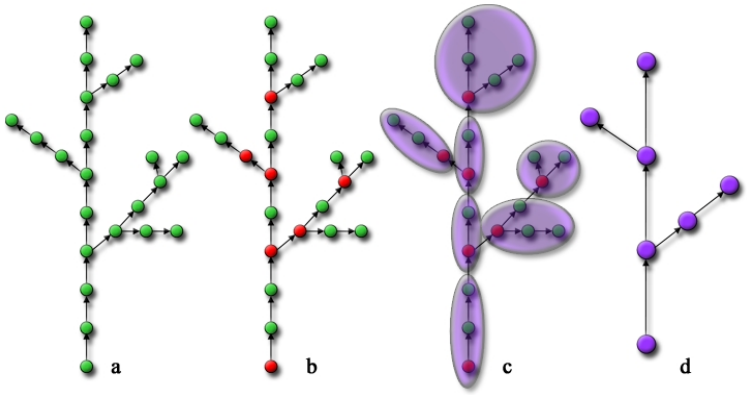


FIG. 5-11 – Contruction d'un graphe arborescent quotient à partir (a) d'un graphe arborescent et (b) d'un prédicat qui colorie les sommets du graphe qui sont les racines des groupes (en rouge). De ces deux informations, on peut reconstruire (c) la fonction de quotientement, qui associe à toutes les sous-arborescences un sommet macroscopique, et finalement (d) le graphe quotienté.

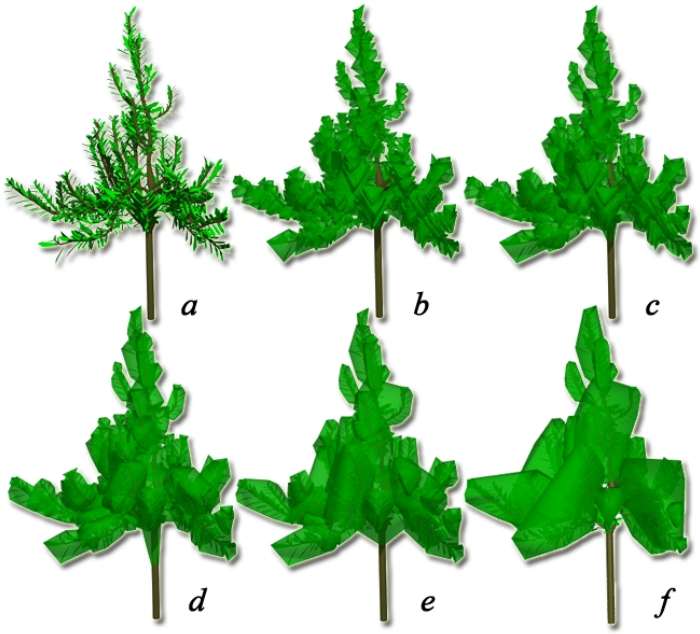


FIG. 5-12 – Exemple de création d'échelles intermédiaires de représentation. Les échelles sont créées algorithmiquement. Les sommets dont la profondeur est un multiple de (b) 5, (c) 10, (d) 20, (e) 30 et (f) 60 sont coloriés créant ainsi de nouveaux groupes à représenter. Données de Costes *et al.* [Costes et al., 1998].

La figure 5-12 illustre cette méthode de construction des groupes. Les groupes sont construits en fonction de la profondeur des entités dans l'arborescence : chaque groupe est constitué des entités dont les profondeurs dans l'arborescence sont comprises entre αp_0 et $(\alpha + 1)p_0 - 1$. Cet exemple de construction, purement algorithmique, se fait en coloriant tous les sommets dont la profondeur est un multiple de p_0. Différentes valeurs de p_0 sont utilisées et représentées.

Nous avons donc présenté un mécanisme flexible de regroupement des entités en structures macroscopiques. Ce mécanisme nous permet d'explorer de manière souple les différentes échelles contenues dans une plante. On remarque que la simplicité de cette méthode est due à la topologie arborescente des plantes.

5.4 Applications

Nous présentons ici différentes applications de la représentation multi-échelles de l'architecture des plantes. L'exploration multi-échelles de l'organisation d'une plante, l'étude de la dynamique d'un houppier à différentes échelles et finalement le calcul de la dimension fractale seront abordées sur différentes espèces de plantes.

5.4.1 Géométrie multi-échelles d'un eucalyptus en croissance

Dans cette première application, nous avons considéré la croissance d'un eucalyptus simulé à différents âges (Figure 5-13), créé avec le simulateur AMAPsim [Barczi et al., 1997]. La période de simulation va de 1 à 8 mois (sur une échelle notée de 10 à 250). Pour chacune des plantes aux différents âges, nous avons calculé l'enveloppe convexe de leur houppier (Figure 5-14). Nous obtenons ainsi un premier type de représentation multi-échelles dynamique. Nous étudions ici les différences de variations de ces deux représentations à deux échelles. Ces différences nous permettent de caractériser la stratégie d'occupation de l'arbre au cours du temps.

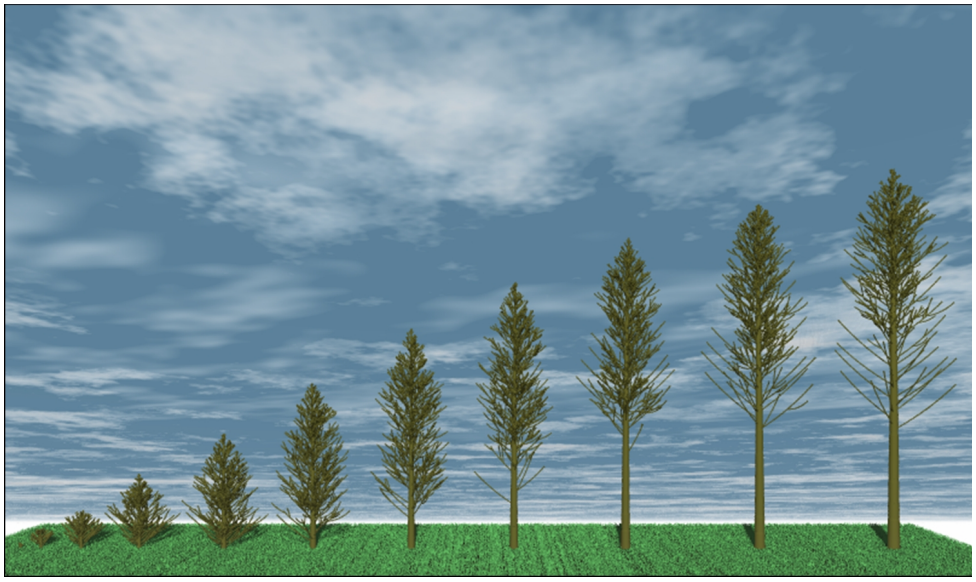


Fig. 5-13 – Eucalyptus simulé à différents âges. Données de T. Coudurier et Y. Caraglio [Coudurier et al., 1994] construites avec le simulateur AMAPsim [Barczi et al., 1997]. La géométrie a été reconstruite avec AMAPmod [Godin et al., 2002] et rendue avec POV-Ray [POV-Ray, 2003].

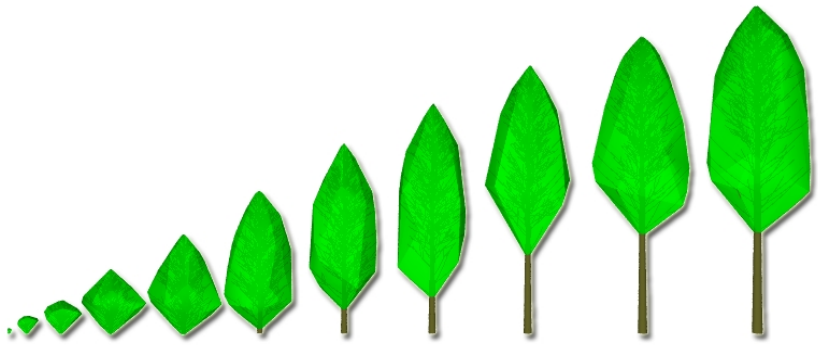


FIG. 5-14 – Calcul d'une représentation globale du houppier des eucalyptus à différents âges par une enveloppe convexe.

Au cours du temps les phénomènes suivants sont observés : les méristèmes créent de nouveaux entre-nœuds qui forment de nouvelles branches (croissance primaire), les branches s'épaississent (croissance secondaire) et finalement les branches les plus basses sont naturellement élaguées. Ces phénomènes sont reflétés par la surface et le volume du bois et de la couronne du houppier. Ces deux variables sont étudiées dans les figures 5-15 et 5-17 caractérisant ainsi la dynamique de la plante au cours du temps et sa stratégie d'occupation de l'espace.

La figure 5-15 donne de manière comparative les dynamiques des surfaces de bois et des couronnes. La première figure (5-15.a) donne les deux types de surfaces au cours du temps. La seconde figure (5-15.b) donne la quantité de surface de bois créée et l'expansion de la surface du houppier à chaque pas de temps. Ces quantités sont calculées de la manière suivante :

$$\Delta S(t) = \frac{S(t) - S(t - \Delta t)}{\Delta t}$$

Finalement, la dernière figure donne le pourcentage de surface de bois créé et d'expansion de surface de houppier :

$$\%S(t) = \frac{(S(t) - S(t - \Delta t)) * 100}{S(t) * \Delta t}$$

On observe qu'au cours du temps la quantité de bois générée est toujours supérieure à la quantité de bois perdue par élagage (Figure 5-15.a). Ensuite, on peut observer différents cycles dans la croissance de la plante (Figure 5-15.b). Des pas de temps 0 à 90, la quantité de bois créée est directement corrélée à l'expansion du houppier. D'un point de vue botanique, cette période correspond à la *phase d'établissement* de la plante. Dans la période $[90, 180]$, la quantité de bois créée est en forte baisse (de $5350cm^2$ à $1000cm^2$). La surface du houppier ne subit pas une baisse aussi importante ($4250cm^2$ à $3330cm^2$). On observe dans cette période le début du phénomène d'élagage. La quantité de bois créée semble principalement destinée à l'élargissement de la couronne plutôt qu'à l'élargissement des branches et à la densification de l'intérieur de la couronne. Autour du pas de temps 200, le phénomène semble s'inverser. Finalement après le pas de temps 225, la quantité de bois créée est en diminution ($2275cm^2$ à $1115cm^2$) tandis que le houppier est au contraire en expansion forte ($1450cm^2$ à $5720cm^2$). Ce dernier phénomène peut être interprété par un élagage important des axes d'ordre supérieur à 1 à la base du houppier[4] comme le montre la Figure 5-13. La couronne, à la base, est encore portée par les axes d'ordre 1 et est en expansion par le sommet.

[4]Cet élagage trop important pour être naturel est certainement un artefact du modèle.

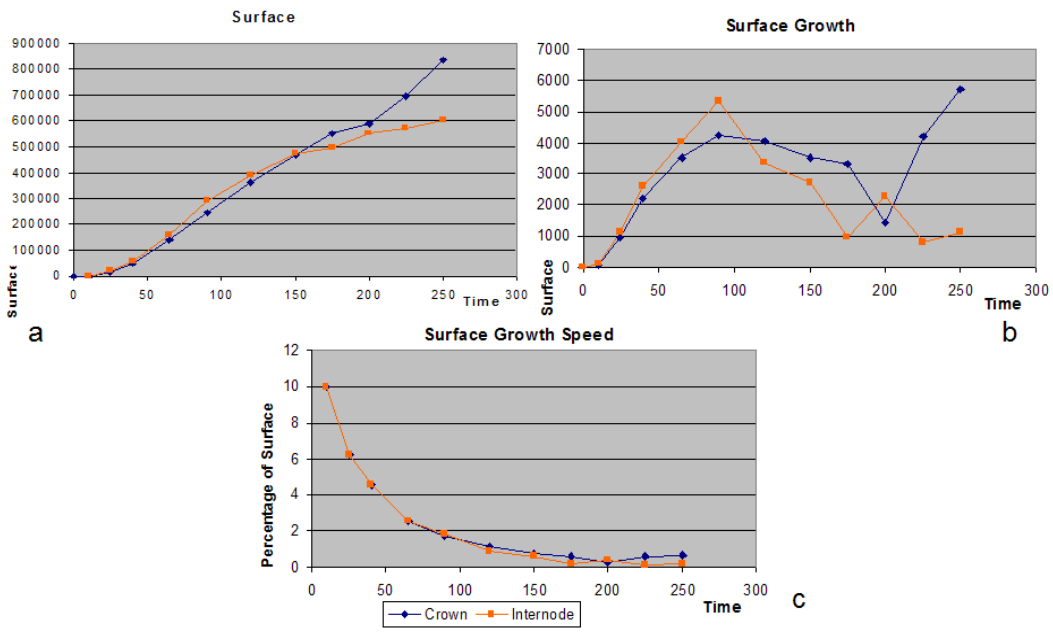


FIG. 5-15 – (a) Surface (en mm^2)de l'enveloppe convexe du houppier en bleu et des entre-nœuds en orange aux différents âges (en pas de temps) . (b) Surface générée entre deux pas de temps pour le houppier et pour les entre-nœuds. (c) Pourcentage de surface générée par rapport à la surface existante à chaque pas de temps.

La Figure 5-15.c nous donne l'ampleur du phénomène de croissance en comparaison à la structure existante. On peut voir que la quantité de bois créée n'est pas proportionnelle à la quantité de bois déjà existante. Ce qui peut être interprété comme le fait que la croissance de la plante est principalement faite par ces méristèmes apicaux : la croissance principale de la branche est due aux nouveaux organes mis en place sur les extrémités plus que par un élargissement des organes existants (croissance secondaire). On note que les phénomènes observés sur la courbe précédente se reflètent aussi sur cette courbe, mais leur amplitude est diminuée en fonction de la structure existante.

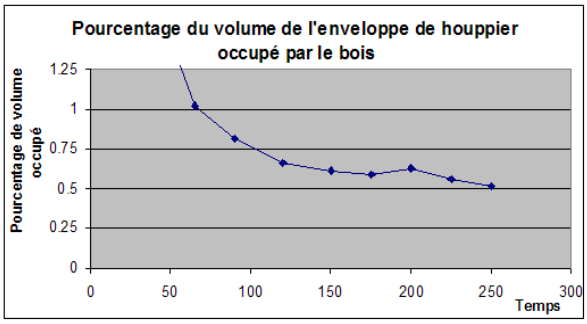


FIG. 5-16 – Densité du houppier. Ce graphique exprime le pourcentage de volume de l'enveloppe du houppier occupé par le bois.

La Figure 5-16 nous donne un autre indicateur de la densification de la couronne. Le pourcentage de volume de la couronne occupé par le bois est exprimé au cours du temps. La courbe

est globalement en diminution, le volume de la couronne augmente donc plus rapidement que le volume de bois. Autour du pas de temps 200, ce ratio augmente légèrement ce qui confirme les observations faites sur les diagrammes précédents. Une explication possible est la mise en place d'un complexe réitéré à l'intérieur de la couronne. Son expansion a d'abord des répercutions sur l'intérieur de la couronne et l'on assiste donc à une densification de celle ci. Ce complexe réitéré se développe ensuite sur la périphérie de la couronne, élargissant le volume occupé par celle ci.

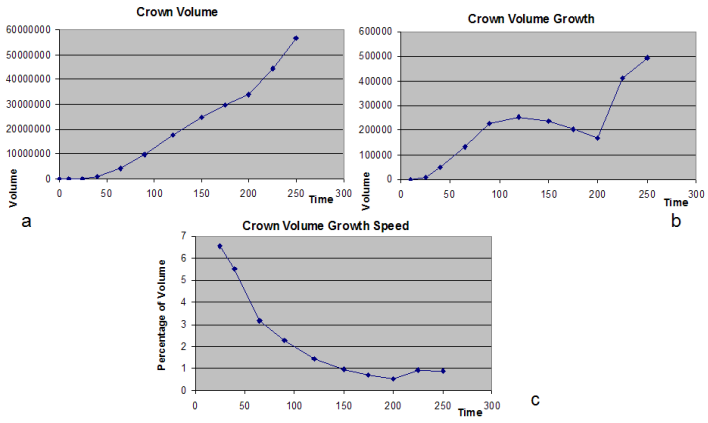


FIG. 5-17 – (a) Volume (exprimé en mm^3) de l'enveloppe convexe du houppier aux différents âges. (b) Volume généré entre deux pas de temps. (c) Pourcentage de volume généré par rapport au volume de bois existant à chaque pas de temps.

Les observations faites avec l'étude de la dynamique de la surface du houppier sont confirmées par l'étude du volume de celui-ci comme le montre la Figure 5-17 .

5.4.2 Dimension fractale de pêchers

A cause de leurs structures particulièrement intriquées, il est difficile d'aborder, avec la géométrie classique (euclidienne), l'analyse de l'espace occupé par le houppier d'une plante. L'exemple précédent, par une étude comparative des surfaces des entre-nœuds et des enveloppes des houppiers, donne une première idée de cette stratégie d'occupation. Elle manque pourtant d'indicateur concret du type de structure et de son occupation de l'espace. La géométrie fractale offre de tels indicateurs avec par exemple la notion de dimension fractale. Plusieurs méthodes existent pour estimer cette dimension, la plus connue étant la méthode des boîtes (pour une revue, voir [Godin, 2003]). Nous allons montrer comment les outils de calculs de géométries multi-échelles que nous avons développés permettent de déterminer cette dimension par une méthode alternative : la méthode des deux surfaces [Zeide and Pfeifer, 1991].

Nous commençons par présenter la base de données de pêchers digitalisés pour lesquels nous avons calculé différentes représentations à différentes échelles. Après une brève présentation de la méthode des deux surfaces, nous montrons finalement comment ces représentations à plusieurs échelles peuvent être utilisées pour calculer la dimension fractale de ces plantes.

Matériel

La base de données est donc constituée de quatre pêchers dont on a digitalisé la structure ramifiée (le bois) [Sonohat et al., 2004]. L'organisation des plantes a été formalisée par une arborescence multi-échelles à trois échelles. Elle est décrite en termes de :

- Segments : segments de bois, sans signification botanique mais utilisés pour la digitalisation. Echelle utilisée pour construire les représentations détaillées (Figure 5-18).

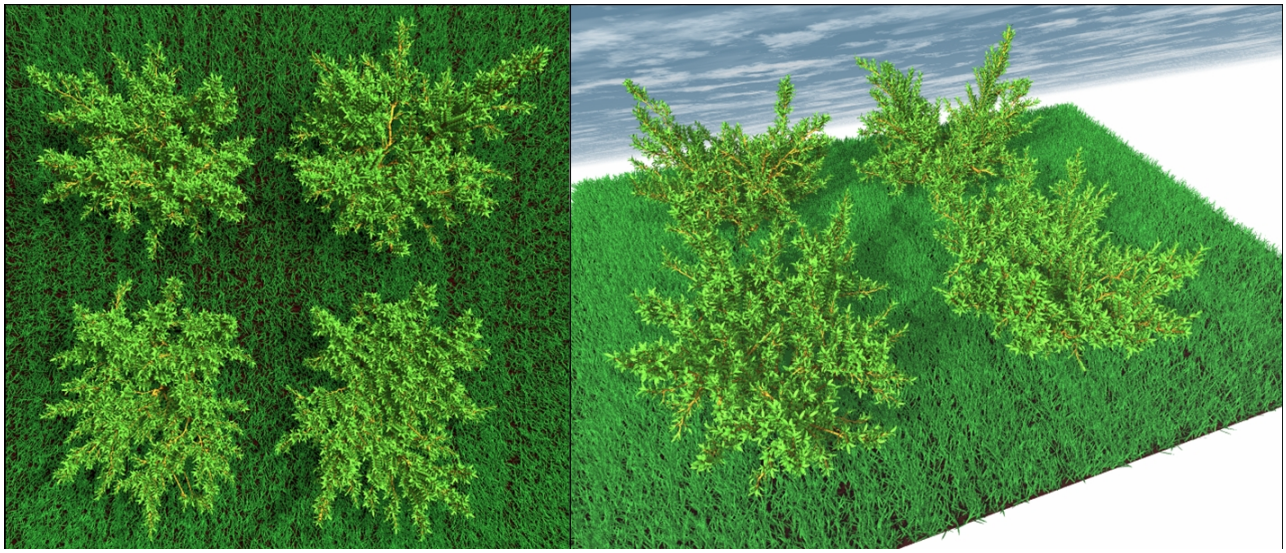


FIG. 5-18 – Quatres pêchers dont on a digitalisé la structure ramifiée (le bois). La géométrie du bois et des feuilles a été reconstituée sous AMAPmod [Godin et al., 2002]. Les maquettes ont finalement été rendues sous POV-Ray [POV-Ray, 2003]. Les données nous ont été fournies par H. Sinoquet.

- Pousses. Différents types de pousses composent ces plantes :
  - Pousses longues de l'année attachées au bois de 1 an.
  - Pousses courtes de l'année attachées au bois de 1 an.
  - Gourmands qui sont des pousses longues attachées à du bois de plus d'un an.
- Vieux bois, initiant les sous-systèmes ramifiés des plantes. Les systèmes ramifiés considérés commencent par des rameaux mixtes, qui sont les pousses de l'année précédente et portent les pousses de l'année. Le rameau mixte est l'unité de production fruitière et peut être vu comme une unité autonome au sein de l'arbre. C'est l'échelle considérée par l'arboriculteur pour la gestion de l'arbre (en particulier, lors de la taille où l'on conserve un certain nombre de rameaux mixtes dans des positions qui leur permettent de recevoir un bon éclairement).

Finalement les maquettes dont la géométrie a été organisée autour de la structure multi-échelles biologique décrite précédemment ont été complétées par des feuilles ajoutées à l'échelle des segments. Ces feuilles ne proviennent pas de la digitalisation mais ont été calculées à partir de densités et relations allométriques mesurées sur le terrain.

Les éléments des différentes échelles macroscopiques ont ensuite été représentés par leur enveloppes convexes (deux premières images de la figure 5-19). Deux autres représentations à des échelles plus macroscopiques ont été inférées algorithmiquement en utilisant la méthode présentée dans la section 5.3. La première est formée par les systèmes ramifiés initiés par les branches charpentières des pêchers (troisième image de la figure 5-19). Ces pêchers sont en effet formés d'un tronc très court qui se divise en quelques branches, dites charpentières, (entre 4 et 6) portant d'importants systèmes ramifiés. La dernière échelle est celle des houppiers, représentant les plants globalement.

Exploration multi-échelles

Ces représentations multi-échelles nous permettent comme le montre la figure 5-19 d'explorer la structure multi-échelles de ces pêcher. En effet, elles nous permettent de visualiser les différents groupes et systèmes ramifiés constituant les plants. L'utilisation de couleurs nous permet de

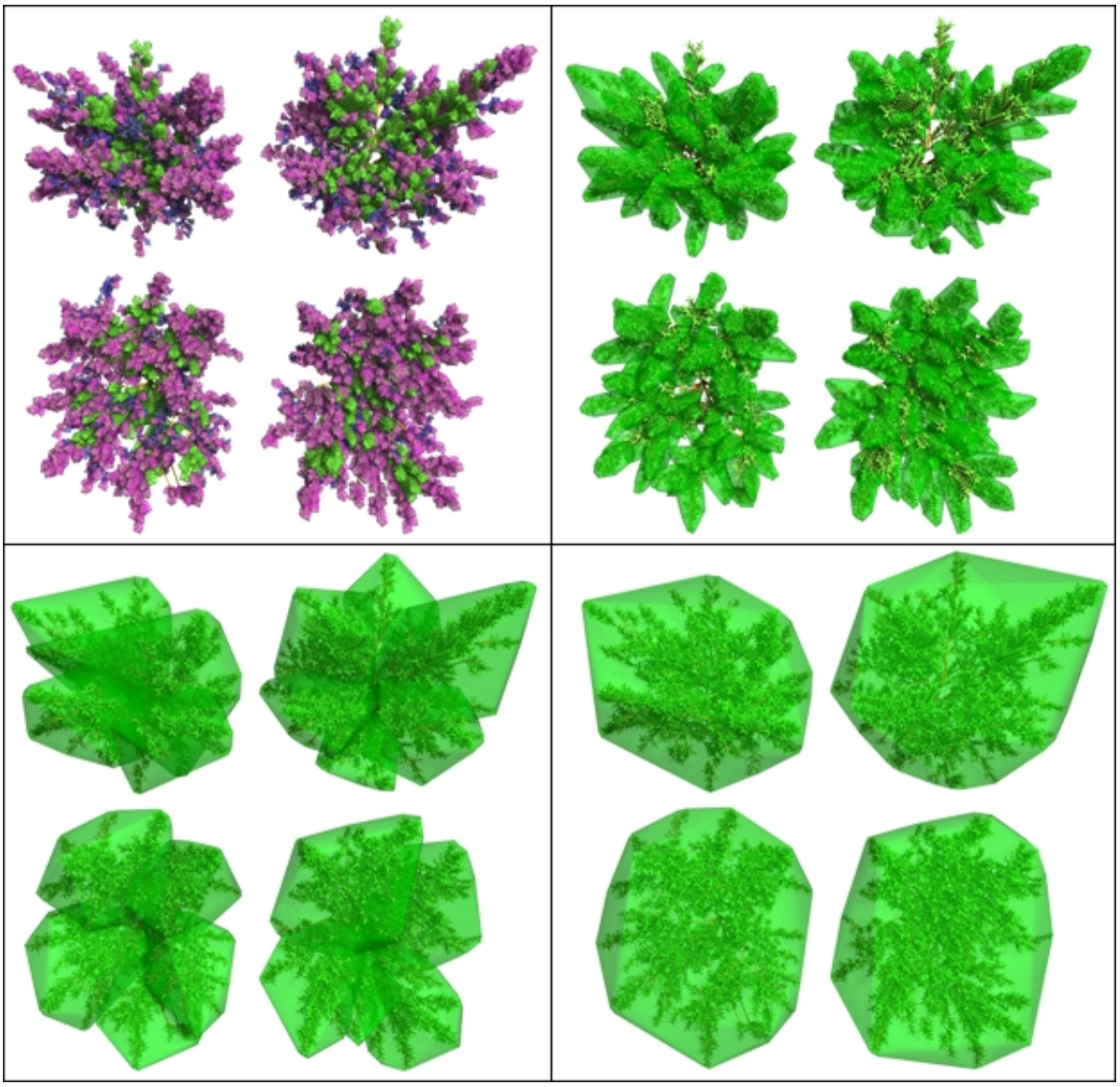


FIG. 5-19 – Représentation multi-échelles des différents pêchers. Quatres échelles macroscopiques ont été choisies : dans le sens de la lecture, représentations à l'échelle des pousses (pousses longues en violet, pousses courtes en bleu et gourmands en vert), des rameaux mixtes, des branches charpentières et des houppiers.

visualiser la répartition des différents types d'éléments à une même échelle. Sur la première image de la figure 5-19 représentant l'échelle des pousses, les pousses courtes sont représentées en bleu, les pousses longues en violet et les gourmands en vert. Ces derniers sont clairement situés vers la centre de la plante, tandis que les deux autres types de pousses se partagent la périphérie du houppier.

D'autres informations sont remarquables dans cette représentation multi-échelles. On peut, sur la troisième image, observer la répartition des principaux systèmes ramifiés de ces pêchers. Ces systèmes ramifiés peuvent s'enchevêtrer (pêcher en haut à gauche) ou au contraire se répartir équitablement l'espace (pêcher en bas à gauche). Cela nous donne donc des premières informations sur la stratégie d'occupation de l'espace. Cette information n'apparaît pas du tout dans une représentation détaillée de la plante. Le nombre important de feuilles masque en effet ces structures. Leurs représentations par une enveloppe globale permettent une claire identification de leurs étendues et de leurs constituants.

La méthode des deux surfaces

Il est possible de relier la mesure de l'extension spatiale (longueur, aire, volume, ...) de deux objets ayant des propriétés d'échelle différentes par une équation faisant intervenir leur dimensions fractales. L'idée consiste à comparer l'extension de deux objets P et Q, de dimensions respectives, d et D, différentes, en utilisant la différence dans leur variation d'extension (liée à leurs dimensions différentes). Les mesures μ_δ de P et Q obtenues à l'échelle δ suivent des lois d'échelle différentes corrélées de la manière suivante [Godin, 2003] :

$$\mu_\delta(P) = k.\mu_\delta(Q)^{\frac{d}{D}} \tag{5.1}$$

où k dépend de la mesure des objets initiaux.

Cette méthode permet de calculer la dimension fractale d'un objet en comparant sa surface et son périmètre (ou bien son volume et sa surface, *etc.*). Par opposition à d'autres méthodes, cette technique n'utilise la mesure de l'extension d'objet qu'à une seule échelle (les objets étant supposés avoir une même dimension fractale). Elle a été utilisée par [Zeide and Pfeifer, 1991] pour estimer les dimensions fractales de différentes espèces d'arbres. Pour l'appliquer, il faut considérer un échantillon d'individus dont on suppose que les couronnes sont caractérisées par une dimension fractale commune d. Pour chaque individu de cet échantillon, on considère deux objets géométriques : P est l'ensemble des feuilles de l'individu et Q est l'enveloppe convexe de sa couronne. Ces deux objets ont a priori des dimensions d et D différentes. L'enveloppe convexe étant une surface rectifiable (sur un intervalle assez petit, elle peut être considérée comme continue et dérivable), elle a une dimension $D = 2$. Si l'on suppose que toutes ces couronnes sont issues d'individus ayant une même dimension fractale, la surface de feuille S_F de chaque individu est reliée à la surface de son enveloppe convexe S_E par l'équation 5.1, où k et d sont des constantes caractéristiques de la population étudiée :

$$S_F = k.\,(S_E)^{\frac{d}{2}}$$

Les facteurs k et d peuvent alors être estimés par régression linéaire entre les variables $\ln S_F$ et $\ln S_E$ obtenues sur l'échantillon d'individus.

L'équation 5.1 permet de comparer la surface de feuille à d'autres caractéristiques de la forme géométrique de la couronne. Par exemple, la surface de feuille peut être comparée au volume total de la couronne V_E, objet de dimension 3. Dans ce cas l'équation 5.1 devient :

$$V_F = k'.\,(S_E)^{\frac{d}{3}}$$

Résultats

Nous avons donc appliqué cette méthode sur les pêchers avec les différentes échelles d'organisation qu'ils contiennent. La qualité de la régression dépend du nombre et de l'homogénéité des mesures sur lesquelles elle s'appuie. Il est difficile d'obtenir un grand nombre de maquettes digitalisées d'une même espèce de plante, car la digitalisation est une technique particulièrement coûteuse en temps. S'appuyer sur les systèmes ramifiés constituants les plants nous permet donc de minimiser cet effort.

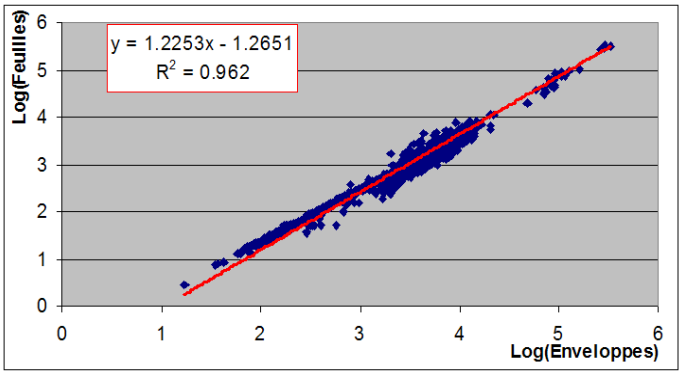


FIG. 5-20 – Calcul de la dimension fractale des pêchers avec la méthode des deux surfaces. Pour cela, une régression linéraire sur le log des surfaces de feuilles contenues dans les systèmes ramifiés et le log de la surface de leurs enveloppes convexes est introduite. La régression nous donne la valeur de $\frac{d}{2}$ et le coefficient de corrélation R, qui permet d'estimer la qualité de la régression. Tous les systèmes ramifiés à toutes les échelles sont utilisés. La dimension fractale obtenue est de 2,45.

La figure 5-20 donne la régression linéraire nécessaire pour calculer la dimension fractale des pêchers. Pour chaque système ramifié considéré, on calcule la surface de l'enveloppe convexe et la surface des feuilles qu'il contient[5]. La régression nous donne donc la valeur $\frac{d}{2}$ (puisque la surface de l'enveloppe est de dimension 2) et un facteur de corrélation R. Ce facteur exprime la variance expliquée par la régression sur la variance réelle des mesures. Il donne ainsi un coefficient de qualité de la régression. L'analyse de variance de ces données nous fournit aussi la *statistique de Fisher* (probabilité qu'un tel résultat soit observé s'il n'existe pas de relation linéaire). Cette probabilité nous confirme que la relation est significative.

Nous avons ensuite appliqué cette méthode séparément sur les différentes échelles d'organisation des pêchers (figure 5-21). En effet, les différents systèmes ramifiés contenus dans les pêchers nous permettent de calculer la dimension fractale à différentes échelles. Il est à noter que la dimension fractale à l'échelle des systèmes ramifiés est différente de celle à l'échelle des houppiers car les structures des systèmes ramifiés et de la plante entière sont différentes. Les plantes ne sont en effet pas des structures strictement auto-similaires. Cependant ces dimensions aux différentes échelles apportent déjà des informations sur le type de structure étudiée et le type de variation en échelle que l'on peut observer.

La probabilité de Fisher, toujours inférieure à 10^{-7} sauf pour le cas des représentations globales où elle est égale à 0.33, nous confirme que les relations sont significatives. Le tableau

[5]Il est à noter que, dans cette application, nous suivons rigoureusement la méthode indiquée par Zeide en ne considérant que la surface des feuilles pour l'échelle détaillée. Nous avons cependant d'autres informations disponibles tel que la surface et le volume du bois. Une suite intéressante de ce travail serait de prendre en compte cette information, voir de déterminer quel est l'influence relative de la structure ramifiée et de l'arrangement spatial des feuilles sur la dimension fractale de la plante.

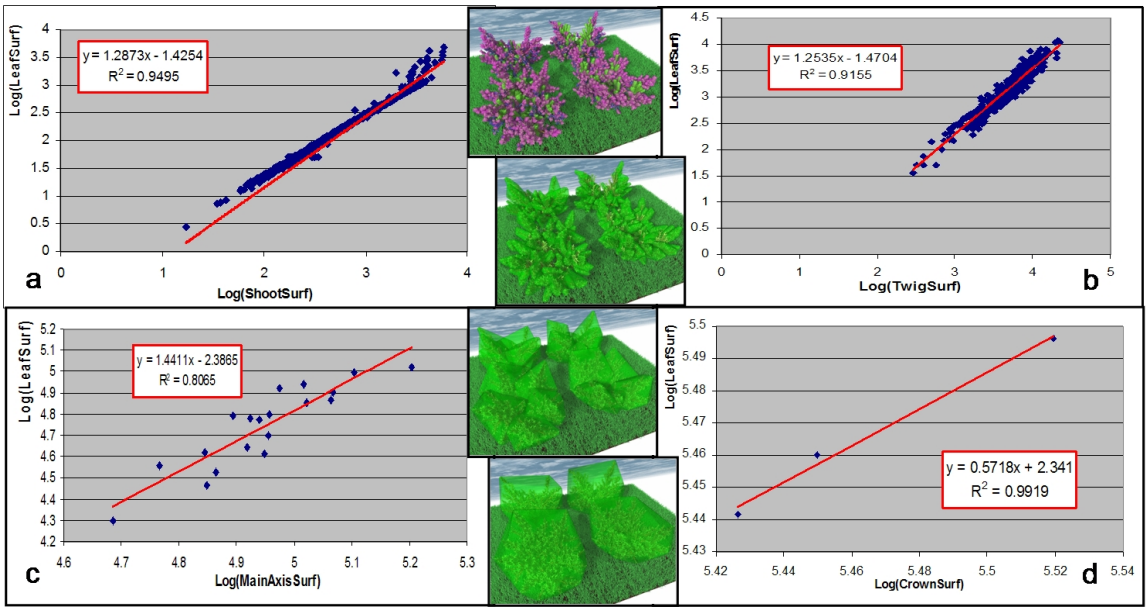


FIG. 5-21 – Calcul de la dimension fractale des systèmes ramifiés constituants les pêchers aux différentes échelles.

suivant résume la figure 5-21 en donnant les différentes dimensions fractales obtenues aux différentes échelles. On voit que le facteur de corrélation est plutôt bon pour les échelles des pousses, des rameaux et des houppiers. Il est moins bon pour l'échelle des branches charpentières. Les trois premières dimensions fractales sont entre 2.5 et 3. Cela exprime le fait que la structure du feuillage à ces trois échelles est entre la surface (dimension 2) et le volume (dimension 3). La quatrième valeur de dimension ne semble pas refléter le type de structure étudiée. Malgré le très bon coefficient de corrélation, cela s'explique par un échantillon beaucoup trop petit (4 individus).

	d	R^2
Pousses	2.57	0.95
Rameaux	2.50	0.92
Charpentières	2.88	0.81
Houppiers	1.14	0.99

Le nombre de pousses et de rameaux par pêcher étant assez important, il est possible de calculer une dimension fractale à partir des individus à ces échelles pour chaque pêcher. Ces valeurs, données dans le tableau suivant nous permettent une première estimation de la sensibilité de cette méthode. On observe des différences de 0.02 entre les dimensions fractales calculées à l'échelle des pousses et de 0.23 à l'échelle des rameaux.

	Pêcher 1		Pêcher 2		Pêcher 3		Pêcher 4	
	d	R^2	d	R^2	d	R^2	d	R^2
Pousses	2.57	0.93	2.58	0.95	2.56	0.96	2.56	0.95
Rameaux	2.66	0.91	2.44	0.92	2.6	0.93	2.43	0.93

Afin de valider et de déterminer plus clairement la sensibilité d'une telle méthode, la calcul de la dimension fractale par d'autre méthode, tel que la méthode des boîtes ainsi que l'estimation par cette méthode de la dimension d'autres types de plantes sont prévus pour poursuivre ce

travail.

5.5 Conclusion

Nous avons présenté, dans ce chapitre, différents outils pour le calcul de représentations volumiques macroscopiques de la géométrie des plantes. Un certain nombre d'algorithmes d'inférence de géométries macroscopiques sont présentés. Une analyse statistique nous a permis de comparer les différents types de représentations obtenues et de caractériser les relations entre la géométrie détaillée et la géométrie macroscopique d'une même plante. Ces deux types de représentation n'ont pas forcement des ordres de grandeurs de dimensions et de variations communes. Le choix du modèle macroscopique est important. Différentes applications pour la biologie, incluant la caractérisation multi-échelles de la géométrie complexe d'une plante ou le calcul de sa dimension fractale, ont été présentées pour illustrer l'intérêt de ces représentations multi-échelles.

Chapitre 6

Représentations géométriques compressées de systèmes ramifiés

Nous avons vu dans le chapitre précédent comment utiliser des représentations globales pour caractériser des structures émergentes de l'architecture des plantes. Dans un contexte plus infographique, nous nous sommes intéressés plus précisément au contrôle de la simplification introduite par ces modèles globaux dans le cas de la représentation de système ramifié.

Avec différents algorithmes d'inférence, nous pouvons calculer plusieurs représentations macroscopiques pour un même ensemble de géométries détaillées. Tous ces modèles ne sont pas équivalents. Ils offrent des représentations plus ou moins complexes de plus ou moins bonne qualité. Ces deux critères varient généralement dans des directions opposées. En effet, un modèle complexe, plus souple, permet généralement une représentation de meilleure qualité tandis qu'un modèle simple offre généralement une représentation peu précise.

Le problème du contrôle de la simplification est assez simple lorsqu'on considère une représentation globale qui inclue tous les composants détaillés. On est alors capable de quantifier la complexité et la qualité des différentes représentations possibles, et de choisir une représentation qui propose le meilleur compromis entre ces deux quantités. Le problème devient plus complexe lorsqu'on cherche à représenter une plante à une échelle intermédiaire. En effet, considérons le cas d'une représentation détaillée à l'échelle des entre-nœuds sur laquelle nous voulons calculer une représentation à l'échelle des axes (branches et tronc). Pour calculer une représentation à l'échelle des axes d'une certaine complexité globale, différentes stratégies sont possibles. On peut représenter tous les modèles avec la même réduction de complexité ou on peut favoriser les axes les plus importants. De plus la réduction de complexité globale ne permettant pas forcément de représenter toutes les entités, il faut parfois choisir quels composants représenter. Une certaine combinatoire apparaît donc dans le choix des modèles macroscopiques de toutes les branches.

Après avoir formalisé plus précisément ce problème, nous présentons une méthode générale de résolution. Nous présentons ensuite une heuristique, qui résout en temps linéaire ce problème en faisant des choix basés sur un ordonnancement des branches en fonction de leur importance visuelle et botanique. La validité de cette approche est enfin illustrée sur un noyer, dont on calcule différentes représentations compressées.

6.1 Calcul d'une représentation simplifiée

6.1.1 Choix d'un modèle

Le problème consiste à définir pour chacune des I entités v_i d'une plante à une échelle macroscopique, un modèle géométrique pour la représenter. Nous disposons donc d'un certain nombre de familles de modèles géométriques M_l (cylindre généralisé, enveloppes, *etc.*). Nous notons $\mathcal{M}$

l'ensemble de tous les modèles contenus dans toutes les familles de modèles géométriques :

$$\mathcal{M} = \{m_p : | : \exists l_p, \lambda_p \quad m_p = M_{l_p}(\lambda_p)\}$$

Chacune de ces familles est paramétrique, et en fixant la valeur du paramètre, nous obtenons un modèle géométrique particulier $m_0 = M_{l_0}(\lambda_0)$. Grâce aux différentes familles de modèles, nous pouvons trouver P modèles géométriques particuliers m_p pour représenter une entité.

Pour chaque modèle, nous définissons une fonction de coût $c(v_i, m_p)$ et une fonction de "consigne" $k(m_p)$. La fonction de coût $c(v_i, m_p)$ estime la qualité de la représentation de v_i par le modèle m_p. Pour calculer des représentations macroscopiques, nous utilisons des algorithmes d'inférence (comme par exemple ceux donnés au chapitre précédent). Une méthode classique pour estimer la qualité d'un modèle inféré est d'utiliser la distance des points approchés au modèle. La fonction de consigne $k(m_p)$ donne la complexité du modèle m_p, définie comme la quantité d'information nécessaire pour définir m_p. Cette fonction permet de définir le niveau de cette caractéristique que l'on veut imposer à la solution.

6.1.2 Contrôler la complexité de la représentation

Notre problème consiste à attribuer à chaque entité v_i un modèle particulier m_{p_i}. L'ensemble des modèles choisis forme le modèle représentant la plante

$$m(g) = \bigcup_{i=1}^{I} m_{p_i}$$

Le modèle de plante a une fonction de consigne et une fonction de coût définies par

$$K(m(g)) = k(\bigcup_{i=1}^{I} m_{p_i}) = \sum_{i=1}^{I} k(m_{p_i}) \text{ et } C(m(g)) = \sum_{i=1}^{I} c(m_{p_i})$$

Le choix du modèle attribué à chaque entité est fait en se servant des fonctions de consigne et de coût définies ci-dessus de manière à minimiser le coût total de la représentation.

En fonction de ses exigences, l'utilisateur va pouvoir contrôler la représentation en fixant une valeur globale K_0. Une contrainte globale est ainsi créée : la somme des valeurs prises par la fonction k pour chaque modèle m_{p_i} représentant l'entité v_i devra être égale ou quasiment égale à K_0 :

$$K(m(g)) = K_0$$

Dans le cas de la compression d'information, K_0 est généralement exprimée en fonction d'un ratio τ (appelé taux de compression) par rapport à la quantité d'information en entrée K_{init} :

$$\tau = \frac{K_{init}}{K_0}$$

L'algorithme devra trouver une représentation qui minimise l'ensemble des coûts et dont la somme des tailles des paramètres des modèles devra être égale à K_0. Le problème revient donc à calculer la représentation qui minimise une fonction de coût global :

$$\min_{\{p_1, p_2, \ldots, p_I\} \in \mathcal{M}^I} \sum_{i=1}^{I} c(v_i, m_{p_i})$$

sous la contrainte

$$\sum_{i=1}^{I} k(v_i, m_{p_i}) = K_0$$

Une solution à ce problème nous donnera un ensemble de couple (v_i, m_{p_i}). L'ensemble des m_{p_i} formant la représentation voulue à l'échelle s de la plante.

$$S = \{p_1^*, p_2^*, ..., p_I^*\} = \underset{\{p_1, p_2, ..., p_I\} \in \mathcal{M}^I}{\arg\min} \sum_{i=1}^{I} c(v_i, m_{p_i})$$

6.1.3 Les modèles

Nous nous sommes plus particulièrement intéressés dans ce chapitre à la représentation de structures linéiques tels que les axes. Notre approche s'appuie sur les cylindres généralisés [Bloomenthal, 1995] et un algorithme d'approximation. A partir d'une représentation détaillée à l'échelle des entre-nœuds, cet algorithme est utilisé pour calculer une représentation simplifiée à l'échelle des axes. Chaque série d'entre-nœuds formant un axe est utilisée pour calculer un cylindre généralisé de complexité et de qualité plus ou moins grande.

Les cylindres généralisés

Les cylindres généralisés (extrusions) sont définis par un axe principal C (ou squelette, une courbe géométrique 3D), une section S (une courbe 2D), et éventuellement une suite de transformations T définies sur un intervalle (souvent normalisé : $[0, 1]$) [Bloomenthal, 1995]. Ce type de modèle fait partie des modèles construits par balayage. La représentation est en effet construite en plaçant et orientant la section S suivant le squelette C (et éventuellement en appliquant la transformation correspondante de T) (voir section 3.2.2). Ce modèle est particulièrement utile pour représenter globalement la géométrie des branches [Bloomenthal, 1985].

Le squelette de ce modèle peut être une courbe géométrique 3D quelconque. Nous considérons 3 types de courbes : les courbes polygonales (Polyline), les courbes de Bézier et les B-Splines. Les *courbes polygonales* sont définies par une suite de points 3D reliés entre eux. D'autres types de courbes auraient pu être considérés (comme par exemple les Cardinal Splines).

Les courbes de Bézier

Une *courbe de Bézier* de degré n est définie par

$$C(u) = \sum_{i=0}^{n} B_{i,n}(u) P_i \text{ avec } 0 \leq u \leq 1$$

où les $B_{i,n}(u)$ sont *les polynômes de Bernstein de degré* n donnés par

$$B_{i,n}(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i}$$

et les coefficients géométriques $\{P_i\}$ appelés *les points de contrôle*. Dans ce modèle, tous les points de contrôle contribuent à la position de tous les points de la courbe.

Les B-Splines

Pour obtenir une influence plus locale des points de contrôle, les *B-Splines* ont été définies de la manière suivante

$$C(u) = \sum_{i=0}^{n} N_{i,p}(u) P_i \text{ avec } a \leq u \leq b$$

avec les $\{P_i\}$ des points de contrôle et les $\{N_{i,p}(u)\}$ des fonctions de base B-Spline de degré p définies sur un vecteur de nœuds non-périodique

$$U = \{ \underbrace{a, ..., a}_{p+1}, u_{p+1}, ..., u_{n-p-1}, \underbrace{b, ..., b}_{p+1} \}$$

où a a généralement pour valeur 0 et b 1. La $i$ème fonction de base d'une B-Spline de degré p, notée $N_{i,p}(u)$ est définie par récurrence de la manière suivante :

$$N_{i,0}(u) = \begin{cases} 1 & si\, u_i \leq u < u_{i+1} \\ 0 & sinon \end{cases}$$
$$N_{i,p}(u) = \frac{u-u_i}{u_{i+p}-u_i} N_{i,p-1}(u) + \frac{u_{i+p+1}-u}{u_{i+p+1}-u_{i+1}} N_{i+1,p-1}(u)$$

Il est à noter que les B-Splines sont en fait une généralisation des courbes de Bézier. Une courbe de Bézier avec un degré n et $n+1$ point de contrôle correspond à une courbe B-Spline avec les mêmes paramètres et un vecteur de nœuds uniforme.

Complexité des différents modèles

La complexité de chacun de ces modèles, que nous définissons comme la quantité d'information nécessaire pour définir leur paramètre, est résumée dans le tableau suivant :

Nom du Modèle	Taille des paramètres
Ligne polygonale	$3n$ Floats
Courbe de Bézier	$3(deg+1)$ Floats + 1 Int
Courbe B-Spline	$3n$ Floats + 1 Int + $(n+deg)$ Floats

avec n le nombre de points (de contrôle pour les courbes de Bézier et les B-Splines) et deg le degré.

Algorithme d'inférence

La géométrie des entre-nœuds donne une suite de points (points base et sommet de chaque tronc de cône) et de sections (que l'on peut considérer de même forme circulaire avec des dimensions différentes) comme le montre la Figure 6-1.a. La suite de points forme une courbe polygonale qui représente l'axe principal du modèle. Afin de lisser les données et d'obtenir une représentation plus continue, les points peuvent être ajustés par une courbe B-Spline. En effet, les données peuvent contenir du bruit, notamment dans le cas de données digitalisées. L'algorithme d'ajustement que nous avons utilisé est celui proposé par Piegl et Tiller [Piegl and Tiller, 1997]. Cet algorithme calcule une *approximation* par une B-Spline de la suite de points.

Contrairement à l'interpolation, la courbe ne va pas forcement passer par les points donnés mais plutôt les approcher et essayer de récupérer la forme générale induite par ces points. Le calcul de l'approximation est contrôlé par un nombre de points de contrôle n et un degré (généralement fixé à 3). Soit $Q_0, ..., Q_p$ $(p > n)$ les p points à approcher. L'algorithme d'approximation consiste à trouver la courbe C avec n points de contrôle qui minimisent la distance e des points en entrée à la courbe (e est égale à la somme des distances des points à la courbe).

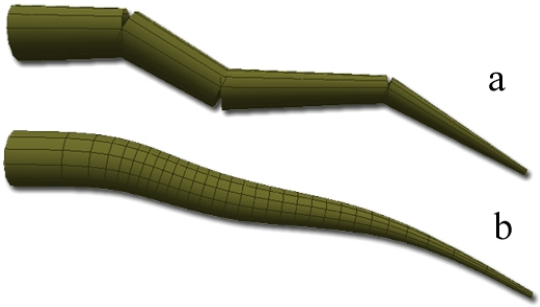


FIG. 6-1 – Calcul d'une extrusion pour représenter une branche (b) à partir d'une série de troncs de cône représentant les entre-nœuds de la branche (a). Les troncs de cône donnent, par leurs points base et sommet, une suite de points qui forme l'axe de l'extrusion. Cet axe peut être lissé en approchant la suite de points par une courbe B-Spline.

L'idée est donc de minimiser la fonction $e = \sum_{i=1}^{p} \|C(u_i) - Q_i\|$ avec $C(u_i)$ le point de la courbe C le plus proche de Q_i. Les inconnues sont les points de contrôle et les valeurs des u_i (qui permettent de calculer les $C(u_i)$). Les u_i sont estimés par la méthode de la longueur de corde :

$$u_i = \frac{\sum_{j=1}^{i} \|Q_j - Q_{j-1}\|}{\sum_{j=1}^{p} \|Q_j - Q_{j-1}\|}$$

On obtient ainsi un système linéaire d'équations bien conditionné dont les inconnues sont les coordonnées des points de contrôle P_i et qui peut être résolu numériquement.

Une courbe de Bézier peut être vue comme un cas particulier des B-Splines dans lequel le degré est égal au nombre de points de contrôle +1. Elle peut donc être ajustée à une série de points de la même manière qu'une B-Spline mais en fixant le nombre de points de contrôle en fonction du degré. Les Polylines peuvent être ajustées à une série de points en ajustant une B-Spline et en la discretisant en un nombre défini de points.

Nous disposons donc de plusieurs algorithmes d'inférence pour ajuster différents modèles de complexités et de précisions différentes à la géométrie d'une branche.

6.2 Calcul optimal : Modélisation par un problème de flot

Nous présentons maintenant une méthode générale de résolution du problème du calcul d'une représentation compressée de plante. Cette méthode nous permet de trouver une solution optimale mais coûteuse en temps de calcul pour une complexité globale K_0 donnée.

Pour obtenir une approche générique, nous avons rapproché ce problème d'un problème global d'appariement. Nous voulons attribuer à chaque entité v_i un modèle géométrique m_p en minimisant le coût $c(v_i, m_p)$. Pour chaque famille de modèles, nous sommes capables de trouver un ou plusieurs modèles particuliers représentant chaque v_i. Ils n'ont pas la même valeur pour les fonctions $c(m_p, v_i)$ et $k(m_p)$. Pour ce problème d'appariement, nous ne pouvons pas considérer tous les modèles $m_p \in \mathcal{M}$ possibles pour toutes les entités v_i. Le nombre de modèles serait évidemment infini. Pour cela, nous considérons des classes d'équivalence de modèles par rapport à la fonction k. Pour une classe k_0 et une entité v_i, nous supposons que les algorithmes d'inférences sont capables de donner le modèle m_p, tel que $k(m_p) = k_0$ et $c(m_p, v_i)$ soit minimal. Ceci limite donc le nombre de cas à considérer. Cependant le nombre de classes possibles reste toujours infini. Dans notre cas, nous pouvons borner ce nombre. En effet, dans le cas de l'application de

compression de la géométrie d'une plante, suivant le taux de compression, nous pouvons estimer une borne maximum de taille de modèle pour une entité, et donc le nombre de classes.

Le problème est modélisé par un graphe de flot avec multiplicateurs, comme illustré dans la figure 6-2 (Pour plus de détails sur les problèmes de flot, voir [Gondran and Minoux, 1990]). Nous avons un sommet source s avec un flot en entrée égal au nombre d'entité v_i. Ce sommet source est relié par des arcs de capacité 1 aux I entités v_i. Cela nous garantit qu'un flot unitaire arrive en entrée de chaque sommet v_i. Une deuxième série de sommets que nous notons $M_{l,j}$ représente les différentes classes de modèle. Nous affectons à un arc a reliant un sommet v_i et un sommet $M_{l,j}$ une capacité de 1 et un coût $c(M_{l,j}, v_i)$. Ces arcs représentent les affectations d'une classe de modèle $M_{l,j}$ à une entité v_i. A partir de ces classes et des algorithmes d'inférence, nous pouvons trouver les modèles géométriques particuliers. L'ensemble de ces modèles constituera la représentation. Tous les sommets $M_{l,j}$ sont reliés à un sommet K qui représente la fonction de contrôle globale sur les modèles attribués. L'arc qui relie K à une classe $M_{l,j}$ a une capacité infinie, ce qui signifie que chaque sous-famille peut être appariée à toutes les entités v_i, et un coefficient multiplicatif $\gamma(a)$ égal à $k(M_{l,j})$ (la taille du modèle $M_{l,j}$). Le flot en sortie d'un sommet $M_{l,j}$ indique le nombre de fois que la sous-famille est attribuée. Ce flot est multiplié par la valeur pour la fonction caractéristique de cette classe de modèle. Le flot en entrée du sommet K est donc égal à la valeur globale $k(m(g))$ de la représentation. Ce sommet K est ensuite relié au sommet puits t. L'arc qui relie ces deux sommets a une capacité K_0, ce qui garantit que la taille de la représentation sera inférieure ou égale à K_0. Le flot en sortie $f(t)$ sera égal à $k(m(g))$.

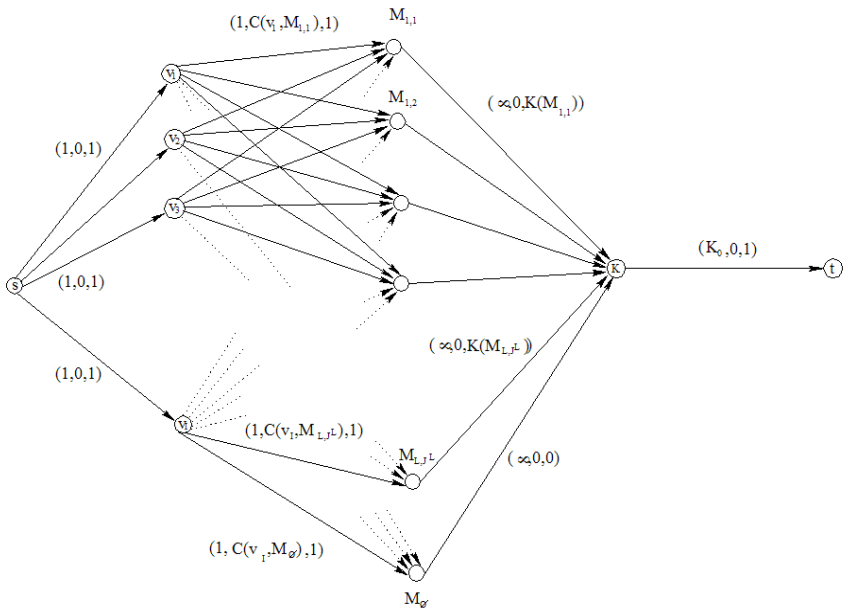


FIG. 6-2 – Graphe à flot multiplicatif. On attribue à chaque arc e un triplet qui donne les valeurs de la capacité $\kappa(e)$, du coût unitaire $c(e)$ et le coefficient multiplicatif $\gamma(e)$. Le sommet source s est relié par un arc de capacité 1 à tous les v_i (représentant les branches à représenter). Les v_i sont ensuite reliés aux différents sommets représentant les sous-familles de modèle $M_{l,j}$ avec des arcs de capacité unitaire et de coût correspondant au coût d'attribution d'un modèle $M_{l,j}$ à l'entité v_i. Finalement, les sous familles de modèles sont reliées au sommet K, par des arcs avec des coefficients multiplicatifs égaux à la taille des paramètres des $M_{l,j}$. Le flot arrivant à K donne en fait la somme des tailles des paramètres des modèles sélectionnés. Pour contraindre cette taille, le sommet K est relié au sommet puits t par un arc de capacité K_0.

Sommet dont la géométrie n'est pas représentée : Une des sous-familles de modèles utilisées est $M_\emptyset$ qui représente le modèle nul. En effet, une entité v_i peut ne pas être représentée et cela sera modélisé par la saturation de l'arc reliant v_i à $M_\emptyset$. Les coûts associés à ces arcs sont importants et difficiles à déterminer. D'une manière générale, de tous les arcs qui sortent d'une entité v_i, ce sont ceux qui doivent avoir le coût le plus important. Dans le cas inverse, on risque de favoriser la non représentation des entités et obtenir une représentation par l'élément nul de la plante. De plus, si on a $c(M_\emptyset, v_i) > c(M_\emptyset, v_j)$, on va favoriser davantage la représentation de l'entité v_i que celle de l'entité v_j. On peut se servir de cette propriété pour donner de l'importance à la représentation de certaines entités.

Le flot f^* solution de ce problème nous donne une représentation de la plante optimale pour une valeur K_0, donnée [Boudon, 2000].

Différents algorithmes existent pour résoudre un tel problème. Gondran et Minoux [Gondran and Minoux, 19 en proposent trois : l'algorithme primal proposé par Maurras en 1972, l'algorithme primal-dual proposé par Jewel en 1958 et l'algorithme dual de Jensen et Bhaumik en 1977.

La modélisation de notre problème comme un problème de flot avec multiplicateurs à coût minimum nous garantit des solutions qui minimisent la fonction de coût global. Dans la pratique cet approche a plusieurs inconvénients :

- L' utilisation de tous les coûts $c(M_{l,j}, v_i)$ pour le calcul de la solution nous oblige à calculer tous les modèles particuliers pour toutes les entités, ce qui peut prendre beaucoup de temps. Néanmoins, si ces calculs ont été faits une fois, nous pouvons trouver une représentation pour n'importe quelle valeur de K_0 seulement en trouvant un flot f^*.
- Cette approche ne prend pas en compte la connection des entités, *i.e.* une entité v_i peut être représentée sans que l'entité qui la porte le soit. Intuitivement, il nous semble cependant qu'en fixant des valeurs particulières aux $c(M_\emptyset, v_i)$, nous pouvons diriger l'algorithme vers une solution connexe.
- Cet algorithme ne garantit aucune continuité entre les différentes représentations calculées pour différents K_0. En effet, pour des valeurs successives de K_0, des branches peuvent apparaître puis disparaître.

Pour résoudre ces problèmes, nous proposons une heuristique plus rapide que nous présentons à la section suivante.

6.3 Une heuristique : ordonnancement des branches

Notre heuristique se compose principalement de deux procédures. Une première consiste à découper la quantité d'informations disponible K_0 en k_i tel que $\sum_{i=1}^{I} k_i = K_0$. Cette procédure va distribuer séquentiellement les I quantités k_i aux entités v_i de la plante à l'échelle s. La seconde procédure ajuste à partir des familles de modèles étudiées un modèle particulier m_{p_i} à chaque v_i tel que $K(m_{p_i}) \leq k_i$.

6.3.1 La sélection de modèle

L'hypothèse sur laquelle repose la procédure est la suivante : plus la taille mémoire k_i attribuée à une entité est grande, plus le modèle ajusté peut être complexe augmentant les chances de trouver une valeur de paramètres qui corresponde à une approximation de bonne qualité. Nous avons exprimé un ordre de préférence sur les familles de modèles à sélectionner. Ainsi nous préférons utiliser des B-Splines, puis des courbes de Bézier et enfin les lignes polygonales. Ces préférences tiennent compte d'abord des caractéristiques géométriques des modèles (continuité, ...), et de leur flexibilité pour s'ajuster à une série de points. En effet, les B-Splines, de part leurs propriétés géométriques, permettent un ajustement de meilleure qualité pour moins de paramètres, qu'une Polyline, sur des séries de points de grande taille.

Pour chaque entité v_i, les différentes familles de modèles vont être examinées séquentiellement par ordre de préférence, et s'il y en a une qui est compatible avec la valeur de k_i, nous l'inférons sur les données en entrée. Deux modèles peuvent être équivalents s'ils représentent les mêmes points de l'espace. Si le modèle obtenu est équivalent à un modèle appartenant à une autre famille qui a un coût inférieur, nous sélectionnons ce modèle équivalent. Par exemple, une B-Splines de degré deg avec $deg + 1$ points de contrôle est équivalente à une courbe de Bézier de même degré et nombre de points de contrôle mais moins coûteuse en espace mémoire. Dans ce cas, la courbe de Bézier sera préférée.

Dans notre implémentation, l'algorithme qui infère une B-Spline avec k_i octets de paramètres sur une série de points, commence par inférer une B-Spline de complexité inférieure et augmente progressivement la complexité en comparant à chaque fois le gain en qualité d'ajustement obtenu. Si ce gain est nul ou très faible, l'algorithme s'arrête. On évite ainsi d'utiliser des ressources pour des gains trop faibles. Cette procédure coûte malheureusement cher en temps de calcul puisque un grand nombre de B-Splines peuvent être ajustées pour une même série de points. Sur nos résultats, la plus grande partie du temps de calcul est dépensée dans cette procédure.

L'algorithme (Algo 1) pour représenter l'entité v_i avec une taille mémoire k_i est de la forme suivante :

Algorithme 1 Algorithme pour choisir un modèle à l'entité v_i avec une taille mémoire k_i

```
selectModel(ki,inputs){
  /// Une B-Spline de degré 3 a besoin au minimum de 4 points de contrôle et un
vecteur
  /// de 8 noeuds, soit environ 80 octets ( cela dépend du compilateur), pour
être définie.
 /// On vérifie donc d'abord que ki est suffisament important pour permettre
 /// l'inférence de ce modèle.
   Si ( compatibilitéBSpline(ki) ){
     Trouver la B-Spline avec ki octets de paramètres max ;
     Si la B-Spline a (deg+1) points de contrôle
          Transformer B-Spline en courbe de Bézier ;
   }
   Sinon{
    /// Une courbe de Bézier de degré 3 a besoin de 4 points de contrôle,
    ///  soit environ 52 octets, pour être définie. On vérifie donc d'abord que
ki est
   /// suffisament important pour permettre l'inférence de ce modèle.
     Si ( compatibilitéBezier(ki) )
       Trouver la courbe de Bézier avec ki octets de paramètres max ;
     Sinon{
        /// Il faut au moins 2 points pour définir une ligne polygonale, soit
environ 24 octets.
       Si ( compatibilitéPolyline(ki) )
           Trouver la ligne polygonale avec ki octets de paramètres ;
      /* Sinon l'entité v_i ne sera pas représentée */
     }
   }
}
```

6.3.2 La procédure de distribution de la ressource K_0

Nous commençons par ordonner totalement les entités de la plante à l'échelle s. Toutes les entités ne seront pas forcément représentées. L'ordre des entités doit refléter l'importance qu'il y a à les représenter. La dernière entité sera celle que l'on veut le plus représenter et inversement, la première sera celle qui nous paraît la moins importante à représenter.

L'importance visuelle et botanique d'une branche est souvent reflétée par son ordre (au sens botanique). Nous pouvons donc nous servir de l'ordre partiel topologique, c'est-à-dire celui induit par l'arborescence représentant la structure topologique à l'échelle s, pour déterminer un ordre total. Cela implique, par exemple, de mettre le tronc en dernier, puis les branches portées par le tronc (dites d'ordre 1 dans l'ordre topologique), puis celles portées par les branches d'ordre 1 (dites d'ordre 2), *etc.* L'intérêt est d'obtenir une représentation *connexe* c'est à dire dont l'ensemble des modèles forme un objet géométriquement connexe (Si une branche est représentée, alors la branche qui la porte l'est aussi).

Pour obtenir un ordre total, les branches de même ordre (botanique) sont classées suivant leur importance visuelle. Cette importance peut être évaluée par différents critères. Nous avons choisi d'utiliser simplement le nombre d'entre-nœuds qui composent la branche. D'autres critères faisant intervenir, par exemple, la longueur ou le diamètre peuvent être utilisés.

Cet algorithme examine séquentiellement les différentes entités en deux passes et distribue à chacune une partie de la quantité K_0. Il attribue à chaque entité v_i une première valeur k_i :

$$k_i = \frac{K_{0,i} * K'_i}{\sum_{j=i}^{I} K'_j}$$

avec $K_{0,i}$ la taille mémoire qui n'est pas encore distribuée après avoir examiné les $(i-1)$ premières entités, et K'_i la taille des paramètres des données en entrée pour l'entité v_i.

Avec cette taille mémoire, l'algorithme sélectionne puis infère un modèle géométrique avec la procédure `selectModel`. Il est possible que le modèle calculé n'utilise pas toute la taille mémoire. k_i prend alors comme valeur la taille des paramètres du modèle choisi. puis $K_{0,i+1} = K_{0,i} - k_i$. La différence entre la valeur initiale et la valeur finale de k_i va en fait être gardée comme taille mémoire à distribuer aux entités suivantes. Cela explique l'importance de l'ordre dans lequel la procédure examine les entités. Les dernières entités utilisent les ressources laissées par les premières entités.

Les tests ont montré qu'en général deux passes suffisent à utiliser quasiment toute la taille mémoire K_0.

6.4 Résultats

Nous avons testé notre algorithme sur un noyer digitalisé par Hervé Sinoquet *et al.* [Sinoquet et al., 1997] (Figure 6-4.a) à différents taux de compression. Nous obtenons les résultats présentés dans la Figure 6-4. Ce noyer est composé de 8131 points répartis en 1875 branches. L'histogramme du nombre de branches en fonction du nombre de segments est donné dans la Figure 6-3. Une majorité de branches avec peu de segments et quelques branches longues (jusqu'a 34 segments) composent cet arbre.

La Figure 6-4.b est une compression à 50% du noyer. L'algorithme a principalement simplifié la géométrie des grands axes et éliminé quelques petites branches. Le résultat reste néanmoins assez proche de la figure d'origine. La Figure 6-4.c présente une compression à 80%. La suppression des petits axes commence à être perceptible. Les axes plus importants sont légèrement moins sinueux. Les trois dernières figures présentent des compressions plus fortes. A 95%, la majorité des axes courts ont disparu. A 99%, seuls les axes les plus importants restent. Et finalement à 99.7%, les huit derniers axes les plus importants sont représentés, de manière assez simpliste.

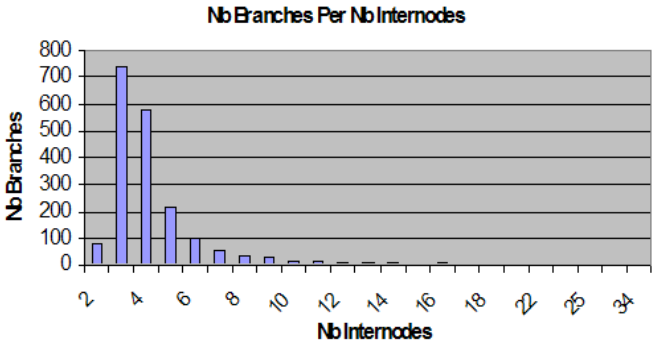


FIG. 6-3 – Composition du noyer : Le nombre de branches par nombre de segments. Le noyer est principalement composé de branches courtes de trois ou quatre segments.

Ces représentations donnent quand même l'idée principale de la géométrie de ce noyer, malgré des taux de compression importants.

Nous ne considérons pas, dans le calcul des représentations macroscopiques, les contraintes intra-échelle induites par la topologie. De ce fait, les branches, lorsqu'elles sont représentées par des modèles simples, ne sont plus obligatoirement connectées. En effet une approximation un peu forte faite par une B-Spline ne passera pas par tous les points de branchement des branches latérales. Une procédure simple pour satisfaire ces contraintes a été implémentée et est appelée après que tous les modèles macroscopiques aient été choisis. Le premier point de chacune des branches est modifié pour correspondre à sa projection sur la branche porteuse.

La figure 6-5 donne les taux de compression réellement obtenus par rapport aux taux de compression demandés. Pour les taux les plus faibles, une compression plus importante est obtenue. Les courbes B-Splines ont tendance à compresser les représentations polygonales données par les segments sans diminuer la qualité de la représentation.

Ces tests ont été effectués sur un PC avec un processeur Pentium4 2GHz. Les temps pour calculer des représentations compressées du noyer vont de 20 à 0.01 secondes (Figure 6-6). La majeure partie du temps de calcul est prise par l'algorithme d'approximation qui choisi la B-Spline ayant le meilleur rapport nombre de paramètres, qualité de l'approximation. La procédure de distribution des ressources prend moins d'une seconde à chaque fois. Une amélioration simple, pour diminuer les temps de calcul, consisterait donc à diminuer le nombre de B-Splines calculées.

6.5 Conclusion

Nous avons utilisé, dans ce chapitre, une approche montante pour calculer des représentations compressées de systèmes ramifiés. Ces représentations sont calculées à l'échelle des axes avec des cylindres généralisés plus ou moins complexes. Nous avons formalisé le problème du contrôle global de la simplification de la géométrie des systèmes ramifiés comme un problème d'appariement sous contrainte. Ce problème peut être modélisé comme un problème de flot avec multiplicateurs. Différents algorithmes donnant une solution optimale existent. Cependant, ces algorithmes nécessitent l'évaluation d'un grand nombre de modèles pour chaque entité à représenter, ce qui dans notre cas peut coûter cher en temps de calcul. De plus, ces solutions ne sont pas continues lorsqu'on augmente progressivement le taux de compression.

Pour palier à ces problèmes, nous proposons une heuristique qui trouve bonne solution en temps linéaire. Cette heuristique utilise un ordonnancement des branches pour distribuer au mieux la complexité globale de la représentation aux différents modèles de branches. Les résultats

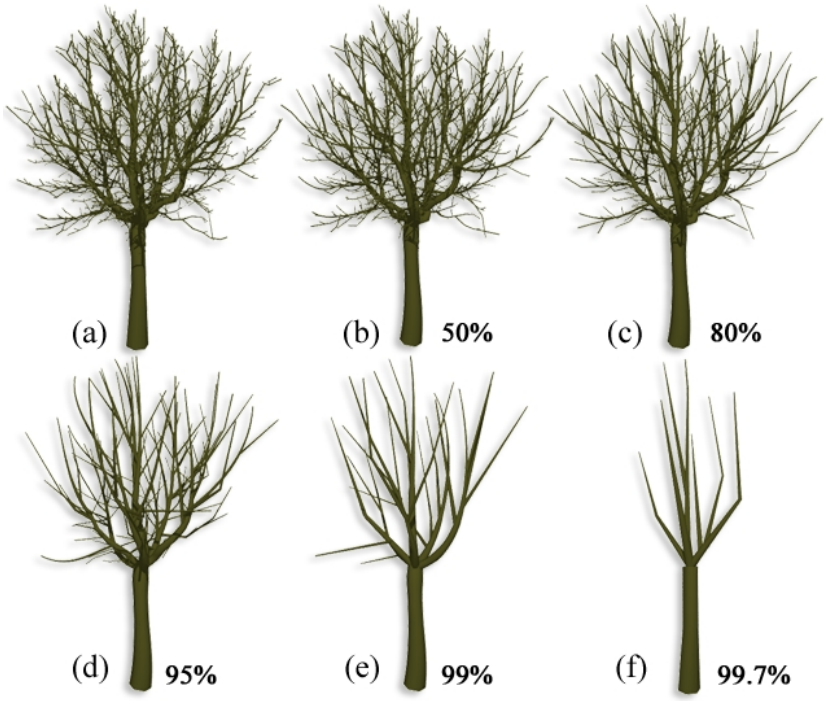


FIG. 6-4 – Une représentation détaillée d'un noyer digitalisé (les paramètres des modèles géométriques occupent environ 157.5Ko en mémoire, soit 2.2Mo d'informations géométriques passées à OpenGL) compressée à différents taux pour finalement obtenir une représentation très symbolique compressée à 99.7% dont les paramètres n'occupent que 4,7 Ko (correspondant à 111 Ko d'informations pour OpenGL)

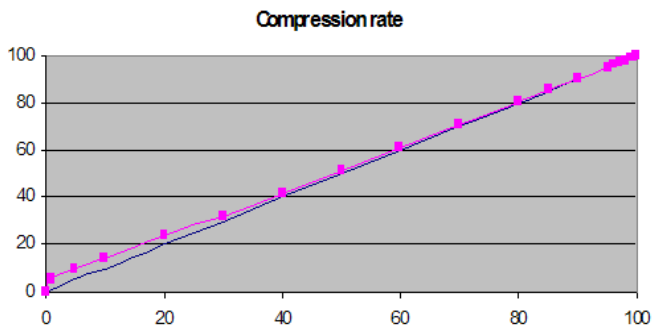


FIG. 6-5 – Taux de compression réellement obtenu (courbe violette) en fonction du taux de compression demandé (courbe bleue). Pour les taux faibles de compression ($< 30\%$), la géométrie est toujours un peu plus compressée car l'utilisation des courbes B-Splines permet de diminuer la quantité de paramétres utilisés sans réellement amoindrir la qualité de la représentation.

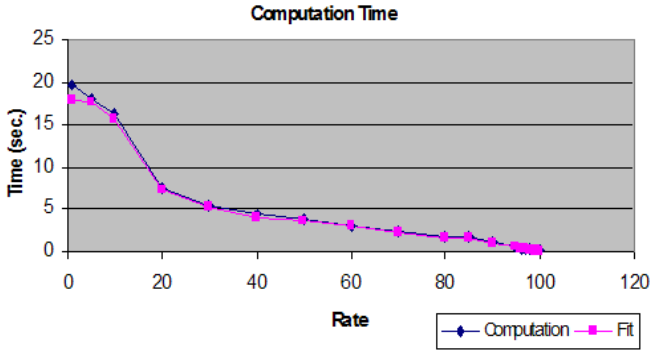


FIG. 6-6 – Temps de calcul pour des taux de compression allant de 1% à 99.99%. En bleu, le temps de calcul total. En violet, le temps de calcul pour ajuster des courbes B-Splines et simplifier les axes des extrusions. La différence des deux courbes est le temps utilisé par l'algorithme pour distribuer la ressource K_0 à chacune des branches.

obtenus sur un modèle de noyer semblent intéressants puisque, même en cas de compressions importantes, les représentations calculées donnent toujours de manière symbolique l'impression générale de la géométrie de la plante. La qualité visuelle de nos résultats mérite cependant d'être étudiée plus attentivement, notamment par une évaluation qualitative.

Troisième partie

Approche descendante : Design interactif de végétaux

Les Olives Noires. Guibert & Sfar. Dupuis.

Chapitre 7

L-systèmes et design interactif

Cette partie présente notre travail sur un modèle de génération multi-échelles de plantes construit à partir des L-systèmes. Une des propriétés de cette approche multi-échelles est de permettre la spécification à différentes échelles des informations utilisées pour générer un modèle. Ceci nous a amené à développer un système interactif qui permet de manipuler à différent niveaux de détail un modèle de plante.

Nous commençons cette partie par une présentation générale des L-systèmes et des différents concepts existants sur la manipulation globale et interactive des modèles générés par ce système. Ces concepts serviront de point de départ à notre propre travail. Nous présentons dans le chapitre suivant un système graphique de design de plante, XFrog, qui est confronté aux différents problèmes inhérents à la manipulation interactive de modèles complexes de plantes. Finalement au dernier chapitre nous présentons notre extension multi-échelles des L-systèmes pour le design interactif de modèles de plantes.

7.1 Les principes généraux des L-systèmes

Les systèmes de Lindenmayer (ou L-systèmes) sont un formalisme mathématique introduit par Aristid Lindenmayer en 1968 [Lindenmayer, 1968] pour modéliser le développement d'organismes multi-cellulaires. Leur capacité à simuler des systèmes dont la structure évolue dans le temps, appelés aujourd'hui systèmes dynamiques à structure dynamique [Giavitto and Michel, 2001], les rendent particulièrement bien adaptés à la modélisation de la croissance des plantes.

Ce système de simulation est basé sur un ensemble de règles de réécriture qui s'appliquent sur une chaîne de lettres représentant les composants d'une *arborescence axiale* (section 2.1.2). Un représentation tridimensionnelle de cette chaîne est possible en l'interprétant géométriquement via une tortue de type LOGO (voir section 3.1.5). Les L-systèmes offrent de nombreuses possibilités pour la synthèse d'images.

7.1.1 Les D0L-systèmes

La forme la plus simple de L-systèmes, introduite par Lindenmayer [Lindenmayer, 1968], correspond aux D0L-systèmes (**D**éterministes, sans contexte (**0**), **L**-systèmes) :

Définition 18
- *Un* alphabet *est un ensemble fini de* lettres *noté V. Les lettres sont aussi appelées* modules.
- *Un* mot *est une séquence de lettres sur un alphabet. L'ensemble de tous les mots sur l'alphabet V est noté V^*.*
- *Une* production *est une paire (a, u) notée $a \rightarrow u$, où a est une lettre et u est un mot, a est appelé prédécesseur et u est appelé successeur.*

- *Un D0L-système est un triplet $G = \langle V, \omega, P \rangle$, où V est un alphabet, $\omega \in V^*$ est un mot appelé* l'axiome, *et P est l'ensemble des productions tel que $\forall a \in V : \exists! p_a \in P$, où p_a représente une production qui a un module a comme prédécesseur.*

On dit que la production $a \rightarrow u$ s'applique au module a. Par convention, on suppose que si aucune production n'est spécifiée pour le module a explicitement alors la production *Identité* ($a \rightarrow a$) est ajoutée implicitement. Sous cette hypothèse, un 0L-système $G = \langle V, \omega, P \rangle$ est dit *déterministe* (D0L-système) si et seulement si pour chaque module $a \in V$ l'ensemble des productions inclue exactement une seule production applicable à ce module. Une production peut aussi spécifier qu'un module doit être supprimé de la chaîne. On écrit $a \rightarrow \varepsilon$ où ε est le symbole "vide".

Le processus d'application des productions et de création de nouvelles chaînes est appelé *réécriture de chaîne*. L'exécution d'un L-système consiste en une série de réécritures de chaînes, qui sont alors appelées *dérivations*. Dans la phase de réécriture des L-systèmes, les productions sont appliquées en parallèle à tous les modules de la chaîne. C'est ce qui les distingue des grammaires plus classiques de type Chomsky.

Si un module a produit un mot χ quand on lui applique les productions d'un L-système G, on écrit $a \mapsto \chi$. Soit un mot $w = a_1 a_2 ... a_m$, on dit que le mot $v = \chi_1 \chi_2 \cdots \chi_m$ est directement dérivé de w et on écrit $w \Rightarrow v$ si et seulement si $a_j \mapsto \chi_j$ pour tout $j = 1, 2, ..., m$. Un mot v est généré par G dans une dérivation de longueur n s'il existe une séquence de mots $w_0, w_1, ..., w_n$ tel que $w_0 = \omega$, $w_n = v$ et $w_0 \Rightarrow w_1 \Rightarrow ... \Rightarrow w_n$.

Un D0L-système G produit une séquence de développement infinie unique de chaînes $S(G) = \{w_0, w_1, w_2, ...\}$. L'ensemble de tous les mots dans la séquence de développement d'un D0L-système définit le langage $L(G)$ engendré par le système.

Exemple 19 *Un exemple simple de D0L-système avec pour axiome ω et deux productions étiquetées p_1 et p_2 [Mech, 1997] est présenté ici.*

L-système			*Séquence de développement*
ω	:	A	A
p_1	:	$A \rightarrow AB$	AB
p_2	:	$B \rightarrow BB$	$ABBB$
			$ABBBBBBB$

Les quatre premières chaînes de la séquence de développement S générées par ce L-système sont présentées sur la droite. La n-ième chaîne de la séquence de développement S est de la forme AB^{2^n-1}, où B^k représente un mot de k modules B. Ainsi le langage L engendré par ce L-système est :

$$L = \{AB^{2^n-1} \mid n \geq 0\}$$

Les chaînes ainsi obtenues permettent de représenter des structures linéaires.

7.1.2 Les D0L-systèmes parenthésés

Pour étendre l'utilisation des L-systèmes aux structures arborescentes, en utilisant les arborescences axiales, deux modules spécifiques ont été introduits dans la définition originale des L-systèmes [Lindenmayer, 1968]. Ces modules sont le crochet gauche ([ ) et le crochet droit ( ]) ; ils spécifient respectivement le début et la fin d'une branche. La définition des L-systèmes parenthésés est généralement étendue par deux conditions supplémentaires [Prusinkiewicz and Kari, 1996] :

- Les crochets sont implicitement remplacés par eux-mêmes ([$\rightarrow$ [et] $\rightarrow$]).
- La partie droite de chaque production est une chaîne parenthésée bien formée (*i.e.* pour chaque crochet gauche [, il existe un crochet droit] correspondant et le nombre de crochets gauches et droits entre chaque paire est le même).

Cette extension des L-systèmes permet la réécriture de chaîne de caractères représentant des arborescences axiales.

7.1.3 Les 0L-systèmes paramétriques

Les 0L-systèmes, décrits dans les sections précédentes, peuvent représenter des informations qualitatives : par exemple les différents types de modules représentent les différents types de composants du modèle, tels que des cellules ou des organes. Des informations quantitatives (tel que la longueur des entre-nœuds ou la valeur des angles de rotations) peuvent aussi être représentées par le formalisme 0L en utilisant plusieurs modules pour exprimer les différentes longueurs. Cependant il est impossible d'exprimer une forme aussi simple qu'un triangle isocèle droit dont les longueurs d'arêtes n'ont pas de dénominateur commun. Ce problème a été résolu en introduisant les L-systèmes paramétriques [Prusinkiewicz and Lindenmayer, 1990, Hanan, 1992].

L'essence des L-systèmes paramétriques est dans l'association de paramètres aux différents modules de la chaîne, appelés alors *lettres paramétriques*. Chaque module consiste en un symbole appartenant à un alphabet V et des paramètres numériques appartenant à l'ensemble des réels $\mathbb{R}$. Un module avec une lettre $A \in V$ et les paramètres $a_1, a_2, ..., a_n \in \mathbb{R}$ sera noté $A(a_1, a_2, ..., a_n)$. Chaque module appartient à l'ensemble $V \times \mathbb{R}^*$, ou $\mathbb{R}^*$ est l'ensemble de tous les séquences finies de paramètres.

Les paramètres réels avec des valeurs définies qui apparaissent dans les mots peuvent être référencés par des paramètres formels, qui sont alors utilisés dans la spécification des règles de productions des L-systèmes. Les paramètres formels sont utilisés dans des expressions logiques et arithmétiques, pour exprimer des conditions d'application ou calculer de nouvelles valeurs aux paramètres des successeurs dans une production. Si Σ est l'ensemble des paramètres formels, alors $C(\Sigma)$ représente une expression logique avec les paramètres de Σ, et $E(\Sigma)$ est une expression arithmétique avec les paramètres du même ensemble. Les ensembles de toutes les expressions logiques et arithmétiques correctement construites avec les paramètres de l'ensemble Σ sont notés $\mathcal{C}(\Sigma)$ et $\mathcal{E}(\Sigma)$

Définition 20 *Un D0L-système paramétrique est un quadruplet ordonné $G = \langle V, \Sigma, \omega, P \rangle$, avec*
- *V l'alphabet du système,*
- *Σ l'ensemble des paramètres formels,*
- *l'axiome $\omega \in (V \times \mathbb{R}^*)^+$ un mot paramétrique non vide, et*
- *P, un ensemble fini de productions, est un ensemble de triplets (pred : cond $\rightarrow$ succ) tel que pred $\in V \times \Sigma^*$, cond $\in \mathcal{C}(\Sigma)$, succ $\in (V \times \mathcal{E}(\Sigma)^*)^*$. Les trois composants des productions sont appelés respectivement* prédécesseur, condition *et* successeur.

Par exemple, une production avec un prédécesseur $A(t)$, une condition $t > 5$, et un successeur $B(t+1)CD(t\hat{}0.5, t-2)$ s'écrit de la manière suivante :

$$A(t) : t > 5 \rightarrow B(t+1)CD(t\hat{}0.5, t-2)$$

Une production pour être appliquée sur un module avec des paramètres est sélectionnée de la manière suivante :
- la lettre du module et du prédécesseur doivent être les mêmes,
- le nombre de paramètres réels du module doit correspondre au nombre de paramètres formels du prédécesseur,
- la condition doit être évaluée à *vrai*, après que les paramètres réels du modules soient substitués aux paramètres formels de la production.

Si une production peut être appliquée sur un module, elle créera un chaîne de modules spécifiée par le successeur de la production. Les valeurs des paramètres réels remplacent les paramètres formels en fonction de leurs positions. Par exemple, la production de l'exemple

précédent s'applique sur le module $A(9)$, puisque la lettre A dans le module est la même que dans le prédécesseur de la production, il y a un seul paramètre réel dans le module $A(9)$ ainsi qu'un seul paramètre formel dans le prédécesseur $A(t)$, et l'expression logique $t > 5$ est vraie pour t égal à 9. Le résultat de l'application de la production, dans ce cas, sera le mot paramétrique $B(10)CD(3,7)$.

7.1.4 Le symbole Coupe

Les L-systèmes considérés jusqu'à présent étaient de type propagatif. Chaque module, une fois créé, continue d'exister indéfiniment ou engendre un ou plusieurs enfants, mais ne disparaissent jamais sans laisser de trace. Le processus naturel du développement d'une plante implique pourtant la disparition de certains de ses composants. L'approche présentée jusqu'à présent pour simuler la suppression d'un module passe par l'utilisation de productions de la forme $A \rightsquigarrow \varepsilon$. Modéliser la situation où une structure entière est perdue par la plante par cette méthode est cependant beaucoup plus difficile.

La plante peut contrôler ce processus par *élagage* avec parfois la mise en place d'une zone d'abscission (le développement d'une couche de cellules qui affaiblissent la branche à sa base). Il paraît évident que l'élagage est représenté plus fidèlement en coupant une branche qu'en simulant la suppression de tous ces modules. Pour simuler la perte d'organe, Hanan [Hanan, 1992] s'inspire de ce phénomène d'élagage et étend le formalisme des L-systèmes avec le "*symbole Coupe*" %, qui cause la suppression de tout le reste de la branche qui le suit. Par exemple, en l'absence d'autres productions, le pas de dérivation suivant prendra place :

$$a[b\%[cd]e[\%f]]g[h[\%i]j]k \Rightarrow a[b]g[h[]j]k$$

Exemple 21 *Un exemple simple de L-système incorporant le symbole Coupe est donné ici :*

$$\begin{array}{lllll}
\omega & : & A & & \\
p_1 & : & A & & \rightarrow \quad F(1)[-X(3)B][+X(3)B]A \\
p_2 & : & B & & \rightarrow \quad F(1)B \\
p_3 & : & X(d) & : d > 0 & \rightarrow \quad X(d-1) \\
p_4 & : & X(d) & : d == 0 & \rightarrow \quad U\% \\
p_5 & : & U & & \rightarrow \quad F(0.3)
\end{array}$$

D'après la production p_1, à chaque pas de dérivation l'apex de l'axe principal A produit un entre-nœud F de longueur unitaire et une paire d'apex latéraux B. Chaque apex B étend une branche en formant une succession d'entre-nœuds F (production p_2). Après trois pas à partir de l'initiation de la branche (contrôlée par la production p_3), la production p_4 insère un symbole Coupe % et un symbole auxiliaire U à la base de la branche, alors que le symbole U insère un marqueur $F(0.3)$ indiquant une cicatrice laissée par la suppression de la branche.

La séquence de développement résultante est montrée dans la Figure 7-1. Les pas initiaux capturent la croissance d'une structure basitone (développée extensivement à la base). A partir du sixième pas de dérivation, les plus vieilles branches tombent, donnant l'impression d'une couronne d'arbre de forme et de taille constante qui s'élève. La couronne est dans un état d'équilibre dynamique : l'ajout de nouvelles branches à l'apex est compensé par la perte de branches plus bas.

7.1.5 Les 0L-systèmes stochastiques

Dans les sections précédentes, les processus de développement sont supposés être des phénomènes déterministes, modélisés par des D0L-systèmes qui produisent toujours la même séquence de développement. Cependant, des facteurs aléatoires interviennent souvent dans la construction de modèles de plantes. Il est possible d'intégrer ces phénomènes dans la théorie des L-systèmes

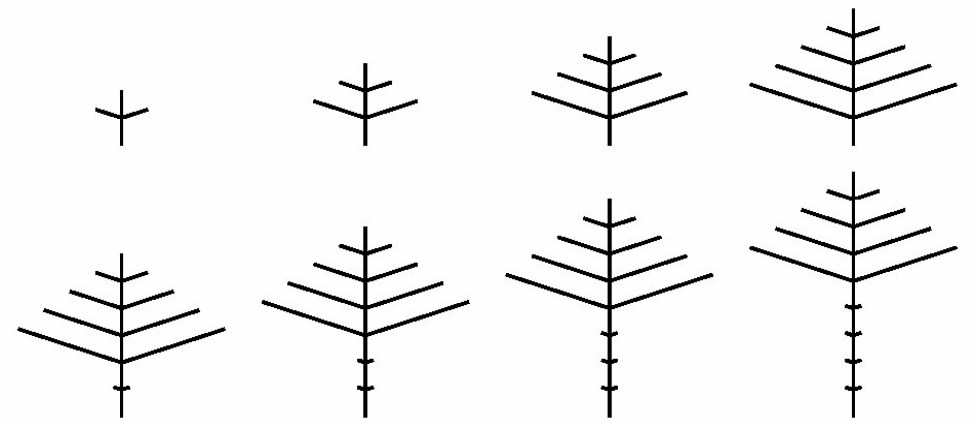


FIG. 7-1 – Une séquence de développement générée par le L-système de l'Exemple 21. Les images représentent les pas de dérivation 2 à 9. Tiré de [Prusinkiewicz et al., 1997].

en associant des probabilités d'application aux règles de production. Différentes extensions stochastiques des L-systèmes ont été proposées dans la littérature [Eichhorst and Savitch, 1980, Jürgensen, 1976, Nishida, 1980, Yomokori, 1980] et appliquées pour exprimer des modèles de développement de plantes [Nishida, 1980, Prusinkiewicz, 1987, Prusinkiewicz and Hanan, 1989]. La définition donnée ici, tirée de [Prusinkiewicz et al., 1997], intègre l'utilisation de l'aléatoire aux L-systèmes paramétriques.

Un *0L-système paramétrique stochastique* est un quintuplet ordonné :

$$G_\pi = \langle V, \Sigma, \omega, P, \pi \rangle .$$

L'alphabet V, l'ensemble de paramètres formels Σ, l'axiome ω, et l'ensemble de productions P sont définis comme pour les D0L-systèmes. La fonction $\pi : P \rightarrow \mathcal{E}(\Sigma)$, assigne une expression arithmétique qui retourne un nombre non négatif appelé le *facteur de probabilité* à chaque production dans P. Une production dans un L-système stochastique s'écrit

$$pred : cond \rightarrow succ : f,$$

avec *pred*, *cond* et *succ* jouant le même rôle que dans les D0L-systèmes, et f une expression retournant le facteur de probabilité.

Si $P_A \subseteq P$ est l'ensemble des productions pouvant s'appliquer sur un module $A(t_1, t_2, ..., t_n)$ donné de la chaîne à réécrire, alors la probabilité $prob(p_k)$ d'appliquer une production particulière $p_k \in P_A$ à ce module est égale à :

$$prob(p_k) = \frac{\pi(p_k)}{\Sigma_{p_i \in P_A} \pi(p_i)}$$

En général, cette probabilité n'est pas une constante associée avec une production, mais dépend de la valeur des paramètres du module à réécrire. On appelle la dérivation $w \Rightarrow v$ une *dérivation stochastique* dans G_π si pour chaque occurrence d'un module a dans le mot w la probabilité d'appliquer la production p_k avec pour prédécesseur a est donnée par l'équation précédente. Pour chaque occurrence d'un module a s'appareillant à plusieurs productions p_k suivant différentes probabilités, un tirage aléatoire qui tient compte de ces probabilités est effectué pour choisir la règle à appliquer. Ainsi, différentes productions avec le même prédécesseur peuvent s'appliquer à différentes occurrences du même module dans un pas de dérivation.

Exemple 22 *Un exemple de L-système stochastique est donné ci-dessous [Prusinkiewicz et al., 1997].*

$$\begin{array}{lcl} \omega & : & A(1)B(3)A(5) \\ p_1 & : & A(x) \rightarrow A(x+1) : 2 \\ p_2 & : & A(x) \rightarrow B(x-1) : 3 \\ p_3 & : & A(x) : x > 3 \rightarrow C(x) : x \\ p_4 & : & B(x) \rightarrow B(2*x)A(x) : 1 \end{array}$$

Les productions p_1, p_2 et p_3 remplacent le module $A(x)$ par $A(x+1)$, $B(x-1)$, ou $C(x)$. Si la valeur du paramètre x est inférieure ou égale à 3, seulement les deux premières productions peuvent s'appliquer sur $A(x)$. Les probabilités d'appliquer chaque production sont : $prob(p_1) = 2/(2+3) = 0.4$, et $prob(p_2) = 3/(2+3) = 0.6$. Si le paramètre x est plus grand que 3, la production p_3 peut aussi s'appliquer sur le module $A(x)$, et les probabilités d'appliquer chaque production dépendent de la valeur de x. Par exemple si x est égal à 5, ces probabilités sont $prob(p_1) = 2/(2+3+5) = 0.2$, $prob(p_2) = 3/(2+3+5) = 0.3$, et $prob(p_3) = 5/(2+3+5) = 0.5$. La production p_4 remplace un module $B(x)$ par la paire de modules $B(2*x)A(x)$. En prenant tout cela en compte, le premier pas de dérivation de ce L-système peut avoir la forme suivante :

$$A(1)B(3)A(5) \Rightarrow A(2)B(6)A(3)C(5)$$

avec la production p_1 qui s'applique sur le module $A(1)$, et la production p_3 sur le module $A(5)$ comme résultat des tirages aléatoires.

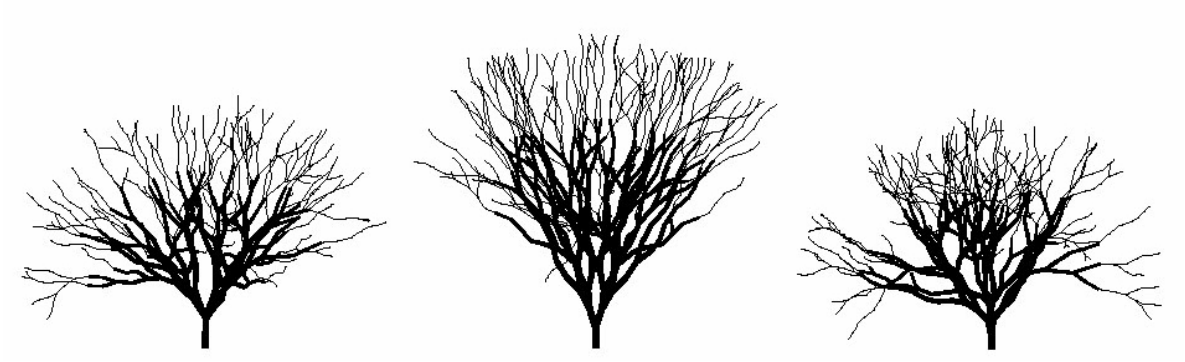


FIG. 7-2 – Exemples de structures arborescentes générées en utilisant le même L-système stochastique. Tiré de [Prusinkiewicz et al., 1994].

7.2 Utilisation du contexte

7.2.1 Les L-systèmes sensibles au contexte

Les productions dans les 0L-systèmes paramétriques ne tiennent pas compte du contexte, *i.e.* elles sont applicables indépendamment du contexte dans lequel le prédécesseur apparaît. Pour les L-systèmes sensibles au contexte, l'application des productions dépend aussi du contexte du prédécesseur. Cette extension est nécessaire pour modéliser l'échange d'information entre modules voisins, pour être capable, par exemple, de simuler le flux d'eau, d'hormones ou de nutriments à travers la plante. Plusieurs extensions des L-systèmes sensibles au contexte ont été proposées et étudiées (par exemple [Herman and Rozenberg, 1975, Lindenmayer and Rozenberg, 1976, Salomaa, 1973]). Nous présenterons ici les L-systèmes parenthésés sensibles au contexte, qui opèrent donc sur les chaînes qui représentent des structures ramifiées [Hanan, 1992, Prusinkiewicz and Hanan, 1990,

Prusinkiewicz and Lindenmayer, 1990]. Dans le cas non stochastique, les productions sont de la forme suivante :

$$cg < pred > cd : cond \rightarrow succ,$$

où les symboles $<$ et $>$ séparent les trois composants du prédécesseur : une chaîne de modules sans crochet (un chemin) *cg* appelé *contexte gauche*, un module *pred* appelé *prédécesseur strict* et une chaîne de modules correctement parenthésée (une arborescence axiale) *cd* appelé *contexte droit*. Les composants restant de la production sont la condition *cond* et le successeur *succ*, définis comme pour les D0L-systèmes.

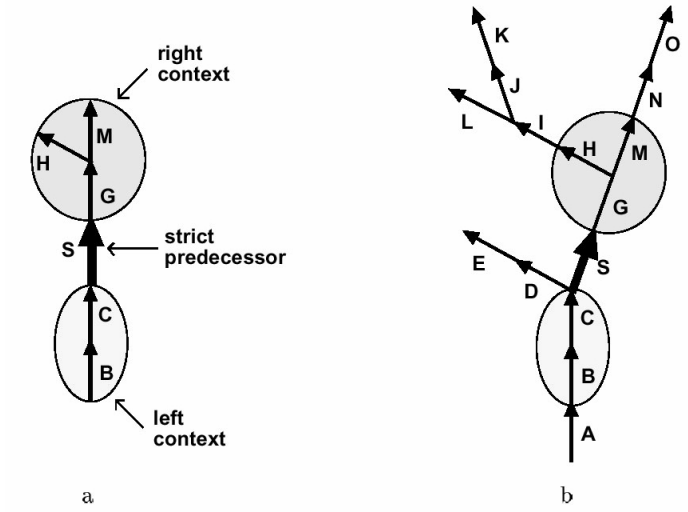


FIG. 7-3 – Le prédécesseur d'une production sensible au contexte (a) s'applique à l'arête S dans l'arborescence T (b). Tiré de [Prusinkiewicz et al., 1988].

Le nouveau mécanisme introduit par les L-systèmes sensibles au contexte est dans le processus d'appariement du prédécesseur d'une production avec un module donné de la chaîne à réécrire. Nous allons décrire ce processus sur les arborescences axiales, en supposant pour l'instant que les modules n'ont pas de paramètres. Comme illustré par la Figure 7-3, le prédécesseur strict S d'une production est situé entre un chemin spécifié par le contexte gauche, et une arborescence axiale spécifiée par le contexte droit. Si aucun paramètre n'est présent, une production p s'applique sur une occurrence donnée du segment S dans l'arborescence axiale T si les conditions suivantes sont rencontrées :

- le prédécesseur strict *pred* est le symbole S
- le contexte gauche *cg* représente un chemin dans T terminant au début de S, et
- le contexte droit *cd* représente une sous-arborescence de T commençant à la fin de S.

La production peut alors s'appliquer en remplaçant S par l'arborescence axiale spécifiée par le successeur de la production.

Quand une représentation par chaîne parenthésée est utilisée pour exprimer les productions et la structure à réécrire, le processus d'appariement des prédécesseurs aux modules dans les L-systèmes parenthésés sensibles au contexte ne peut pas se réduire à un simple appariement de chaîne, car la représentation par chaîne parenthésée des arborescences axiales ne préserve pas le voisinage des segments. En conséquence, la procédure d'appariement du contexte peut avoir à "sauter" des symboles représentant des branches ou des portions de branches. Par exemple, la Figure 7-3 indique que la production avec comme prédécesseur $BC < S > G[H]M$ peut

s'appliquer au symbole S de la chaîne

$$ABC[DE][SG[HI[JK]L]MNO],$$

ce qui implique de "sauter" les symboles $[DE]$ dans la recherche du contexte gauche, et $I[JK]L$ dans la recherche du contexte droit.

Dans le cas paramétrique, une production qui s'applique sur une occurrence du module S dans la structure à réécrire doit satisfaire des conditions supplémentaires, similaires à celles définies pour les 0L-systèmes paramétriques.

- le nombre de paramètres formels dans chaque partie du prédécesseur (*i.e.* le prédécesseur strict et les contextes) doit être le même que le nombre de paramètres réels dans les modules correspondant de l'arborescence T, et
- la condition doit être évaluée à *vrai* lorsque les paramètres formels de la production sont remplacés par les valeurs des paramètres réels.

Par exemple, la production paramétrique sensible au contexte :

$$A(x) < B(y) > C[D(z)]F : x + y + z > 10 \rightarrow U((x + y)/2)[V((y + z)/2]$$

peut s'appliquer au module $B(5)$ apparaissant dans le mot paramétrique

$$...A(4)B(5)C[D(6)E]F...$$

car les lettres et les nombres de paramètres dans tous les modules impliqués dans la comparaison coïncident, et la condition $4 + 5 + 6 > 10$ est vraie. Comme résultat de l'application de cette production, le module $B(5)$ sera remplacé par la structure ramifiée $U(4.5)[V(5.5)]$. Naturellement, les modules restants de la chaîne à réécrire peuvent être aussi remplacés (par d'autres productions) dans le même pas de dérivation.

Les productions dans les 2L-systèmes utilisent le contexte des deux cotés du prédécesseur strict. Les 1L-systèmes sont un cas spécial de 2L-systèmes dans lequel le contexte n'apparaît seulement que d'un seul côté des productions. De manière consistante avec cette convention, aucun contexte n'est considéré dans les 0L-systèmes. Ce mécanisme permet par exemple de modéliser la propagation de flux dans un modèle de plante. Le contexte gauche exprime un flux d'information *acropète* (de la base de la structure ramifiée vers les apex) tandis que le contexte droit exprime un flux *basipète* (des apex vers la racine).

7.2.2 Le transfert rapide d'information

Les L-systèmes sensibles au contexte permettent de transférer des informations entre différents modules de la chaîne. Cependant, une production, lorsqu'elle va s'appliquer, ne pourra modéliser à chaque pas de dérivation que le transfert entre deux modules voisins. Sur une structure linéaire de n modules, il faudra donc au minimum n pas de dérivation pour transférer l'information d'un bout à l'autre de la structure.

Exemple 23 *Le L-système sensible au contexte ci-dessous [Karwowski, 2002] déplace un signal S de gauche à droite d'une série de module A, en comptant le nombre de A. Chacune des dérivations étant en $O(n)$ (elles examinent tous les symboles de la chaîne), la complexité des*

algorithmes de transfert d'information utilisant cette méthode est donc en $O(n^2)$.

		L-système			*Séquence de développement*
ω	:	$S(0)AAAA$			$S(0)AAAA$
p_1	:	$S(n) < A$	$\rightarrow$	$AS(n+1)$	$AS(1)AAA$
p_2	:	$S(n) > A$	$\rightarrow$	ε	$AAS(2)AA$
					$AAAS(3)A$
					$AAAAS(4)$

Karwowski [Karwowski, 2002] propose une nouvelle méthode plus rapide pour transférer l'information à travers une structure ramifiée en utilisant la notion de *nouveau contexte*. En effet, lorsque, la chaîne $\mu_i = a_1a_2...a_i...a_n$ représentant une arborescence axiale est transformée par un L-système en $\mu_{i+1} = w_1w_2...w_j...w_n$, la réécriture se fait séquentiellement. A l'étape j de la réécriture, avant même de réécrire le module a_j de la chaîne μ_i, les w_k pour $k = 1, ..., j-1$ sont déjà instanciés. Le système connaît donc déjà le début de la chaîne μ_{i+1} soit $\mu^j_{i+1} = w_1...w_{j-1}$ qui donne le futur contexte du mot w_j. Ce contexte est appelé le nouveau contexte gauche de a_j.

On notera que, dans le cas de réécriture de la chaîne de la droite vers la gauche, c'est le nouveau contexte droit qui est accessible lors de la réécriture.

Le transfert rapide d'information propose de tenir compte de ce nouveau contexte dans l'application de la production qui réécrira a_j en w_j. Cela permet, par exemple, de réécrire la valeur des paramètres de w_j en fonction d'informations que vient juste de recevoir w_{j-1} sans attendre un nouveau pas de dérivation. L'avantage principal de cette extension est donc de pouvoir transférer des informations sur une chaîne de n éléments en un seul pas de dérivation. La complexité de cet algorithme de transfert est proportionnel à la complexité de la dérivation, soit $O(n)$.

Le formalisme des L-systèmes a donc été étendu pour prendre en compte ces nouveaux contextes. Les productions sont de la forme suivante :

$$cg < ncg << pred > cd : cond \rightarrow succ,$$

avec le contexte gauche cg, le contexte droit cd, le prédécesseur strict $pred$, la condition $cond$ et le successeur $succ$ jouant le même rôle que dans les L-systèmes sensibles au contexte et une chaîne de modules sans crochet ncg représentant le nouveau contexte gauche. Dans le cas paramétrique, pour s'appliquer sur un module a_j donné, cette production devra satisfaire les conditions suivantes :

- le prédécesseur strict $pred$ a le même symbole que a_j,
- le contexte gauche cg est une chaîne qui représente un chemin dans μ_i terminant au début de a_j,
- le nouveau contexte gauche ncg est une chaîne qui représente un chemin dans la sous arborescence $\mu^j_{i+1} = w_1...w_{j-1}$ terminant à la fin de μ^j_{i+1},
- le contexte droit cd représente une sous-arborescence de μ_i commençant à la fin de a_j,
- le nombre de paramètres formels dans chaque partie du prédécesseur (*i.e.* le prédécesseur strict et les contextes) doit être le même que le nombre de paramètres réels dans les modules correspondant des arborescences μ_i et μ^j_{i+1}, et
- la condition doit être évaluée à *vrai* lorsque les paramètres formels de la production sont remplacés par les valeurs des paramètres réels.

La production peut alors s'appliquer en remplaçant a_j par l'arborescence axiale spécifiée par le successeur de la production.

Dans le cas ou la réécriture se fait de droite à gauche, les productions seront de la forme :

$$cg < pred >> ncd > cd : cond \rightarrow succ,$$

avec une chaîne de modules sans crochet *ncd* représentant le nouveau contexte gauche. Les conditions d'application seront similaires.

Exemple 24 *L'exemple suivant [Karwowski, 2002] montre l'utilisation du transfert rapide d'information à travers une structure ramifiée. Le but étant de calculer l'ordre de chaque entre-nœuds. Chaque module I représentant un entre-nœud a un paramètre pour stocker cette information.*

$$\begin{array}{lcllcl} \omega & : & I(0)[I(0)A]I(0)A & & & \\ p_1 & : & I(m) << & I(n) & \rightarrow & I(m+1) \\ p_2 & : & & I(n) & \rightarrow & I(n+1) \end{array}$$

Le processus de réécriture en un pas de dérivation est montré dans la table suivante :

Chaîne originale	*Production appliquée*	*Chaîne résultante*
$\mathbf{I(0)}[I(0)A]I(0)A$	p_2	$\mathbf{I(1)}$
$I(0)[\mathbf{I(0)}A]I(0)A$	p_1	$I(1)[\mathbf{I(2)}$
$I(0)[I(0)\mathbf{A}]I(0)A$	*Identité*	$I(1)[I(2)\mathbf{A}$
$I(0)[I(0)A]\mathbf{I(0)}A$	p_1	$I(1)[I(2)A]\mathbf{I(2)}$
$I(0)[I(0)A]I(0)\mathbf{A}$	*Identité*	$I(1)[I(2)A]I(2)\mathbf{A}$

La production p_2 initialise l'ordre du premier entre-nœud à 1, puis grâce à la production p_1 les I sont réécrits, en fonction de leur nouveau contexte qui contient l'ordre depuis la racine de l'entre-nœud précédent.

7.3 Interaction avec l'environnement

7.3.1 Les L-systèmes sensibles à l'environnement

L'environnement est un facteur important dans le processus de développement d'une plante. Les L-systèmes présentés précédemment peuvent être considérés comme des systèmes fermés, capables de contrôler leur développement sans communiquer avec l'environnement. Les L-systèmes sensibles à l'environnement ont été développés afin de permettre la prise en compte de l'influence des facteurs environnementaux locaux sur un modèle de plante en croissance [Prusinkiewicz et al., 1994, Mech, 1997]. L'idée développée dans cette approche est de fournir à chaque module sa position et son orientation pour pouvoir construire des productions en fonction de ces informations locales.

Dans ce système, la chaîne générée est interprétée graphiquement par la tortue à chaque pas de dérivation. Les attributs de la tortue (position, orientation) trouvés pendant l'interprétation sont alors retournés comme paramètres de modules de requête prédéfinis dans la chaîne. Chaque pas de dérivation se déroule comme pour les L-systèmes paramétriques, excepté pour les valeurs des paramètres de modules de requête qui restent indéfinies. Pendant l'interprétation, le système assigne une valeur à ces modules en fonction de la position et l'orientation de la tortue dans l'espace. Syntaxiquement, les modules de requête sont de la forme $?X(x, y, z)$, où $X = P, H, U$ ou L. Suivant la valeur du symbole X, les valeurs des paramètres x, y et z représentent un vecteur de position ou d'orientation.

Exemple 25 *Un exemple de L-système sensible à l'environnement [Prusinkiewicz et al., 1994] est donné ci-dessous.*

$$\begin{array}{lcll} \omega & : & A & \\ p_1 & : & A \rightarrow & [+B][-B]F?P(x,y)A \\ p_2 & : & B \rightarrow & F?P(x,y)@OB \\ p_3 & : & ?P(x,y) & : 4x^2 + (y-10)^2 > 10^2 \\ & & \rightarrow & [+(2y)F][-(2y)F]\% \end{array}$$

Le module F représente un segment unitaire, et les modules $+$ et $-$ sans paramètre représentent une rotation gauche et droite de 60°. Le développement commence avec un module A, qui crée une séquence de branches opposées $[+B][-B]$ séparées par des entre-nœuds F (production p_1). Les branches s'allongent par addition de segments F, délimités par des marqueurs @0 (représentés par des sphères) (production p_2). L'apex principal A et les apex latéraux B créent des modules de requête $?P(x, y)$, qui retournent pour le prochain pas de dérivation les positions de la tortue correspondantes. Si un module de requête est placé au-delà de l'ellipse $4x^2 + (y - 10)^2 = 10^2$, la production p_3 crée une paire de "tentacules", représentée par la chaîne $[+(2y)F][-(2y)F]$. Les angles $4y$ dépendent de la position verticale y du module de requête. La production p_3 insère aussi le symbole Coupe %, qui termine la croissance de la branche en supprimant son apex. En résumé, ce L-système produit une structure ramifiée confinée dans une ellipse, avec des tentacules placées à la frontière de la structure.

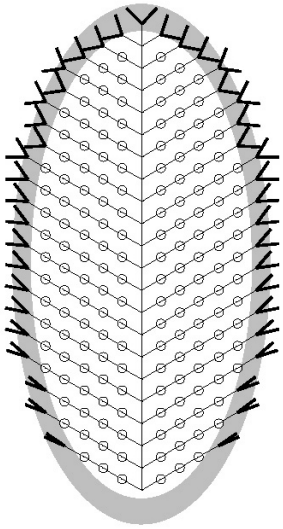


FIG. 7-4 – Une structure ramifiée élaguée sous forme d'ellipse. Tiré de [Prusinkiewicz et al., 1994].

7.3.2 Les L-systèmes ouverts

Si les L-systèmes sensibles à l'environnement permettent une communication de l'environnement vers le modèle de plante, les L-systèmes ouverts [Mech and Prusinkiewicz, 1996, Mech, 1997] permettent de modéliser une interaction bidirectionnelle entre la plante et son environnement. Dans ce cas, la tache de modéliser l'environnement est confiée à un programme externe (généralement écrit en C) qui communique alors avec le processus L-systèmes via un nouveau module de communication (échangé par fichier, pipe ou mémoire partagée). Le développement est supposé être caractérisé par un scalaire ou un vecteur de scalaires. Les modules d'une plante en croissance peuvent demander à l'environnement de tester certaines de ces valeurs pour certaines positions spécifiques. L'environnement pourra modifier ces valeurs en fonction de la position du module ou des valeurs de ces paramètres. Le modèle conceptuel derrière les L-systèmes ouvert est présenté dans la Figure 7-5.

La phase *internal processing* du modèle de plante correspond à un pas de dérivation. A chaque pas de dérivation, la chaîne est scannée et les modules de communication avec des informations additionnelles optionnelles (*e.g.* la position et l'orientation de la tortue dans l'espace) sont envoyés à l'environnement. L'environnement reçoit les informations (*reception*), traite les

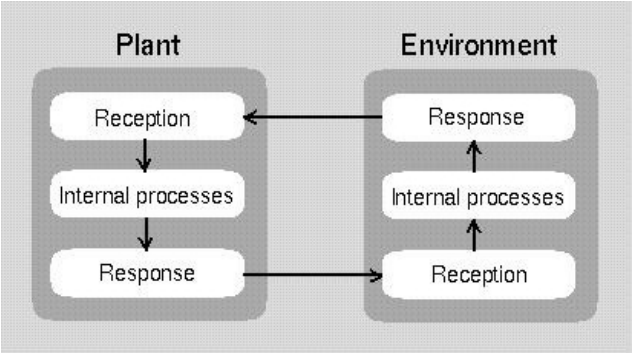


FIG. 7-5 – Modèle conceptuel de l'intéraction entre la plante et son environnement [Mech, 1997].

données, et renvoie sa réponse au modèle de plante. La plante reçoit alors la réponse et est prête pour le prochain pas de dérivation. Cette boucle d'interaction continue tout au long de la simulation.

L'échange d'information entre la plante et son environnement est fait à travers les modules de communication $?E$. Ces modules sont similaires aux symboles de requête des L-systèmes sensibles à l'environnement. La principale différence étant que, lors de la génération des modules de communication, les valeurs de leurs paramètres servent de données à l'environnement. Ces informations sont passées à l'environnement qui détermine alors sa réponse en donnant de nouvelles valeurs aux paramètres. Ces nouvelles valeurs sont alors renvoyées aux modules de communication et ensuite utilisées pour les prochains pas de dérivation.

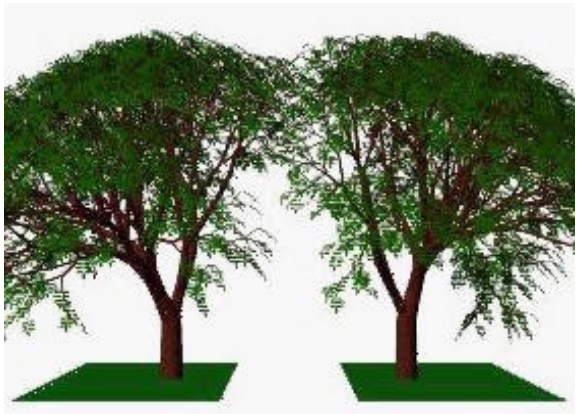


FIG. 7-6 – Un exemple d'utilisation des L-systèmes ouverts. L'environnement est utilisé pour modéliser la compétition pour la lumière entre les différents organes de deux plantes [Mech and Prusinkiewicz, 1996].

Le résultat d'un L-système ouvert où l'environnement gère l'illumination des organes est présenté dans la figure 7-6. Dans cet exemple, les branches sont en compétition pour la lumière. Des groupes de feuilles font de l'ombre aux feuilles situées en dessous. Un apex dans l'ombre ne produit plus de nouvelles branches. Une branche existante qui ne reçoit plus de lumière meurt et tombe de l'arbre. De cette manière, la compétition pour la lumière contrôle la densité de branches et des feuilles dans la couronne des arbres.

Pratiquement, à chaque pas de dérivation, le modèle L-système spécifie au programme en-

vironnemental les positions des groupes de feuilles et des apex. Celui-ci détermine alors les quantités de lumière venant du soleil reçues par chaque apex et les retourne au modèle via les modules de communication. Le L-système choisit ensuite, en fonction de la lumière reçue, d'élaguer ou de continuer la croissance des bourgeons.

7.4 Règles de décomposition et Homomorphisme

Les règles de productions L-systèmes permettent de rendre compte du développement des composants d'une plante dans le temps. Par exemple, la division d'une cellule mère A en deux cellules filles B et C peut être décrite par la règle de production $A \rightarrow BC$. Dans le cas d'organismes multi-cellulaires, les règles de productions L-systèmes sont appliquées en parallèle pour faire évoluer les différentes cellules dans le temps de manière consistante. La simulation est finie lorsque l'organisme a atteint un âge terminal prédéfini, correspondant à un nombre donné de pas de dérivation.

Les grammaires de Chomsky, à l'inverse, permettent de caractériser la structure d'une plante, *i.e.* la distribution de ses composants dans l'espace. Le fait qu'un organisme A consiste en deux sous-parties B et C peut donc aussi être exprimé par une règle de production, par exemple $A \rightsquigarrow BC$. Une telle *règle de décomposition* aura cependant une signification et une fonction différentes de son équivalent en L-système. Puisque des sous-structures qui ne se recouvrent pas peuvent être partitionnées indépendamment, les règles de décomposition peuvent donc être appliquées séquentiellement. De plus, une condition appropriée pour terminer le processus de décomposition est d'atteindre un symbole terminal, représentant un composant qui ne peut pas se diviser.

Les L-systèmes ont donc été étendus pour inclure de telles règles de décomposition, issues des grammaires de Chomsky. Ces règles de décomposition sont à spécifier dans le modèle après les règles de productions L-systèmes. Elles sont appliquées après chaque pas de dérivation, jusqu'à ce qu'aucun symbole ne puisse être réécrit (ou jusqu'à avoir atteint un nombre maximum de pas de dérivation de décomposition).

Les L-systèmes ont aussi été étendus par des *règles d'homomorphisme*, définies uniquement pour des problèmes d'interprétation. L'homomorphisme d'un L-système est défini comme un ensemble de règle donnant la correspondance géométrique des symboles contenus dans la structure. Il est en effet pratique de séparer la structure de la plante et son développement, modélisé par les règles de production L-système, de son interprétation géométrique. Ces règles sont appliquées juste avant la représentation du modèle. Elles font donc la transition entre la chaîne obtenue par la simulation (qui n'est pas modifiée et garde donc l'essence du modèle) et la visualisation.

7.5 Environnement de modélisation

Les modèles créés en utilisant les L-systèmes paramétriques devenant de plus en plus complexes, Hanan [Hanan, 1992] étend ce formalisme pour inclure des constructions de langage de programmation. Ces constructions incluent notamment l'assignement de variables locales ou globales, l'appel à des fonctions prédéfinies, des déclarations conditionnelles, des boucles. Similaire au C, ces constructions permettent une utilisation plus souple des L-systèmes et donnèrent lieu à l'implémentation du langage et de l'interpréteur cpfg (*continuous plant and fractal generator*). Ce programme contient donc un certain nombre de fonctions prédéfinies notamment des fonctions mathématiques, des générateurs de nombres pseudo-aléatoires, des fonctions pour manipuler des fichiers ou écrire sur la console.

Afin d'inclure et d'utiliser simplement ces fonctions, les productions des L-systèmes s'écrivent avec la syntaxe suivante :

$$pred : \{bloc_1\}cond\{bloc_2\} \rightarrow succ,$$

On observe l'addition de deux blocs $bloc_1$ et $bloc_2$ insérés avant la condition *cond* et avant le successeur *succ* par rapport à la définition originale des L-systèmes. Ces blocs sont des séquences de déclarations C permettant d'écrire des calculs complexes sans surcharger les parties *cond* et *succ*. Le premier bloc est exécuté avant l'évaluation de la condition. Si la condition est évaluée à vrai, le second bloc est évalué et la production appliquée.

Récemment, afin de pouvoir utiliser les constructions orientées objet dans le formalisme des L-systèmes, un nouveau langage *L+C* (L-systèmes + C++), et son interpréteur lpfg (*L+C plant and fractal generator*), ont été définis et développés par Karwowski [Karwowski, 2002]. Le langage comprend des instructions et constructions C++ et des déclarations spécifiques aux L-systèmes. Ce code est interprété dans ce cas en C++ en transformant les constructions L-systèmes en code C++ puis compilé et chargé dynamiquement par le programme lpfg. Il est ensuite exécuté pour générer le modèle.

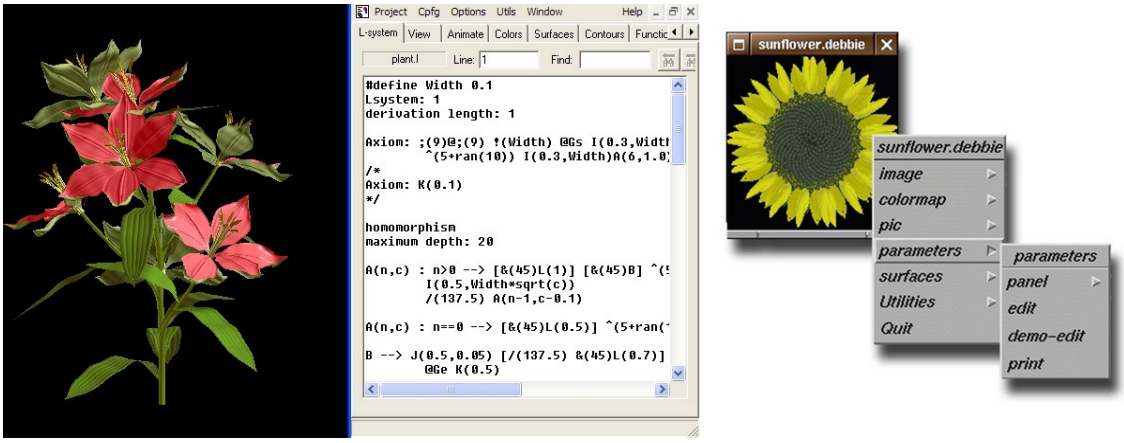


FIG. 7-7 – A gauche, l'environnement Lstudio/cpfg représentant un modèle de lis. A droite, l'environnement Vlab, permettant de personnaliser ses menus et donc les méthodes d'édition de ses objets, affichant un modèle de fleur de tournesol [Prusinkiewicz and Lindenmayer, 1990].

L'ensemble de ces outils a été intégré dans l'environnement de travail L-studio [Prusinkiewicz et al., 2000, Karwowski, 2002] (Figure 7-7). Celui ci comprend donc les deux interpréteurs cpfg et lpfg ainsi qu'un certain nombre d'éditeurs graphiques d'objets tels que des fonctions, des courbes, des surfaces et des matériaux. Ces derniers sont accessibles via les langages cpfg et L+C et peuvent donc être utilisés pour construire les modèles. Cet environnement a été développé pour l'instant uniquement pour les plate-formes Windows. Un autre environnement de modélisation, pour les plate-formes Unix/Linux est disponible : Vlab [Mercer et al., 1990, Mercer, 1991, Federl and Prusinkiewicz, 1999 Ce dernier est construit pour être une architecture modulaire dans laquelle de nouveaux modules peuvent s'intégrer très facilement (Figure 7-7).

7.6 Contrôle global

En utilisant les L-systèmes, une plante est modélisée en utilisant un ensemble de règles (les productions) qui décrivent des lois de croissance et d'interaction des composants locaux. Le plus souvent, la forme globale de la plante est une propriété émergente de ce système de règles locales. Il n'est pas toujours possible de prévoir un comportement macroscopique à partir de règles locales. Les caractéristiques globales de l'apparence de la plante, comme la silhouette générale de la couronne, sa densité d'organe, et la courbure des axes restent donc difficilement contrôlables. Différents travaux ont été réalisés pour résoudre ce problème.

7.6.1 Les topiaires

En utilisant les L-systèmes sensibles à l'environnement et le symbole Coupe, Prusinkiewicz *et al.* [Prusinkiewicz et al., 1994] propose une méthode pour sculpter virtuellement une plante en simulant la réponse d'une plante à l'élagage. Cette méthode, illustrée notamment par la création de topiaires virtuelles, s'appuie sur une forme représentant le volume contenant la plante. Lorsque le modèle croit, le L-système teste la position des différents organes de la plante en utilisant les modules de requête. Si les nouveaux organes sont à l'extérieur de la forme, ils sont alors élagués en insérant un symbole coupe. Dans ce cas, le développement de la plante ne prend pas vraiment en compte la forme globale mais est plutôt ajusté à la forme par élagage. Un exemple d'algorithme est donné dans la définition des L-systèmes sensibles à l'environnement. Des formes végétales variées sont obtenues grâce à cette méthode (voir Figure 7-8).

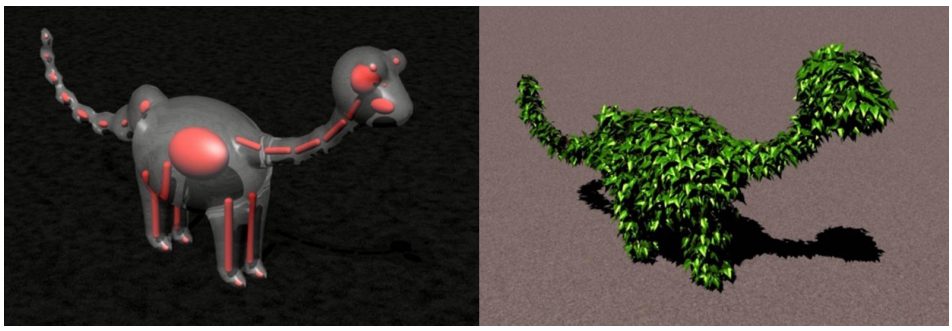


FIG. 7-8 – Une surface implicite, définie par un squelette composé de lignes et d'ellipsoïdes, est utilisée pour créer une topiaire en forme de dinosaure [Prusinkiewicz et al., 1994]

7.6.2 Reconstruction à partir de photographies

Une autre méthode s'appuyant sur des techniques similaires à la méthode précédente a été proposée par Shlyakhter *et al.* [Shlyakhter et al., 2001] afin d'automatiser la reconstruction de plantes à partir de photographies. La méthode part de plusieurs photographies sous différents angles de la plante pour reconstruire un volume englobant de la plante (voir section 1.4.2). A partir de l'enveloppe, ils calculent ensuite le squelette de l'arbre qui peut être défini comme le tronc et les premiers niveaux de ramification. Ce squelette est construit en calculant une bonne approximation de l'axe médian de l'enveloppe. L'axe médian d'un objet 3D A est défini comme un ensemble de points à l'intérieur de A avec la propriété suivante : pour chaque point de l'axe médian, la sphère la plus large centrée sur ce point et contenue dans A n'est contenue dans aucune autre sphère contenue dans A. Comme son nom le suggère, l'axe médian traverse le milieu de l'objet.

Le squelette de la plante est ensuite traduit sous forme de chaîne pour devenir l'axiome de départ d'un L-système, avec les deux derniers niveaux de branches munis de méristèmes. Un L-système ouvert modélisant la croissance d'un arbre d'un type choisi suivant le type du modèle réel est ensuite appliqué. Ce L-système communique avec deux processus environnementaux. Le premier assure que la forme de l'arbre résultant ressemble au modèle réel. Il élague les branches qui poussent en dehors de l'enveloppe. Le second augmente la qualité botanique des motifs de ramification et de la distribution de feuilles. Il calcule la quantité de lumière interceptée par chaque feuille et s'en sert pour simuler la compétition pour la lumière et les flux d'assimilats dans l'arbre.

Une fois la simulation finie, les feuilles du modèle résultant sont coloriées en fonction des photos des différents points de vue, permettant ainsi de donner une apparence au feuillage

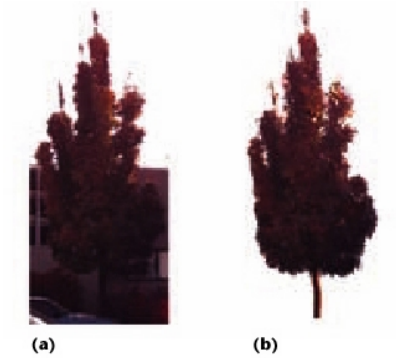


FIG. 7-9 – Un modèle (b) reconstruit à partir de photographies de l'arbre original (a). Tiré de [Shlyakhter et al., 2001].

cohérente avec le modèle réel (Figure 7-9).

7.6.3 Utilisation de l'information positionnelle

Les deux approches précédentes utilisent une enveloppe 3D prédéfinie pour contraindre un modèle de plante en croissance à pousser à l'intérieur. Le contrôle de la forme globale de la plante est donc assuré par cette enveloppe.

Prusinkiewicz *et al.* [Prusinkiewicz et al., 2001] se sont intéressés au contrôle d'autres caractéristiques globales de la géométrie d'une plante comme la géométrie des axes, parfois fortement courbés, ou la densité d'organes dans un système ramifié. L'idée principale de leur méthode est l'utilisation de l'*information positionnelle*, définie comme la position d'un organe le long de l'axe qui le porte. Cette information combinée à des fonctions définies par l'utilisateur permet d'exprimer des *gradients morphogénétiques*, qui décrivent la distribution d'une caractéristique le long des axes. On retrouve cette notion de gradient morphogénétique dans la littérature botanique [Barthélémy et al., 1997] où il est observé par exemple que la longueur des branches peut être exprimée en fonction de leur position sur le tronc.

Un exemple de cette approche est donné dans la Figure 7-10.a où la longueur des feuilles d'une fougère d'eau est définie en fonction de la position u de la feuille le long de la tige et une fonction $\mathcal{F}$. Le L-système responsable de la génération de ce modèle, avant de créer une feuille à la position u (normalisée entre 0 et 1) sur l'axe principal, demande la valeur de la fonction $\mathcal{F}$ en u. Cette valeur donne la longueur de la feuille à créer. Ainsi en manipulant la fonction $\mathcal{F}$, on change globalement les longueurs des feuilles et donc la forme globale de ce système ramifié. Un autre exemple avec une fonction définissant les longueurs de branches d'un houppier est présenté dans la figure 7-10. Cette technique peut s'appliquer pour d'autres caractéristiques telles que l'angle de branchement, le diamètre, *etc.*

Pour modéliser la géométrie d'un axe courbé dans un modèle de plante en croissance écrit en L-système, les auteurs proposent d'utiliser une courbe 3D définie par l'utilisateur comme guide lors de la création de nouveaux entre-nœuds de l'axe. Pour chaque entre-nœud ayant une position u sur l'axe, le système est capable de calculer sa position $\vec{P}$ et son orientation $\vec{H}$, $\vec{U}$, $\vec{L}$ grâce aux modules de requête $?P$, $?H$, $?U$ et $?L$. La tangente $\vec{T}$ de la courbe de référence à la position u peut être demandée via les fonctions $\tan X(id, u)$, $\tan Y(id, u)$, $\tan Z(id, u)$ où id représente l'identifiant de la courbe dans le système. Ce vecteur nous permet de calculer les rotations à appliquer au prochain entre-nœud pour l'orienter correctement : Deux rotations,

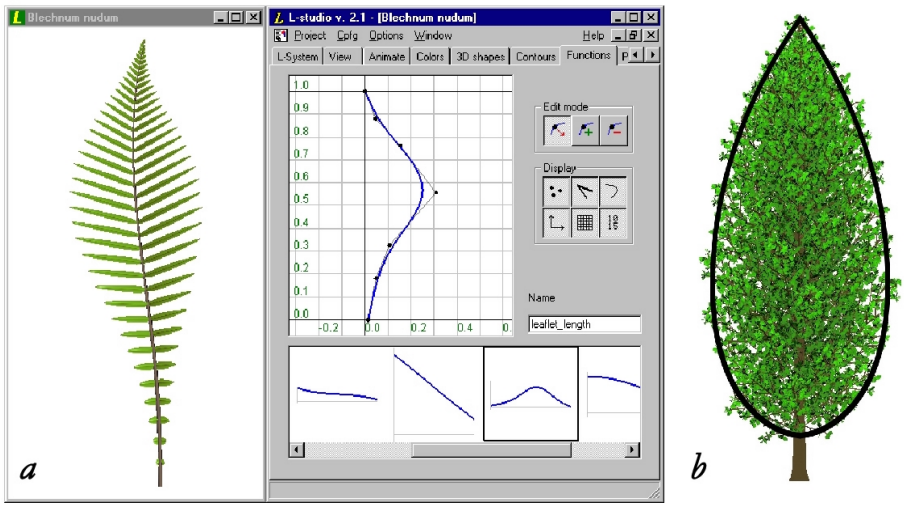


FIG. 7-10 – (a) Un modèle de fougère d'eau dont les longueurs de feuilles sont déterminées par leurs positions sur l'axe principal et une fonction contrôlable par l'utilisateur. (b) Un modèle d'arbre générique où la longueur des branches du houppier est spécifiée par une fonction représentée en noir ici. Tiré de [Prusinkiewicz et al., 2001].

respectivement autour de $\vec{U}$ et $\vec{L}$

$$d\Omega_U = \vec{T}.\vec{L} \text{ et } d\Omega_T = \vec{T}.\vec{U}$$

Algorithme 2 L-système permettant de construire un axe en fonction d'une courbe 3D

```
#define l 1.0  // Longueur totale de l'axe
#define Δs 0.02  // Pas de la tortue
#define P 1  // Identifiant de la courbe
#define K 57.29 // radian en degrés
Axiom : A(0)?U(0,0,0)?L(0,0,0)
A(s) > ?U(u_x,u_y,u_z)?L(l_x,l_y,l_z) : { s' = s + Δs } s' ⩽ l
        {t_x = tan X(P,s') ; t_y = tan Y(P,s') ; t_z = tan Z(P,s') ;
         dΩ_U = K * (t_x l_x + t_y l_y + t_z l_z) ;  dΩ_L = K * (t_x u_x + t_y u_y + t_z u_z) ;}→
        +(dΩ_U)&(dΩ_L)F(Δs)A(s')
```

Dans l'algorithme 2, la structure initiale consiste en un apex A suivi de modules de requête $?U$ et $?L$. Le paramètre de l'apex représente la position courante de l'apex, mesuré comme distance de longueur d'arc à partir de l'origine de la courbe P. La production de ce L-système crée un axe sous forme de séquences de tranches de cylindre généralisé de longueur Δs. Pour chaque tranche (représentant un entre-nœud) les rotations $d\Omega_U$ et $d\Omega_L$ sont calculées en multipliant (produit scalaire) les vecteurs $\vec{U}$ et $\vec{L}$ retournés par les modules de requête $?U$ et $?L$ avec le vecteur tangent $\vec{T}$ de la courbe $\mathcal{P}$ retourné par les fonctions $\tan X$, $\tan Y$, $\tan Z$. Les valeurs $d\Omega_U$ et $d\Omega_L$ orientent le prochain segment de la courbe, représenté par le module $F(\Delta s)$. Un exemple d'application de cet algorithme est montré dans la Figure 7-11.

Pour définir la densité d'organes portés par un axe, Prusinkiewicz *et al.* proposent d'utiliser l'information positionnelle avec une fonction représentant la densité locale de nœuds. Placer les nœuds sur l'axe revient à contrôler la densité d'organes portés. Pour cela, ils se servent d'une

FIG. 7-11 – *Allium vineale* (ail sauvage) modélisé en utilisant l'algorithme 7.6.3. Tiré de [Prusinkiewicz et al., 2001].

fonction λ telle que

$$N(s_0, s) = \int_{s_0}^{s} \frac{ds}{\lambda(s)}$$

représente le nombre d'entre-nœuds entre les positions s_0 et s sur l'axe.

Algorithme 3 Algorithme pour placer les noeuds sur l'axe

Axiom : $A(0,0)$
$p_1 : A(s,a) : \{\ s' = s + \Delta s\ \}\ s' \leq l$
$\quad \{\ a' = a + \Delta s/\lambda(s)\,;$
$\quad\quad$ if($a' < 1$) $\{\ flag = 0\,;\ \}$ else $\{\ a' = a' - 1\,;\ flag = 1\,;\ \}\ \}$
$\quad \rightarrow F(\Delta s)B(flag)A(s', a')$
$p_2 : B(flag) : flag == 0 \rightarrow \varepsilon$
$p_3 : B(flag) : flag == 1 \rightarrow @o$

Dans l'algorithme 7.6.3, la structure initiale consiste en un apex A. Le premier paramètre représente la distance du point courant sur l'axe depuis la base de l'axe, comme pour l'algorithme précédent. Le second paramètre représente la fraction de l'intégrale $N(0, s)$. La production p_1 crée un axe comme une séquence de segments F de longueur Δs, séparés par des marqueurs de nœud potentiel B. Si la variable $flag$ est à zero, le module B est supprimé (production p_2). Quand a est plus grand que 1, $flag$ est défini à 1 pour produire un marqueur de nœud avec le symbole $@o$ (production p_3).

Ces différents algorithmes permettent donc de contrôler une caractéristique globale de la plante par des fonctions définies par l'utilisateur et en utilisant l'information positionnelle. Un exemple d'utilisation combinée de ces différents algorithmes est proposé dans la Figure 7-12.

L'information positionnelle et les fonctions pour définir la forme générale d'un houppier peuvent aussi être utilisées dans la définition d'un écosystème. Différents travaux [Deussen et al., 1998, Lane and Prusinkiewicz, 2002] exposent des méthodes générales pour définir globalement un écosystème où les individus sont représentés très simplement par un diamètre de projection au sol. En utilisant des relations allométriques, les hauteurs des plantes sont instanciées et permettent de définir des profils simples des plantes. Ces profils peuvent être utilisés comme fonction de contrôle de la géométrie des houppiers pour les algorithmes présentés précédemment, générant des représentations détaillées. Ces dernières peuvent être placées ensuite dans l'écosystème pour en donner une représentation réaliste.

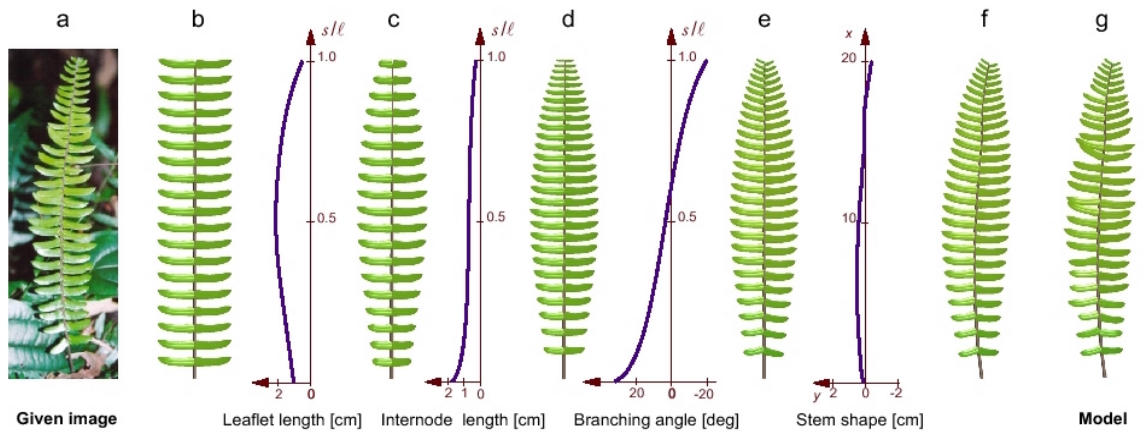


FIG. 7-12 – Utilisation de l'information positionnelle pour contrôler différentes caractéristiques globales d' une feuille de *Pellaea falcata* (fougère faucille). Tiré de [Prusinkiewicz et al., 2001].

7.7 Modélisation interactive

Le besoin d'interaction directe avec un modèle généré par L-système fut observé il y a quelques années. Différentes méthodes ont été proposées pour répondre à ce besoin. Une première méthode consiste à manipuler interactivement les différents paramètres du modèle. Une deuxième consiste à modifier directement la structure générée. Ces différentes méthodes sont présentées dans cette section.

7.7.1 Modélisation continue

La modification de paramètres tel que des fonctions, des courbes et des surfaces passait autrefois par une étape d'édition suivie d'écriture des nouvelles valeurs dans un fichier, et une relecture de ces paramètres par le simulateur pour finalement relancer la simulation et enfin obtenir le modèle modifié. Les progrès récents dans les capacités de calculs des ordinateurs et la vitesse croissante des cartes graphiques ont rendu possible le processus de modélisation continue avec les L-systèmes.

La modélisation continue [Karwowski, 2002] libère l'utilisateur des étapes de sauvegarde et relecture des paramètres en le laissant se concentrer sur la tache de modélisation (analyse de l'effet du changement de valeur des paramètres). Après chaque édition de paramètres (via les éditeurs de L-studio), les modifications sont automatiquement sauvées et une requête est envoyée au programme de modélisation pour les relire et relancer la simulation. De cette manière, les informations mises à jour sont transférées continuellement des éditeurs vers le programme de modélisation. Les valeurs numériques peuvent donc être éditées interactivement avec le Panel de L-studio. Les fonctions, les courbes et les surfaces ont aussi leur propres éditeurs (voir Figure 7-13).

7.7.2 Interaction avec le modèle : ilsa

Power *et al.* [Power et al., 1999] proposent une méthode pour manipuler graphiquement et interactivement des modèles générés par L-système grâce à leur application *ilsa* (interactive L-système string arranger). Pour manipuler une plante en utilisant cpfg/ilsa (Figure 7-14), l'utilisateur doit d'abord créer un modèle dans cpfg. Lorsque le modèle est généré, la chaîne est sauvée dans un fichier qui est ensuite lu dans ilsa. Ilsa fournit une fenêtre graphique où l'utilisateur peut interactivement manipuler les organes en les tournant ou en les tordant. Les branches

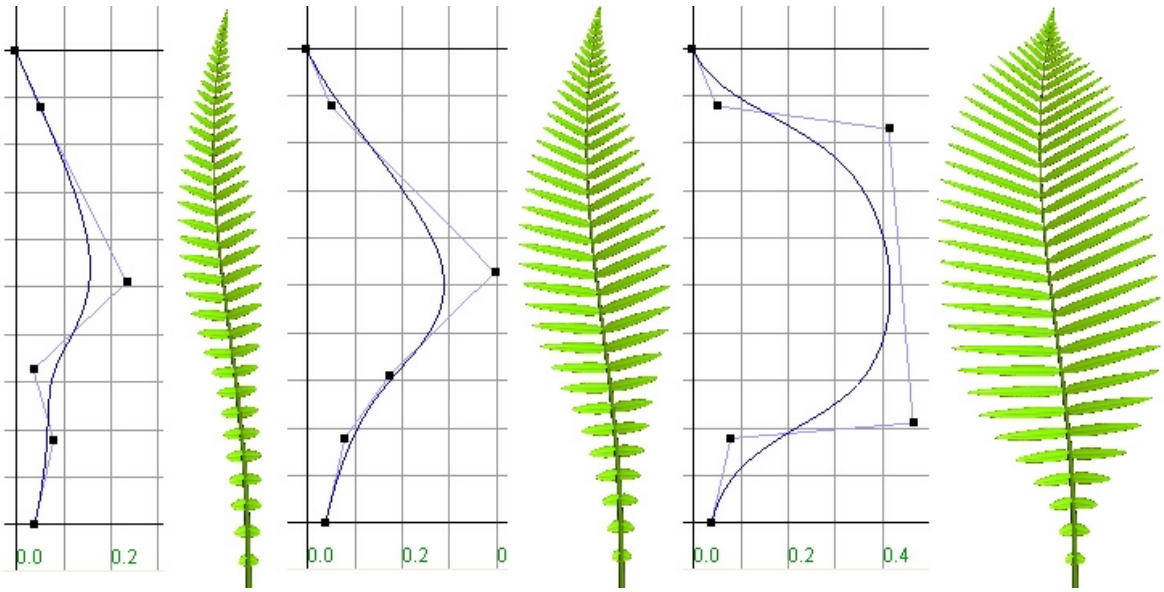


FIG. 7-13 – Ajustement de la fonction de longueur de feuilles sur un modèle de fougère d'eau (voir Figure 7-10). Cet ajustement est fait interactivement en manipulant une fonction 2D dans l'éditeur de L-studio.

sont modélisées comme une succession de segments rigides connectés par des articulations de type ressorts qui se tordent et se courbent en fonction de l'action appliquée par l'utilisateur. Ce système insère des commandes pour la tortue correspondant à ces opérations dans la chaîne. La chaîne modifiée peut alors être sauvée et relue par cpfg pour continuer la simulation.

FIG. 7-14 – L'application ilsa travaillant avec cpfg. La représentation détaillée est donnée par cpfg sur la gauche. ilsa fournit une représentation simplifiée permettant de manipuler interactivement le modèle de plante.

7.7.3 Interaction avec le modèle : le module prédéfini X

Karkowski *et al.* [Karwowski, 2002] proposent une méthode générale pour interagir graphiquement avec un modèle généré par L-système. Cette méthode est basée sur l'insertion d'un module prédéfini X. L'utilisateur peut cliquer sur un élément du modèle dans la fenêtre gra-

phique et le module X est alors inséré dans la chaîne juste devant le module sélectionné. Les productions peuvent ainsi spécifier la réaction du modèle lorsqu'un module X est trouvé dans la chaîne. Par exemple, la règle très simple

$$p : X \rightarrow \%$$

permet à l'utilisateur d'élaguer interactivement son modèle de plante.

7.8 Conclusion

Nous avons présenté dans ce chapitre les principaux concepts de la théorie des L-systèmes. Ce système, par des règles de croissance locale, permet de construire simplement des modèles détaillés de plantes.

Un problème majeur de cette approche est le contrôle du modèle généré. La forme émergente du modèle est difficilement contrôlable par les règles locales. Plusieurs extensions permettent néanmoins de contrôler globalement un modèle généré par L-systèmes : l'utilisation de l'information positionnelle et de fonction 2D, l'utilisation d'enveloppe globale et de l'élagage, *etc.* Ces approches n'abordent cependant que la manipulation globale du modèle. La possibilité de contrôler spécifiquement des sous-parties du modèle (un système ramifié de la plante par exemple) n'a pas été abordée.

L'interaction avec un modèle généré par L-systèmes a aussi donné lieu à des premiers travaux. La modélisation continue permet de manipuler interactivement les différents paramètres d'un modèle. Elle ne permet cependant pas d'agir directement avec le modèle. Le logiciel ilsa permet une interaction directe sur le modèle. Il ne permet cependant d'interagir que sur les composants locaux du modèle, rendant difficile le contrôle de la forme globale.

Chapitre 8

XFrog, un éditeur graphique interactif de modèles de plante

Les L-systèmes, présentés précédemment, sont un formalisme simple particulièrement adapté pour la modélisation de la croissance de plantes. Cependant, ils ne permettent pas aux infographistes par exemple, de créer, de manière simple et interactive, des modèles 3D de plantes, basés principalement sur des critères géométriques. Deussen et Lintermann [Deussen and Lintermann, 1997, Lintermann and Deussen, 1999] proposent une méthode de modélisation interactive qui permet la génération d'une large variété de structures arborescentes. Cette méthode a été implémentée sous la forme d'un logiciel, XFrog.

Un ensemble de *composants* décrivant des éléments géométriques et/ou structurels de la plante sont assemblés en un *graphe* qui forme la description d'un modèle spécifique de plante et génère sa géométrie.

Cette méthode propose de combiner le pouvoir d'expression des approches procédurales tel que les L-systèmes avec l'intuitivité des méthodes d'arborescences génériques. Dans ce système interactif, l'utilisateur peut ainsi manipuler directement les composants du graphe avec un retour immédiat de ses actions.

8.1 La méthode de modélisation

Ce système est basé sur un graphe qui contient des composants permettant de générer des parties de plantes. Ces composants représentent des algorithmes optimisés qui génèrent et multiplient de la géométrie. Ainsi, cette description reste très intuitive car elle est composée uniquement d'unités de haut niveau.

8.1.1 Les différents types de composants

Dans cette approche, un ensemble de composants contenant des informations géométriques et structurelles a été défini. Ces composants encapsulent des données et des algorithmes pour générer des éléments de la plante. Ils ont différentes fonctionnalités. Ils peuvent produire des primitives géométriques, définir des propriétés de matériaux tel qu'une couleur ou une texture, une profondeur de récursion ou des paramètres de transformations.

Certains composants permettent de multiplier et placer algorithmiquement d'autres composants. Ces composants permettent par exemple de placer des objets sur une surface en fonction de la section d'or (tel que la distribution de graines sur une feuille de tournesol). Un des paramètres de ces composants est le nombre de composants à multiplier, d'autres contrôlent leur placement et leur orientation.

Certains composants peuvent aussi être utilisés pour définir des contraintes globales ou locales. Des composants spécialisés encapsulent des opérations complexes ou génèrent des formes

spéciales. Les composants peuvent être classifiés de la manière suivante :

- **Génération de géométrie**

Simple : Un composant qui produit une géométrie simple (cube, sphère, cylindre, etc.).

Revo : Produit une surface de révolution. La définition est faite en éditant une spline qui décrit le contour.

Horn : Un composant Sweep qui place d'autres composants sur une courbe définie par l'utilisateur. Ce composant est utilisé pour définir des tiges, coudes, etc.

Leaf : Ce composant est utilisé pour la construction de feuilles (botaniques). La feuille est définie par son contour, plusieurs courbures et son matériel. Les textures peuvent être utilisées pour définir la structure interne.

- **Itération et arrangement de composants**

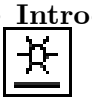
Tree : Composant de base pour les arbres. Crée la géométrie d'une tige et multiplie les composants fils comme branches. Les paramètres sont la distribution de branches, leurs tailles, angles, etc.

Hydra : Multiplie les composants fils sur un cercle avec des angles uniformes et une direction perpendiculaire à la direction du composant père.

Wreath : Multiplie les composants fils de manière similaire à Hydra, mais avec une direction parallèle à la direction du composant père.

PhiBall : Multiplie les composants fils sur la section d'une sphère en fonction d'une section d'or.

- **Introduction de contraintes**



World : Introduit des contraintes liées aux champs de lumière et de gravitation sur la géométrie composants fils de type Tree.

- **Transformation de sous hiérarchies**

FFD : Permet des déformations de forme libre de la géométrie produite par les composants fils et est aussi utilisé pour désactiver les FFD parentes. Différentes méthodes de définition peuvent être utilisées.

HyperPatch : Permet des déformations de forme libre de la géométrie produite par les composants fils et permet aussi de désactiver les FFDs parentes sur ses fils. La déformation est contrôlée en bougeant les points de contrôle d'un patch 3D.

Un composant particulier, qui sert de racine au graphe représentant le modèle, est le composant *caméra* qui permet de contrôler la vue et les positions des sources lumineuses.

Les composants sont représentés graphiquement par des icônes. Ces icônes peuvent être

combinés dans le graphe décrivant le modèle. L'utilisateur les sélectionne à partir d'une boîte à outils et les combine en les accrochant sur l'icône racine. En cliquant sur un icône, l'ensemble des paramètres du composant correspondant peut être édité via une boite de dialogue. Par exemple, le composant *feuille* utilise une courbe pour spécifier le contour d'une feuille (Figure 8-1). L'édition d'une Spline permet à l'utilisateur de définir la courbure latérale et longitudinale.

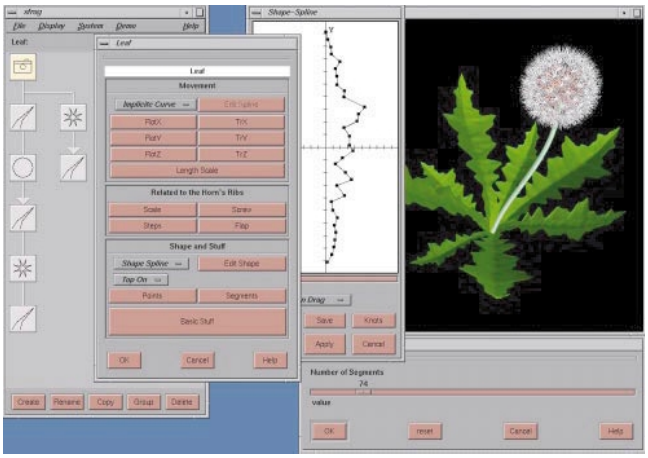


FIG. 8-1 – Les fenêtres principales de XFrog. La structure de la plante est représentée dans la fenêtre en haut à gauche, et le résultat apparaît sur la droite. Les paramètres de chaque composant peuvent être édités via des éditeurs. Ici, le contour d'une feuille est édité, comme le montre la boîte de dialogue "Shape-spline" au centre.

8.1.2 Représentation par un graphe

La description de la plante complète est construite en connectant les prototypes de composants en un graphe orienté appelé le graphe des prototypes ou *p-graph*. Les sommets du graphe sont des composants qui représentent des parties de la plante, et les arêtes les dépendances de création. Le graphe représente ainsi les règles de création et définit un système de règles sans contexte. L'information structurelle, dans cette approche, est représentée de deux manières. La première représentation est donnée par la structure du p-graphe, et l'autre par la classe spéciale de composants qui multiplient leurs enfants algorithmiquement.

Pendant la création d'un p-graphe, l'utilisateur choisit un composant, à partir d'une boîte à outils graphique, qui est alors automatiquement placé près du graphe. En sélectionnant un composant et en le bougeant sur un autre composant, l'utilisateur établit ainsi un lien. Trois types de liens peuvent être utilisés, de la manière suivante :

- *Lien "enfant"* : Le lien standard. La représentation géométrique du composant est placée relativement au modèle géométrique du père. Le p-graphe affiche ce type de lien sous forme de lignes fines.
- *Lien "branche"* : Le composant fils est multiplié sous forme de branches si le parent est un composant de type Tree. Dans les autres cas, ce lien est traité comme lien enfant. Ces liens apparaissent sous forme de lignes épaisses, dans le p-graphe.
- *Lien "feuille"* : Si le composant père fait partie d'une récursion, le fils est crée seulement après que la récursion soit terminée (la cible est une feuille au sens structurel). Ces liens apparaissent en pointillé, dans le p-graphe.

Quand le système traverse le p-graphe pour générer la géométrie, il construit d'abord un arbre temporaire composé des instances des composants appelé *i-arbre*. Chaque lien entre deux composants dans le p-graphe correspond à une dépendance de création. Le prototype parent

est d'abord copié pour créer une instance. Le système développe ensuite le prototype fils puis connecte l'instance du prototype parent à l'instance (ou aux instances s'il y en a plusieurs) du fils par un lien. Toutes les instances vont former le i-arbre.

Si un composant multiplicateur fait partie du graphe, pendant l'expansion du p-graphe en i-arbre, le système crée autant d'instances de tous les composants fils que définit dans le composant multiplicateur. Afin d'obtenir des fils avec des géométries différentes, le prototype d'un composant multiplicateur contient plusieurs rangs de valeurs pour différents paramètres (par exemple la taille de ses enfants). Le système interpole les valeurs pour obtenir la valeur spécifique pour chaque instance d'un fils. Si le rang d'un paramètre est $[v_0, v_1]$, la valeur de ce paramètre pour la i-ème instance parmi n est $v_i = v_0 + i(v_1 - v_0)/(n-1)$. Avant que le système utilise cette valeur de paramètre, il applique une fonction arbitraire donnée par l'utilisateur. Cela permet ainsi d'introduire de l'aléatoire dans le modèle et de faire varier la forme individuelle de chaque composant fils.

Chaque prototype de composant a un paramètre qui définit la plus grande profondeur de récursion. Si elle est définie dans le p-graph, une récursion se transforme en un sous-arbre en produisant un certain nombre d'instances en fonction de la profondeur de récursion.

Une fois le p-graphe expansé en i-arbre, le système traverse le i-arbre pour produire la géométrie. Le composant racine du i-arbre engage la production de la géométrie en sortie en demandant à ses enfants à faire de même. Les propriétés géométriques sont contrôlées par les paramètres des composants.

8.1.3 Un exemple

Un exemple simple permet de clarifier l'utilisation des composants et du graphe. Nous allons voir comment générer une plante composée d'un tronc suivi de trois branches, composées chacune de trois segments (Figure 8-2.c). Le racine du p-graphe modélisant une telle structure est un composant prototype A. Il est suivi de trois composants B, C et D. Le prototype B est un composant simple qui génère de la géométrie. Ce composant produit un segment pour représenter le tronc de la plante. C est un composant multiplicateur qui doit produire trois instances de ses enfants et aucune géométrie. Le prototype D produit un segment et a une profondeur de récursion égale à trois.

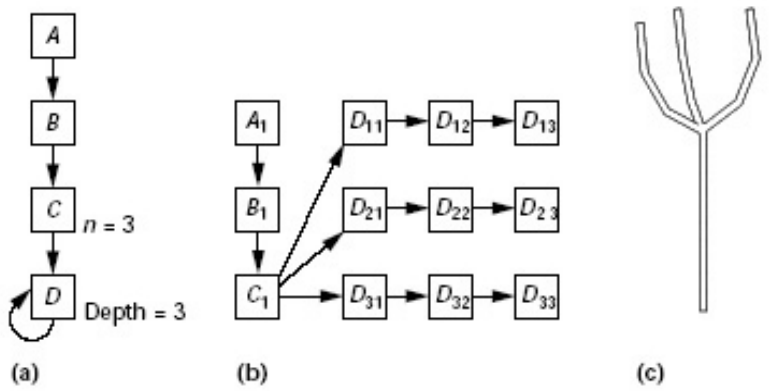


FIG. 8-2 – Pour générer une plante, l'utilisateur construit un p-graphe (a), qui est développé en i-arbre (b). L'arbre produit ensuite la géométrie.

La Figure 8-2.a montre les composants (A à D) et leurs connections qui forment le p-graphe, et la Figure 8-2.b le i-graphe résultant. Le prototype du composant C génère trois instances de

D, soit D_1, D_2 et D_3, en fonction de ses paramètres de multiplication et les connectent à C_1. La récursion définit sur le prototype D forme une séquence de trois instances : D_{i1}, D_{i2}, et D_{i3}. Cela donne neuf instances de D dans l'arbre final. La Figure 8-2c donne le résultat géométrique.

8.2 Modélisation de forme naturelle

La conception d'une scène végétale nécessite souvent la modélisation d'une forme particulière de plante, tel que celle d'un arbre poussant derrière un mur ou une plante qui est partiellement dans l'ombre ou qui a subit l'effet du vent. Pour répondre à ce besoin, quatre méthodes additionnelles ont été définies pour générer une grande variété de formes et d'artefacts.

Ces techniques sont l'utilisation d'exceptions, la modélisation fonctionnelle, le tropisme et les déformations de formes libres.

8.2.1 Utilisation des exceptions

Dans le cas de la multiplication algorithmique d'un même composant, on peut vouloir distinguer une instance particulière et ne pas la représenter. Une méthode pour résoudre cela est l'utilisation d'exceptions. Chaque composant multiplicateur contient une liste de valeurs booléennes pour indiquer si le composant doit générer ou pas la géométrie pour une certaine itération. La valeur booléenne est mise à vrai pour les itérations qui sont des exceptions (et donc ne doivent pas générer de géométrie) et à faux pour les autres. Une sorte de sensibilité au contexte est ainsi introduite car le composant multiplicateur spécifie un contexte en spécifiant à chaque itération si un composant à multiplier doit être généré ou non suivant son rang.

8.2.2 Modélisation fonctionnelle.

Comme expliqué précédemment, les composants multiplicateurs utilisent des intervalles de valeurs pour certains paramètres de leurs enfants (par exemple le facteur de dilatation ou la courbure d'une feuille). Avant que le système utilise la valeur d'un de ces paramètres pour un rang particulier, une fonction est préalablement appliquée sur la valeur. Ceci permet par exemple d'introduire de l'aléatoire dans le modèle. Des fonctions mathématiques standards, tel que sinus, cosinus, etc, ou des générateurs de nombres aléatoires peuvent ainsi être appliqués. L'utilisateur peut définir des fonctions arbitraires pour chaque intervalle de paramètres. Les paramètres tels que la profondeur de récursion ou le nombre d'itérations peuvent être utilisés à l'intérieur de ces fonctions.

FIG. 8-3 – Un modèle d'agave. La courbure des feuilles peut être définie en fonction du nombre d'itérations et d'une fonction aléatoire.

La figure 8-3 montre un modèle d'agave qui illustre cet effet. Les feuilles presque verticales sont moins courbées que les autres. Cela a été fait en utilisant le nombre d'itérations que le

composant Phiball donne pendant la multiplication. L'ajout d'aléatoire permet de donner une apparence plus naturelle.

8.2.3 Tropismes

Le tropisme spécifie la sensibilité d'une plante ou d'une partie de la plante à la gravité ou à la direction de la lumière. Le tropisme peut modéliser des effets très différents. Un tropisme traditionnel a été utilisé pour créer le modèle de saule pleureur de la Figure 8-4a, un tropisme cylindrique pour le modèle de philodendron autour d'une tige de la Figure 8-4d. De la même manière, l'influence du vent (Figure 8-4b) peut être simulée par un tropisme horizontal, ou une plante peut être forcée à pousser contre un mur (Figure 8-4c).

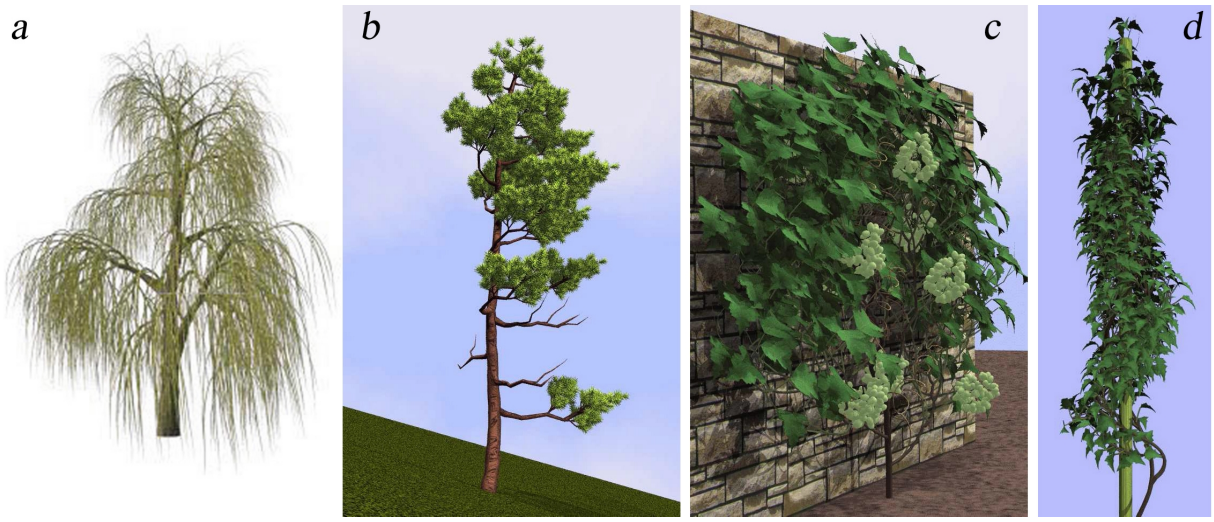


Fig. 8-4 – Quatre exemples d'utilisation du tropisme. De gauche à droite, un saule pleureur affecté par un fort tropisme gravitationnel, deux tropismes horizontaux qui simulent l'influence du vent et forcent une vigne à pousser sur un mur et un tropisme cylindrique pour modéliser un philodendron qui pousse le long d'une tige

8.2.4 Les déformations de formes libres

Les déformations de formes libres (FFDs) peuvent aussi être utilisées pour changer globalement la forme d'un modèle entier. Deux types de composants ont été créés dans ce but. Le composant FFD permet à l'utilisateur de définir des déformations indépendantes suivant les 3 axes, tandis que, comme le montre la Figure 8-5, le composant hyperpatch utilise un cube 3D pour contrôler la déformation. L'utilisateur sélectionne et bouge les points du cube parallèlement au plan de vue. Ces points sont des points de contrôle d'une fonction de Bézier 3D qui définit la déformation. Placés dans le p-graphe, ces composants modifient toute la sous-arborescence formée par leurs enfants.

Il est souvent nécessaire de ne modifier qu'une partie du modèle, comme par exemple les branches d'un arbre sans les feuilles ou les aiguilles. Placer un autre composant FFD ou HyperPatch dans le sous-arbre affecté par une déformation libre de forme permet de limiter la portée de cette dernière. Ce composant est utilisé comme délimiteur d'effet de la déformation définie précédemment. Un exemple de déformation libre de forme partielle est donné par la figure 8-6. Dans la figure 8-6b les branches sont déformées, mais les aiguilles restent inchangées.

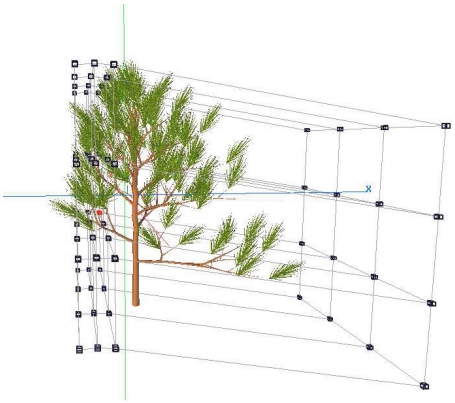


FIG. 8-5 – L'interface utilisateur du composant HyperPatch avec affichage du patch 3D de contrôle

Dans la figure 8-6c qui montre le résultat de la déformation totale de la plante, les aiguilles sur la droite apparaissent trop grosses, ce qui est le résultat de cette forte déformation.

FIG. 8-6 – Une déformation libre de forme appliquée sur un pin. (a) le modèle non déformé ; (b) déformation des branches (les aiguilles restent inchangées) ; et (c) déformation de tout l'arbre.

8.3 Evaluation de l'interface

Le logiciel XFrog a été évalué par différents types d'utilisateurs, des novices et des experts.

Les novices ont globalement trouvé l'utilisation de la structure de graphe et des composants intuitive, claire et facile à éditer. L'espace de paramètres particulièrement large semble cependant poser quelques problèmes. L'influence de certains paramètres, comme par exemple différentes rotations, a été considérée comme difficile à comprendre par ces utilisateurs. Ce problème est toutefois contrebalancé par le retour direct des actions par le système qui permet aux utilisateurs de tester les différents paramètres de manière interactive.

Les utilisateurs expérimentés sont capables de modéliser des objets complexes en peu de temps. La réutilisation de modèles existants pour créer de nouveaux modèles rajoute de l'efficacité à la méthode. Les composants comme le tropisme ou les déformations de formes libres permettent de rapidement changer la forme globale de la plante. Les problèmes rapportés par ces utilisateurs concernent l'influence et la combinaison de plusieurs paramètres de l'arbre. La création d'arbres complexes avec cette technique est possible mais reste tout de même compliquée.

8.4 Représentations géométriques des plantes

Cette plate-forme de design de plantes étant développée dans le but de créer intuitivement des modèles de plantes de bonne qualité infographique, différentes études ont été menées pour améliorer la représentation 3D finale. Une première étude propose d'utiliser des points comme primitives graphiques pour améliorer la vitesse d'affichage de ces modèles à la complexité importante. Une deuxième propose des représentations simplifiées et schématiques de ces modèles permettant de faire ressortir différentes caractéristiques visuelles.

8.4.1 Réduction de la complexité du modèle géométrique

Différentes techniques ont été proposées par Deussen *et al.* [Deussen et al., 2002] pour réduire la complexité des modèles de plantes, notamment ceux générés par XFrog. L'utilisation de primitives *points* et *lignes* pour le rendu permet de réduire efficacement la quantité de données et le temps de calcul pour l'affichage [Pfister et al., 2000, Rusinkiewicz and Levoy, 2000]. La hiérarchisation de ces points permet de n'afficher qu'un nombre minimal de primitives suivant la résolution de la visualisation (Pour plus de détails, voir section 3.3.2).

8.4.2 Représentation non-photoréaliste de plantes

Depuis quelques années, des chercheurs en infographie s'orientent vers la création informatique de dessins avec de simples lignes. Différentes techniques ont été proposées pour faire des croquis et du rendu non-photoréaliste à partir d'objets 3D. La recherche dans ce domaine a été motivée par le fait que des croquis permettent de transmettre des informations visuelles de manière alternative et souvent plus claire que par des représentations réalistes [Strothotte and Strothotte, 1997]. C'est une des raisons pour laquelle un large pourcentage des images dans les livres sont des dessins [Strothotte et al., 1994]. Une partie de la recherche en infographie consiste donc à générer des images similaires aux dessins artistiques, et dans certains cas à proposer de nouveaux styles.

Les croquis de plantes étant fortement utilisés en botanique ou pour des applications comme le design de paysage, Deussen *et al.* [Deussen and Strothotte, 2000] ont donc appliqué ces nouvelles techniques de représentation sur des modèles de plantes. Dans ces représentations, les objets sont représentés par leurs contours importants. Ces derniers sont calculés à partir d'une représentation 3D réaliste. Pour cela, l'utilisation du Z-buffer et l'analyse des différences de profondeurs entre chaque pixel suivant différents seuils [Saito and Takahashi, 1990] permettent de faire ressortir des caractéristiques plus ou moins globales (Figures 8-7). Le calcul de ces représentations est pour l'instant trop complexe pour permettre une visualisation interactive.

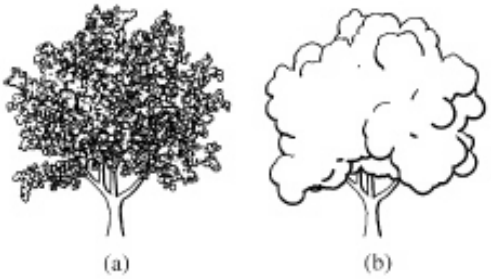


FIG. 8-7 – Un arbre rendu de manière non-photoréaliste. Différents seuils peuvent être utilisés pour calculer les contours importants de la forme à représenter.

8.5 Conclusion

Deussen et Lintermann ont développé et implémenté un système pour modéliser interactivement des modèles 3D de plantes. Pour cela, ils utilisent une structure de graphe qui contient des composants représentant des parties de plantes capables de générer de la géométrie. L'interactivité et l'intuitivité du système ainsi que la qualité visuelle des représentations finales ont été les buts principaux de leur démarche. La qualité des résultats montre la pertinence de cette approche.

Ce travail soulève cependant une problématique importante de la modélisation interactive de structures ramifiées : un grand espace de paramètres, nécessaires à la définition de modèles de plantes, semble être difficile à appréhender par un utilisateur. Nous montrons dans le chapitre suivant comment une approche multi-échelles permet de résoudre ce problème.

Ces auteurs ont aussi abordé différents problématiques liées à la visualisation des modèles générés. L'utilisation de structures hiérarchiques et de points permet un affichage optimisé. L'utilisation de méthodes de rendu non photoréaliste permet de faire ressortir des caractéristiques plus ou moins globales des modèles. Cette deuxième démarche confirme l'intérêt qu'il y a à représenter des structures macroscopiques pour permettre une meilleure perception de la géométrie d'une plante (problématique que nous avons abordée au chapitre 5). Dans ce travail, des représentations simplifiées de type croquis sont calculées tandis que nous proposons une solution en utilisant des enveloppes 3D.

Fig. 8-8 – Une scène virtuelle représentée sous forme de croquis.

Chapitre 9

Génération multi-échelles de modèles de plantes

Ce chapitre présente un travail effectué en collaboration avec l'équipe du professeur Przemyslaw Prusinkiewicz de l'université de Calgary et qui a donné lieu à une publication à la conférence Eurographics 2003 [Boudon et al., 2003].

En infographie, les modèles de plantes sont généralement construits par des méthodes procédurales, qui génèrent des structures ramifiées complexes à partir d'un ensemble restreint de données fournies par l'utilisateur. Les modèles procéduraux de plantes peuvent être divisés en deux classes, les modèles construits du *local vers le global* et ceux construits inversement du *global vers le local* [Prusinkiewicz et al., 2001]. Dans l'approche local-vers-le-global, l'utilisateur caractérise les composants d'une plante, et l'algorithme intègre ces composants dans une structure complète. Cette approche est particulièrement bien adaptée à la modélisation et la simulation du développement d'une plante dans le contexte biologique. La forme globale de la plante est, dans ce cas, une propriété émergente de ces modèles et est difficilement contrôlable. Une exception notable est la modélisation des topiaires [Prusinkiewicz et al., 1994], qui est basée sur la simulation de la réponse d'une plante à l'élagage. Dans l'approche global-vers-local, au contraire, l'utilisateur caractérise des aspects globaux de la forme de la plante, tels que sa silhouette globale ou la densité de la distribution de branches. L'algorithme de modélisation utilise ces informations pour inférer les détails de la structure de la plante. L'approche global-vers-local fournit un contrôle plus direct et intuitif des aspects visuellement importants de la forme d'une plante, et est donc préférable pour des applications où le résultat visuel est de première importance. Ces applications incluent notamment l'inférence de la structure de plantes à partir de photographies [Shlyakhter et al., 2001] et le design interactif de modèles de plantes, qui est le sujet de ce chapitre.

L'utilisation d'information globale dans le design de modèles de plantes se retrouve dans le travail de Reeves et Blau [Reeves and Blau, 1985]. Avec leur méthode, l'utilisateur spécifie une surface de révolution qui définit la silhouette globale d'un arbre. L'algorithme génératif utilise cette information pour déterminer la longueur des branches d'ordre 1. La technique de Reeves et Blau a été successivement améliorée par Weber et Penn [Weber and Penn, 1995], Lintermann and Deussen avec XFrog (voir chapitre 8) et Prusinkiewicz *et al.* (voir section 7.6.3), qui introduisent des paramètres numériques et des fonctions définies graphiquement pour contrôler la densité de branches, la progression des angles de branchement, le changement de diamètre ou de courbure des axes, et d'autres caractéristiques du modèle.

Dans ces approches la sélection des paramètres (numériques, fonctionnels ou composés, tel que l'enveloppe de la plante entière) qui peuvent être directement contrôlés a un rôle important. Si le nombre de ces paramètres est petit, l'algorithme de modélisation doit alors en réutiliser certains quand il génère différentes parties de la structure. Reeves et Blau avaient déjà remarqué cela et notèrent que les branches des ordres les plus grands ont "certains paramètres

hérités des parents" dans leurs modèles [Reeves and Blau, 1985]. Une réutilisation judicieuse des paramètres permet effectivement de contrôler des modèles dont la structure est fortement répétitive, comme par exemple celle des fougères, de nombreuses inflorescences, et de jeunes arbres [Prusinkiewicz et al., 2001]. D'autres modèles de plantes nécessitent toutefois un contrôle direct de chaque composant pour capturer leurs particularités distinctes ; ce qui implique donc d'utiliser un ensemble de paramètres plus grand. Malheureusement, la manipulation de tels ensembles produit un problème en soit : c'est un processus pénible dans lequel l'utilisateur peut être facilement dépassé par le nombre important de paramètres et peut perdre la maîtrise de leur effet. De plus, avoir beaucoup de paramètres rend plus difficile le contrôle des caractéristiques globales des modèles. On peut trouver une analogie avec l'édition interactive des courbes et des surfaces, où un nombre important de points de contrôle rend difficile le contrôle de la géométrie globale. Une solution connue à ce problème est l'édition multi-résolution, introduite par Forsey et Bartels [Forsey and Bartels, 1988], puis généralisée dans différents contextes mathématiques (e.g. [Stollnitz et al., 1996, Zorin et al., 1997]) ou applicatifs, tel que la modélisation de chevelures [Kim and Neumann, 2002]. Dans ce chapitre, nous étendons le paradigme de modélisation multi-résolution à la conception de modèles de plantes.

Nous utilisons une structure multi-échelles de représentation de la plante qui nous sert à stocker les différents paramètres du modèle. Les valeurs de ces paramètres sont décomposées suivant différents niveaux de détails, donnant à l'utilisateur la possibilité de les modifier plus ou moins localement.

Les opérations d'édition sont effectuées avec plusieurs outils logiciel, tel que des explorateurs (browsers) de la structure de la plante et des éditeurs des différents paramètres. Un composant particulièrement important est l'*éditeur de silhouette*, qui permet de manipuler directement des silhouettes 3D, potentiellement asymétriques, des systèmes ramifiés.

Le graphe de décomposition sert de source pour les valeurs des paramètres utilisés par le modèle procédural. Nous utilisons des algorithmes génératifs de type global-vers-local, dont la structure générale est décrite par Prusinkiewicz *et al.* [Prusinkiewicz et al., 2001], implémentés avec la plate-forme logicielle de modélisation `L-studio` [Prusinkiewicz et al., 2000]. Le langage de modélisation `L+C` [Karwowski, 2002] basé sur les L-systèmes, étendu par des fonctions permettant d'accéder et de manipuler le graphe de décomposition, permet à l'utilisateur de redéfinir ou de modifier l'algorithme génératif si cela est nécessaire pour un modèle particulier.

Cette méthode est illustrée par la création de modèles d'arbres bonsaïs. Les bonsaïs réels ont souvent des formes particulièrement irrégulières, dont les irrégularités inhérentes au développement biologique ont été accentuées par l'intervention humaine. Les modèles de bonsaïs sont des exemples typiques de besoin en manipulations pour des formes de plantes caractérisées par un grand nombre de paramètres.

9.1 Structure multi-échelles de la plante

9.1.1 Représentation de la plante

Pour représenter nos modèles de plantes, nous utilisons le formalisme de représentation géométrique multi-échelles, présenté dans le chapitre 4, basé sur les graphes multi-échelles [Godin and Caraglio, 1998]. Dans ce travail, nous considérons des graphes multi-échelles construits en utilisant la structure hiérarchique d'une plante induite par son ordre de ramification (Figure 9-1). Les nœuds du graphe sont utilisés pour contenir les différents paramètres qui décrivent les parties de l'arbre aux différents niveaux de la hiérarchie, et ainsi à différents niveaux de détails. Dans ce processus de design interactif d'un modèle de plante, les paramètres décrivant les branches des ordres les plus grands sont initialement hérités des paramètres décrivant la plante entière. L'utilisateur introduit de la diversité dans la structure générée en cassant ce système d'héritage de paramètres et en éditant les paramètres d'un composant sélectionné au niveau

choisi de la hiérarchie. De cette manière, la plante est graduellement raffinée avec une expansion minimale du jeu de données. Dans cette approche, un accent particulier est mis sur le graphe de décomposition contenu dans le graphe multi-échelles.

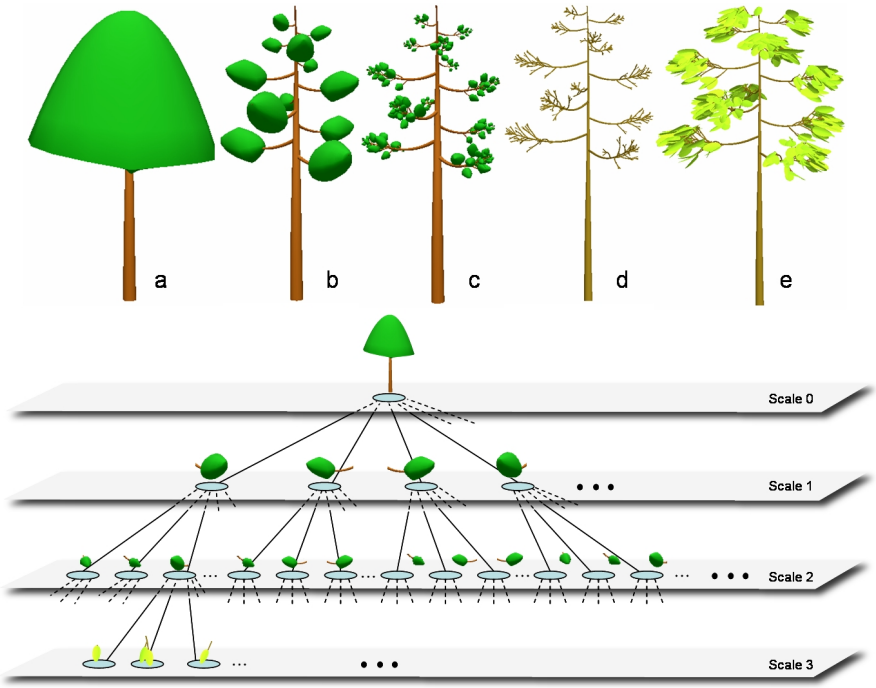


FIG. 9-1 – En haut : Représentation approximative de l'échelle 0 à 3 (a à d) et la représentation finale (e). En bas : Le graphe multi-échelles correspondant.

9.1.2 Les paramètres

Pour permettre de définir flexiblement la portée de chacun des paramètres stockés dans le graphe, nous avons défini différents types de paramètres. Tous les paramètres sont identifiés par un nom, qui fournit un lien avec le programme génératif, et contiennent une valeur qui peut être de différente nature (scalaire, fonction, matériel, surface, *etc.*). Le *type* d'un paramètre spécifie son statut d'héritage. Un paramètre peut avoir une valeur définie *explicitement* pour un nœud donné du graphe, *héritée* d'un autre nœud, ou définie *relativement* par rapport à la valeur héritée. La valeur d'un paramètre peut être la propriété *privée* d'un nœud, et donc ne peut pas être héritée, ou *publique* et donc héritable. Finalement, la valeur d'un paramètre peut être *partagée* par plusieurs nœuds qui existent au même niveau du graphe de décomposition. Le mécanisme de partage s'applique seulement aux valeurs de paramètres qui sont explicitement définies (*i.e.* ne sont pas héritées).

La distinction entre les paramètres privés et publics affecte le mécanisme d'héritage de la manière suivante. Considérons la situation dans laquelle un paramètre particulier d'un nœud donné est hérité, le paramètre correspondant du nœud parent est privé, et celui du nœud grand-parent est public. La valeur de paramètre du nœud courant sera alors héritée du grand-parent plutôt que du parent. Plus généralement, le paramètre hérité reçoit sa valeur du premier nœud supérieur dans le graphe de décomposition dans lequel le paramètre correspondant est déclaré public. Par définition, tous les paramètres de la racine du graphe sont publics.

9.1.3 Simulation multi-échelles

Le processus de modélisation commence par la spécification d'un algorithme génératif (donné par l'utilisateur) dans le langage `L+C` basé sur les L-systèmes [Karwowski, 2002]. Nos modèles sont construits en utilisant le paradigme global-vers-le-local, et utilisent donc des règles de décomposition plus que des règles de croissance. Ces règles sont mieux décrites par les grammaires de Chomsky (qui appliquent séquentiellement les productions) que par les L-systèmes (application parallèle des productions) [Prusinkiewicz et al., 2001], Cependant, nous continuons à utiliser le terme "L-systèmes", car les L-systèmes et les règles de production de Chomsky peuvent être combinés dans le même modèle, rendant la distinction difficile.

Pendant sa première exécution, l'algorithme fait appel aux fonctions qui créent le graphe multi-échelles et définissent les paramètres pour certains nœuds. L'algorithme se sert de ces paramètres pour générer les différents composants aux différentes échelles, et spécifie, lors de la première exécution, leurs valeurs par défaut dans le graphe. Les nœuds dont les valeurs de paramètres ne sont pas explicitement définies héritent leurs valeurs des nœuds parents. Plus spécifiquement, si l'exécution initiale de l'algorithme définit les valeurs de paramètres seulement pour la racine du graphe de décomposition, tous les nœuds du graphe partageront le même ensemble de valeurs.

Une fois le graphe multi-échelles initial créé, les paramètres qu'il contient peuvent être interactivement édités. Le graphe multi-échelles est passé à un processus parallèle qui permet à l'utilisateur par différents explorateurs et éditeurs de modifier cette structure.

Ensuite, l'algorithme génératif doit être re-exécuté pour reconstruire la structure de la plante, en fonction des paramètres mis-à-jour. Dans cette troisième phase, l'algorithme accède aux valeurs stockées dans le graphe multi-échelles pour regénérer les différents composants (et ainsi reconstruire un nouveau graphe multi-échelles). Pour associer les nœuds du graphe avec les systèmes ramifiés spécifiques de la structure générée, les systèmes ramifiés et les nœuds sont identifiés par leurs chemins jusqu'à la racine dans l'arbre de décomposition. Le chemin d'un nœud est récursivement défini par trois composants :

- le *chemin* du nœud parent ;
- la *position* normalisée du système ramifié sur l'axe principal du père ;
- un *nombre* identifiant le système ramifié, si plusieurs systèmes ramifiés sont attachés au même point sur l'axe porteur.

Une fois la structure regénérée, l'utilisateur a donc un retour sur les modifications qu'il a effectuées (généralement de manière interactive en fonction de la complexité du modèle). Il peut alors réitérer le processus en éditant de nouveau les paramètres du graphe multi-échelles, jusqu'à obtenir le modèle souhaité.

Nous présentons, dans la section suivante, les différents outils que nous avons développés pour explorer et éditer cette structure multi-échelles au cours de la création d'un modèle de plante.

9.2 Navigation et édition d'une structure de plante

Dans cette approche, un modèle de plante est généré algorithmiquement, avec les paramètres stockés dans le graphe multi-échelles. Comme le nombre de nœuds dans le graphe multi-échelles peut être grand, des outils sont nécessaires pour explorer commodément le graphe et accéder aux paramètres associés aux nœuds. Nous avons développé deux outils dans ce but : l'*explorateur de graphes de décomposition* et l'*explorateur de structures ramifiées*.

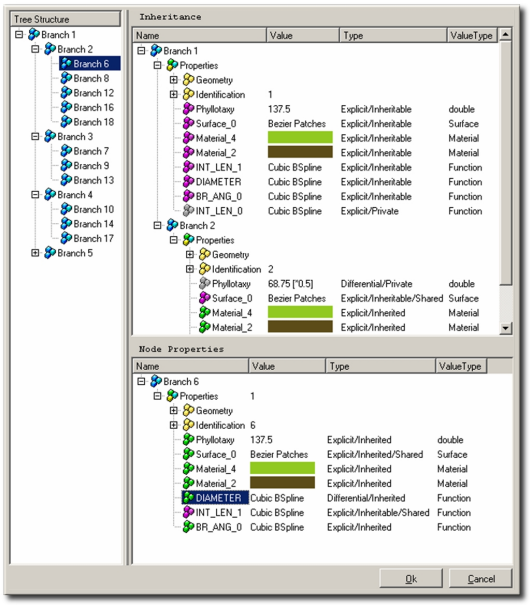


FIG. 9-2 – L'explorateur de graphes de décomposition.

9.2.1 L'explorateur de graphes de décomposition

L'explorateur de graphes de décomposition se présente à l'écran sous la forme d'une fenêtre en trois parties (Figure 9-2). Le cadre de gauche représente la structure hiérarchique du graphe. Il est visuellement similaire aux outils d'exploration de fichier du système Windows, et fournit des opérations similaires d'expansion/contraction pour contrôler quelle partie du graphe est montrée. Cette partie permet aussi de sélectionner un nœud spécifique dans le graphe de décomposition.

Les attributs du nœud sélectionné sont affichés dans le cadre en bas à droite. Les aspects du statut d'héritage sont visualisés par la colonne type et par les couleurs des icônes associées à chaque paramètre et nœud.

Pour n'importe quelle valeur de paramètre du modèle, l'utilisateur a besoin de connaître le nœud duquel elle provient si elle est héritée. Cette information est disponible dans le cadre en haut à droite, qui affiche les paramètres de tous les nœuds dans le chemin de la racine au nœud courant sélectionné. En inspectant quels nœuds sont publics ou privés, l'utilisateur peut identifier les sources des valeurs de paramètres hérités par le nœud courant.

La définition et la redéfinition du statut d'héritage des nœuds sont des aspects importants du processus de modélisation de la plante. Initialement, tous les nœuds créés héritent leurs paramètres des nœuds supérieurs dans la hiérarchie, éventuellement la racine du graphe de décomposition. En accédant et éditant un paramètre hérité, l'utilisateur créé sa copie et y assigne une nouvelle valeur. De cette manière, le nombre de paramètres contrôlés indépendamment augmente, conduisant à une diversification graduelle des composants du modèle. Par un menu, l'utilisateur peut aussi redonner à un paramètre sa valeur par défaut (celle héritée) et plus généralement changer le statut d'héritage de n'importe quel paramètre. En définissant soigneusement la structure d'héritage du graphe de décomposition, l'utilisateur construit graduellement un jeu de paramètres qui inclut tous les paramètres nécessaires pour capturer la diversité des composants de la plante modélisée, sans inclure de paramètres superflus.

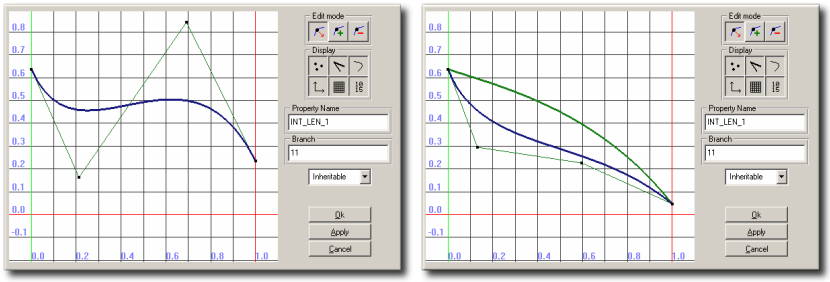


FIG. 9-3 – Editeur de fonction 2D en mode explicite (gauche) et en mode relatif (droite). En mode relatif, la valeur de référence de la fonction est dessinée en vert et la valeur relative, que l'on édite, en bleu.

9.2.2 Les éditeurs de paramètres

Les valeurs des paramètres sont modifiées via des *éditeurs* spécialisés. Un éditeur est ouvert en sélectionnant un paramètre dans la partie affichant le nœud courant. La liste exacte des paramètres, et donc des éditeurs, associés avec les nœuds dépend de l'algorithme génératif sous-jacent. Dans le cas simple des *scalaires*, l'interface de l'éditeur est composée d'une glissière (slider) et d'un champ numérique éditable. Des éditeurs plus évolués sont utilisés pour les paramètres composés du nœud, comme par exemple, un éditeur de *matériel* pour définir les propriétés optiques de la branche, des éditeurs de *courbes* et de *surfaces*, et un éditeur graphique de *fonctions* (Figure 9-3), comme décrit dans [Prusinkiewicz et al., 2000].

Certains paramètres (les scalaires et les fonctions dans la présente implémentation) peuvent être déclarés comme relatifs par rapport à la valeur héritée (Ce sont des paramètres différentiels). Dans ce cas, la valeur du paramètre est une combinaison de la valeur héritée d'un autre nœud et de la valeur définie localement. Dans le cas des scalaires, cela signifie que la valeur réelle du paramètre est obtenue en appliquant une différence définie localement (combinaison additive), ou en prenant un fraction définie localement (combinaison multiplicative) à la valeur héritée. Dans le cas des fonctions, les mêmes combinaisons sont appliquées en prenant la somme ou le produit des fonctions héritées et définies localement. Pour faciliter le processus d'édition, l'éditeur de fonctions peut afficher simultanément la valeur héritée et la valeur modifiée (Figure 9-3, à droite).

9.2.3 L'explorateur de structures ramifiées

L'explorateur de structures ramifiées (Figure 9-4) fournit une vue multi-résolution alternative de la structure de la plante. Il utilise une représentation iconique des systèmes ramifiés pour visualiser un niveau choisi de la structure de la plante, et ainsi montre certains de ces aspects géométriques, mais ne montre pas explicitement la relation d'héritage dans le graphe de décomposition.

Cet explorateur est construit autour de la notion de *silhouette de système ramifié*, qui décrit l'axe principal et le contour (l'enveloppe) du système ramifié qu'il contient. L'explorateur arrange ces silhouettes en une structure ramifiée qui représente la géométrie de la plante à l'échelle sélectionnée par l'utilisateur. Ainsi, en plus des silhouettes elles-mêmes, l'explorateur affiche la taille et l'orientation des branches (définies par les angles phyllotaxiques et de branchement).

L'utilisateur peut changer la taille et l'orientation d'un système ramifié en le sélectionnant et en le manipulant avec la souris (pour une discussion générale sur la manipulation interactive des branches, voir [Power et al., 1999]). L'utilisateur peut aussi invoquer un éditeur externe pour le nœud sélectionné. Cet éditeur de silhouettes, particulièrement important dans le contexte de l'édition multi-échelles, est présenté dans la section suivante. Utilisés conjointement, l'explora-

teur de structures ramifiées et l'éditeur de silhouettes fournissent un moyen adapté à l'édition interactive de la géométrie d'une plante.

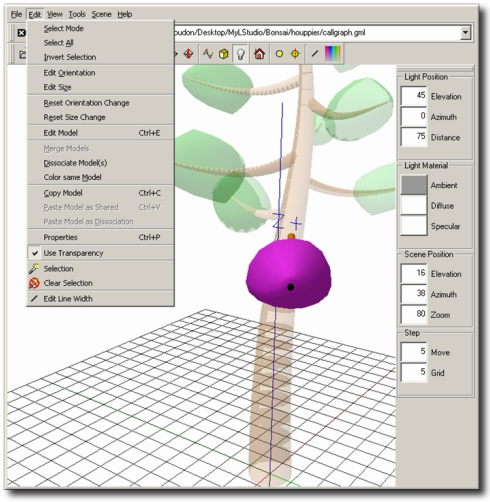


FIG. 9-4 – Une capture d'écran de l'explorateur de structure ramifiée. La plante est représentée à l'échelle 2. L'orientation de l'instance coloriée en pourpre est en cours d'édition. Les autres instances deviennent transparentes, et donnent ainsi à l'utilisateur un focus sur l'opération courante. Le menu Edit affiche toutes les opérations d'édition possibles.

9.2.4 L'éditeur de silhouettes

La géométrie globale d'un système ramifié est spécifiée par sa silhouette (Figure 9-5, voir aussi Figure 9-1). La silhouette consiste en une courbe 3D (polyline, courbe de Bézier ou B-spline), qui spécifie l'*axe* de la silhouette. La silhouette inclue aussi une *enveloppe* qui représente les longueurs des branches latérales. La littérature en modélisation des plantes contient une grande variété de modèles d'enveloppes asymétriques pour représenter la couronne d'un système ramifié (voir le chapitre 1). Nous avons choisi d'implémenter le modèle d'enveloppe proposé par Cescatti [Cescatti, 1997], qui a été défini pour représenter de façon souple une grande variété de couronnes d'arbres de manière intuitive.

L'enveloppe de Cescatti est définie par six points de contrôle et deux facteurs de formes (voir section 1.3.1). Ces différents paramètres ainsi que les points de contrôle de la courbe sont interactivement manipulables via l'éditeur de silhouette, permettant à l'utilisateur un design flexible de la forme globale d'un système ramifié.

9.3 Contraintes multi-échelles

Le mécanisme d'héritage induit des contraintes inter-échelles (voir la section 4.4) sur les nœuds du graphe : il établit des relations simples entre les valeurs des paramètres des nœuds à différentes échelles du graphe multi-échelles. Des relations plus complexes, induites par la signification des paramètres, peuvent aussi exister. La relation entre la forme d'une enveloppe d'un système ramifié et la taille des silhouettes associées avec les branches latérales (Figure 9-6) en est un exemple. Clairement, on ne peut pas modifier la silhouette globale sans affecter la taille des silhouettes des systèmes ramifiés latéraux, et vice et versa. L'information dans le nœud père du graphe multi-échelles est reliée à celle contenue dans les nœuds enfants, car le père et les enfants décrivent le même système ramifié.

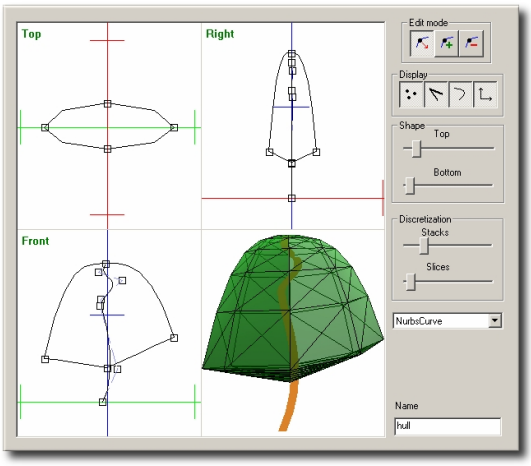


FIG. 9-5 – Editeur de silhouettes

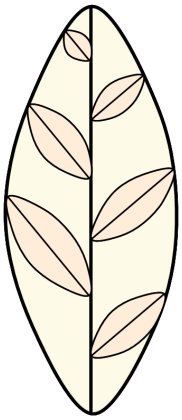


FIG. 9-6 – Relation entre silhouettes à deux échelles différentes de la hiérarchie de la plante. La taille des silhouettes à l'échelle la plus fine est déterminée par la forme de la silhouette à l'échelle la plus globale.

Comme discuté dans la section 4.4.5, de telles contraintes peuvent être satisfaites en utilisant soit une approche montante, du local vers le global, soit une approche descendante, du global vers le local. Ces termes qualifient la direction dans laquelle l'information contrainte se propage sur le graphe multi-échelles. La direction descendante est plus adaptée au design interactif de plantes, qui commence par la silhouette générale d'une plante, et la raffine graduellement [Prusinkiewicz et al., 2001].

Pratiquement, le problème revient à instancier une silhouette E_c à l'intérieur de la silhouette parent E_p. Pour cela, nous ajoutons un point de contrôle T supplémentaire à la description de la silhouette, comme le montre la Figure 9-7. Le vecteur $\overrightarrow{BT}$ qui connecte la base B de la silhouette E_c au point T est utilisé pour orienter la silhouette dans l'espace et déterminer sa taille. Le point B est d'abord positionné sur le point de branchement spécifié par l'algorithme génératif. Ensuite, le vecteur $\overrightarrow{BT}$ est aligné avec la direction de la branche donnée par les angles de branchement et de phyllotaxie. Finalement, la silhouette E_c est dimensionnée de manière à ce que le point T soit placé sur l'enveloppe de la silhouette parente. Optionnellement, un repère associé à la silhouette peut aussi être utilisé pour contrôler et déterminer l'orientation de celle-ci

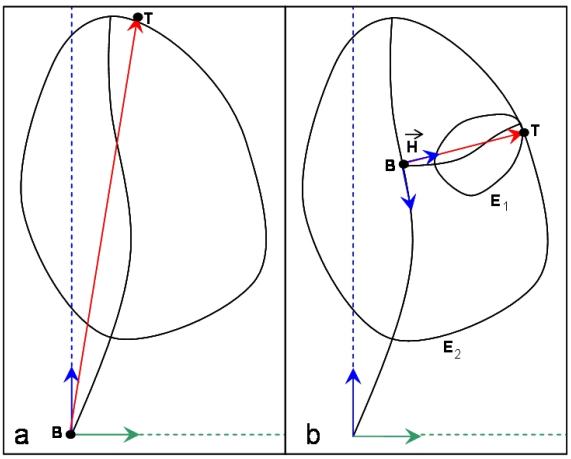


FIG. 9-7 – Instanciation de la silhouette enfant E_c dans la silhouette parent E_p. a) La forme des deux silhouettes. b) Le résultat de l'instanciation.

lors d'une instanciation.

Du point de vue de l'implémentation, la résolution de telles contraintes exige donc de déterminer, pour un vecteur $\overrightarrow{H}$ et un point B donnés, le point T pour calculer la distance $\left\|\overrightarrow{BT}\right\|$. Ceci se résout par un lancer de rayon. Une technique classique permettant d'accélérer les calculs liés au lancer de rayon est l'utilisation d'un octree pour hiérarchiser la géométrie dans l'espace (voir section 3.3.2). Cet octree permet, lors du lancer de rayon, de minimiser le nombre de polygones à examiner. Pour chaque modèle de silhouette, un octree est précalculé avant la simulation et peut être réutilisé pour les différentes instances de ce modèle.

Les contraintes inter-échelles discutées auparavant donnent la relation entre des paramètres à différentes échelles dans le graphe multi-échelles, mais n'affectent pas sa structure (topologie). Une situation plus complexe peut se produire quand l'utilisateur manipule la densité de branches construites sur l'axe d'un système ramifié. La fonction de densité associée avec le nœud parent détermine le nombre de nœuds enfants, et donc affecte la structure du graphe multi-échelles. Dans cette approche, cette structure est générée algorithmiquement ce qui signifie que l'algorithme génératif doit être re-exécuté pour satisfaire les contraintes liées aux densités de branches. Le couplage entre l'algorithme génératif et la manipulation interactive des paramètres est schématisé dans la figure 9-8.

9.4 Le processus de modélisation

9.4.1 Algorithme génératif

Nous présentons ici le canevas des algorithmes utilisant un arbre de décomposition pour stocker les valeurs de leurs paramètres. Cet algorithme L-système utilise les conventions du langage `L+C`. Nous utilisons cependant ici les noms des fonctions du langage `cpfg`, qui ont toutes un équivalent en `L+C`, et sont plus compacts. Cet algorithme combine l'utilisation du graphe multi-échelles avec les techniques développées dans [Prusinkiewicz et al., 2001] pour, par exemple, contrôler globalement la densité des branches et les angles de branchement d'un système ramifié ainsi que celle pour contrôler le tracé d'un axe par la tortue suivant une courbe de référence (ici l'axe de la silhouette). Pour un rappel sur ces techniques, voir la Section 7.6.3.

Différentes fonctions relatives à l'utilisation du graphe multi-échelles sont utilisées. Les fonctions

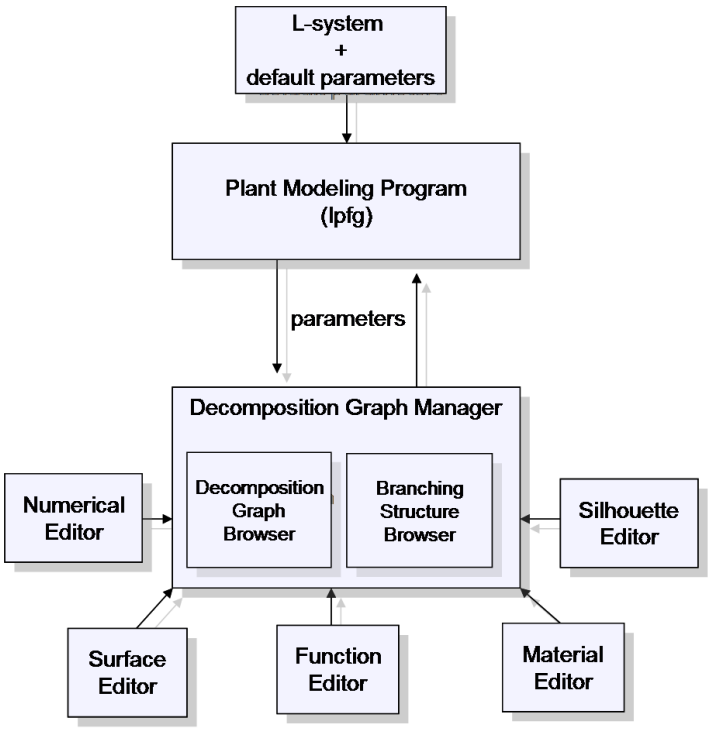


FIG. 9-8 – Les interactions des différents composants pendant le processus de modélisation.

- `node(chemin_parent, position, id, silhouette,` $\overrightarrow{h}$`,` $\overrightarrow{u}$`, longueur)` et
- `initialNode(silhouette,longueur)`

permettent d'accéder ou de créer (s'ils n'existent pas) respectivement les nœuds et la racine du graphe. Les différents paramètres d'appel de ces fonctions permettent d'identifier les nœuds et d'instancier un modèle de silhouette suivant les méthodes décrites précédemment. `position` donne la position sur l'axe principal du parent, `id` permet de distinguer différents nœuds créés à la même position, `silhouette` est l'identifiant du modèle de silhouette, $(\overrightarrow{h},\overrightarrow{u})$ l'orientation de la silhouette et `longueur` la longueur de l'enveloppe sur sa direction principale $\overrightarrow{BT}$ (cette longueur nécessite de calculer la distance à l'enveloppe parent dans la direction h, ce qui est donné par la fonction `distance` décrite plus loin). Le chemin identifiant le nœud est retourné. Pratiquement, ce chemin est résumé par un identifiant unique lors de l'exécution de l'algorithme génératif.

Les fonctions `setFunction(chemin,func_id)`, `setParameter(chemin,name,value)`, etc., permettent d'associer des paramètres de différents types aux nœuds du graphe. Dans l'algorithme présenté ici, par défaut, seule la racine reçoit des paramètres qui seront ainsi hérités par tous les autres nœuds du graphe. Si les paramètres existent déjà, notamment dans le cas d'une mise à jour, ces fonctions sont sans effet. Dans cet exemple, deux fonctions contrôlant respectivement la densité de branches portées par l'axe principal du système ramifié et les angles de branchement des branches latérales sont associées à la racine (ligne 11), ainsi qu'un paramètre scalaire, contrôlant la phyllotaxie des axes. Pour récupérer la valeur de ces paramètres dans le graphe pour les différents nœuds, les fonctions `func(chemin,func_id)`, `param(chemin,nom)`, *etc.*, sont utilisées (ligne 18 et 23).

Finalement, certaines caractéristiques de la géométrie des silhouettes sont données par trois fonctions :

- $\tan\Delta$`(chemin, pos,` $\overrightarrow{u}$`,` $\overrightarrow{l}$`,`Δu`,`Δl`)` calcule la tangente de l'axe principal de la silhouette à la position *pos* et retourne les valeurs des rotations nécessaires pour orienter la tortue

dans la direction de la tangente en partant du repère $[\overrightarrow{u} \times \overrightarrow{l}, \overrightarrow{u}, \overrightarrow{l}]$ (ligne 17) ;

- `distance`(`chemin`,`pos`,$\overrightarrow{dir}$) calcule, pour la silhouette du nœud `chemin`, la distance du point positionné en `pos` sur l'axe principal à l'enveloppe dans la direction $\overrightarrow{dir}$ (ligne 34) ;
- `axisLength`(`chemin`) donne la longueur de l'axe principal de la silhouette du nœud `chemin` (ligne 38).

L'algorithme 9.4.1 présenté ici construit récursivement les systèmes ramifiés. La première règle, appelée `P`, (ligne 9 à 13) initie le graphe multi-échelles en donnant les valeurs par défaut des paramètres à la racine du graphe. La règle `A` (ligne 15 à 29) construit récursivement l'axe principal d'un système ramifié suivant la courbe de la silhouette (ligne 17, 22 et 26). Cette règle initie, en fonction de la fonction de densité $\mathcal{D}$, les systèmes ramifiés latéraux par un appel à la règle `N` (ligne 23). La fonction $\mathcal{D}$, stockée dans le graphe et accessible depuis celui-ci (ligne 11 et 18), peut donc être redéfinie pour chacun des systèmes ramifiés. La règle `N` (ligne 31 à 43) procède à l'instanciation d'un nœud dans le graphe suivant la technique décrite dans la section 9.3. A la ligne 34, la distance du point de branchement à l'enveloppe est calculée. Si cette longueur est assez grande et que l'échelle de décomposition est inférieure à une constante, un nœud et sa silhouette sont instanciés (ligne 37). La géométrie détaillée de ce nouveau système ramifié est alors générée grâce à la règle `A` (ligne 38).

Cet algorithme est un exemple d'utilisation du graphe multi-échelles et des silhouettes de systèmes ramifiés. Cette approche générique permet à d'autres paramètres d'être passés au graphe et de personnaliser l'algorithme en fonction du type de plante. L'utilisation du graphe multi-échelles et des silhouettes guide cependant l'algorithme vers une forme proche de celle présentée ici. Pour les modèles de bonsaïs présentés plus bas, les algorithmes utilisés diffèrent peu de celui-ci si ce n'est par l'intégration d'une fonction définissant les diamètres des axes.

9.4.2 Le problème de découplage

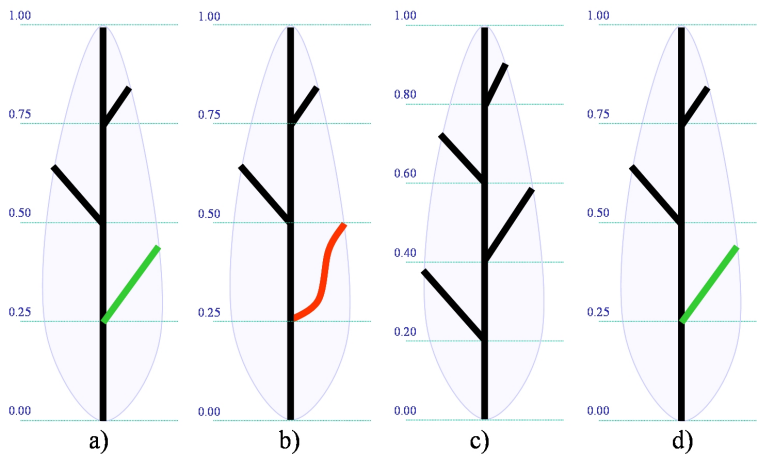


FIG. 9-9 – Effet de découplage. a) La structure de la plante, avec une forme par défaut pour la branche du bas. b) La forme de la branche du bas (position 0.25) a été manuellement modifiée. c) La densité de branches est augmentée. d) La valeur d'origine pour la densité de branches est rétablie, et toutes les branches sont reconstruites avec une géométrie par défaut (ligne droite).

Stocker les paramètres d'un algorithme dans un arbre de décomposition amène un certain nombre de problèmes. Comme conséquence de la manipulation de certains paramètres, les chemins assignés aux branches pendant la ré-exécution de l'algorithme génératif peuvent différer des chemins stockés dans le graphe multi-échelles. Cela se produit, par exemple, si l'utilisateur

Algorithme 4 L-systèmes qui utilise le graphe multi-échelles comme source de valeur de paramètres.

```
1.  #define ℒ 100        /* longueur du tronc */
2.  #define Δs 2         /* pas de la tortue */
3.  #define 𝒮 1          /* id de la silhouette utilisée*/
4.  #define ℬ 2          /* id de la fonction d'angle de branchement */
5.  #define 𝒟 3          /* id de la fonction de densité*/
6.  #define Ψ 137.5      /* valeur par défaut de la phyllotaxie */
7.  #define ℳ𝒮 3         /* Echelle maximale de décomposition */
8.  Axiom : P
9.  P : {
10.     path = initialNode(𝒮,ℒ);
11.     setFunction(path,ℬ); setFunction(path,𝒟); setParameter(path,"Phyllotaxy",
Ψ);
12.     produce A( 0, 0, 0, 0, ℒ, path ) ?U(0⃗) ?L(0⃗);
13. }
14. /// Apex de l'axe principal d'un système ramifié
15. A( s, a, φ, o, l, path ) > ?U(u⃗) ?L(l⃗) : {
16.     if( s <= l) {
17.         tan Δ(path, s/l, u⃗, l⃗, Δu, Δl);
18.         s += Δs; a += Δs/func(path, 𝒟, s/l);
19.         if(a > 1){
20.             a− = 1;
21.             Δφ = param(path ,"Phyllotaxy");
22.             produce +(Δl)&(Δu)F(Δs)
23.                         [/(φ)+(90*func(id,ℬ,s/l))N(s/l,o + 1,path) ?H(0⃗) ?U(0⃗)]
24.                 A(s,a,φ + Δφ,o,l,path);
25.         }
26.         else produce +(Δl) &(Δu) F(Δs) A(s,a,φ,o,l,path);
27.     }
28.     else produce;
29. }
30. // Créer un nouveau noeud du graphe d'ordre "o" à la position "u" sur l'axe dans le complexe
"c_path"
31. N( u, o, c_path ) > ?H(h⃗) ?U(u⃗) :{
32.     // demande la distance du point de branchement (position u sur l'axe de c_path)
33.     // à l'enveloppe de la silhouette du noeud c_path dans la direction h⃗.
34.     l = distance(c_path,u, h⃗);
35.     if(l > 0) {
36.         if( o < ℳ𝒮 ){
37.             path = node(c_path, u, 0, 𝒮, h⃗, u⃗, l);
38.             produce A(0,0,0,o + 1, axisLength(path),path) ?U(0⃗) ?L(0⃗);
49.         }
40.         else produce FinalBranchAndLeaf(l);
41.     }
42.     else produce;
43. }
44. FinalBranchAndLeaf(l) ... /// Code des branches finales
```

a modifié la fonction qui définit la densité de branches le long d'un axe (*c.f.* Section 9.3). Dans le cas d'un tel découplage, l'algorithme génératif supprime les nœuds qui ne sont plus utilisés, et ajoute de nouveaux nœuds dans le graphe multi-échelles s'ils n'ont pas de correspondance. Ces ajustements peuvent avoir des conséquences contre-intuitives pour l'utilisateur. Considérons l'exemple illustré dans la Figure 9-9. La fonction de densité de branches pour la structure de plante initiale (a) est définie pour que l'axe principal ait trois branches latérales. Les positions normalisées de ces branches latérales sont respectivement 0.25, 0.5 et 0.75. Par défaut, toutes les branches sont droites. Supposons maintenant que l'utilisateur change la forme de la branche du bas pour lui donner une allure plus courbée, comme illustré dans la Figure (b). Ensuite l'utilisateur change la densité de branches de l'axe principal, augmentant le nombre de branches latérales à quatre. Quand l'algorithme régénère la structure de la plante, il essaye d'obtenir les valeurs de paramètres pour les branches dont les positions normalisées sont 0.2, 0.4, 0.6 et 0.8. Ces chemins, toutefois, ne correspondent à aucun des nœuds existant dans le graphe multi-échelles. En conséquence, de nouveaux nœuds sont créés pour toutes les branches latérales, tandis que les anciens nœuds sont retirés du graphe. Des valeurs par défaut sont assignées aux nouveaux nœuds, ce qui donne la structure représentée dans la Figure (c). La forme de l'axe associé au nœud en 0.25 est définitivement perdue. Et donc, même si la valeur d'origine de la densité de branche est rétablie, l'algorithme ne redonne pas sa forme courbée à la branche du bas (Figure d).

Le problème décrit ci-dessus peut être attribué au fait que la gestion des paramètres est découplée de l'algorithme qui les utilise pour construire la plante. Nous percevons cela comme un problème fondamental : pour interagir avec la plante, nous personnalisons chacune des branches, permettant ainsi de les sélectionner et les modifier. Malheureusement, il n'existe pas de méthode robuste pour maintenir l'identité des branches à travers diverses modifications qui peuvent les déplacer, voir même les retirer temporairement de la structure. En pratique, nous réduisons l'impact d'un tel problème en définissant d'abord la distribution de branches, et en appliquant ensuite des modifications à leurs formes.

9.5 Résultats

Nous avons appliqué notre méthode pour modéliser un certain nombre d'arbres bonsaïs. Ces arbres représentent un challenge à modéliser à cause de leurs structures hautement irrégulières. A l'image des arbres bonsaïs réels qui sont le résultat de l'interaction entre le développement biologique et l'intervention humaine, nos modèles sont le résultat de l'interaction entre l'algorithme génératif basé sur la biologie et les manipulations interactives.

Les résultats sont présentés sur les Figures 9-11 à 9-15. Pour référence, nous montrons aussi certaines des plantes réelles que nous avons essayé de modéliser. Tous les modèles ont été créés en approximativement trois heures.

Sur un PC avec un processeur 1 GHz Pentium III, le processus de génération des modèles détaillés de bonsaïs montrés en Figures 9-11, 9-14, 9-13 et 9-15 ont pris entre 1 et 2.5 secondes. Les modèles de la Figure 9-12 ont été les plus longs à générer (10 et 12.5 secondes) à cause du grand nombre d'aiguilles (modélisées par des cylindres généralisés).

Nous présentons à la figure 9-10 un extrait du processus de construction d'un de ces modèles. Cette construction est faite par une approche global vers le local. La structure initiale utilisée est construite avec les mêmes valeurs de paramètres à toutes les échelles. Elle est d'abord éditée à des échelles macroscopiques (Figure 9-10.a). L'édition se fait ensuite sur des aspects de plus en plus détaillés de la structure. La forme, l'orientation et les dimensions des silhouettes de chaque système ramifié sont modifiées. Des fonctions contrôlant la densité de branches latérales et les diamètres des branches peuvent aussi être modifiées. Ces paramètres permettent de modifier rapidement et intuitivement la forme et le nombre de systèmes ramifiés à toutes les échelles.

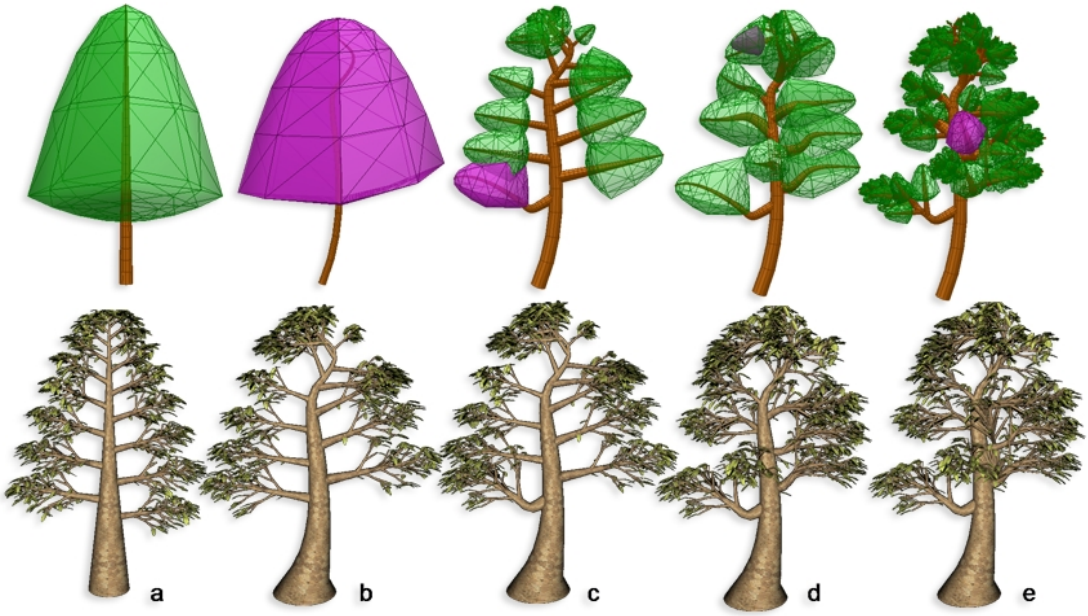


FIG. 9-10 – Construction d'un modèle de bonsai. Le résultat et le modèle sont donnés à la Figure 9-15. (a) Le processus de construction part d'une structure autosimilaire. L'édition est faite en utilisant une approche global vers le local : (b) le modèle est édité globalement, (c et d) puis à l'échelle des systèmes ramifiés d'ordre 1, et finalement, (e) au niveau le plus détaillé.

FIG. 9-11 – Bonsai 1 : style *Bunjinji*, photographie tirée de [Clark and Voynovich, 1989]

FIG. 9-12 – Bonsai 2 : Style *nejikan* (En cascade, tortueux) et Bonsai 3 : Style *fukinagashi* (Battu par les vents)

9.6 Conclusion

Nous avons présenté une approche pour modéliser des plantes basée sur une méthodologie de design du global-vers-le-local, s'inspirant des techniques artistiques. Pour cela, nous avons utilisé un modèle multi-échelles de plante, définit sous la forme d'un arbre de décomposition dont les nœuds représentent les différents systèmes ramifiés de la plante. Les paramètres nécessaires à la construction de la plante sont alors associés aux nœuds de l'arbre de décomposition. Nous proposons l'utilisation de mécanismes d'héritage et de partage de paramètres pour minimiser le nombre total de paramètres nécessaires. L'utilisateur peut ainsi raffiner différents aspects plus ou moins globaux du modèle. Notre approche résout le problème de la gestion et de la navigation à travers un espace complexe de paramètres, qui semblait particulièrement problématique dans le design interactif de plantes [Deussen and Lintermann, 1997, Lintermann and Deussen, 1999].

Pratiquement, nous avons implémenté un ensemble logiciel de design de modèles de plantes basé sur ces paradigmes. Ce système consiste en plusieurs outils qui permettent à l'utilisateur de sélectionner une structure ramifiée à n'importe quel niveau de l'organisation hiérarchique d'une plante, et d'éditer interactivement ses paramètres. Ces outils permettent le design de modèles de plantes relativement rapidement et de manière intuitive. Finalement, nous avons démontré l'efficacité de notre système en modélisant plusieurs bonsaïs, dont la forme tortueuse est réputée difficile à modéliser.

Il reste cependant des améliorations possibles à notre système. Nous pensons que le problème des attributs découplés de l'algorithme procédural mérite d'être exploré plus profondément. Par exemple, nous pouvons associer les nœuds du graphe multi-échelles à des rangs de positions, plutôt que des positions seules, réduisant potentiellement les artefacts de découplage discutés en Section 9.4 (Figure 9-9).

Le mécanisme d'héritage des paramètres que nous avons considéré dans ce chapitre ne relie que des branches à différentes échelles. Cette approche semble particulièrement bien adaptée pour la modélisation de plantes monopodiales, ayant une distinction claire entre les axes parents et les axes latéraux. Cependant, notre mécanisme d'héritage doit être étendu aux relations intra-échelle entre les nœuds pour modéliser des plantes sympodiales.

FIG. 9-13 – Bonsai 4 : Style *sankan* (plusieurs troncs), avec trois branches partant du même point.

FIG. 9-14 – Bonsai 5 : Style *kengai* (Cascade, retombant), photographie tirée de [Clark and Voynovich, 1989]

FIG. 9-15 – Bonsai 6 : Style *chokkan* (Droit, classique), photographie tirée de [Clark and Voynovich, 1989]

Conclusion

Le modèle que nous avons développé dans cette thèse permet d'exprimer la géométrie d'une plante comme une fonction de l'échelle. L'idée d'utiliser une structure hiérarchisée en échelles pour organiser des modèles de plantes se retrouvait dans différents travaux récents, aussi bien dans le domaine de l'analyse de l'architecture des plantes avec les arborescences multi-échelles qu'en infographie où différents travaux organisent leurs modèles géométriques en niveaux de détails pour optimiser le rendu de scènes complexes. Les premiers n'abordent cependant que la topologie et les seconds sont principalement basés sur la notion de décomposition.

Nous avons donc défini un modèle géométrique multi-échelles combinant les deux types d'information, *i.e.* prenant en compte aussi bien les relations entre entités à la même échelle que les relations de décomposition. Les relations entre entités à une même échelle peuvent être utilisées pour vérifier la cohérence du modèle à cette échelle, *i.e.* des contraintes *intra-échelle* peuvent être imposées sur les paramètres des modèles géométriques à une même échelle pour exprimer l'adjacence ou la continuité entre deux entités. De manière similaire, une même entité pouvant être représentée à différents niveaux de détails, des règles de cohérence, sous forme de contraintes *inter-échelles* entre modèles géométriques, peuvent être formalisées. Ce modèle permet un point de vue unifié des différentes représentations géométriques de l'architecture végétale que l'on retrouve dans la littérature.

Nous avons abordé l'algorithmie liée à l'utilisation d'un tel modèle, tel que la navigation, l'édition ou le choix d'une représentation. En utilisant les contraintes intra et inter-échelles et différents algorithmes d'inférence, nous avons montré comment inférer de nouvelles représentations à de nouvelles échelles.

Cette dernière propriété peut être utilisée suivant deux approches.

La première, dite *montante*, consiste à utiliser des informations géométriques détaillées pour calculer des représentations globales. Ces nouvelles représentations approchent globalement les données détaillées en en faisant ressortir les caractéristiques émergentes. Par une première étude statistique, nous avons pu caractériser l'influence du choix du modèle global utilisé sur les résultats obtenus. Dans un contexte biologique, l'étude comparative de la dynamique du modèle global et du modèle détaillé nous permet de caractériser et ainsi de mieux comprendre l'espace occupé par une plante. Dans un contexte plus infographique, nous avons utilisé l'inférence de structures macroscopiques pour calculer des représentations à différents niveaux de détails du bois de structures ramifiées.

Une deuxième approche d'inférence, dite *descendante*, consiste à générer de l'information géométrique détaillée en fonction d'information géométrique globale. Nous l'avons utilisée pour la création d'un système génératif multi-échelles qui permet une manipulation interactive et intuitive des modèles de plantes, même complexes. Les paramètres nécessaires à la génération de la structure 3D de plantes sont associés aux nœuds d'un modèle multi-échelles. Un mécanisme d'héritage des paramètres des nœuds des entités les plus macroscopiques de la plante vers les nœuds des entités les plus fines permet de minimiser le nombre de paramètres et de garder un contrôle global sur le modèle. D'autre mécanismes permettent de raffiner localement les valeurs des paramètres tout en gardant un contrôle global. Cette approche résout le problème de la gestion et de la navigation à travers un espace complexe de paramètres, qui semblait

particulièrement problématique dans le design interactif de plantes. Pratiquement, nous avons réalisé une application logicielle permettant à l'utilisateur de créer et d'éditer interactivement la géométrie d'une structure ramifiée. Cette édition pouvant se faire à n'importe quel niveau de l'organisation hiérarchique, elle permet la conception de modèles de plantes relativement rapidement et de manière intuitive.

Ces deux approches ouvrent plusieurs pistes de recherche et permettent d'envisager de nouvelles applications.

Dans le cadre de l'approche montante, nous donnons des premières pistes pour comprendre comment une forme complexe occupe l'espace. Les premiers résultats, notamment sur le calcul de la dimension fractale d'une plante, doivent être comparés avec d'autres méthodes existantes, telle que la méthode des boîtes, pour être validés. L'application de ces méthodes sur différents types d'architectures végétales permettrait de caractériser et comparer leurs stratégies d'occupation de l'espace. Ce travail nous conduira à aborder le problème du changement d'échelles en modélisation à travers la prise en compte explicite du changement de la géométrie de la plante en fonction de l'échelle.

Dans le contexte de la synthèse d'images, notre méthode de calcul de représentations simplifiées consiste à éliminer ou simplifier la géométrie des branches d'un arbre. Pour la poursuite de ce travail, il serait necessaire d'effectuer une évaluation des résultats obtenus par différents utilisateurs. Leurs perceptions de la simplification apportée permettraient de dégager les critères visuels importants à conserver dans une représentation simplifiée et donc éventuellement d'orienter la méthode vers de nouvelles solutions. Cette méthode ne s'intéresse pour l'instant qu'à la géométrie du bois d'un arbre. Le feuillage, lors de la visualisation, joue aussi un role très important et sa géométrie reste difficile à simplifier de manière convaincante. Pour obtenir des représentations dégradées utilisables en synthèse d'images, il est donc necessaire d'intégrer à notre méthode la simplification du feuillage.

Dans le cadre de l'approche descendante, la méthode de simulation multi-échelles présentée dans cette thèse, apporte une plus grande flexibilité dans le design de modèles de plantes. Dans cette application, nous avons considéré des structures multi-échelles construites sur la hiérarchisation des composants en fonction de leur ordre. Ce travail pourrait être généralisé à des structures multi-échelles plus générales avec la prise en compte de relations inter-échelles, ce qui permettrait de modéliser des structures de plantes plus complexes comme des plantes sympodiales. D'autre part, cette application est confrontée au problème de l'identification des entités au cours des modifications de la structure (voir section 9.4.2). Ce point mérite d'être exploré plus profondément. Une dernière piste ouverte par cette étude concerne la navigation à l'intérieur d'une structure multi-échelles. Nous proposons deux méthodes de représentation de la structure multi-échelles : la première donnant les relations de décomposition (explorateur du graphe de décomposition) et la deuxième une représentation de la plante à une échelle donnée (explorateur de systèmes ramifiés). Des représentations plus sophistiquées, mixant des entités à différentes échelles pour par exemple mettre en relief les parties de la plante en cours d'édition, peuvent être envisagées (vue Fisheye [Noik, 1996]).

Ce modèle géométrique multi-échelles et ses deux approches d'utilisation permettent de créer de manière cohérente différentes représentations à différentes échelles d'un même objet. Ils peuvent donc être utilisés pour adapter la complexité d'un modèle géométrique de plante pour son utilisation dans différents contextes. Pour l'étude de l'interception lumineuse d'une plante par exemple, pour simplifier les calculs, une méthode consiste à assimiler une plante à un volume contenant une distribution uniforme de feuilles (milieu turbide) auquel on peut associer une probabilité d'interception d'un rayon lumineux [Norman and Welles, 1983]. L'hypothèse d'homogénéité est cependant souvent mis à mal dans la réalité : les feuilles d'un arbre sont généralement réparties sur la périphérie de la couronne. Une application de notre modèle pourrait être de calculer, à partir d'une représentation détaillée et les algorithmes présentés au chapitre 5, des représentations volumiques macroscopiques qui optimisent cette hypothèse d'homogénéité

(en calculant des clusters homogènes de feuilles en fonction de l'architecture de la plante).

Finalement, ce modèle multi-échelles permet de relier les modèles de simulation à l'échelle du peuplement et ceux à l'échelle de l'individu. Une première application de ce modèle permettrait d'améliorer les modèles à l'échelle du peuplement. En effet, pour ces modèles, l'interaction entre individus est reflétée par la modification de grandeurs tel que la hauteur et le diamètre de certains individus. A l'échelle détaillée, des modèles d'interaction plus précis ont été élaborés [Greene, 1989, Blaise, 1991, Mech and Prusinkiewicz, 1996] mais, pour des raisons de complexité, ne peuvent prendre en compte que peu d'individus. Ces modèles, couplés avec notre méthode de calcul de géométries macroscopiques, peuvent être utilisés pour déterminer des déformations de la couronne d'un arbre en fonction de son environnement, et ainsi des règles d'interactions macroscopiques plus précises. Ces règles, difficiles à déterminer de la mesure, peuvent ensuite être utilisées pour calculer l'interaction entre individus dans un peuplement complet, et ainsi rendre plus précis les simulations à l'échelle d'un peuplement. Dans une approche inverse, une deuxième application de notre modèle consiste à calculer des représentations détaillées d'un peuplement à partir des données d'une simulation macroscopique et des algorithmes présentés au chapitre 9.

Pratiquement, nous avons développé, pendant cette thèse, une librairie dédiée à la représentation des plantes [Boudon et al., 2001a], contenant les différents algorithmes et modèles géométriques présentés dans cette thèse. Cette librairie a été couplée avec les logiciels AMAPmod et L-studio/lpfg pour la réalisation des différentes applications présentées dans ce document.

Bibliographie

[Abelson and diSessa, 1984] Abelson, H. and diSessa, A. (1984). Turtle geometry. *The MIT Press*.

[Baker, 1979] Baker, J. K. (1979). Trainable grammars for speech recognition. In Klatt, D. H. and Wolf, J. J., editors, *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pages 547–550.

[Barber et al., 1996] Barber, C. B., Dobkin, D. P., and Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4) :469–483.

[Barczi et al., 1997] Barczi, J. F., Reffye, P. d., and Caraglio, Y. (1997). Essai sur l'identification et la mise en oeuvre des paramètres nécessaires à la simulation d'une architecture végétale : le logiciel amapsim. In Bouchon, J., Reffye, P. d., and Barthélémy, D., editors, *Modélisation et Simulation de l'Architecture des Végétaux*, Science Update, pages 205–254. INRA Editions, Paris, France.

[Barnsley, 1988] Barnsley, M. (1988). *Fractals everywhere*. Academic Press, Boston.

[Barnsley et al., 1988] Barnsley, M., Jacquin, A., Malassenet, F., Reuter, L., and Sloan, A. (1988). Harnessing chaos for image synthesis. In *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 131–140. ACM Press.

[Barthélémy et al., 1997] Barthélémy, D., Caraglio, Y., and Costes, E. (1997). Architecture, gradients morphogénétiques et âge physiologique chez les végétaux. In Bouchon, J., Reffye, P. d., and Barthélémy, D., editors, *Modélisation et Simulation de l'Architecture des Végétaux*, Science Update, pages 89–136. INRA Editions, Paris, France.

[Birnbaum, 1997] Birnbaum, P. (1997). *Modalités d'occupation de l'espace par les arbres en forêt guyanaise*. Thèse de doctorat biologie végétale tropicale, Université Paris VI.

[Blaise, 1991] Blaise, F. (1991). *Simulation du parallélisme dans la croissance des plantes et applications*. Thèse de doctorat informatique, Université Louis Pasteur (ULP).

[Blaise et al., 1998] Blaise, F., Barczi, J. F., Jaeger, M., Dinouard, P., and Reffye, P. d. (1998). Simulation of the growth of plants. modeling of metamorphosis and spatial interactions in the architecture and development of plants. In Kunii, T. L. and Luciani, A., editors, *Cyberworlds*, pages 81–109. John Wiley and Sons, Ltd, Tokyo, Japon.

[Bloomenthal, 1985] Bloomenthal, J. (1985). Modeling the mighty mapple. In *Computer Graphics (Proceedings of SIGGRAPH'85)*, volume 19, pages 305–311, San Francisco.

[Bloomenthal, 1995] Bloomenthal, J. (1995). *Skeletal Design of Natural Forms*. PhD thesis, University of Calgary.

[Bloomenthal and Shoemake, 1991] Bloomenthal, J. and Shoemake, K. (1991). Convolution surfaces. In *Computer Graphics, Proceedings of Siggraph'91*, volume 25(4), Las Vegas.

[Bloomenthal and Wyvill, 1990] Bloomenthal, J. and Wyvill, B. (1990). Interactive techniques for implicit modeling. In *Computer Graphics (Proceedings of Symposium on Interactive 3D Graphics)*, volume 24(2), pages 109–116.

[Borchert and Honda, 1985] Borchert, R. and Honda, H. (1985). Control of development in the bifurcating branch system of tabeduia rosea : a computer simulation. *Botanical Gazette*, 145(2) :184–195.

[Boudon, 2000] Boudon, F. (2000). Développement d'un modèle géométrique paramétrique pour la représentation de l'architecture des plantes. DEA, Université Montpellier II. Ecole doctorale Information, Structures, Systèmes, LIRMM.

[Boudon et al., 2001a] Boudon, F., Nouguier, C., and Godin, C. (2001a). Geom module manual. I. user's guide. Document de travail du Programme Modélisation des plantes 3-2001, CIRAD.

[Boudon et al., 2001b] Boudon, F., Nouguier, C., and Godin, C. (2001b). Geom module manual. ii. developper guide. Document de travail du Programme Modélisation des plantes 4-2001, CIRAD.

[Boudon et al., 2003] Boudon, F., Prusinkiewicz, P., Federl, P., Godin, C., and Karwowski, R. (2003). Interactive design of bonsai tree models. In Brunet, P. and Fellner, D., editors, *Computer Graphics Forum (Proceedings of Eurographics 2003)*, volume 22(3), pages 591–599, Grenada, Spain.

[Cescatti, 1997] Cescatti, A. (1997). Modelling the radiative transfert in discontinuous canopies of asymetric crowns. i. model structure and algorithms. *Ecological modelling*, 101(2-3) :263–274.

[Chelle, 1997] Chelle, M. (1997). *Développement d'un modèle de radiosité mixte pour simuler la distribution du rayonnement dans les couverts végétaux.* Thèse de doctorat informatique, Université de Rennes I.

[Chen et al., 1994] Chen, S., Ceulemans, R., and Impens, I. (1994). A fractal-based populus canopy structure model for the calculation of light interception. *Forest Ecology and Management*, 69(01-mars) :97–110.

[Chen and Williams, 1993] Chen, S. E. and Williams, L. (1993). View interpolation for image synthesis. *Computer Graphics*, 27(Annual Conference Series) :279–288.

[Chen, 1976] Chen, W. (1976). *Applied graph theory.* North Holland Publ. Co., Amsterdam, The Netherlands.

[Christophe, 2002] Christophe, A. (2002). *Photorègulation du déploiement spatial chez une espèce clonale, le trèfle blanc (Trifolium repens L.) : architecture 3D de la plante et quantification de la colonisation de l'espace.* PhD thesis, I.N.P.L. Nancy.

[Clark and Voynovich, 1989] Clark, R. R. and Voynovich, P. (1989). *Outstanding american bonsai : A photographic essay on the works of fifty american bonsai artists.* Timber Press.

[Cluzeau et al., 1995] Cluzeau, C., Dupouey, J. L., and Courbaud, B. (1995). Polyhedral representation of crown shape. a geometric tool for growth modelling. *Annales des Sciences Forestières*, 52 :297–306.

[Constable, 1991] Constable, G. A. (1991). Mapping the production and survival of fruit on field grown cotton. *Agronomy Journal*, 83(2) :374–378.

[Costes et al., 1998] Costes, E., Sinoquet, H., Godin, C., and Kelner, J. J. (1998). La digitalisation 3d de l'architecture aérienne de pommiers : son utilisation pour étudier la variabilité de la qualité des pommes dans l'arbre, en relation avec leur contexte végétatif. In *Architecture et modélisation en arboriculture fruitière, 11ème colloque sur les recherches fruitières*, pages 278–285, Montpellier, France. INRA-Ctifl.

[Coudurier et al., 1994] Coudurier, T., Barczi, J. F., Barthélémy, D., and Reffye, P. d. (1994). Modélisation et simulation de l'architecture de l'eucalyptus hybride saligna en plantation au congo. Rapport atp 37/93 evaluation de la biomasse des plantations industrielles clonales par télédétection radar au congo, CIRAD - GERDAT Unité de Modélisation des Plantes.

[Courbaud, 1997] Courbaud, B. (1997). *Modélisation de l'éclairement et de la croissance de l'épicéa (Picea abis L. Karst) en forêt irrégulière de montagne.* PhD thesis, Université de Lyon I.

[Dauzat, 1993] Dauzat, J. (1993). Simulated plants and radiative transfer simulations. In Varlet-Grancher, C., Bonhomme, R., and Sinoquet, H., editors, *Colloque Structure du Couvert Végétal et Climat Lumineux : méthodes de caractérisation et applications*, Crop Structure and Light Microclimate. Characterization and Applications, pages 271–278, Saumane, France. INRA Editions.

[Dauzat et al., 2001] Dauzat, J., Rapidel, B., and Berger, A. (2001). Simulation of leaf transpiration and sap flow in virtual plants : description of the model and application to a coffee plantation in costa rica. *Agricultural Forest and Meteorology*, 109 :143–160.

[Demko et al., 1985] Demko, S., Hodges, L., and Naylor, B. (1985). Construction of fractal objects with iterated function systems. In *Computer Graphics, Proceedings of Siggraph 1985*, volume 19(3), pages 271–278. ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[Desbrun, 1997] Desbrun, M. (1997). *Modélisation et Animation de Matériaux Hautement Déformables en Synthèse dŠImages.* PhD thesis, Institut National Polytechnique de Grenoble.

[Deussen et al., 2002] Deussen, O., Colditz, C., Stamminger, M., and Drettakis, G. (2002). Interactive visualization of complex plant ecosystems. In *Proceedings of the IEEE Visualization Conference.* IEEE.

[Deussen et al., 1998] Deussen, O., Hanrahan, P., Lintermann, B., Mech, R., Pharr, M., and Prunsinkiewicz, P. (1998). Realistic modeling and rendering of plant ecosystems. In *SIGRAPPH'98*, Proceedings of SIGRAPPH'98, Orlando, Florida, USA. ACM.

[Deussen and Lintermann, 1997] Deussen, O. and Lintermann, B. (1997). A modelling method and user interface for creating plants. In *Proceedings of Graphics Interface '97*, pages 189–197.

[Deussen and Strothotte, 2000] Deussen, O. and Strothotte, T. (2000). Computer-generated pen-and-ink illustration of trees. In Akeley, K., editor, *Siggraph 2000, Computer Graphics Proceedings*, pages 13–18. ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[Eberly, 2001] Eberly, D. H. (2001). *3D Game Engine Design. A practical approach to real-time computer graphics.* Morgan Kaufmann Publishers, San Francisco, CA USA.

[Eichhorst and Savitch, 1980] Eichhorst, P. and Savitch, W. J. (1980). Growth functions of stochastic lindenmayer systems. *Information and Control*, 45 :217–228.

[Federl and Prusinkiewicz, 1999] Federl, P. and Prusinkiewicz, P. (1999). Virtual laboratory : an interactive software environment for computer graphics. In *Computer Graphics International 1999*, volume 242, pages 93–100.

[Ferley et al., 1997] Ferley, E., Cani, M., and Attali, D. (1997). Skeletal reconstruction of branching shapes. *Computer Graphics Forum*, 16(5) :283–293. Published under the name Marie-Paule Cani-Gascuel.

[Ferraro, 2001] Ferraro, P. (2001). *Méthodes algorithmiques de comparaison d'arborescences. Applications à la comparaison de l'architecture des plantes.* PhD thesis, Thèse de Doctorat en Informatique de l'Institut National Polytechnique de Toulouse.

[Foley et al., 1995] Foley, J., Dam, A. v., Feiner, S., Hugues, J., and Edition, C. (1995). *Interactive computer graphics : principles and practice.* Addison-Wesley, Reading, MA USA.

[Forsey and Bartels, 1988] Forsey, D. R. and Bartels, R. (1988). Hierarchical B-spline refinement. In *Computer Graphics (Proceedings of SIGGRAPH'88)*, volume 22, pages 205–212, Atlanta, Georgia. ACM Press.

[Früh, 1997] Früh, T. (1997). Simulation of water flow in the branched tree architecture. *Silva Fennica*, 31(3) :275–285.

[Gardner, 1984] Gardner, G. Y. (1984). Simulation of natural scenes using textured quadric surfaces. In *ACM/SIGGRAPH Computer Graphics*, volume 18(3), pages 11–20.

[Giavitto and Michel, 2001] Giavitto, J.-L. and Michel, O. (2001). Mgs : A programming language for the transformation of topological collections. Research Report 61-2001, CNRS - Université dŠEvry Val dŠEsonne, Evry, France.

[Godin, 2000] Godin, C. (2000). Representing and encoding plant architecture : a review. *Annals of Forest Science*, 57(05-juin) :413–438.

[Godin, 2003] Godin, C. (2003). *Introduction aux structures multi-échelles. Application à la représentation des plantes.* Habilitation à diriger les recherches, Université de Montpellier II.

[Godin et al., 1996] Godin, C., Bellouti, S., and Costes, E. (1996). Restitution virtuelle de plantes réelles : un nouvel outil pour l'aide à l'analyse de données botaniques et agronomiques. In *L'interface des mondes réels et virtuels, 5èmes Journées Internationales Informatiques*, pages 369–378, Montpellier, France.

[Godin and Caraglio, 1998] Godin, C. and Caraglio, Y. (1998). A multiscale model of plant topological structures. *Journal of Theoretical Biology*, 191 :1–46.

[Godin et al., 1999a] Godin, C., Costes, E., and Sinoquet, H. (1999a). A method for describing plant architecture which integrates topology and geometry. *Annals of Botany*, 84 :343–357.

[Godin et al., 2002] Godin, C., Guédon, Y., Adam, D., Bellouti, S., Boudon, F., Caraglio, Y., Costes, E., Dones, N., Ferraro, P., Nouguier, C., and Sinoquet, H. (1997-2002). AMAPmod v1.8. introduction and reference manual. Manuel de logiciels, CIRAD.

[Godin et al., 1999b] Godin, C., Guédon, Y., and Costes, E. (1999b). Exploration of plant architecture databases with the AMAPmod software illustrated on an apple-tree hybrid family. *Agronomie*, 19(3-4) :163–184.

[Godin et al., 1997] Godin, C., Guédon, Y., Costes, E., and Caraglio, Y. (1997). Measuring and analyzing plants with the AMAPmod software. In Michalewicz, M., editor, *Plants to ecosystems - Advances in Computational Life Sciences, 2nd International Symposium on Computer Challenges in Life Science*, pages 53–84. CSIRO Australia, Melbourne, Australia.

[Gondran and Minoux, 1990] Gondran, M. and Minoux, M. (1990). *Graphes et algorithmes.* Collection de la Direction des Etudes et Recherches d'Electricité de France, 37. Eyrolles.

[Gottschalk et al., 1996] Gottschalk, S., Lin, M. C., and Manocha, D. (1996). OBBTree : A hierarchical structure for rapid interference detection. *Computer Graphics*, 30(Annual Conference Series) :171–180.

[Greene, 1989] Greene, N. (1989). Voxel space automata : modeling with stochastic growth processing in voxel space. *Computer Graphics*, 23(3) :175–184.

[Hallé and Oldeman, 1970] Hallé, F. and Oldeman, R. A. A. (1970). *Essai sur l'architecture et la dynamique de croissance des arbres tropicaux.* Masson, Paris, France.

[Hallé et al., 1978] Hallé, F., Oldeman, R. A. A., and Tomlinson, P. B. (1978). *Tropical trees and forests. An architectural analysis.* Springer-Verlag, New-York.

[Hanan, 1988] Hanan, J. S. (1988). Plantworks : A software system for realistic plant modelling. Master's thesis, University of Regina.

[Hanan, 1992] Hanan, J. S. (1992). *Parametric L-systems and their application to the modelling and visualization of plant.* PhD thesis, University of Regina.

[Herman and Rozenberg, 1975] Herman, G. and Rozenberg, G. (1975). *Developmental systems and languages.*

[Hill, 1996] Hill, D. (1996). *Object-oriented analysis and simulation.* Addison-Wesley Publ. Co., Harlow, UK.

[Honda, 1971] Honda, H. (1971). Description of the form of trees by the parameters of the tree-like body : Effects of the branching angle and the branch length on the shape of the tree-like body. *J. Theor. Biol.*, 31 :331–338.

[Hoppe, 1998] Hoppe, H. (1998). Efficient implementation of progressive meshes. *Computers and Graphics*, 22(1) :27–36.

[Horn, 1971] Horn, H. S. (1971). *The adaptive geometry of trees.* Princeton University Press.

[Jaeger, 1987] Jaeger, M. (1987). *Représentation et simulation de la croissance des végétaux.* Thèse de Doctorat Informatique, Université Louis Pasteur (ULP).

[Jakulin, 2000] Jakulin, A. (2000). Interactive vegetation rendering with slicing and blending. In de Sousa, A. and Torres, J., editors, *Proceedings of Eurographics 2000 (Short Presentations).*

[Janey, 1992] Janey, N. (1992). *Modélisation et synthèse d'images d'arbres et de bassins fluviaux associant méthodes combinatoires et plongement automatique d'arbres et cartes planaires.* PhD thesis, Université de Nancy.

[Janssen and Lindenmayer, 1987] Janssen, J. and Lindenmayer, A. (1987). Models for the control of branch positions and flowering sequences of capitula in mycelis muralis (l.) dumont (compositae). *New Phytologist*, 105 :191–220.

[Jürgensen, 1976] Jürgensen, H. (1976). Probabilistic l systems. *Automata, languages, development*, pages 211–225.

[Kajiya, 1985] Kajiya, J. (1985). Anisotropic reflection models. In Barsky, B., editor, *Computer Graphics (Proceedings of SIGGRAPH'85)*, volume 19(3), pages 15–21.

[Kajiya and Timothy, 1989] Kajiya, J. and Timothy, L. (1989). Rendering fur with three dimensional textures. In *Computer Graphics (Proceedings of SIGGRAPH'89)*, volume 23(3), pages 271–280.

[Karwowski, 2002] Karwowski, R. (October 2002). *Improving the process of plant modeling : The L+C modeling language.* PhD thesis, University of Calgary.

[Kawaguchi, 1982] Kawaguchi, Y. (1982). A morphological study of the form of nature. *Computer Graphics*, 16(3) :223–232.

[Kim and Neumann, 2002] Kim, T. and Neumann, U. (2002). Interactive multiresolution hair modeling and editing. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 620–629. ACM Press.

[Koop, 1989] Koop, H. (1989). Silvi-star : A comprehensive monitoring system. *Forest Dynamics*, page 229.

[Lane and Prusinkiewicz, 2002] Lane, B. and Prusinkiewicz, P. (2002). Generating spatial distributions for multilevel models of plant communities. In *Proceedings of Graphics Interface 2002*, pages 69–80, Calgary, Alberta.

[LeDizès et al., 1997] LeDizès, S., Cruiziat, P., Lacointe, A., Sinoquet, H., LeRoux, X., Balandier, P., and Jacquet, P. (1997). A model for simulating structure-function relationships in walnut tree growth processes. *Silva Fennica*, 31 :313–328.

[Levoy and Whitted, 1985] Levoy, M. and Whitted, T. (1985). The use of points as display primitives. Technical Report TR 85-022, Univ. of North Carolina, Chapel Hill.

[Lindenmayer, 1968] Lindenmayer, A. (1968). Mathematical models for cellular interaction in development, part i and ii. *Journal of theoretical biology*, 18 :280–315.

[Lindenmayer, 1971] Lindenmayer, A. (1971). Adding continuous components to l-systems. In Rozenberg, G. and Salomaa, A., editors, *L Systems*, volume 15, pages 53–68. Springer-Verlag, Berlin.

[Lindenmayer and Rozenberg, 1976] Lindenmayer, A. and Rozenberg, G. (1976). *Automata, languages, development.* North-Holland, Amsterdam.

[Lintermann and Deussen, 1999] Lintermann, B. and Deussen, O. (1999). Interactive modeling of plants. *IEEE Computer Graphics and Applications*, 19(1) :56–65.

[Marshall et al., 1997] Marshall, D., Fussell, D., and III, A. T. C. (1997). Multiresolution rendering of complex botanical scenes. In Davis, W. A., Mantei, M., and Klassen, R. V., editors, *Graphics Interface '97*, pages 97–104. Canadian Human-Computer Communications Society.

[Max, 1996] Max, N. (1996). Hierarchical rendering of trees from precomputed multi-layer z-buffers. In Pueyo, X. and Schorder, P., editors, *Eurographics Rendering Workshop*.

[Max et al., 1999] Max, N., O., D., and Keating, B. (1999). Hierarchical image-based rendering using texture mapping hardware. In *Eurographics Workshop on Rendering*, pages 57–62. Springer Verlag.

[Max and Ohsaki, 1995] Max, N. and Ohsaki, K. (1995). Rendering trees from precomputed Z-buffer views. In *Proceedings of the 6th Eurographics Workshop on Rendering*.

[McClelland, 1996] McClelland, C. K. (1996). On the regularity of blooming in the cotton plant. *Science*, XLIV :578–581.

[McReynolds, 1997] McReynolds, T. (1997). *Programming with OpenGL : Advanced Rendering*. SGI. Siggraph 1997, Course Note.

[Mech, 1997] Mech, R. (1997). *Modeling and simulation of the interaction of plants with the environment using-L-systems and their extensions*. Phd thesis, University or Calgary.

[Mech and Prusinkiewicz, 1996] Mech, R. and Prusinkiewicz, P. (1996). Visual models of plants interacting with their environment. In Rushmeier, H., editor, *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 397–410, New Orleans (Louisiana, USA). Addison-Wesley.

[Mercer, 1991] Mercer, L. (1991). The virtual laboratory. Master's thesis, University of Regina.

[Mercer et al., 1990] Mercer, L., Prusinkiewicz, P., and Hanan, J. (1990). The concept and design of a virtual laboratory. In *Graphics Interface 1990*, pages 149–155.

[Meyer and Neyret, 1998] Meyer, A. and Neyret, F. (1998). Interactive volumetric textures. In Drettakis, G. and Max, N., editors, *Eurographics Rendering Workshop 1998*, pages 157–168, New York City, NY. Eurographics, Springer Wein. ISBN 3-211-83213-0.

[Meyer and Neyret, 2000] Meyer, A. and Neyret, F. (2000). Multiscale shaders for the efficient realistic rendering of pine-trees. In *Graphics Interface*, pages 137–144. Canadian Information Processing Society, Canadian Human-Computer Communications Society. ISBN.

[Meyer et al., 2001] Meyer, A., Neyret, F., and Poulin, P. (2001). Interactive rendering of trees with shading and shadows. In *Eurographics Workshop on Rendering*.

[Mitchell, 1975] Mitchell, K. J. (1975). Dynamics and simulated yield of douglas-fir. *Forest Science*, 21(4) :janv–39.

[Monsi and Saeki, 1953] Monsi, M. and Saeki, T. (1953). Über den lichtfaktor in den pfanzengesellschaften und seine bedeutung für die stoffproduktion. *Jap. J. Bot.*, 14 :22Ű–52.

[Moravie et al., 1997] Moravie, M. A., Pascal, J., and Auger, P. (1997). Investigating canopy regeneration processes through individual-based spatial models : application to a tropical rain forest. *Ecological Modelling*.

[Mortier, 1990] Mortier, F. (1990). Sylvigénèse et structures spatiales en forêts tempérées. Master's thesis, Université de Nancy I.

[Neyret, 1996] Neyret, F. (1996). *Textures Volumiques pour la Synthèse d'images*. PhD thesis, Université Paris-XI - INRIA. http ://www-imagis.imag.fr/Membres/Fabrice.Neyret/publis/thesefabrice-fra.html.

[Neyret, 1998] Neyret, F. (1998). Modeling, animating, and rendering complex scenes using volumetric textures. *IEEE Transactions on Visualization and Computer Graphics*, 4(1) :55–70.

[Neyret, 2001] Neyret, F. (2001). *Complexité Naturelle et Synthèse d'Images*. Habilitation à diriger les recherches, Université Joseph Fourier.

[Nikinmaa, 1992] Nikinmaa, E. (1992). Analyses of the growth of scots pine ; matching structure with function. *Acta Forestalia Fennica*, 235 :mars–68.

[Nishida, 1980] Nishida, T. (1980). K0l-systems simulating almost but not exactly the same development - the case of japanese cypress. *Memoirs Fac. Sci., Kyoto University, Ser. Bio*, 8 :97–122.

[Noik, 1996] Noik, E. (1996). *Dynamic Fisheye Views : Combining Dynamic Queries and Mapping with Database Views*. PhD thesis, University of Toronto.

[Norman and Welles, 1983] Norman, J. M. and Welles, J. M. (1983). Radiative transfer in an array of canopies. *Agronomy Journal*, 75 :481–488.

[Oosterhuis et al., 1982] Oosterhuis, L., Oldeman, R., and Sharik, T. (1982). Architectural approach to analysis of north american temperate deciduous forests. *Canadian Journal of Forest Research*, 12(4) :835–847.

[Oppenheimer, 1986] Oppenheimer, P. E. (1986). Real time design and animation of fractal plants and trees. In Evans, D. C. and Athay, R. J., editors, *SIGGRAPH'86*, volume 20(4) of *Computer Graphics*, pages 55–64, Dallas, Texas USA. ACM.

[O'Rourke, 1985] O'Rourke, J. (1985). Finding minimal enclosing boxes. *Internat. J. Comput. Inform. Sci.*, 14 :183–199.

[Perbet and Cani, 2001] Perbet, F. and Cani, M. (2001). Animating prairies in real-time. In *ACM Interactive 3D Graphics*, USA.

[Perttunen et al., 1996] Perttunen, J., Sievänen, R., Nikinmaa, E., Salminen, H., Saarenmaa, H., and Väkevä, J. (1996). Lignum : a tree model based on simple structural units. *Annals of Botany*, 77 :87–98.

[Pfister et al., 2000] Pfister, H., Zwicker, W., Baar, J. v., and Gross, M. (2000). Surfels : surface elements as rendering primitives. In *SIGGRAPH*, pages 335–342, Los angeles. ACM.

[Piegl and Tiller, 1997] Piegl, L. and Tiller, W. (1997). *The Nurbs book*. Monographs in visual communication. Springer, Berlin, ALL, 2nd edition.

[Popovic and Hoppe, 1997] Popovic, J. and Hoppe, H. (1997). Progressive simplicial complexes. In *SIGGRAPH*, pages 217–223, Los angeles. ACM.

[POV-Ray, 2003] POV-Ray (1991–2003). Persistence of vision raytracer.

[Power et al., 1999] Power, J., Bernheim-Brush, A. J., Prusinkiewicz, P., and Salesin, D. (1999). Interactive arrangement of botanical L-system models. In *Proceedings of the 1999 ACM Symposium on Interactive 3D Graphics*, pages 175–182.

[Preparata and Shamos, 1985] Preparata, F. and Shamos, M. (1985). *Computational geometry*. Springer-Verlag.

[Press et al., 1996] Press, W., Teukolsky, S., Vetterling, W., and Flannery, B. (1996). *Numerical recipes in C. The art of scientific computing*. Cambridge University Press, Cambridge, 2nd edition.

[Prusinkiewicz, 1986] Prusinkiewicz, P. (1986). Graphical applications of l-systems. In *Vision Interface'86 (Proceedings of Graphics Interface '86)*, pages 247–253.

[Prusinkiewicz, 1987] Prusinkiewicz, P. (1987). Applications of l-systems to computer imagery. In Ehrig, H., Nagl, M., Rosenfeld, A., and Rozenberg, G., editors, *Graph grammars and their application to computer science ; Third International Workshop*, pages 534–548. Springer-Verlag, Berlin.

[Prusinkiewicz et al., 1997] Prusinkiewicz, P., Hammel, M., Hanan, J., and Mech, R., editors (1997). *Visual models of plant development*, volume 3 of *Handbook of Formal Languages*. Springer-Verlag, Berlin.

[Prusinkiewicz and Hanan, 1989] Prusinkiewicz, P. and Hanan, J. (1989). *Lindenmayer systems, fractals, and plants*, volume 75 of *Lecture Notes in Biomathematics*. Springer Verlag, New-York.

[Prusinkiewicz and Hanan, 1990] Prusinkiewicz, P. and Hanan, J. (1990). Visualization of botanical structures and processes using parametric l-systems. *Scientific visualization and graphics simulation*, pages 183–201.

[Prusinkiewicz et al., 1994] Prusinkiewicz, P., James, M., and Mech, R. (1994). Synthetic topiary. In *Computer Graphics Proceedings*, pages 351–358.

[Prusinkiewicz and Kari, 1996] Prusinkiewicz, P. and Kari, L., editors (1996). *Subapical bracketed L-systems*, volume 1073 of *Graph Grammar and their Application to Computer Science ; Fifth International Workshop. Lecture Notes in Computer Science*. Springer-Verlag, Berlin.

[Prusinkiewicz et al., 2000] Prusinkiewicz, P., Karwowski, R., Měch, R., and Hanan, J. (2000). L-studio/cpfg : A software system for modeling plants. In Nagl, M., Schürr, A., and Münch, M., editors, *Applications of graph transformation with industrial relevance*, Computer Science 1779, pages 457–464. Springer-Verlag, Berlin.

[Prusinkiewicz et al., 1988] Prusinkiewicz, P., Lidenmayer, A., and Hanan, J. (1988). Developmental models of herbaceous plants for computer imagery purposes. In *SIGGRAPH'88*, volume 22(4) of *Computer Graphics*, pages 141–150, Atlanta, Georgy USA. ACM.

[Prusinkiewicz and Lindenmayer, 1990] Prusinkiewicz, P. and Lindenmayer, A. (1990). *The algorithmic beauty of plants*. Springer Verlag, New York.

[Prusinkiewicz et al., 2001] Prusinkiewicz, P., Mündermann, L., Karwowski, R., and Lane, B. (2001). The use of positional information in the modeling of plants. In *SIGGRAPH'01*, Computer Graphics, pages 36–47, Los Angeles, California. ACM.

[Reeves and Blau, 1985] Reeves, W. T. and Blau, R. (1985). Approximate and probabilistic algorithms for shading and rendering structured particle systems. In *Proceedings of SIGGRAPH 1985*, pages 313–322. ACM Press.

[Reffye et al., 1988] Reffye, P. d., Edelin, C., Françon, J., Jaeger, M., and Puech, C. (1988). Plant models faithful to botanical structure and development. In *SIGGRAPH'88*, volume 22 of *Computer Graphics*, pages 151–158, Atlanta, USA.

[Reffye et al., 1997] Reffye, P. d., Fourcaud, T., Blaise, F., Barthélémy, D., and Houllier, F. (1997). A functional model of tree growth and tree architecture. *Silva Fennica*, 31(3) :297–311.

[Remolar et al., 2003] Remolar, I., Chover, M., Ribelles, J., and Belmonte, O. (2003). View-dependent multiresolution model for foliage. In *In Journal of WSCG (WSCGŠ2003 Proceedings)*.

[Remphrey and Prusinkiewicz, 1997] Remphrey, W. R. and Prusinkiewicz, P. (1997). Quantification and modelling tree architecture. In *Advances in computational life sciences, Vol I : Plants to Ecosystems*, pages 45–52, Australia. CSIRO.

[Robert, 1998] Robert, A. (1998). Représentation et prise en compte du houppier dans diffèrents types de mo'dèles forestiers. Master's thesis, Université Lyon I.

[Rusinkiewicz and Levoy, 2000] Rusinkiewicz, S. and Levoy, M. (2000). QSplat : A multiresolution point rendering system for large meshes. In Akeley, K., editor, *Siggraph 2000, Computer Graphics Proceedings*, pages 343–352. ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[Saito and Takahashi, 1990] Saito, T. and Takahashi, T. (1990). Comprehensive rendering of 3-d shapes. In *Siggraph 1990, Computer Graphics Proceedings*, volume 24(4), pages 197–206. ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[Salminen et al., 1994] Salminen, H., Saarenmaa, H., Pertunen, J., Sievänen, R., Väkevä, J., and Nikinmaa, E. (1994). Modelling trees using and object-oriented scheme. *Mathematical Computer Modeling*, 20(8) :49–67.

[Salomaa, 1973] Salomaa, A. (1973). *Formal languages.* Academic Press, New York.

[Schaufler, 1998] Schaufler, G. (1998). Per-object image warping with layered impostors. In *Eurographics Rendering Workshop 1998*, pages 145Ű–156.

[Shade et al., 1998] Shade, J., Gortler, S., He, L., and Szeliski, R. (1998). Layered depth images. In Akeley, K., editor, *Siggraph 1998, Computer Graphics Proceedings*, pages 231Ű–242. ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[Shinozaki et al., 1964a] Shinozaki, K., Yoda, K., Hozumi, K., and Kira, T. (1964a). A quantitative analysis of plant form. the pipe model theory. i. basic analyses. *Japanese Journal of Ecology*, 14(3) :97–105.

[Shinozaki et al., 1964b] Shinozaki, K., Yoda, K., Hozumi, K., and Kira, T. (1964b). A quantitative analysis of plant form. the pipe model theory. ii. further evidence of the theory and its application in forest ecology. *Japanese Journal of Ecology*, 14(4) :133–139.

[Shlyakhter et al., 2001] Shlyakhter, I., Rozenoer, M., Dorsey, J., and Teller, J. (2001). Reconstructing 3d tree models from instrumented photographs. *IEEE Computer Graphics and Applications*, 21(3) :53–61.

[Sillion, 1995] Sillion, F. X. (1995). Hierarchical solution techniques for realistic rendering. In *State of the Art Report - Graphicon'95 Conférence*, St Petersburg, Russia.

[Sinoquet and Bonhomme, 1992] Sinoquet, H. and Bonhomme, R. (1992). Modeling raditive tranfer in mixed and row intercropping systems. *Agricultural and Forest Meteorology*, 62 :219–240.

[Sinoquet et al., 1991] Sinoquet, H., Moulia, B., and Bonhomme, R. (1991). Estimating the three-dimensional geometry of a maize crop as an input of radiation models : comparison between three-dimensional digitising and plant profiles. *Agric For Meteorol*, 55 :233–249.

[Sinoquet et al., 1997] Sinoquet, H., Rivet, P., and Godin, C. (1997). Assessment of the three-dimensional architecture of walnut trees using digitising. *Silva Fennica*, 31(3) :265–273.

[Sinoquet et al., 1998] Sinoquet, H., Thanisawanyangkura, S., Mabrouk, H., and Kasemsap, P. (1998). Characterisation of the light environment in canopies using 3d digitising and image processing. *Annals of Botany*, 82 :203–212.

[Smith, 1984] Smith, A. (1984). Plants, fractals, and formal languages. *Computer Graphics*, 18(3).

[Smith, 1994] Smith, W. R. (1994). An empirical evaluation of a three dimensionnal crown model for predicting volume growth. *Forest Ecology and Management*, 69 :199–209.

[Smith et al., 1992] Smith, W. R., Farrar, J. R., Murphy, P. A., Yeiser, J. L., Meldhal, R. S., and Kush, J. (1992). Crown and basal area relationship of open-grown southern pines for modelling competition and growth. *Canadian Journal of Research*, 22 :341–347.

[Soler et al., 2003] Soler, C., Sillion, F. X., Blaise, F., and Reffye, P. d. (2003). An efficient instantiation algorithm for simulating radiant energy transfer in plant models. *ACM Transactions On Graphics*, 22(2).

[Sonohat et al., 2004] Sonohat, G., Sinoquet, H., Kulandaivelu, V., Combes, D., and Lescourret, F. (2004). Three-dimensional reconstruction of partially 3d digitised peach tree canopies. In Godin, C. e. a., editor, *Proceedings of the 4TH International Workshop on Functional-Structural Plant Models.*

[Stollnitz et al., 1996] Stollnitz, E. J., DeRose, T. D., and Salesin, D. H. (1996). *Wavelets for computer graphics : theory and applications.* Morgan Kaufmann, San Francisco.

[Strothotte and Strothotte, 1997] Strothotte, C. and Strothotte, T. (1997). *Seeing Between the Pixels : Pictures in Interactive Systems.* Springer-Verlag, Berlin-Heidelberg-New York.

[Strothotte et al., 1994] Strothotte, T., Preim, B. andRaab, A., Schumann, J., and Forsey, D. R. (1994). How to render frames and influence people. In *Computer Graphics Forum*, volume 13(3), pages 455Ű–466.

[Szilard and Quinton, 1979] Szilard, A. L. and Quinton, R. E. (1979). An interpretation for dol systems by computer graphics. *The Science Terrapin*, 4 :8–13.

[Valentine, 1985] Valentine, H. T. (1985). Tree-growth models : derivations employing the pipe-model theory. *Journal of Theorethical Biology*, 117(4) :579–585.

[Weber and Penn, 1995] Weber, J. and Penn, J. (1995). Creation and rendering of realistic trees. In *SIGGRAPH 1995, Computer Graphics Proceedings*, pages 119–128.

[Welzl, 1991] Welzl, E. (1991). Smallest enclosing disks (balls and ellipses). *New Results and New Trends in Computer Science*, 555 :359–370.

[Yomokori, 1980] Yomokori, T. (1980). Stochastic characterizations of eol languages. *Information and Control*, 45 :26–33.

[Zeide and Pfeifer, 1991] Zeide, B. and Pfeifer, P. (1991). A methode for estimation of fractal dimension of tree crown. *Forest Science*, 37(5) :1253–1265.

[Zhang and Blaise, 2003] Zhang, X. and Blaise, F. (2003). Progressive polygon foliage simplification. In Hu, B. and Jaeger, M., editors, *Plant Growth Modeling and Applications (Proceedings of PMA03)*, Beijing, China. Springer.

[Zorin et al., 1997] Zorin, D., Schröder, P., and Sweldens, W. (1997). Interactive multiresolution mesh editing. In *Computer Graphics (Proceedings of SIGGRAPH'97)*, volume 31, pages 259–268, Los Angeles, California. ACM Press.

Annexe A

Glossaire sur l'architecture végétale

Ce travail étant à l'interface de l'informatique et de la botanique, de nombreux termes liés à l'architecture végétale sont utilisés tout au long du document. Nous donnons ici quelques explication relatives à l'architecture des plantes.

Un des éléments fondamental de l'architecture d'un arbre est l'axe (branche et tronc). La croissance de tout axe résulte de l'activité du méristème situé à son extrémité, appelé méristème apical. Un axe est constitué d'un ensemble d'entre-nœuds créés les uns après les autres. Entre deux entre-nœuds existe un nœud qui donne naissance à des feuilles et à des méristème latéraux. La naissance d'un axe à partir d'un méristème latéral est appelée ramification donnant un nouvel ordre d'axe. L'ordre d'un axe est défini de la manière suivante. L'axe d'ordre 1 est l'axe associé au nœud racine. Un axe d'ordre k est édifié par un méristème latéral d'un axe d'ordre $k-1$ (Figure A-1).

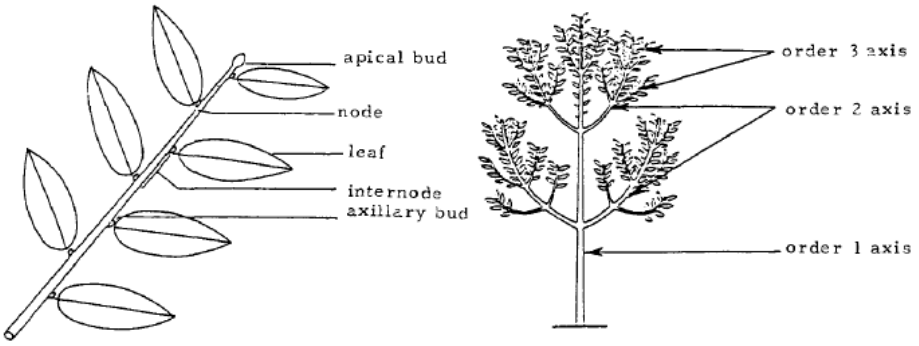


Fig. A-1 – A gauche, décomposition d'un axe en feuilles, entre-nœuds et méristèmes. A droite, notion d'ordre en botanique. Tiré de [Reffye et al., 1988].

La ramification au niveau des méristèmes latéraux s'effectue selon plusieurs modes :

- ramification continue : tout méristème latéral d'un axe donne naissance à un nouvel axe,
- ramification rythmique : seuls certains méristèmes latéraux répartis régulièrement donnent naissance à un nouvel axe,
- ramification diffuse : seuls certains méristèmes latéraux répartis aléatoirement donnent naissance à un nouvel axe.

Le mode de ramification latérale et de croissance apicale de chaque axe sont fonction de l'ordre de l'axe considéré suivant les espèces modélisées. Au niveau géométrique les axes adoptent généralement une tendance *orthotropique* (croissance des axes en direction verticale) ou *plagiotropique* (croissance des axes en direction horizontale).

Un arbre *monopodial* d'ordre k est un arbre possédant un seul axe d'ordre 1 et un nombre

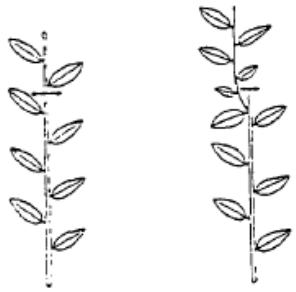


FIG. A-2 – Réitération après l'élagage d'un axe. Exemple de croissance sympodiale. [Reffye et al., 1988]

fini d'axes d'ordre compris entre 1 et k. La croissance des axes peut ne pas être uniquement monopodiale, en particulier si un méristème apical meurt (mort naturelle ou taille de l'axe), il peut y avoir réitération et création à partir de méristèmes latéraux de nouveaux axes à la place de ce nœud A-2. On dit qu'on a alors une croissance *sympodiale*.

Finalement, Hallé, Oldeman et Tomlinson [Hallé et al., 1978] ont recensé 23 modèles architecturaux pour les arbres des forêts tropicales (Figure A-3). Leur classification est principalement basée sur :

- la présence ou l'absence de croissance sympodiale,
- le type de ramification,
- la croissance continue ou non,
- la tendance plagiotropique ou orthotropique.

Pour aider à leur compréhension, nous donnons ici un glossaire du vocabulaire lié à l'architecture. Ce glossaire est basé sur celui fournit par laboratoire AMAP (consultable en ligne à l'adresse suivante : http ://amap.cirad.fr/architecture/glossaire.html).

- A -

Acrotonie L'acrotonie est définie par Troll (1937) comme le développement préférentiel des rameaux placés au sommet de la tige porteuse, celle-ci restant toutefois la plus développée. Selon que le système est monopodial ou sympodial, c'est-à-dire que l'axe porteur est à croissance indéfinie ou définie, Rauh (1939) puis Champagnat (1947) distinguent respectivement une acrotonie primaire et une acrotonie secondaire. Bell (1991) envisage la ramification acrotone («acrotonic branching») comme étant le développement des rameaux les plus distaux d'une tige porteuse sans que celle-ci soit obligatoirement la plus développée.

Allongement L'allongement d'une tige est la manifestation directement observable de la croissance primaire. Il est essentiellement le résultat d'un allongement cellulaire qui prend naissance un peu en arrière du dôme apical, d'où le qualificatif de «subapical growth» employé par Hackett (1985). D'anciens auteurs, faisant allusion à une croissance qui éloignait les différentes pièces élaborées par la croissance terminale, parlaient de croissance intercalaire (Sachs, 1874). Cette expression a été conservée par Rivals (1966) et Kozlowski («intercalary growth», 1971). Ce phénomène, qui concerne essentiellement l'étirement des entre-nœuds, est encore appelé croissance internodale (Van Tieghem, 1884 ; Rivals, 1966).

Aubier Partie extérieure du tronc, sous l'écorce, où circule la sève (bois «jeune»)

- B -

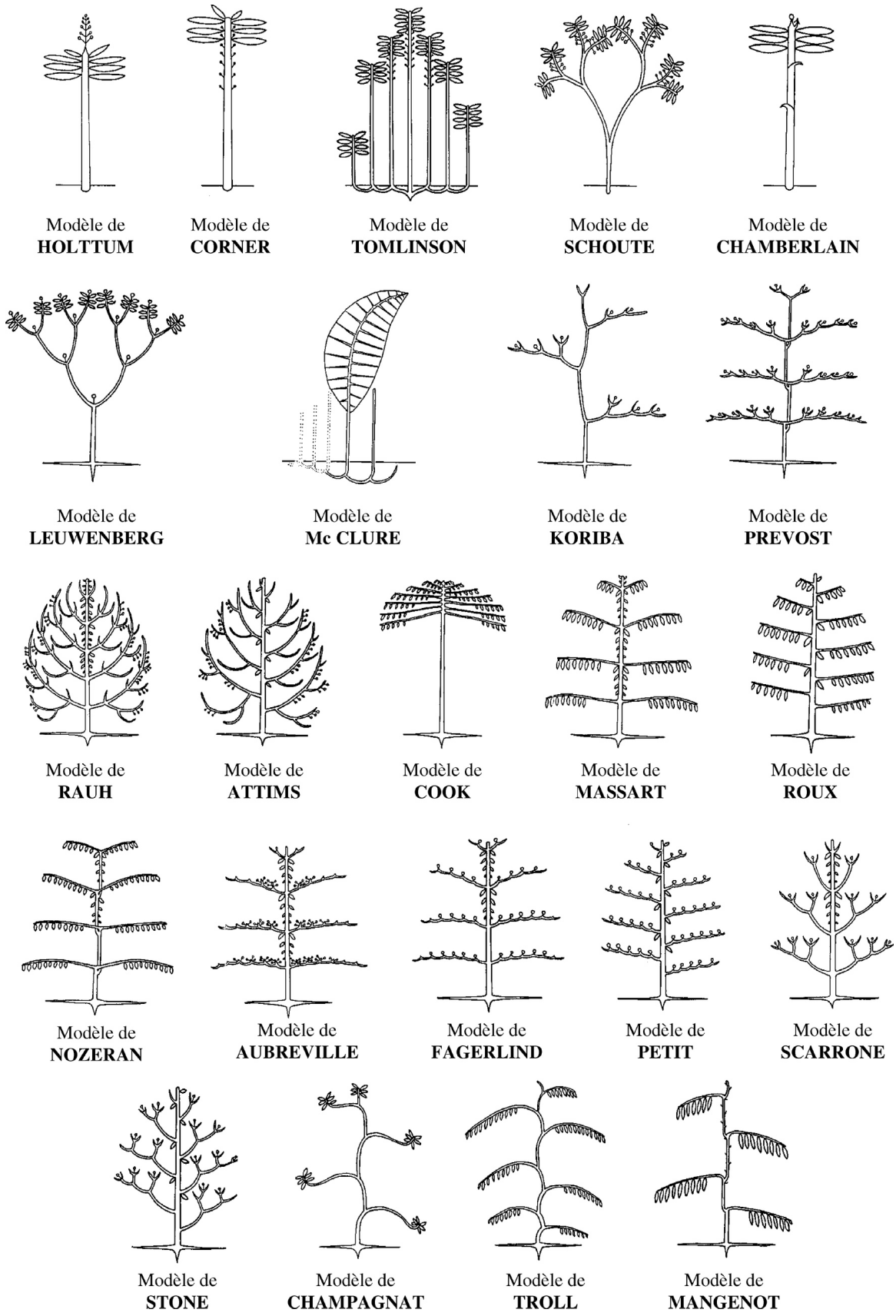


FIG. A-3 – Les différents modèles architecturaux (d'après Hallé et Oldeman [Hallé and Oldeman, 1970]). Tiré de [Ferraro, 2001].

Basitonie Le développement préférentiel de rameaux latéraux vigoureux à la base d'une tige est qualifié de basitonie (Troll, 1937). Bell (1991) parle de "basitonic branching" quand les rameaux sont d'autant plus développés que l'on s'éloigne de l'apex de la tige.

- C -

Cambium Le cambium ou assise cambiale est un méristème qualifié de secondaire car les cellules qui le constituent ne sont pas uniquement des cellules méristématiques primaires mais aussi des cellules qui ont retrouvé l'aptitude à se diviser (cellules à caractère embryonnaire ou totipotentes) par dédifférenciation de cellules parenchymateuses. Le cambium produit des tissus (fonction histogène) dédiés à la vascularisation (bois et liber), à la mise en réserve (rayons) et au soutien (fibre). L'activité de division cellulaire et les tissus qui en résultent par différenciation permettent l'accroissement en diamètre des axes (racinaires et caulinaires) d'une plante.

Croissance Lorsque le méristème terminal d'un axe feuillé se transforme à un moment de sa vie en une structure incapable de poursuivre un fonctionnement végétatif ultérieur (fleur, inflorescence, épine, vrille, tissu parenchymateux,...) ou lorsqu'il meurt (seul, avec l'ensemble des pièces du bourgeon ou avec la portion distale de l'axe qui le porte), on parle de croissance définie («definite extension» Bell, 1991). Si le méristème terminal garde la potentialité de fonctionner indéfiniment, on parle de croissance indéfinie («indefinite extension» Bell, 1991). Impliquant une idée de durée potentielle du fonctionnement d'un méristème, ce qualificatif d'indéfinie est un abus de langage justifié par la nécessité d'opposer ce cas à la croissance définie (Guinochet, 1965). Dans la pratique, il n'est pas toujours facile de trancher entre ces deux expressions dont la qualification, se référant à une durée relative, est soumise dans une large mesure à la subjectivité de l'observateur. Hallé *et al.* (1978) distinguent :

- des axes qui ne présentent pas de périodicité d'allongement endogène marquée et qui sont dits à croissance continue («continuous growth»),
- des axes qui montrent une périodicité d'allongement endogène marquée et qui sont dits à croissance rythmique («rhythmic growth»).

Couronne voir houppier.

Charpentières Appelé aussi branches maîtresses : ce sont celles qui partent directement du tronc. Les plus grosses sont généralement les plus basses.

- E -

Entre-nœud Intervalle entre deux nœuds c'est-à-dire entre deux points d'insertion de feuilles sur une tige.

- H -

Houppier Ensemble des branches et du feuillage d'un arbre. Cela correspond à la partie supérieure de l'arbre.

- M -

Méristème Le mot méristème provient du grec meristos=division. Dans les années 1750 où se situe un renouveau pour la connaissance du végétal et la formation de la botanique, les naturalistes s'intéressent à l'extrémité de la tige, partie où s'effectue le développement. C'est en effet une caractéristique du végétal que de croître par ses extrémités. Cette partie où se produit la croissance caulinaire a été appelée punctum vegetationis par De Wolf en 1759. Avec le développement des moyens optiques, les tissus puis les cellules on été délimités

puis caractérisés, on parle alors de point végétatif ou zone apicale actuellement. Au sommet de cette zone apicale se situe le méristème. Le méristème est l'ensemble des cellules situées à l'extrémité d'un axe qui par leur aptitude à se diviser génère les différentes parties du végétal (feuille, tige, racine, organe de reproduction). Chez l'embryon, des cellules perdent leur aptitude à se diviser pour se spécialiser dans certaines fonctions, c'est le phénomène de différenciation tandis que d'autres conservent cette faculté de division, elles restent totipotentes et forment les tissus méristématiques caulinaires et racinaires.

Mésotonie La mésotonie est caractérisée par le développement préférentiel des rameaux latéraux dans la partie médiane d'une tige (Troll, 1937).

Monopode Dans la ramification monopodiale (Emberger, 1960) ou monopodique (Sachs, 1874), les rameaux latéraux se développent sans qu'il y ait arrêt définitif du fonctionnement du méristème de la tige principale : l'ensemble ramifié qui en découle est qualifié de monopode.

Multiplication végétative La multiplication est un procédé très fréquent chez les végétaux permettant de dupliquer l'organisme puis de le rendre autonome et indépendant de l'organisme géniteur en évitant la phase reproductrice, c'est-à-dire en évitant des recombinaisons génétiques, autrement dit aboutissant au clonage de l'individu.

- N -

Nœud Point d'insertion de feuilles sur une tige.

- O -

Organogenèse L'organogénèse se déroule au niveau de l'apex d'une tige. C'est dans sa partie la plus distale ou méristème terminal (Nougarède et Rembur, 1985) que se produisent des phases d'intense activité mitotique au cours desquelles sont initiés de nouveaux éléments de la tige ou métamères Ce phénomène, trouvant son origine au sommet de l'axe feuillé, a été qualifié de croissance terminale par Van Tieghem (1898) ou encore de croissance apicale par Rivals (1966).

- P -

Polycyclisme Au cours d'une même année de végétation, une tige peut résulter d'une ou plusieurs unités de croissance ou être constituée d'un ou de plusieurs cycles morphogénétiques. La pousse sera alors respectivement qualifiée de monocyclique ou polycyclique. Cette notion de cycle morphogénétique (C.M.), défini à partir des Pins (Prat, 1936), peu usitée, peut toutefois s'employer pour désigner une périodicité spatiale chez tout type de plantes quand on ignore tout de ses liens avec la cinétique d'allongement. Pour des plantes dont les unités de croissance successives sont toutes identiques, la notion de polycyclisme n'a aucun intérêt. Par contre, chez les espèces originaires de la zone tempérée de l'hémisphère nord, le niveau d'organisation plus complexe qui peut en résulter correspond à la notion de pousse annuelle. Cette notion assez bien connue recouvre différents types de structures qui sont caractérisables par de nombreux paramètres et notamment par leur mode de croissance ou de ramification.

Préformation Une tige ou une portion de tige peut être entièrement préformée (croissance prédéterminée, au sens de Hallé *et al.*, 1978) ou entièrement néoformée (croissance indéterminée, au sens de Hallé *et al.*, 1978), elle peut aussi comporter une partie préformée suivie d'une partie néoformée. Dans ce dernier cas, des marqueurs morphologiques permettent, le plus souvent, de distinguer, a posteriori, les deux parties de l'unité de croissance ou le passage entre les deux. Les notions de préformation et de néoformation peuvent s'appliquer à plusieurs niveaux d'organisation de la plante, qu'il convient alors de préciser.

Phyllotaxie Ordre dans lequel sont implantés les feuilles ou les rameaux sur la tige d'une plante, ou, par extension, la disposition des éléments d'un fruit, d'une fleur, d'un méristème. On appelle aussi phyllotaxie la science qui étudie ces arrangements.

- R -

Ramification Pour Bugnon (1980) la ramification est « ...le fait qu'une unité morphologique du corps [du végétal] donne naissance à une ou plusieurs unités nouvelles de même nature fondamentale qu'elle-même. En principe, l'unité génératrice et les unités apparues restent associées en un ensemble dit système ramifié. » Si cette définition a le mérite d'éviter toute confusion entre la ramification (le processus) et le résultat de ce processus (le rameau ou le système ramifié), elle présente, en outre, l'avantage d'attirer notre attention sur la nécessité, pour décrire un système ramifié, de distinguer l'«unité génératrice» des «unités apparues». La nécessité pour ces deux types d'entités d'être de même nature permet de faire une distinction nette entre la croissance d'un axe feuillé (ou élaboration de nouveaux métamères par un méristème apical) et la ramification caulinaire qui correspond à la production d'un axe feuillé à partir d'un autre. Cette définition écarte également, de fait, les tiges qui naissent sur des organes de nature non caulinaire à partir du développement de méristèmes néoformés (drageons, bulbilles développées sur des feuilles,...) (voir multiplication végétative).

Ramification latérale La ramification latérale résulte de la différenciation, sur les flancs de l'apex d'une tige, d'un territoire de cellules à caractères embryonnaires. Ce territoire se développe à l'aisselle d'une ébauche foliaire et forme un méristème latéral. Par son fonctionnement, ce méristème latéral pourra alors édifier à son tour un nouvel axe feuillé, dont il deviendra le méristème terminal et qui sera qualifié d'axe latéral ou rameau. La ramification latérale correspond au mode de ramification le plus répandu chez les végétaux vasculaires. Chez les Angiospermes, ce processus aboutit à la formation de rameaux, généralement axillaires et qu'il est possible d'identifier, la plupart du temps, par la présence du (ou des deux) premier(s) organe(s) foliaire(s) qui occupe(nt) une position précise : les préfeuilles.

Ramification terminale Chez les végétaux dont l'apex présente un méristème apical (Buvat, 1989 ; Sinnott, 1960), la ramification terminale (Gatin, 1924) correspond à une division de ce dernier directement au niveau du dôme méristématique. Selon que cette division entraîne la formation de deux rameaux ou plus, on parlera respectivement de dichotomie ou de polytomie (Gatin, 1924). Ce mode de division peut aboutir à la formation de deux (ou plusieurs) parties égales ou inégales, et on parlera alors respectivement de dichotomie (polytomie) isotone ou anisotone. Ce mode de division se rencontre chez les Ptéridophytes, les Bryophytes et les Thallophytes mais il intéresse alors seulement une ou un faible nombre de cellules apicales (Emberger, 1960). Chez les Angiospermes, la ramification terminale est rare. Elle a été décrite chez quelques Monocotylédones appartenant à des familles comme les Arecaceae (Hyphaene thebaica (L.) C. Martius, Schoute, 1909 ; Nypa fruticans Wurmb, Tomlinson, 1971 ; Chamaedorea cataractarum Martius, Fisher, 1974), les Flagellariaceae (Flagellaria indica L., Tomlinson et Posluszny, 1977) ou les Musaceae (Strelitzia reginae Banks ex Dryander, Fisher, 1976), et de manière plus occasionnelle chez les Dicotylédones, dans la famille des Cactaceae (Mammillaria spp., Boke, 1976) et des Asclepiadaceae (Asclepias syriaca L., Nolan, 1969).

Réitération Même si quelques arbres restent conformes à leur unité architecturale tout au long de leur vie, il est facile de constater dans la nature que, chez la plupart des espèces, l'arbre jeune, pyramidal, entièrement hiérarchisé autour d'un tronc unique, cède la place, à des âges plus avancés, à un arbre plus complexe dont la cime s'arrondit et dont le houppier se structure autour de nombreuses et puissantes branches maîtresses. Cette transformation

résulte, le plus souvent, d'une duplication de l'unité architecturale au cours de l'ontogénèse. Définie pour la première fois par Oldeman (1972, 1974) sous le terme de réitération, ce mode particulier de duplication peut être de natures diverses et revêtir bien des aspects chez les végétaux (Hallé *et al.*, 1978). Il n'en reste pas moins le processus essentiel par lequel s'édifie la cime de la plupart des arbres (Edelin, 1977, 1984, 1986 ; Barthélémy *et al.*, 1989, 1991 ; Sanoja, 1992) au cours d'une métamorphose architecturale (Hallé et Ng, 1981 ; Edelin, 1984).

Réitération partielle Lorsque la duplication ne reprend qu'une partie de l'unité architecturale (une branche qui porte une branche), la structure dupliquée résulte de la réitération partielle de l'unité architecturale.

- S -

Sympode Dans la ramification sympodiale (Emberger, 1960) ou sympodique (Sachs, 1874), le méristème terminal de l'axe porteur meurt ou se transforme en une structure qui perd sa faculté de croître végétativement. La poursuite de la croissance est alors assurée par le fonctionnement d'un ou de plusieurs méristèmes latéraux qui édifieront autant d'axes latéraux ou axes relais, et l'ensemble ramifié sera qualifié de sympode. Selon que cette ramification aboutit à la formation de un, deux ou plus de deux relais, on parlera respectivement de sympode monochasial, dichasial, ou polychasial. Dans la ramification sympodiale, la mort du méristème terminal et le développement des relais peuvent être très précoces et s'effectuer à l'intérieur même du bourgeon. Dans ce cas, le sympode qui en résulte peut présenter une physionomie rappelant une dichotomie et on parle alors de fausse dichotomie ou pseudo-dichotomie (Tomlinson, 1970). Lors de la ramification sympodiale d'un axe, un relais peut se placer exactement dans le prolongement de la tige porteuse donnant alors au système ramifié sympodial une physionomie monopodiale. Troll (1937) puis Rauh (1939) parlent, pour désigner ce type de structure, de «pseudomonopodium».

- U -

Unité architecturale La formation, par le méristème apical du tronc et des branches, de pousses annuelles de plus en plus vigoureuses au cours des premières étapes de l'ontogénie, conduit les structures les plus "différenciées" à être "repoussées" à la périphérie de l'arbre. Quel que soit le mode de croissance spécifique, ce phénomène aboutit à un stade architectural relativement stable et fortement hiérarchisé au niveau duquel il est possible de distinguer un nombre fini de catégories d'axes différenciables entre elles par des caractéristiques morphologiques et fonctionnelles. Le nombre de ces catégories d'axes, leur nature et la manière dont elles s'agencent entre elles définissent alors le «diagramme architectural» (Edelin, 1977) ou encore «l'unité architecturale» (Barthélémy *et al.*, 1989, 1991) d'une espèce donnée.

Unité de croissance L'unité de croissance (U.C. ; unit of extension) correspond à la structure mise en place par la tige au cours d'une phase d'allongement ininterrompue (Hallé et Martin, 1968). Sa détermination requiert la connaissance de l'aspect temporel de l'allongement, c'est-à-dire la cinétique de mise en place des éléments de la tige, elle est matérialisée ou non par des marqueurs morphologiques (écailles foliaires, entre-nœuds courts...).

Unité de morphogénèse L'unité de morphogénèse (U.M. ; unit of morphogenesis) correspond à l'ensemble des organes initiés par le méristème apical durant une phase d'organogénèse ininterrompue (Hallé et Martin, 1968). Elle est unique dans le cas de plantes à croissance continue (Palmier) ou chez certaines plantes à croissance rythmique (Thé), tandis que chez les plantes qui montrent une alternance de phases d'activité et d'inactivité mitotique, les tiges résultent de plusieurs U.M. successives. Notons que les études cytolo-

giques nécessaires pour déterminer cette U.M. n'ont été entreprises que pour de trop rares cas et que son existence et sa structure restent à identifier chez la plupart des espèces.

Annexe B

Symboles interprétés par la tortue

Nous donnons ici pour information un ensemble basique de symboles interprétés par la tortue qui servent à la compréhension de ce document.

- *Symboles qui font avancer et dessiner la tortue*

$F(s), G(s)$ Avance d'un pas de longueur s et trace un segment à partir de la position d'origine jusqu'à la nouvelle position.

$f(s), g(s)$ Avance d'un pas de longueur s sans tracer de segment.

$@O(r)$ Dessine une sphère de rayon r à la position courante.

- *Symboles qui contrôlent l'orientation de la tortue dans l'espace*

$+(\theta)$ Tourne à droite d'un angle θ autour de l'axe U.

$-(\theta)$ Tourne à gauche d'un angle θ autour de l'axe U.

$\&(\theta)$ Tourne vers le bas d'un angle θ autour de l'axe L.

$\hat{}(\theta)$ Tourne vers le haut d'un angle θ autour de l'axe L.

$/(\theta)$ Roule vers le gauche d'un angle θ autour de l'axe H.

$\backslash(\theta)$ Roule vers le droite d'un angle θ autour de l'axe H.

| Tourne de 180° autour de l'axe U. Ce qui est équivalent à $+(180)$ ou $-(180)$.

- *Symboles pour modéliser des structures ramifiés*

[Pousse l'état courant de la tortue (position, orientation et attribut de traçage) dans une pile.

] Sort l'état de la pile et l'attribue à la tortue.

Aucune ligne n'est dessinée, même si en général la position et l'orientation de la tortue ont changé.

- *Symboles pour créer et incorporer des surfaces*

{ Commence à sauver les positions successives de la tortue comme des sommets d'un polygone.

} Rempli le polygone sauvé.

$\tilde{}X(s)$ Dessine la surface identifiée par le symbole X, dillatée d'un facteur s, à la position courante et dans l'orientation de la tortue. Cette surface est généralement définie par un patche bi-cubique.

- *Symboles qui changent les attributs de traçage*

$\#(w)$ Définit l'épaisseur de trait à w, ou augmente la valeur courante d'épaisseur de trait de la valeur d'incrément par défaut si aucun paramètre n'est donné.

$!(w)$ Définit l'épaisseur de trait à w, ou réduit la valeur courante d'épaisseur de trait de la valeur de décrément par défaut si aucun paramètre n'est donné.

; (n) Définit l'index de la palette de couleur à n, ou augmente la valeur de l'index courant par la valeur d'incrément par défaut si aucun paramètre n'est donné.

, (w) Définit l'index de la palette de couleur à n, ou réduit la valeur de l'index courant par la valeur de décrément par défaut si aucun paramètre n'est donné.

Représentation géométrique multi-échelles de l'architecture des plantes

La géométrie d'une plante est un objet complexe, combinant les géométries de nombreux organes, dépendant de l'échelle d'analyse considérée et très variable dans le temps ainsi que d'une espèce à l'autre. Malgré cette complexité et cette plasticité, la géométrie d'une plante est devenue un enjeu de modélisation majeur ces dernières années aussi bien en éco-physiologie, qu'en synthèse d'images.

Cette étude commence par une analyse des différents types de représentations et systèmes génératifs de plantes existants. Deux catégories d'approches pour modéliser la géométrie d'une plante sont utilisées : les approches globales, qui considèrent la plante comme un tout et la représentent par une enveloppe, et les approches modulaires, dans lesquelles la plante est représentée par l'ensemble des modèles géométriques de ses composants. Certains travaux récents introduisent l'idée d'organiser des modèles de plantes en différents niveaux de détails pour adapter la complexité de la représentation ou pour une meilleure compréhension du développement de la plante.

Dans ce travail nous développons et formalisons un modèle géométrique multi-échelles général de représentation de l'architecture des plantes. Ce modèle intègre de manière cohérente différentes représentations géométriques à différents niveaux de détails. Il permet ainsi de contrôler et d'adapter la complexité de la représentation au cours des calculs et d'acquerir une compréhension plus précise de la géométrie complexe des plantes.

Nous utilisons ce modèle dans une approche « montante » pour caractériser des formes globales (émergentes) de la structure détaillée d'une plante et ainsi l'espace qu'elle occupe. Il est ensuite utilisé dans une approche « descendante » pour le développement d'un modèle de croissance des plantes fondé sur un nouveau paradigme de simulation, permettant une spécification des connaissances biologiques à différents niveaux d'organisation. Le principe de cette approche s'appuie sur une hiérarchisation en échelles des informations qui permet la manipulation interactive et intuitive de modèles complexes de plantes.

Mots-Clés : Géométrie multi-échelles - Modélisation de l'Architecture des plantes - Simulation - Cohérence multi-échelles.

Multiscale geometric representation of plant architecture

Plant geometry is a complex object, combining geometries of numerous different organs, depending of the considered scale of analysis and variable in time and from one species to another. Because of this complexity and plasticity, modeling plants geometry become a challenging multi-disciplinary research topic for eco-physiological modeling to computer graphics.

This study begins with the analysis of the various existing types of representation and generative systems of plants models. We identified two categories of approaches for modeling plant geometry : the global approaches, that consider plant as a whole and represents it by an envelop, and the modular one, where a plant is represented by all the geometric models of its components. Some recent works introduce the idea to organize plants models in different levels of details to adapt the complexity of the representation or for a better understanding of plant growth.

In this context, the need for formalism has lead us to define a general multiscale geometric model of plant architecture, and to show its efficiency in different contexts. This model integrates coherently several geometric representations at different levels of details. It allows thus controlling and adapting the complexity of the representation during computations and a better understanding of the complex geometry of plants.

We used this model in a « bottom-up » manner to characterize global shapes emerging from the detailed structure of a plant and consequently to study at different scale the space occupied by the plant. This model is then used in a « top-down » way to develop a model of plant growth based on a new simulation paradigm which allows specifying biological knowledges at different levels of organization. A hierarchization at different scales of the information makes an interactive and intuitive manipulation of complex plant models possible.

Key Words : Multiscale Geometry - Plants Architecture Modeling - Simulation - Multiscale Coherency.

Discipline : Informatique

Laboratoire : UMR **AMAP** bot**A**nique et bioinfor**M**atique de l'**A**rchitecture des **P**lantes
TA 40 / PS 2 - boulevard de la lironde - 34398 Montpellier - France