



HAL
open science

Le raisonnement spatial dans les systèmes à base de connaissances : application à l'analyse de sites avalancheux

Laurent Buisson

► To cite this version:

Laurent Buisson. Le raisonnement spatial dans les systèmes à base de connaissances : application à l'analyse de sites avalancheux. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 1990. Français. NNT: . tel-00338689

HAL Id: tel-00338689

<https://theses.hal.science/tel-00338689>

Submitted on 14 Nov 2008

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.

THESE

présentée par Laurent BUISSON

**pour obtenir le titre de docteur
de l'Université Joseph Fourier - Grenoble 1**

(arrêté ministériel du 5 juillet 1984)

Spécialité informatique

*** * ***

**Le raisonnement spatial dans les
systèmes à base de connaissances**
Application à l'analyse de sites avalanches

*** * ***

Date de soutenance : 5 novembre 1990

Composition du Jury :

M. Yves Chiaramella (président)

Mme Marie-Odile Cordier (rapporteur)

M. Henri Farreny (rapporteur)

M. Gérard Brugnot

M. Pierre Haren

M. François Rechenmann

Thèse préparée au sein du laboratoire ARTEMIS/IMAG - Projet SHERPA

REMERCIEMENTS

Mes remerciements iront tout d'abord aux membres du jury de cette thèse.

Yves Chiaramella me fait l'honneur d'assurer la présidence de ce jury et je lui en sais gré.

Marie-Odile Cordier et Henri Farreny ont accepté, malgré leurs charges de travail et leurs responsabilités, d'être rapporteurs de cette thèse. Je les remercie aussi pour le temps qu'ils ont bien voulu y consacrer ainsi que pour les conseils qu'ils m'ont donnés.

Pierre Haren a bien voulu me faire l'honneur de participer à ce jury malgré un emploi du temps lui aussi très dense. Je tiens à l'en remercier.

Gérard Brugnot a, le premier, eu l'idée d'utiliser les techniques de l'intelligence artificielle pour les appliquer à l'analyse de sites avalanches. Il a su par ailleurs obtenir du ministère de l'Agriculture mon affectation à la Division Nivologie du CEMAGREF, où il m'a accordé toute sa confiance et m'a laissé toute latitude pour l'organisation de mes travaux. Qu'il soit ici remercié.

François Rechenmann nous a accepté, mon sujet et moi-même, au mois de septembre 1987, alors que nous étions à la recherche d'un laboratoire d'accueil. Il a assuré l'encadrement scientifique de cette thèse. Pendant ces quatre ans, ses idées sur les systèmes à base de connaissances et les environnements de résolution de problèmes, sa culture informatique ou, plus exactement, la profondeur de ses analyses de l'évolution de l'informatique et, par dessus tout, la conception qu'il a de la recherche ont été une source permanente d'enrichissement. De plus, il a su me donner les conseils qui se sont révélés être les bons. Je tiens à le remercier de tout cela et aussi de la très bonne ambiance qu'il fait régner dans son équipe grâce à son humour - mes roulettes sont désormais bien rodées - , à sa disponibilité, à son ouverture d'esprit et à l'enthousiasme dont il fait preuve sur les choses qui lui tiennent à cœur.

Je tiens aussi à remercier les deux directeurs successifs du laboratoire ARTEMIS qui m'a hébergé pendant quatre ans : Jean Mermet et Jean Fonlupt.

Jean Wolsack, durant mon année d'option à l'ENGREF, m'a accordé son soutien, sa confiance et sa sympathie. Ils ne m'ont pas fait défaut depuis.

Ce document et le système ELSA n'existeraient pas sans Claude Charlier. Il a été l'expert, sans cesse interrogé, dont les connaissances et l'expérience sont à la base d'ELSA. Plus encore, sa rigueur, sa modestie et son invraisemblable capacité d'observation m'ont guidé dans ma recherche sur le raisonnement spatial. Enfin que dire de la gentillesse, de la subtilité et de la patience dont il a fait preuve à mon égard dans son bureau ou sur le terrain...

Jérôme "Breton" Euzenat a partagé mon bureau pendant trois ans. J'espère qu'il s'en remettra. Le sachant avare en remerciements je vais essayer, pour une fois, de ne pas le provoquer. Qu'il sache néanmoins que je le remercie de tout ce qu'il m'a appris : entre autres, les bases d'UNIX, les

incantations magiques de vi et une partie importante de ce que je sais de Le-Lisp (pour Pascal et C, je crois qu'il faudra que je recommence une thèse).

Bertrand "Mac-Rousseau" m'a initié au matériel Apple ; sur *le Lisa* d'abord, puis sur le Macintosh. Il nous a montré aussi la voie à suivre en matière d'interface et de convivialité. La qualité de l'interface de Dynamac reste l'objectif à atteindre dans ELSA.

Patrice "Hugh" Uvietta et son calumet ont présidé à la confection de ce document. Mettre à jour les versions des logiciels du Macintosh, récupérer les disques mal en point, sortir la hache de guerre quand les virus rôdaient, découvrir les passages secrets de Word et ses vertigineux "raccourcis claviers". Il a aussi chasser le troupeau de fautes d'orthographe et l'anglais ou, pire, le métis. Patrice est un grand Sachem ... mais je crois bien que le grand Manitou - et la Comtesse - ne lui sera, cette fois-ci, d'aucun secours dans sa chasse à la contrepèterie.

Merci aussi à Claudine Meyrieux pour sa gentillesse, son efficacité, son empressément à dépanner ou à aider et ... pour les innombrables canards aussi.

Mes remerciements vont aussi aux membres du projet SHERPA du laboratoire ARTEMIS : Jose-Luis Aguirre-Cervantes, Pierre Fontanille, Mam'zelle Olga Mariño-Drews, Ana Simonet, Danielle "Bouchon" Ziebelin et les stagiaires : Mathias Chaillot, Jérôme Gensel, Pierre Girard, Bruno Orsier, Thierry Poncabaré...

Je tiens aussi à remercier l'ensemble de la Division Nivologie qui, malgré ma "délocalisation", m'a accepté en son sein. L'état d'esprit tout à fait particulier qui y règne, les fortes personnalités qui l'ont créé, l'ont conforté ou l'entretiennent ont toujours fait de mes passages "à la Division" ou de mes sorties sur le terrain de très bons moments. Outre Gérard Brugnot et Claude Charlier, je pense à Pierre Béghin, Saad Benharbit, Gilles Borrel, Florence Bouvet, Frédéric Ousset, Roland Burnet, Françoise Gay, Michel Gay, Gilbert Martinet, Géraldo Maciel, Fabrice Moutte, Mohamed Naaïm, François Rapin, Martine Roussel, Claire Solignac, Dominique Strazzeri, Jean-Michel Taillandier, Anne-Marie Uvietta, François Valla, Jean-Paul Vila.

Je tiens enfin à remercier Jacques Sauvenée, que j'ai encadré en 1989 pour ses stages de DESS-IDC. L'interface graphique d'ELSA lui doit beaucoup. L'équipe SHERPA lui doit, quant à elle, une caisse de Bordeaux (ceux qui n'ont pas pu patienter ont su l'apprécier).

Différents organismes publics ont participé financièrement au développement du système ELSA et ont ainsi permis le bon déroulement de cette recherche : il s'agit de la Région Rhône-Alpes en 1989 et 1990 dans le cadre de l'appel d'offre "Intelligence Artificielle" et la Délégation aux Risques Majeurs du Ministère de l'Environnement.

Je remercie aussi la Ville de Grenoble pour le second Prix du Jeune Chercheur qu'elle m'a décerné en 1990.

Laurent Buisson

RESUME

Les connaissances spatiales sont définies à partir d'une classification qui met en évidence l'importance de la topologie. L'examen d'applications différentes montre que le raisonnement spatial revêt souvent trois formes particulières non exclusives les unes des autres : la construction de topologies, l'inférence de propriétés spatiales et la localisation d'objets.

Les données et les mécanismes d'inférence qui sont utilisés peuvent être introduits dans des systèmes à base de connaissances qui servent alors de support au raisonnement spatial. Son graphe de dépendances possède, par ailleurs, des particularités qu'il est intéressant et, parfois, nécessaire d'exploiter pour gérer la dynamique de ces bases de connaissances spatiales. Les résultats d'inférences doivent en particulier être enregistrés et le maintien de la vérité qu'exige cet enregistrement peut être facilité en exploitant une forme de localité au sein de ce graphe.

Ces considérations prennent appui sur l'expérience du développement du système ELSA, destiné à l'analyse de sites avalancheux. D'autres utilisations des bases de connaissances spatiales ont également été étudiées : modélisation symbolique de phénomènes spatiaux, environnements de résolution de problèmes et bases de connaissances géographiques.

ELSA utilise une représentation de connaissances par objets, qui se révèle bien adaptée même si des extensions apparaissent utiles. Il s'agit notamment de la prise en compte, au sein du modèle de représentation, des relations entre objets, de l'insertion d'un filtrage constructif et d'une ouverture plus grande du langage de requête.

Le fait que ces extensions ne soient pas propres à l'espace et qu'elles puissent concerner d'autres types de connaissances suggère que le raisonnement sur l'espace, à défaut d'être spécifique, apparaît comme un révélateur des besoins du raisonnement en général.

ABSTRACT

Spatial knowledge can be defined through a classification which shows the large part played by topology. Three forms of spatial reasoning can be identified in several different application domains : topology building, inferences of spatial properties and localisation of objects.

Knowledge based systems can handle data and inference mechanisms involved in those three kinds of spatial reasoning and, as a result, support such a reasoning. Its dependancies graph has some features which imply a management of the dynamicity of spatial knowledge bases. The results of inferences must be recorded and the truth maintenance requested by this caching can take advantage of a locality concept defined in the graph.

These conclusions are based on the experience of the development of ELSA, a system dedicated to avalanche paths analysis. Some other domains can use spatial knowledge bases : symbolic modelling of spatial phenomena, problem solving environments and geographic knowledge bases.

ELSA uses an object-centered knowledge representation which is well adapted, even if some extensions appear useful, such as inter-objects relations, constructive pattern-matching and open query language.

Those extensions are interesting not only for spatial information handling but also for other kind of knowledge. This suggests that, far from being specific, spatial reasoning reveals some needs of general reasoning.

TABLE DES MATIERES

Introduction.....	1
Une thèse sur les systèmes à base de connaissances ?	1
Une application sous-jacente : l'analyse de sites avalanches.....	1
Les systèmes à base de connaissances	2
Traiter l'espace.....	3
Structure et présentation du document.....	3
1. Bases de connaissances, espace et informatique	5
1.1. Bases de connaissances	5
1.1.1. Connaissances	6
1.1.2. Systèmes à base de connaissances.....	7
1.1.3. Différents types de représentation et d'inférences	10
1.2. Traiter de l'espace n'est pas raisonner sur l'espace.....	13
1.2.1. Modélisation numérique.....	13
1.2.2. Raisonnement géométrique et robotique.....	15
1.2.3. Systèmes de traitement de l'information géographique.....	16
1.3. Représenter et raisonner sur l'espace	21
1.3.1. Description d'environnements.....	21
1.3.2. Modélisation symbolique.....	24
1.3.3. Conception.....	29
Conclusion.....	30
2. Raisonnement spatial.....	33
2.1. Connaissances spatiales.....	33
2.1.1. Une définition et une classification	34
2.1.2. Connaissances "pictorielles" ?	39
2.1.3. Une échelle de pertinence	41
2.2. Une définition.....	44
2.2.1. A propos de cette définition.....	44
2.2.2. Applications aux systèmes à base de connaissances.....	45
2.3. Description de l'espace et création de topologie	47
2.3.1. Connaissances "données" = connaissances euclidiennes ?	47
2.3.2. Création de topologie	48
2.4. D'une connaissance à l'autre.....	52
2.4.1. D'un type de connaissance à un autre	52
2.4.2. D'une représentation à une autre	53

2.5.	Localisation d'un objet dans une topologie.....	54
2.5.1.	Le problème.....	54
2.5.2.	Raisonnement ensembliste	54
2.5.3.	Raisonnement topologique sur les frontières	55
2.5.4.	Raisonnement métrique	55
2.5.5.	Raisonnement vectoriel	56
2.5.6.	Raisonnement géométrique euclidien.....	56
	Conclusion.....	56
3.	Dynamique des bases de connaissances spatiales	57
3.1.	Acquisition des connaissances	57
3.1.1.	Connaissances données et connaissances inférées.....	57
3.1.2.	Acquisition et inférence.....	58
3.2.	Inférences spatiales	60
3.2.1.	Attachements procéduraux.....	60
3.2.2.	Choix de la représentation et de l'inférence.....	62
3.2.3.	Enregistrement des résultats d'inférences spatiales	63
3.3.	Maintien de la vérité.....	65
3.3.1.	Un "formalisme".....	66
3.3.2.	Modification ≠ suppression + ajout.....	71
3.3.3.	Exploitation de la localité	74
	Conclusion.....	77
4.	ELSA et l'analyse de sites avalancheux	79
4.1.	Analyse de sites avalancheux.....	79
4.1.1.	Avalanches	79
4.1.2.	Analyser des sites	82
4.1.3.	Modéliser pour protéger.....	83
4.1.4.	Evolutions nécessaires.....	88
4.2.	Approche choisie	89
4.2.1.	Sciences de la terre, risques naturels et systèmes à base de connaissances.....	89
4.2.2.	Environnements de résolution de problèmes	90
4.3.	ELSA : un environnement de résolution de problèmes.....	92
4.3.1.	Base de méthodes.....	92
4.3.2.	Méthodes concurrentes.....	93
4.3.3.	Méthodes complémentaires.....	94
4.3.4.	Contrôle de programme numérique	95
4.4.	Communiquer avec des experts	96
4.4.1.	Objets de l'analyse.....	96
4.4.2.	Interface utilisateur.....	99
	Conclusion.....	101

5.	ELSA : une base de connaissances spatiales.....	103
5.1.	Inférer des propriétés spatiales pertinentes	103
5.1.1.	Des connaissances données aux connaissances pertinentes.....	104
5.1.2.	Représentations et inférences spatiales dans ELSA.....	108
5.1.3.	Utilisation des connaissances pertinentes.....	111
5.2.	Construire une décomposition du terrain	115
5.2.1.	Données disponibles et résultats désirés	115
5.2.2.	Différentes solutions.....	116
5.2.3.	Construction manuelle.....	117
5.2.4.	Construction procédurale	117
5.2.5.	Construction déclarative.....	123
5.3.	Dynamique de la base de connaissances	124
5.3.1.	Enregistrement	124
5.3.2.	Maintien de la vérité.....	126
5.3.3.	Evolution de la base au cours d'une analyse	127
5.4.	Situation actuelle du projet ELSA et perspectives	128
5.4.1.	Historique	128
5.4.2.	Etat actuel.....	129
5.4.3.	Perspectives	129
	Conclusion.....	130
6.	Intérêt des bases de connaissances spatiales.....	131
6.1.	Modélisation symbolique.....	131
6.1.1.	Construction de la représentation de l'espace.....	132
6.1.2.	Inférence des connaissances spatiales pertinentes	133
6.2.	Environnements de résolution de problèmes	134
6.2.1.	Partager une base de connaissances spatiales.....	134
6.2.2.	Inférer un maillage	135
6.3.	Bases de connaissances géographiques	136
6.3.1.	Dynamique des connaissances données.....	137
6.3.2.	Inférences et contraintes	138
	Conclusion.....	139
7.	Représentation des connaissances spatiales.....	141
7.1.	Des objets à représenter	141
7.1.1.	Objets	142
7.1.2.	Ensembles et listes.....	143
7.1.3.	Relations.....	145
7.2.	Des inférences efficaces	147
7.2.1.	Filtrage.....	147
7.2.2.	Attachement procédural.....	149
7.2.3.	Variables et passage d'arguments	150

7.3. Des requêtes	150
7.3.1. Consultation ou requête ?	151
7.3.2. Des filtres ?	152
7.4. Gestion de la dynamique	152
7.4.1. Modifications	152
7.4.2. Enregistrement sélectif	153
7.4.3. Implantation	153
7.5. Une grande base de connaissances	154
7.5.1. Limites de la taille de la base	154
7.5.2. Persistance et sécurité	156
7.5.3. Rapidité et accès	156
7.5.4. Partage de la base	157
Conclusion	158
Conclusion	159
Références	161
Annexe : les espaces mathématiques	173
Espaces topologiques	173
Espaces métriques	174
Espaces vectoriels	175
Espaces euclidiens	176

INTRODUCTION

La prise en compte de connaissances spatiales dans des systèmes à base de connaissances permet des formes intéressantes et nouvelles de raisonnement spatial dans ce qu'il convient d'appeler des bases de connaissances spatiales. Celles-ci peuvent concerner des domaines d'application entièrement neufs ou d'autres qui, jusqu'à présent, exploitaient des outils informatiques différents et plus classiques. Cette prise en compte met aussi en évidence les caractéristiques d'une des formes de raisonnement les plus importantes chez l'être humain. Elle permettra peut-être de se prononcer sur une spécificité éventuelle du raisonnement spatial par rapport à d'autres formes de raisonnement.

Une thèse sur les systèmes à base de connaissances ?

Les travaux sur les systèmes à base de connaissances peuvent être classés en deux grandes catégories. D'une part, des chercheurs définissent de nouveaux systèmes de gestion de base de connaissances, de nouveaux modèles de représentation. D'autre part, des ingénieurs cherchent à utiliser ces outils pour résoudre des problèmes particuliers : les applications.

Le présent travail se situe à mi-chemin entre ces deux approches. L'objectif est d'étudier la situation d'un système à base de connaissances chargé de manipuler des connaissances d'un type particulier : les connaissances spatiales. Il ne s'agit donc pas de proposer un nouveau modèle de représentation de connaissances ni d'expliquer comment un système opérationnel destiné à une application a été construit (même si c'est le cas), mais plutôt d'étudier comment un système à base de connaissances réagit (ou devrait réagir) lorsqu'on y introduit des connaissances spatiales qui présentent des traits qui leur sont propres et qui sont susceptibles de concerner de nombreuses applications.

Une application sous-jacente : l'analyse de sites avalanches

Ce document n'est donc pas consacré à la réalisation d'un système informatique destiné à une application particulière. Il n'en reste pas moins qu'une telle réalisation a, en permanence, accompagné la recherche.

Elle l'a tout d'abord rendu possible par la présence au laboratoire ARTEMIS de l'auteur de ces lignes, présence justifiée, pour la division Nivologie du CEMAGREF, par la réalisation d'un système d'aide à l'analyse de sites avalanches : le système ELSA (Etude des Limites de Sites Avalanches).

Cette application a aussi guidé ce travail de recherche car elle est à l'origine de l'essentiel des thèmes étudiés ici. Ceci ne signifie pas qu'ELSA exploite toutes les techniques présentées ici. La volonté de construire un système opérationnel a en effet souvent interdit l'utilisation de solutions intéressantes à terme mais dont l'implantation efficace aurait demandé trop de temps. A l'opposé, ce document ne contient pas l'ensemble des techniques

informatiques utilisées à l'occasion du développement d'ELSA : interface graphique, logiciel de maillage, codes numériques n'apparaîtront donc qu'occasionnellement. Néanmoins les problèmes abordés ici ont tous été soulevés à un moment ou à un autre par cette application.

Il ne faut donc pas être surpris de rencontrer tout au long de ce document des exemples liés, de près ou de loin, aux avalanches. Un effort a cependant été fait pour présenter d'autres applications. Mais les avalanches ne changent-elles pas des habituelles applications "bureautique" ou "conception de VLSI" de la recherche en informatique ?

Les systèmes à base de connaissances

Les systèmes à base de connaissances constituent un des thèmes les plus riches de recherche en informatique. Ils se trouvent aujourd'hui au confluent de deux courants de recherche.

Tout d'abord, le développement des systèmes experts, implantés le plus souvent suivant une architecture de système à base de connaissances, a mis en évidence l'intérêt qu'il y avait à construire en tant que tel un corpus de connaissances (qu'elles soient expertes ou de bon sens) et à le faire évoluer au cours du développement d'un système informatique. Il a aussi été montré que ce corpus de connaissances était un bon support pour faire réaliser par la machine une forme de raisonnement.

De plus, la capacité à expliquer le raisonnement constitue un des objectifs les plus ambitieux de la recherche en informatique. Comme cette capacité a été entr'aperçue dans quelques systèmes experts bâtis sur des architectures de systèmes à base de connaissances, ces dernières sont apparues comme une des voies les plus prometteuses.

Parallèlement à ces recherches qui concernent essentiellement l'intelligence artificielle, l'évolution des systèmes de gestion de bases de données met en évidence trois axes d'effort qui tendent tous à rapprocher futurs systèmes à base de connaissances et futurs systèmes à base de données :

- Des modèles de données différents du modèle relationnel apparaissent. En particulier, des modèles extensibles offrent à l'utilisateur la possibilité de définir ses propres types de données comme c'est le cas dans certains systèmes à base de connaissances.

- Par ailleurs, depuis quelques années, des chercheurs tentent d'installer la notion de calcul ou de production d'information nouvelle dans les systèmes à base de données. En effet, si dans certaines applications les calculs étaient initialement exécutés par des programmes extérieurs fonctionnant en entrées-sorties, cet aspect calculatoire est désormais présent dans les systèmes de gestion de base de données. C'est le concept de base de données déductive qui tend à se rapprocher, lui aussi, de celui de base de connaissances.

- Enfin, la gestion de la dynamique des bases de données implique des mécanismes capables de prendre en compte l'évolution des structures de données, des contraintes, les valeurs ou l'existence de ces données. Là encore c'est de raisonnement dont il s'agit. Les systèmes à base de connaissances semblent constituer une des solutions.

De la rencontre de ces différents courants naît donc une nouvelle génération de systèmes à base de connaissances dont les domaines d'application recouvrent ceux de ces deux courants et sont appelés à les dépasser.

Traiter l'espace

Le problème de l'introduction de connaissances spatiales dans de tels systèmes a été peu étudié ; d'abord parce que ces systèmes constituent une technique informatique récente et encore mal maîtrisée ; ensuite parce que les domaines d'applications des premiers systèmes experts étaient très symboliques et ne se préoccupaient pas de l'espace (à quelques notables exceptions près) ; enfin parce que les applications qui peuvent être concernées par l'introduction de connaissances spatiales dans un système à base de connaissances disposaient d'outils qui apparaissaient jusqu'à présent comme suffisants.

En effet, la modélisation de phénomènes spatiaux était la plupart du temps prise en compte par des "modèles numériques". Ces modèles sont bâtis sur un modèle physique (les équations du phénomène, conservatives ou comportementales), un schéma de résolution (issu de l'analyse numérique) et un programme procédural qui code le tout. Jusqu'à présent, ce type d'approche a permis d'étudier et de traiter un très grand nombre de phénomènes à dimension spatiale (aérodynamique, météorologie, physique, hydraulique, thermique, etc.). L'espace manipulé dans ce cas est un espace euclidien qui donne une signification aux équations. Les propriétés euclidiennes de cet espace sont exploitées de façon intensive.

A l'autre bout de l'univers informatique, c'est-à-dire du côté du stockage et non plus du calcul, des systèmes à base de données destinés à la géographie ont été développés ces vingt dernières années et donnent jusqu'à présent satisfaction. Le calcul des propriétés qui ne sont pas numérisées est réalisé par des programmes externes.

Cette situation est en complète évolution.

Tout d'abord, le développement des systèmes à base de connaissances rend leur utilisation de plus en plus répandue, en particulier dans des domaines où d'autres techniques étaient utilisées.

Ensuite, les applications où l'expertise s'exprime dans l'espace sont très nombreuses : elles concernent par exemple la modélisation dynamique de phénomènes spatiaux à l'aide du calcul symbolique. Les systèmes à base de connaissances apparaissent comme une solution possible surtout depuis le développement de la physique qualitative.

Enfin, dans le même temps, les applications spatiales qui se satisfaisaient des outils fondés sur les bases de données se font plus exigeantes : le modèle de données spatiales n'est pas satisfaisant ; les utilisateurs souhaitent introduire des capacités d'apprentissage ou de diagnostic utilisant les données spatiales ; enfin la gestion de la dynamique de ces données devient un objectif important.

Structure et présentation du document

Cette thèse est composée de 7 chapitres constitués de sections elles-mêmes détaillées en sous-sections.

L'approche choisie pour la présentation de ce document n'est pas linéaire. Il eût été possible de commencer par des séries de définitions relatives à l'espace, puis au raisonnement spatial, puis aux systèmes à base de connaissances, puis au raisonnement spatial dans de tels systèmes en distribuant dans chaque partie les considérations sur les applications... Une certaine logique eût sans doute été respectée mais la patience du lecteur par trop sollicitée et le rôle de l'application masqué.

L'approche retenue privilégie plutôt, dans un premier temps, une appréhension intuitive du raisonnement spatial, quitte à le formaliser ensuite, et conserve son unité à la description de l'application.

Dans le premier chapitre, la notion de *système à base de connaissances* est précisée afin de donner des contours nets à cette étude mais sans pour autant proposer une théorie unifiée des bases de connaissances. Une description de systèmes informatiques traitant de l'espace est ensuite présentée. Elle ne contient pas de définition mais a pour objectif de faire ressortir les *traits communs* et les *différences* entre ces systèmes. En particulier, la notion de raisonnement spatial est donnée de façon intuitive à l'occasion des exemples.

Au cours du chapitre suivant, après une classification des *connaissances spatiales*, une définition du raisonnement spatial est proposée. Il est montré que dans les systèmes à base de connaissances, ce raisonnement n'est possible qu'avec des systèmes qui manipulent de la connaissance profonde et des modèles de l'espace étudié. Certains aspects de ce raisonnement sont ensuite abordés : construction de topologie, enrichissement de descriptions spatiales et localisation d'objets dans une topologie.

Dans le chapitre 3, l'utilisation de systèmes à base de connaissances pour le raisonnement spatial est présentée sous l'aspect dynamique. A cette occasion, et dans un but de formalisation, différentes notions sont introduites qui concernent les systèmes à base de connaissances en général mais qui sont utilisées dans cette partie. Il s'agit des notions de connaissances données, inférées et implicites. L'accent est mis ensuite sur l'importance de l'enregistrement dans les bases de connaissances spatiales et sur la notion de modification dans un système de maintien de la vérité. Elle permet, en effet, une exploitation de la localité pour limiter la tâche d'un tel système.

Dans le quatrième chapitre, un système particulier est présenté. Il s'agit du système ELSA qui a été développé dans le cadre de cette thèse. Après une présentation générale du contexte de travail, les fonctionnalités recherchées sont présentées ; il s'agit de celles d'un environnement de résolution de problèmes.

Les différents aspects du raisonnement spatial dans ELSA sont détaillés dans le chapitre 5 : inférences spatiales, construction de la représentation du terrain mais aussi les problèmes d'implantation rencontrés qui illustrent les aspects dynamiques développés dans le chapitre 3.

Trois types d'applications sont détaillées dans le chapitre suivant pour lesquelles un système à base de connaissances qui raisonne dans l'espace est ou pourrait être utilisé. Ces applications concernent la modélisation symbolique, les environnements de résolution de problèmes et les systèmes d'information géographique.

Dans le chapitre 7, l'utilisation d'une représentation centrée objet pour les connaissances spatiales est discutée. A partir des besoins dégagés au cours des chapitres précédents, des lacunes sont mises en évidence et des propositions développées.

La conclusion reprend les acquis relatifs au raisonnement spatial, les applications concernées et évoque des ouvertures possibles. Elle tente, enfin, de se prononcer sur la spécificité éventuelle du raisonnement spatial dans les systèmes à base de connaissances par rapport à d'autres formes de raisonnement.

Des définitions importantes mais qui ne relèvent pas directement du thème de cette thèse sont présentées en annexe.

1. BASES DE CONNAISSANCES, ESPACE ET INFORMATIQUE

De nombreux systèmes informatiques traitent de l'espace. Ils calculent les caractéristiques de phénomènes qui s'y déroulent, stockent des données qui y sont réparties, analysent des distributions ou des évolutions, déterminent les trajectoires de véhicules ou de robots... Les techniques utilisées sont très nombreuses et très variées. Mettent-elles toutes en œuvre des formes de raisonnement spatial ? Manipulent-elles l'espace en tant que tel ? Certaines techniques sont-elles mieux adaptées à la représentation de l'espace et au raisonnement spatial ?

Il est impossible d'examiner toutes ces techniques et de répondre de façon exhaustive à ces questions qui débordent largement du cadre du raisonnement spatial dans les systèmes à base de connaissances. Il est cependant intéressant, pour mieux mettre en évidence ce cadre, de montrer en quoi ces systèmes à base de connaissances se différencient des autres approches.

C'est précisément l'objectif de ce premier chapitre. Pour cela, dans une première section, la notion de système à base de connaissances est définie. Ensuite, différents systèmes informatiques qui traitent de l'espace sans pour autant y raisonner sont passés en revue. L'objectif n'est pas encore de donner une définition précise du raisonnement spatial mais plutôt, à la faveur de quelques exemples ou applications qui seront repris plus tard, de faire sentir ce qui ne relève pas du raisonnement spatial. La dernière section traite d'autres systèmes informatiques pour lesquels, à la différence des précédents, l'approche retenue est proche de celle choisie ici. Elle concerne soit la représentation de l'espace avec une classification des connaissances spatiales soit des formes intéressantes de raisonnement spatial.

1.1. BASES DE CONNAISSANCES

Ce qui suit n'a pas pour objectif de constituer une théorie générale des systèmes à base de connaissances ni d'examiner les ressemblances ou les différences avec d'autres types de systèmes informatiques. Néanmoins ce document fait appel de façon systématique à un ensemble de définitions, de concepts, de situations et de comparaisons qu'il est utile de définir ou de décrire de façon précise.

Un certain nombre de notions ou d'approches originales seront introduites ultérieurement dans le chapitre 3 relatif aux bases de connaissances spatiales : connaissances données, connaissances implicites, modification, etc. Elles ne concernent pas seulement les bases de connaissances spatiales mais il est plus logique de les introduire dans le chapitre où elles seront exploitées.

Les chercheurs en intelligence artificielle ont compris, dès les années soixante, l'intérêt qu'il y avait à rassembler l'ensemble des connaissances relatives à un domaine d'application pour résoudre un problème sur ce domaine. Néanmoins, il faut attendre le développement, dans les années 70, des premiers systèmes experts, DENDRAL puis MYCIN, pour voir apparaître l'expression "base de connaissances". Le concept de système à base de connaissances n'a été utilisé, en tant que tel, qu'à partir des années 80 et reste

encore très lié à la notion de système expert. Celle-ci n'est pas pourtant exempte d'ambiguïtés selon que la définition fait référence à la fonction, à l'architecture, aux connaissances contenues ou aux trois à la fois [BUCH&84].

Pour les uns, un système expert est chargé de traiter des problèmes complexes que des approches ou des algorithmes classiques ne peuvent résoudre [CHAT&87]. Pour d'autres, les systèmes experts se caractérisent par la nature des connaissances, insuffisamment structurées et sujettes à évolution, qui ont été utilisées pour son développement [FARR85]. Ces connaissances peuvent provenir, la plupart du temps, d'un ou de plusieurs experts. Cette théorie est sans doute à l'origine du nom de système expert.

La notion de système expert est aussi parfois liée à un type d'architecture qui distingue base de faits et base de règles (qui dans ce contexte est aussi qualifiée de base de connaissances).

Par ailleurs, dans la littérature, les systèmes experts sont très souvent associés à un type de programmation particulier : la programmation par règles [CHAR&85] même si les auteurs précisent qu'il est possible d'utiliser d'autres types de langages de représentation de connaissances [GOND83].

Enfin de nombreux auteurs utilisent plus ou moins pour définir les systèmes experts la définition qui sera ici retenue de système à base de connaissances...

Ces ambiguïtés conduisent à préférer, dans le présent travail, la notion de système à base de connaissances à celle de système expert, même s'il apparaît clairement que les systèmes à base de connaissances constituent de bons outils pour atteindre les objectifs qui ont pu être assignés aux systèmes experts. Le fait que des systèmes à base de connaissances puissent ne pas être des systèmes experts incite d'ailleurs à maintenir cette séparation.

1.1.1. Connaissances

Il ne s'agit pas ici de définir le concept de connaissance mais simplement de considérer la définition suivante qui pose qu'une unité de connaissance est "la représentation symbolique d'un aspect de l'univers du discours" d'un individu donné [FROS86]. En d'autres termes, cette unité *en tant que telle* est perçue par cet individu comme une unité dotée de signification. Il faut noter que seule cette définition peut donner un sens à l'affirmation "cette connaissance est spatiale". Elle sera perçue en tant que telle comme élément du discours.

Dans la littérature, la connaissance est parfois considérée comme plus générale que la donnée car, à la différence de celle-ci, elle peut contenir des méthodes de calcul ou de raisonnement [FROS86].

Exemple 1.1 : En s'inspirant de Frost, il est possible de présenter des exemples de connaissances et de données :

"Jean est le mari de Marie" est une donnée (donc une connaissance).

"Jean est chercheur au laboratoire ARTEMIS"

"La moyenne des salaires au laboratoire ARTEMIS est de 9000,00 f par mois".

"Jean ne pense pas qu'il ait une vie saine"

"Pour obtenir la moyenne des salaires au laboratoire ARTEMIS, il faut faire la division de la masse salariale par le nombre de salariés"

"Si X travaille au laboratoire ARTEMIS et si X n'est pas salarié, alors X est étudiant"

Contrairement aux précédents, les deux derniers exemples de connaissances permettent d'obtenir des connaissances nouvelles ; le premier fait référence à une procédure de calcul ; le second à une règle logique.

1.1.2. Systèmes à base de connaissances

La plupart des définitions ne se rattachent qu'à un type bien particulier de systèmes à base de connaissances : les systèmes à règles de production. Pourtant le développement de PROLOG [GIAN&85] puis celui des représentations à base d'objets ou de schémas [MASI89] ont prouvé que d'autres représentations de connaissances pouvaient être utilisées. La définition doit donc s'affranchir du type choisi d'expression des connaissances. Celle qui est présentée ici est la plus communément admise.

Définition

Un système à base de connaissances est une machine virtuelle composée d'une base de connaissances et d'un moteur d'inférence. La base de connaissances est, à un instant donné, un ensemble dont les éléments sont des unités de connaissances de l'univers du discours d'un individu qui sera appelé dorénavant utilisateur¹. En exploitant les connaissances déjà présentes dans la base, le moteur d'inférence est capable de produire éventuellement de nouvelles unités de connaissance, donc de modifier la base. Il y a alors inférence ou raisonnement.

Le moteur d'inférence est construit par des informaticiens mais ne contient aucune unité de connaissance de l'utilisateur telle qu'elle a été définie ci-dessus. L'utilisateur (il peut naturellement y avoir plusieurs utilisateurs) n'accède donc pas à ce moteur d'inférence puisqu'il n'y rencontre aucun élément de son univers du discours.

En revanche, l'utilisateur construit la base de connaissances en y introduisant des éléments de son univers du discours. Si on souhaite parler de programmation, il ne peut que s'agir de programmation *déclarative* puisque les connaissances sont simplement ajoutées à la base sans que leurs modes ou procédures d'utilisation ne figurent dans la base.

La notion de raisonnement utilisée dorénavant dans ce document sera liée à cette définition. Il n'y aura raisonnement que si le producteur de ce raisonnement peut en présenter une décomposition ou une justification du type : "pour obtenir cette unité de connaissances j'ai utilisé les unités de connaissances suivantes".

Il convient enfin de préciser, d'emblée, que deux thèmes importants de la recherche sur les systèmes à base de connaissances ne seront pas évoqués dans ce document afin de limiter le thème d'étude. Il s'agit du problème du contrôle du raisonnement et des connaissances floues.

Inférences et modèles de connaissances

La capacité à inférer des connaissances nouvelles à partir des connaissances déjà présentes dans la base est une des fonctionnalités

¹Dans le dernier chapitre on introduira une distinction entre le concepteur de la base et l'utilisateur non concepteur. Cette distinction n'est pas utile pour l'instant. S'il y a plusieurs utilisateurs, ils partagent, tout au moins pour l'application, le même univers du discours.

fondamentales des systèmes à base de connaissances qui les distingue des systèmes de gestion de bases de données. *Ils créent de la connaissance*. Il faut préciser cependant que des ensembles (base de données + programmes de calcul) permettaient déjà d'enrichir des bases de données et qu'avec l'apparition des bases de données déductives [NICO&83], cette distinction a tendance à disparaître.

Une autre différence importante provient du fait que le moteur d'inférence *ne sait rien de l'application qui ne soit écrit* explicitement dans la base de connaissances². En effet, pour une base de données, les utilisateurs sont censés connaître plus de choses qu'il n'en est écrit dans la base, ce qui permet aux constructeurs de celle-ci d'exploiter ces connaissances et, par conséquent, de stocker moins d'information dans la base elle-même.

Exemple 1.2 : Frost donne un exemple de cette différence entre une approche base de données et une approche base de connaissances à propos de la date de naissance des patients d'un hôpital dans un système informatique chargé de la gestion de cet hôpital [FROS86].

- Dans une approche base de données, la date de naissance est enregistrée dans un champ de l'enregistrement patient. De plus, les programmes chargés d'exploiter la base de données ont été écrits pour exploiter cette information. La structure de donnée contient l'information qu'il y a une date de naissance pour chaque patient.
- A l'opposé, dans un système à base de connaissances, l'existence de la date de naissance pour chaque patient est déclarée de façon explicite dans la base de connaissances elle-même.

En particulier, pour revenir sur le problème de la création de connaissances, cette propriété interdit au système à base de connaissances d'inférer sans que la technique d'inférence soit décrite dans la base de connaissances elle-même.

Exemple 1.3 : Dans le système d'information géographique ARC-INFO [ESRI89] qui est un ensemble (base de données + programmes de calcul), l'inférence de certaines propriétés de couvertures spatiales (topologie, périmètres, surfaces) est réalisée par les programmes CLEAN ou BUILD qui acceptent en entrées des limites de polygones qui viennent d'être numérisées. Ces limites sont enregistrées dans la base de données ARC et sont exploitées par le programme CLEAN. La base de données a été construite pour contenir des lignes polygonales et les programmes pour les calculer. A aucun endroit dans ARC-INFO n'est déclaré que les commandes à exécuter pour obtenir la topologie d'une couverture sont CLEAN ou BUILD.

Ce qui précède dépend d'ailleurs de la différence entre concepteur du système (l'informaticien de la définition ci-dessus) et l'utilisateur dont les connaissances figurent dans la base.

Cet utilisateur bénéficie certes de ce travail déjà fait mais ne peut, en contre-partie, revenir dessus. En particulier il ne peut modifier les contraintes sur le type ou sur certaines méthodes d'accès. Et c'est regrettable car il y a, pour un même type de données, différentes façons de le représenter et de le manipuler, et chacune de ces façons peut être la mieux adaptée dans une application spécifique. Il n'est pas possible de choisir une de ces approches et

² Les moteurs d'inférence de "shell" destinés à une application générique particulière (le diagnostic technique par exemple) peuvent néanmoins contenir des connaissances sur la stratégie employée.

de maintenir la généralité. En même temps, il est clairement impossible d'offrir toutes ces approches dans le même système de gestion de bases de données [MANO&86].

Exemple 1.4 : Dans des systèmes dédiés aux applications spatiales, les types points ou rectangles sont souvent introduits. Dans de nombreux cas, pour des raisons d'efficacité, le stockage des données prend en compte, au niveau le plus bas, ces nouveaux types. Des opérations ou des requêtes sont possibles sur ces types : intersection, superposition, etc. La plupart se ramènent à des méthodes de localisation de points dans l'espace. L'informaticien qui a implanté chacun de ces systèmes a choisi des algorithmes qu'il a ensuite installés dans le système de gestion de la base de données. Ce choix a été fait selon des critères qui sont ceux de cet informaticien. Rien ne garantit que les utilisateurs auront des critères identiques. Kriegel [KRIE90] présente par exemple un grand nombre de méthodes dont aucune ne semble faire l'unanimité. Pour un système donné, il est impossible aux utilisateurs de changer les méthodes ou d'en rajouter.

Les bases de données extensibles sont une réponse à ce défaut. Un système tel que PROBE [MANO&86] a été précisément développé à cette fin. Il faut d'ailleurs remarquer que PROBE est aussi bien qualifié, dans la littérature, de système de gestion de bases de données extensible que de système de gestion de bases de connaissances.

Requêtes

A la différence des systèmes de gestion de bases de données, la plupart des systèmes à base de connaissances ont un langage de consultation relativement pauvre. Dans les systèmes de gestion de bases de données, le langage de manipulation de données permet d'émettre des requêtes dont la réponse s'obtient par une forme de raisonnement qui est en fait décrit dans la requête. Dans un système à base de connaissances, le langage de consultation est beaucoup plus pauvre.

Exemple 1.5 : Une application informatique manipule des polygones. Il est supposé ici que cette application est réalisée de deux façons différentes : d'une part, à l'aide d'un système de gestion de bases de données et, d'autre part, à l'aide d'un système à base de connaissances.

- "Quelle est la superficie du polygone1 ?" est une requête à laquelle le système de gestion de bases de données pourra répondre si la superficie est connue.

"Quels sont les polygones dont la superficie est inférieure à 4 m² ?" est une requête à laquelle pourra répondre le système de gestion de bases de données. Son langage de manipulation de données lui permet d'exécuter le raisonnement qui consiste à consulter l'ensemble de la base et à ramener les polygones dont la superficie est inférieure à 4 m².

- Dans un système à base de connaissances, il n'est possible de lancer depuis le langage de requête que des raisonnements décrits dans la base de connaissances.

"Quelle est la superficie du polygone1 ?" est une requête à laquelle le système à base de connaissances pourra répondre si la superficie est connue ou si, dans la base de connaissances, figure une indication d'une méthode applicable au polygone1 et qui permet d'en connaître la superficie.

"Quels sont les polygones dont la superficie est inférieure à 4 m² ?" est une requête à laquelle le système à base de connaissances ne pourra répondre sauf si la base de connaissances a été écrite en prévoyant cette requête (c'est-à-dire en indiquant une méthode d'inférence rendant ce résultat). PROLOG constitue d'ailleurs une exception car son système d'interrogation permet l'usage de variables et la composition de prédicats.

L'importance du langage de manipulation dans les bases de données par rapport à celle du langage de consultation dans les bases de connaissances explique certaines difficultés d'évolution des premières. Pour autoriser des requêtes efficaces, des algorithmes ont été installés dans le système de gestion de bases de données. Souvent, même, dans des buts d'efficacité, des structures de données cachées de l'utilisateur sont créées (chemins d'accès dans les systèmes de gestion de bases de données classiques, rectangle ou boîte enveloppant pour les bases de données spatiales). A aucun moment l'utilisateur ne peut accéder aux méthodes liées au langage de manipulation de données. Il n'en reste pas moins que ces langages offrent un confort d'utilisation bien plus grand que les langages de consultation minimaux des systèmes à base de connaissances.

Exemple 1.6 : Un système bâti sur une architecture de gestionnaire de bases de données a été enrichi pour prendre en compte des *types* particuliers comme le type *rectangle*. Le langage de manipulation de données permet des requêtes typiquement spatiales telles que l'intersection de rectangles. L'information utilisée pour ces requêtes n'est pas présente dans la base de données en tant que telle, elle est cachée dans le gestionnaire de bases de données. A l'opposé un système à base de connaissances réalisera ce type d'inférences en n'exploitant que les connaissances de la base. Celle-ci devra donc contenir le fait que R1 et R2 sont deux rectangles mais aussi que l'intersection de deux rectangles peut être réalisée en utilisant telle technique d'inférence avec tel type de contrôle.

Explication ?

Parce que toutes les connaissances de la base font partie de l'univers du discours de l'utilisateur, les connaissances utilisées pour les inférences lui sont aussi connues. Si ces inférences ne sont pas trop complexes, l'utilisateur peut retrouver lui-même les connaissances qui ont conduit à ces inférences. Si l'enchaînement de celles-ci est trop compliqué, il est possible de mettre en évidence le chemin qui a été suivi pour parvenir à ces inférences. Des graphes de dépendances peuvent être utilisés à cette fin. C'est cette propriété qui justifie l'utilisation du terme raisonnement et qui peut offrir des capacités d'explication. Bien qu'elle ne soit que rarement réellement exploitée dans des systèmes opérationnels, cette capacité à expliquer est, pour certains, la meilleure signature de l'intelligence artificielle. Kodratoff pense "que l'approche "intelligence artificielle" est caractérisée par la prééminence de l'explicabilité sur l'efficacité" [KODR86].

L'explication basée sur la seule *trace* du raisonnement n'est cependant pas la seule forme d'explication possible dans les systèmes à base de connaissances.

1.1.3. Différents types de représentation et d'inférences

Il n'est pas envisageable de passer en revue l'ensemble des formalismes et des techniques informatiques qui peuvent être utilisés pour la construction d'un système à base de connaissances. Il apparaît néanmoins aujourd'hui que ces systèmes peuvent être regroupés en trois familles :

- les systèmes de représentation à règles
- les systèmes de représentation à objets
- les systèmes mixtes qui correspondent à l'offre commerciale la plus importante.

Dans la suite de ce document, deux systèmes seront particulièrement présents. Il s'agit de PROLOG, système de représentation de connaissances basé sur la logique du premier ordre et assimilé ici, et par moi-même, de façon abusive, aux représentations à règles [FARR85], et de SHIRKA [RECH&89], représentation à objets. Si SHIRKA sera particulièrement étudié à la fin de ce document, PROLOG sera utilisé tout au long de celui-ci pour fournir un formalisme aux systèmes à base de connaissances.

Pour ce qui concerne les types d'inférence, seuls sont mentionnés ici ceux qui jouent un rôle dans la manipulation de connaissances spatiales : le filtrage (la règle logique de l'exemple 1.1) et l'attachement procédural (la procédure de l'exemple 1.1).

Le premier consiste à chercher, dans la base, des connaissances particulières vérifiant certaines propriétés. L'attachement procédural consiste à lancer une procédure externe.

Le *filtrage* est un mécanisme fondamental. Dans SHIRKA, il consiste à ramener l'ensemble des instances d'un filtre [RECH&89]. Ce filtre est une spécialisation de classe qui contient des restrictions de types et des contraintes. Dans PROLOG, une forme de filtrage peut être réalisée à l'aide de l'unification. Le filtrage offre aux systèmes à objets la puissance de la plupart des systèmes à règles avec malheureusement trop souvent des limites d'implantations importantes qui seront évoquées dans le dernier chapitre.

L'*attachement procédural* consiste à appeler des modules ou des fonctions correspondant à des programmes procéduraux. Les calculs géométriques et numériques sont installés dans la base de connaissances à l'aide de telles fonctions. Le problème de l'efficacité de ces algorithmes n'entre donc pas en considération de même que celui de leur programmation. Ces problèmes constituent d'ailleurs d'immenses thèmes de recherche. L'attachement procédural sera plus détaillé à l'occasion de la définition des connaissances implicites (cf. chapitre 3). En particulier, une définition en sera proposée.

Une remarque s'impose cependant d'ores et déjà qui concerne l'utilisation de l'attachement procédural. Son utilisation systématique semble rapprocher les systèmes à base de connaissances des techniques de programmation classiques et procédurales. En fait, il n'en est rien : en programmation procédurale, toute information qui n'est pas donnée (au sens de Von Neumann, c'est-à-dire qui n'est ni entrée ni sortie) est inaccessible à l'utilisateur même si cette information est pertinente. L'utilisation d'attachements procéduraux dans un système à base de connaissances est simplement un moyen de passer d'informations pertinentes à d'autres informations pertinentes à l'aide de méthodes d'inférence ou de calcul décrites en tant que telles par ailleurs mais qui, du point de vue de l'utilisateur, sont des boîtes noires.

Exemple 1.7 : Soit le problème de la détermination de la précipitation neigeuse critique sur un site avalancheux constitué de trois facettes triangulaires et recouvert d'une végétation connue.

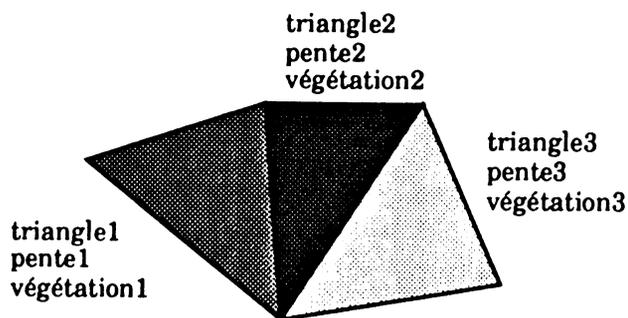


Figure 1.1 : Un site avalancheux fictif

• La résolution de ce problème peut se faire au sein d'un attachement procédural unique. Les données d'entrée sont les trois pentes et les trois végétations. La sortie est la précipitation critique. Si l'utilisateur désire évaluer l'importance relative de la pente ou de la végétation dans la stabilité d'un petit panneau, il ne peut trouver la réponse car ces informations demeurent internes à la procédure. La figure 1.2 indique la procédure et les intermédiaires de calcul. Le rectangle représente la procédure. Les variables en petits caractères sont internes à cette procédure donc inaccessibles.

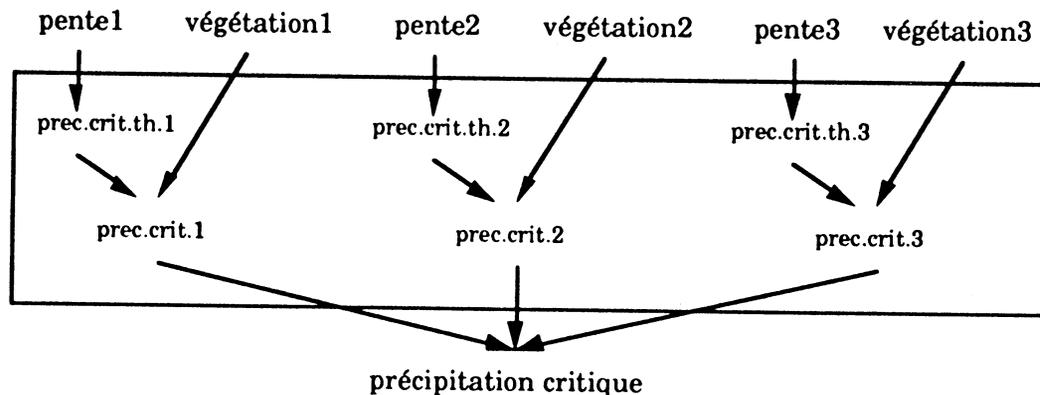


Figure 1.2 : La détermination de la précipitation critique par un module "boîte noire". Seules les entrées et les sorties sont accessibles à l'utilisateur.

• Avec une approche "systèmes à base de connaissances" toutes les informations pertinentes existent en tant que telles dans la base de connaissances. Les attachements procéduraux permettent simplement d'effectuer les inférences menant des informations connues à celles qui ne le sont pas. Il s'agit encore de procédures, mais cette fois, les informations internes à ces procédures sont sans signification autre que celle d'un intermédiaire de calcul ; si l'utilisateur n'y a pas accès, c'est, cette fois-ci, sans importance. Le développeur d'un tel système doit donc décider de ce qui doit être masqué dans l'attachement procédural et de ce qui doit être accessible à l'utilisateur. Sur cet exemple d'attachement procédural, l'approche base de connaissances se distingue nettement d'une autre approche... et montre tout son intérêt.

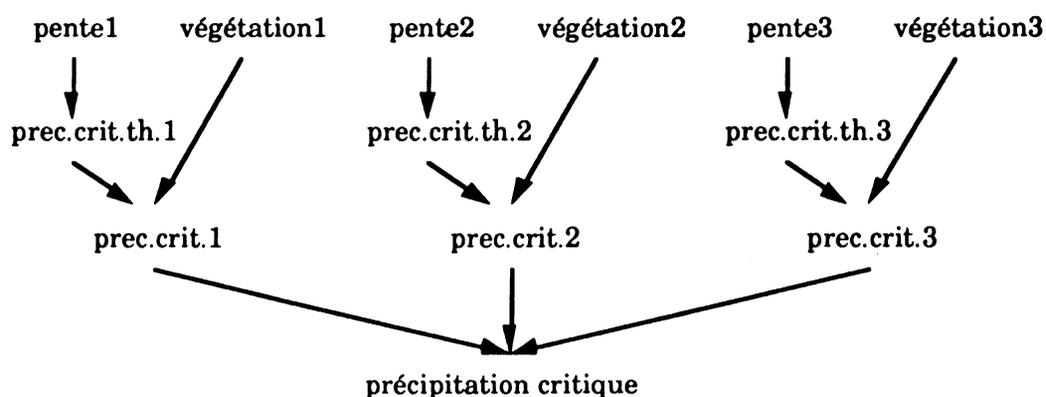


Figure 1.3 : La détermination de la précipitation critique par un système à base de connaissances. Toutes les caractéristiques importantes sont cette fois accessibles.

Les moteurs d'inférence de la plupart des systèmes à base de connaissances possèdent des mécanismes de contrôle complexes et très évolués. Un mécanisme de contrôle intervient lorsqu'une alternative se présente pour le choix d'une inférence. Ce choix peut concerner différentes règles ou clauses, il peut aussi concerner différentes méthodes d'inférence [CORD&85].

Dans ce document le problème du contrôle ne sera pas abordé. Les exemples de gestionnaires de bases de connaissances choisis pour illustrer ce document, PROLOG et SHIRKA, ne possèdent d'ailleurs pas de mécanismes de contrôle ou tout au moins de mécanismes de contrôle paramétrables.

1.2. TRAITER DE L'ESPACE N'EST PAS RAISONNER SUR L'ESPACE

Tout être doué de raison est naturellement capable de raisonner plus ou moins sur l'espace. La question qui est posée ici ne concerne donc que les systèmes informatiques existants et non pas les concepteurs de ces systèmes qui savent assurément raisonner sur l'espace... En d'autres termes, que le concepteur d'un système donné exploite sa capacité à raisonner sur l'espace lors de la conception de ce système ne suffit pas à dire que ledit système réalise un raisonnement spatial.

Un certain nombre de situations ou de systèmes sont maintenant passés en revue dans cette deuxième section. Cet exposé n'a aucune prétention d'exhaustivité. La diversité des disciplines évoquées l'interdit. Il s'agit, simplement, de mettre en évidence des points communs entre des applications, a priori, fort différentes et de montrer, sans se servir encore d'une définition, que les systèmes présentés ne mettent pas en œuvre un véritable raisonnement spatial.

1.2.1. Modélisation numérique

De nombreux programmes numériques modélisent des phénomènes ayant une réalité spatiale. Bâties sur des jeux d'équations différentielles ou d'équations aux dérivées partielles (cf. chapitre 4, les modèles numériques pour les avalanches), ils permettent de simuler des phénomènes telles que la météorologie, l'écoulement d'air, la température d'un fuselage, tout ceci dans l'espace. Toutefois, à aucun moment, de tels programmes ne sont capables de mettre en évidence la démarche suivie, les résultats intermédiaires obtenus. Si

leurs concepteurs ont mené un raisonnement complexe pour leur développement, ce raisonnement n'apparaît pas au cours de leur fonctionnement.

Exemple 1.8 : Soit le problème de Dirichlet homogène :

Sur Ω un ouvert borné par Γ de \mathbb{R}^n , on cherche une fonction u vérifiant :

$$-\Delta u + u = f \quad \text{sur } \Omega$$

$$u = 0 \quad \text{sur } \Gamma.$$

à résoudre avec $\Omega =]0, 1[$.

La résolution de ce problème peut se faire en appliquant la méthode des différences finies qui passe par la résolution d'un système linéaire.

Différentes solutions informatiques sont possibles pour implanter un tel système :

La première solution consiste à écrire un programme de façon procédurale qui accepte en entrée la description de f sur Ω et donne en sortie la description de la solution u (en fait une approximation de cette solution). Il est bien entendu impossible de mettre en évidence à l'aide de ce programme le raisonnement suivi par le numéricien pour obtenir cette solution. Le programme est une boîte noire. C'est la solution la plus répandue.

Certains systèmes, développés depuis peu sont basés sur une autre approche. Ils tentent de décomposer le raisonnement du spécialiste et de le faire reproduire par la machine. Ils mènent une forme de raisonnement. Ce type d'approche reste cependant ultra-minoritaire dans le domaine de la modélisation numérique.

Exemple 1.9 : La deuxième solution informatique pour traiter le problème de l'exemple 1.8, est celle qui a été retenue par Chaillot et Girard pour le développement du logiciel EVE [CHAILL&89]. EVE est bâti en effet sur l'outil de représentation de connaissances SHIRKA selon une architecture de systèmes à base de connaissances. Dans la résolution du problème, EVE suit les étapes du raisonnement du numéricien. Les propriétés du système, la preuve éventuelle de l'existence et de l'unicité de la solution, le maillage retenu, la matrice à inverser sont des connaissances accessibles en tant que telles à l'utilisateur. EVE réalise une forme de raisonnement.

Pour ce qui concerne la prise en compte de l'espace dans les systèmes de modélisation numérique, elle est assurée par une modélisation à base d'équations et de maillage, c'est-à-dire d'une *décomposition* de l'espace. Et de même que le raisonnement qui conduit à appliquer un théorème ou à choisir une méthode d'inversion de matrice n'est, la plupart du temps, pas apparent dans l'exécution du programme, de même, le raisonnement sur les conditions aux limites, sur le maillage ou sur les équations n'est pas mis en évidence dans les programmes numériques les plus classiques. De même l'espace, décomposé en maillage, n'est pas manipulé autrement que sous la forme de tableau.

Un système à base de connaissances tel que EVE est cependant capable de modifier un maillage uni-dimensionnel en fonction des résultats de calcul. En d'autres termes, il *modifie une représentation de l'espace* à partir d'un phénomène donné. Dans la mesure où il explicite les raisons de cette modification, il est possible de parler de raisonnement spatial. Cet exemple ne constitue toutefois qu'une exception et il faut considérer qu'à défaut de raisonner et de représenter l'espace, les systèmes de modélisation numérique sont incapables de raisonner sur l'espace.

1.2.2. Raisonnement géométrique et robotique

La robotique est vue généralement comme scindée en deux groupes [LAUM88] [ALAM&90]:

- La robotique manufacturière se préoccupe de saisie, de transport, et d'assemblage. Elle concerne en général un univers fortement structuré.
- La robotique d'intervention qui doit faire face à des univers inconnus.

Ces deux groupes ne sont pas séparés par une frontière très nette ; de nombreuses applications relèvent des deux groupes (robotique sous-marine, robotique spatiale).

Un grand nombre de systèmes développés en robotique (navigation ou assemblage) sont considérés classiquement comme dotés de capacités de raisonnement spatial. De fait, leur performance en précision et en rapidité exige des représentations de l'espace et des raisonnements dans cet espace bien adaptés [CROW&86] [LAUG88]. Mais ces raisonnements sont ceux des programmeurs qui modélisent l'espace et qui écrivent les algorithmes ; ce ne sont pas ceux que peuvent exhiber les machines sur lesquels tournent ces algorithmes. Ces algorithmes implantent souvent des raisonnements qui n'ont plus lieu dans l'espace physique à trois dimensions mais dans des espaces d'états, créations mathématiques, de dimension souvent supérieure (égale en général au nombre de degrés de liberté). Le raisonnement géométrique que tient l'informaticien lui permet d'implanter des systèmes particulièrement efficaces qui donnent des résultats concrets intéressants. Parler de raisonnement spatial au sein de la machine est par conséquent, dans la plupart des systèmes, délicat. L'objectif des chercheurs en robotique n'est pas, d'ailleurs, de faire faire, à des machines, du raisonnement spatial semblable à celui d'êtres humains, mais de commander des robots.

Il en va de même pour les systèmes dits de vision par ordinateurs. Nécessaires à un système complet chargé d'imiter l'homme dans l'espace, les capacités de vision se situent en amont du raisonnement spatial et concernent également la robotique. Elles débordent donc aussi largement du cadre de ce travail.

Ceci étant posé, le problème de la représentation de l'espace en robotique et en vision est parfois très proche de celui rencontré dans les systèmes à base de connaissances. En particulier, la robotique dite d'intervention s'intéresse à la navigation dans un espace inconnu qu'il faut *structurer* donc *décomposer*. De plus, certaines approches distinguent différents types de descriptions spatiales qu'elles exploitent de façon sélective, par exemple modèles *géométrique* et *topologique*. Les descriptions de l'espace ne sont plus alors toutes sur le même plan. Le système distingue les descriptions l'une de l'autre. Il n'y pas nécessairement raisonnement spatial mais la prise en compte de l'espace est beaucoup plus fine.

Enfin, certains robots utilisent des bases de connaissances pour la représentation de leur environnement et il en va de même pour certains systèmes d'acquisition et d'interprétation d'image tri-dimensionnelles [WALK&88]. Ces systèmes raisonnent sur l'espace, ils en manipulent une décomposition.

Les solutions qui viennent d'être présentées se distinguent de la plupart des systèmes de robotique qui traitent de raisonnement géométrique mais qui ne raisonnent pas sur l'espace parce qu'ils ne raisonnent pas. Elles seront abordées dans la section suivante de ce chapitre.

1.2.3. Systèmes de traitement de l'information géographique

La conservation et la manipulation d'informations géographiques constituent aujourd'hui un secteur de l'informatique en plein développement. L'essor des systèmes cartographiques ou des systèmes d'information géographique le confirment amplement. Les informations géographiques ont toutes des caractéristiques spatiales fortement marquées car elles n'ont de sens que sur la surface de la terre c'est-à-dire dans un espace. Ces informations sont liées à une ou plusieurs décompositions de l'espace ou à des réseaux répartis dans le plan.

La question qui est posée ici est la suivante : les systèmes de traitement de l'information géographique réalisent-ils un raisonnement spatial ? Si ce n'est pas le cas, quel type de description spatiale manipulent-ils ?

Damier [DAMI89] dresse un inventaire complet et très intéressant de ces systèmes. Il s'agit, d'une part, des systèmes d'information géographique et, d'autre part, des systèmes de gestion de bases de données géographiques. Selon Damier, les premiers sont issus des besoins des géographes en matière de stockage et de manipulation de données géographiques. Ils se caractérisent par leur développement poussé et une prise en compte très adaptée des caractéristiques géographiques. Les seconds correspondent aux travaux de chercheurs en bases de données spécialisés dans les applications géographiques. Cette distinction n'est pas retenue ici et, de façon abusive, il ne sera pas fait de différence entre systèmes d'information géographique et système de traitement de l'information géographique.

Systèmes d'information géographique

Un système d'information géographique est un système informatique destiné à enregistrer, stocker, analyser et représenter graphiquement des données réparties dans l'espace géographique.

Ce concept est apparu à la fin des années soixante en réponse à une demande de plus en plus forte d'outils destinés à la manipulation de données réparties dans l'espace. Aujourd'hui, les plus gros consommateurs de systèmes d'information géographique sont les sociétés qui ont un réseau à gérer (eau, gaz, électricité, télécommunications, transports) ou les organisations en charge d'aménagement (parcs naturels, agences d'urbanisme ...). Il faut noter qu'en France, l'Institut Géographique National compte utiliser des systèmes d'information géographique pour ses bases de données cartographiques et topographiques [IGN90].

Un système d'information géographique est d'abord un système à base de données, même si tous ne respectent pas les critères de sécurité, de partage, de séparation entre niveau physique et niveau logique (c'est d'ailleurs ici que se fait la distinction entre systèmes d'information géographique et bases de données géographiques).

Le plus célèbre des systèmes d'information géographique, ARC-INFO, est constitué de deux bases de données ARC et INFO (cf. Figure 1.4 et [ESRI89]). La première est une base de données spatiales qui contient la définition des objets spatiaux (points, arcs et polygones) et leurs caractéristiques topologiques (points de départ et d'arrivée, frontières, polygone à droite et à gauche, etc.) ; la seconde est une base de données descriptives qui contient les informations sur

les propriétés des différents objets spatiaux. La base de données topologiques permet d'éviter les redondances (une frontière commune à deux polygones n'est définie qu'une fois) et d'optimiser certaines requêtes (sur l'adjacence de polygones par exemple).

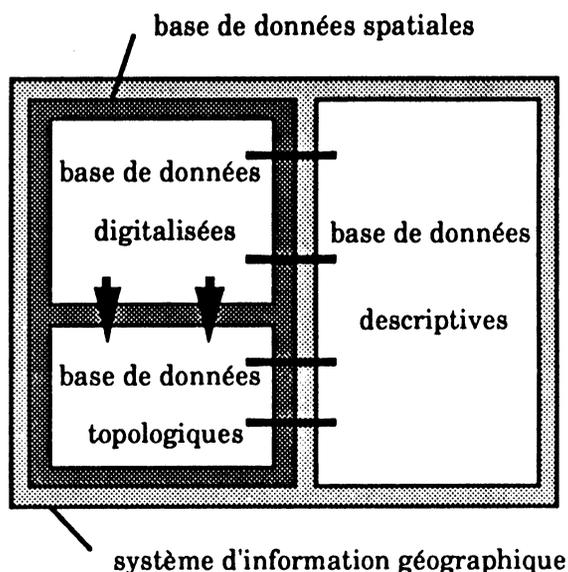


Figure 1.4 : L'architecture du système d'information géographique ARC-INFO. La base de données spatiales contient les informations relatives aux caractéristiques spatiales des objets manipulés. Ces objets ont par ailleurs d'autres propriétés contenues dans la base de données descriptives.

Les systèmes d'information géographique ne connaissent leur essor que depuis le milieu des années 1980. En effet, la nature des traitements à effectuer conduit à des calculs importants qu'il n'est possible de réaliser en interactif que depuis le développement des stations de travail (c'est-à-dire des machines possédant approximativement les caractéristiques suivantes : écran graphique, mémoire vive supérieure à 5 Mo, puissance de calcul supérieure à 1 Mips ; le tout pour moins de 100 000 francs 1990).

L'architecture des systèmes d'information géographique reflète la réflexion poussée qu'ont menée leurs concepteurs en ce qui concerne les structures de données spatiales et surtout l'importance de la description *topologique* [ESRI89]. Des travaux se poursuivent actuellement sur les modèles de données spatiales et, plus particulièrement, sur les modèles de données topologiques [EGEN&90].

Il faut noter que le calcul de cette description topologique est assuré par des programmes procéduraux externes à la base de données. Comme en modélisation numérique ou en robotique, *ces programmes ne raisonnent pas* et, a fortiori, ne raisonnent pas sur l'espace, même si leurs concepteurs ont raisonné sur l'espace pour les écrire.

Stockage des données

De nombreux travaux concernent directement les problèmes liés au stockage des données et à leur accès dans les systèmes de traitement de l'information géographique. Des structures de données particulières et dédiées aux données spatiales ont été développées. C'est le cas des "quadrees", des R-arbres et autres index spatiaux [SAME90]. Ces structures de données ne sont développées que pour les données spatiales. Il n'y a plus séparation entre le niveau logique et le niveau physique. Certains auteurs croient que cette séparation n'est qu'une chimère à partir du moment où l'on désire des systèmes efficaces et rapides. En d'autres termes, un stockage physique efficace de données spatiales doit exploiter la nature spatiale des données

stockées [NIEV90]. La plupart du temps, les techniques qui exploitent cette nature spatiale des données sont bâties sur des analyses du raisonnement spatial : décomposition hiérarchique, boîte enveloppante...

Cette approche revient à cacher dans la structure physique des données leur nature spatiale et, par conséquent, le raisonnement spatial. L'utilisateur ou les programmes qui consultent la base savent que les données sont spatiales mais cette nature spatiale n'est indiquée nulle part dans la base. Cette solution s'éloigne de l'approche base de connaissances et, partant, de l'explicitation d'un raisonnement spatial (cf. exemple 1.4).

Modèles de données et requêtes

Il n'existe aucun modèle prenant en compte les différentes formes de relations spatiales et les structures de données associées dans les systèmes d'information géographique [NCGI89]. Par exemple, le modèle relationnel est peu utilisé. En effet, les systèmes d'information géographique les plus répandus ayant vu leur développement commencer avant l'apparition du concept de base de données relationnelle, ils ne sont pas implantés sur des systèmes de gestion de bases de données relationnels. Depuis, des interfaces entre certains systèmes d'information géographique et des systèmes de gestion de bases de données relationnels ont commencé à apparaître (interface entre Arc-Info et les systèmes de gestion de bases de données relationnels par le biais de SQL par exemple).

Certains chercheurs ont implanté des systèmes d'information géographique sur des systèmes de gestion de bases de données relationnels. Ces tentatives ont mis en évidence l'inadéquation du modèle relationnel.

Les travaux les plus récents sur les systèmes d'information géographique (liens avec le langage de requêtes relationnel SQL, implantations sur des systèmes de gestion de bases de données relationnels entre autres) les ont rapprochés de la recherche sur les bases de données. Devant les lacunes du modèle relationnel, les chercheurs proposent des implantations utilisant d'autres types de modèles de données, ou des extensions des systèmes de gestion de bases de données actuels.

Ces recherches présentent en général deux caractéristiques. D'une part, elles proposent des formalismes plus complets pour prendre en compte la nature spatiale des données, d'autre part, elles définissent des types de données supplémentaires, des organisations de données et des requêtes spatiales. Les opérations réalisées jusqu'à présent dans les systèmes d'information géographique par des programmes externes sont alors prises en charge par le système de gestion de bases de données.

David construit ainsi le système SPATIAREL [DAVI88] à partir d'un système de gestion de bases de données relationnel en l'étendant grâce à un nouveau type "spatial" défini dans un espace topologique (qui peut être le plan ou d'ailleurs un espace 3D). David donne un cadre mathématique précis à ses recherches ; il exploite en particulier des définitions issues de la topologie. Par exemple, les éléments de son domaine spatial sont les ouverts d'un espace métrique (cf annexe). Il définit ensuite les opérations classiques sur ce domaine spatial et crée de nouveaux opérateurs de fusion et de décomposition qui agissent sur la topologie de la carte. A partir de l'extension du langage SQL ainsi définie, David peut émettre des requêtes spatiales qui constituent des questions dont la réponse passe par des formes de raisonnement spatial.

Exemple 1.10 : David propose de sélectionner des polygones du plan par une sélection par chevauchement. Il s'agit de déterminer quels sont les polygones recouverts en partie par un rectangle donné. Le raisonnement qui donne la réponse est un raisonnement spatial. Il part d'une décomposition de l'espace et envisage la superposition de différents objets. Toutefois, David ne propose pas de décrire le raisonnement à suivre dans la base de données. SPATIAREL n'est pas un système à base de connaissances.

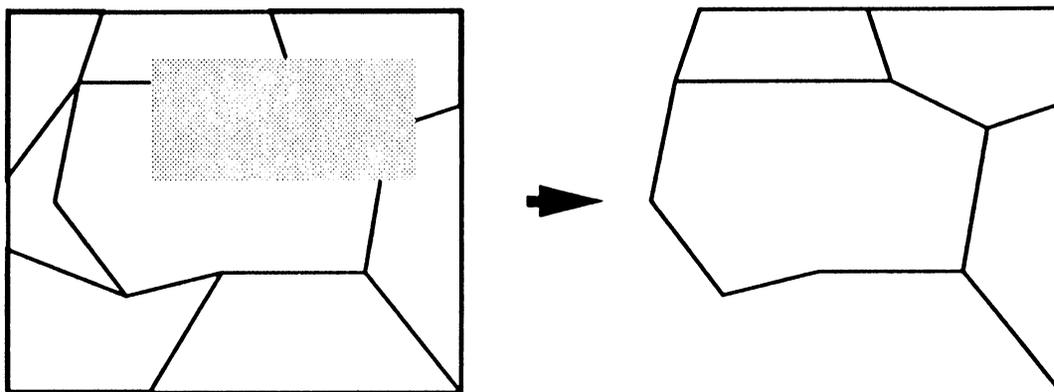


Figure 1.5 : Un exemple de sélection par chevauchement dans le système SPATIAREL.

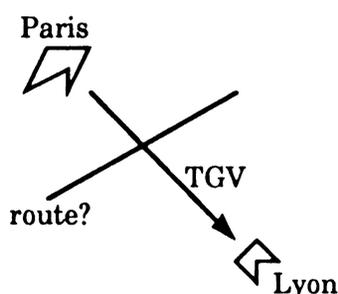
Scholl et Voisard [SCHO&89] proposent de leur côté un langage de requêtes extensible et adaptable pour la manipulation de cartes thématiques. Les requêtes auxquelles ils se proposent de répondre sont du type :

“Représenter les communes de la région Midi-Pyrénées”

“Représenter les communes qui votent à plus de 20% pour le Parti des Biomanes Branchés”.

Pour cela ils définissent les notions de régions et de cartes et des opérations sur ces entités assez proches de celles définies par David. A cette occasion une algèbre d'objets complexes est définie. L'espace manipulé est l'espace géométrique. Les régions qui sont à la base du type “spatial” sont en effet des polygones.

Mainguenaud et Portier [MAIN&89] travaillent sur un langage graphique pour l'interrogation de système d'information géographique. Ce langage, CIGALES est chargé de traduire la requête graphique dans le langage fonctionnel choisi par les auteurs.



Exemple 1.11 : “Quelles sont les routes croisant la voie du TGV entre Paris et Lyon ?”

Cette requête est exprimée graphiquement par la figure 1.6.

Figure 1.6 : Une requête graphique exprimée dans le système CIGALES

Par ailleurs, ils se proposent d'utiliser un modèle de données orienté objet pour implanter leur système. Leur base de données devrait ainsi supporter des méthodes de calcul “encapsulées” dans les objets (par exemple, le calcul de la densité de population d'une ville).

Les trois travaux de recherche présentés concernent tous le développement d'un langage de manipulation de données qui permet à l'utilisateur de décrire *un raisonnement spatial assez simple* grâce à des opérateurs spatiaux particuliers. Ce raisonnement est ensuite appliqué à la base et son résultat constitue la réponse à la requête. Mises à part celles auxquelles se proposent de répondre Mainguenaud et Portier qui peuvent être basées sur des mécanismes de déduction³, les requêtes spatiales sont "basées sur l'aspect général de la carte" [MAIN&89] c'est-à-dire qu'elles ne font pas appel à d'autres informations que celles présentes dans la base de données.

Par ailleurs, d'autres tentatives sont actuellement en cours pour exploiter les propriétés des nouveaux systèmes de gestion de bases de données (les systèmes de gestion de bases de données orientés objets et les systèmes de gestion de bases de données extensibles).

C'est le cas du système PROBE. Le système de gestion de bases de données PROBE est chargé de manipuler de façon efficace différents types de données sans préjuger de leur nature [MANO&86]. PROBE doit donc être rangé dans la catégorie des systèmes de gestion de bases de données extensibles (il est parfois également considéré comme un système de gestion de base de connaissances). Cependant son modèle de données, fonctionnel, peut être alimenté par des descriptions orientées objets. PROBE admet deux types d'objets : les entités et les fonctions. Ces dernières représentent les attributs ou les relations des premières. Les entités sont des objets complexes. Manola et Orenstein définissent ainsi des objets spatiaux du type ensemble-de-points (points, lignes, surfaces et volumes) qui peuvent être le co-domaine de certaines fonctions comme forme ou extension-spatiale. Ils définissent ensuite des opérations propres aux entités spatiales. Pour l'implantation de leur système, ils exploitent une "géométrie approximative" qui a pour objectif d'accélérer certaines requêtes. Chaque entité spatiale est approchée par sa boîte enveloppante. A la différence des précédentes cette approche tente d'utiliser un outil de représentation général pour la représentation de descriptions spatiales.

Pour sa part, Damier voit dans ces systèmes extensibles une solution au problème du "multi-média". C'est l'approche qu'il a retenue pour le développement du système OMEGA [DAMI89]. Ce problème du multi-média est cependant assez éloigné de ceux que ce document se propose d'étudier.

Enfin, depuis quelques années, des chercheurs tentent d'exploiter des systèmes à base de connaissances pour réaliser des systèmes d'information géographique. C'est le cas des travaux développés à Santa-Barbara et présentés dans [SMIT87b]. Malheureusement dans les articles sur ces travaux, l'exploitation des capacités "intéressantes" des systèmes à base de connaissances est passée sous silence. Il est par conséquent impossible de déterminer s'il y a raisonnement spatial ou non.

De leur côté, Mohan et Kashyap se proposent d'utiliser, dans une approche base de connaissances, des solutions basées sur un modèle orienté objet pour la

³ "Quelle est la prévision d'extension du feu de forêt localisé à 50 km au nord-ouest de Toulon sachant que le vent souffle force 6 direction sud-sud-est ?" Cette requête exige du système qui doit y répondre des connaissances qui ne sont pas seulement celles d'un système d'information géographique.

manipulation de description spatiale (en l'occurrence des images à base de pixels) [MOHA&88]. Ils exploitent dans la structuration des connaissances des notions de multi-héritage destinées à rendre compte de certaines propriétés spatiales. Ils envisagent ensuite d'introduire des mécanismes d'inférence capables de produire de nouvelles connaissances spatiales à partir de connaissances données au système. Ces mécanismes d'inférence sont présentés sous forme de prédicats. La solution de Mohan et Kashyap peut par conséquent être considérée comme un support pour le raisonnement spatial.

En conclusion sur les systèmes de traitement de l'information géographique, s'ils manipulent une décomposition de l'espace en sous-parties, et s'ils sont le produit d'une réflexion poussée sur la nature des informations spatiales (les descriptions topologiques sont traitées en tant que telles pour des raisons d'efficacité), leur implantation les rattache à des systèmes de gestion de bases de données qui interdisent de parler de raisonnement au sein du fonctionnement du système informatique. Toutefois, ils offrent la possibilité de raisonner sur l'espace dans la mesure où la requête, que permet de formuler le langage de manipulation de données, contient un raisonnement. De plus, les approches les plus récentes sur les bases de données orientées objet incorporent des mécanismes de calcul dans la base elle-même.

Les systèmes de traitement de l'information géographique actuels ne réalisent donc pas, dans leur immense majorité, de raisonnement spatial. Il n'était cependant pas inutile de les décrire en détail car le traitement de l'information géographique constitue un domaine d'application pour les systèmes à base de connaissances spatiales. Ceci fera l'objet d'une section dans le chapitre 6. Par ailleurs, ils partagent tous une décomposition de l'espace et exploitent en tant que telles des notions topologiques.

1.3. REPRESENTER ET RAISONNER SUR L'ESPACE

Les systèmes qui viennent d'être présentés ne réalisent pas pour l'essentiel de raisonnement spatial parce qu'ils ne raisonnent pas. De plus, les descriptions de l'espace qu'ils manipulent - même si elles mettent en évidence, pour certaines, la topologie - ne traitent pas d'autres types de structures ni ne les exploite de façon sélective.

Cette section s'intéresse à des systèmes d'intelligence artificielle qui, soit manipulent différentes structures d'espace, soit raisonnent sur des connaissances apparemment spatiales. Raisonnent-ils pour autant tous sur l'espace ?

L'ordre de présentation choisi reprend une classification de Charniak et McDermott. Dans leur ouvrage de présentation générale de l'Intelligence Artificielle [CHAR&85], ils consacrent un court paragraphe au raisonnement spatial. Ils énumèrent trois types de problèmes : la détermination de chemin, les problèmes de "naive physics" ou de modélisation symbolique et la conception de pièces ou d'outils. C'est cet ordre qui a été repris dans cette section.

1.3.1. Description d'environnements

Les systèmes qui suivent ont en général deux fonctions : construire des cartes et des représentations d'environnements ou déterminer des chemins. Comme Charniak et McDermott l'expliquent, la détermination de chemins ou navigation possède deux aspects distincts selon que l'espace est structuré ou non [CHAR&85]. Dans le cas où il est structuré le problème peut consister à construire un graphe puis à le parcourir. Cette solution peut cependant être

enrichie par des techniques de raisonnement. Dans le cas où l'espace n'est pas structuré, sa structuration est une solution possible. En tout état de cause, le parcours d'un graphe déjà existant ne relève pas du raisonnement spatial. C'est d'ailleurs un problème qui dépasse largement le cadre de cette étude.

Les systèmes qui vont être présentés ne sont pas tous des systèmes à base de connaissances tels qu'ils ont été définis précédemment. La plupart des articles ne mentionnent que peu de choses quant à l'implantation et il est par conséquent difficile de se prononcer sur la réalité d'un raisonnement dans ces systèmes.

TOUR, QUALNAV et NX : les travaux de Kuipers

Kuipers a développé, en collaboration avec d'autres chercheurs (Levitt, Byun...), une série de trois systèmes TOUR, QUALNAV et NX [KUIP&88^a]. Le contexte d'utilisation est la navigation de robot mobile. Kuipers définit, à chaque fois, l'objet de ses études, le "large-scale space", comme un espace dont la structure ne peut être observée d'un seul point de vue. L'approche choisie est celle d'un robot mobile qui se déplace dans cet espace en enregistrant des informations.

TOUR est le plus ancien des systèmes développés par Kuipers [KUIP78]. TOUR est en quelque sorte un voyageur dans un environnement fixe qui perçoit, à un instant donné, par des capteurs (qui ne sont pas précisés) une vue "égocentrique" de son environnement. A côté de ces vues, TOUR manipule des actions A qui représentent des déplacements (translation ou rotation). Il s'agit de changements d'état d'une vue V à une autre V'.

TOUR acquiert ainsi, au gré de ses déplacements des séries de procédures qui relient des vues à d'autres vues par des actions. Il constitue ainsi une carte procédurale de son environnement.

Il lui est ensuite possible de construire des cartes dites topologiques qui peuvent être de deux types : un réseau des lieux et des chemins et une carte des régions. Le premier type s'obtient par une dégradation des schémas $\langle V, A, V' \rangle$. La carte des régions se construit à partir des chemins. Ces deux cartes topologiques peuvent être exploitées ensuite pour trouver des chemins en exploitant des heuristiques inutilisables dans un graphe classique (en particulier grâce à la carte des régions).

TOUR construit également des cartes appelées métriques qui exploitent l'aspect quantitatif et mesuré des actions (vecteur de la translation et angle de la rotation). Ces cartes peuvent être locales (description quantifiée des chemins qui rayonnent d'un lieu, description quantifiée des lieux sur un chemin) ou plus globales (insertion des lieux et des chemins dans un système de coordonnées).

Bien qu'il soit difficile d'identifier la technique de programmation de TOUR (visiblement très ad-hoc, vu le nombre de versions...), TOUR est un système intéressant par la réflexion que Kuipers a menée sur la représentation de l'espace et sa *décomposition* en différentes cartes.

QUALNAV à la différence de TOUR est moins orienté vers la représentation d'environnements urbains et concerne plutôt des espaces larges, ouverts et dotés de singularités qui sont définies par l'utilisateur [KUIP&88^a] [LAWT&88]. Comme l'espace est moins structuré, QUALNAV crée une structure en reliant entre eux ces singularités par des LPBs ("landmark-pair boundaries").

QUALNAV manipule des points de vue qui sont constitués des singularités qu'il peut observer d'une position. Il manipule aussi des régions d'orientation constituées des LPBs d'un point de vue. Ces points de vue et ces régions d'orientation peuvent être acquis par les capteurs de QUALNAV. Les déplacements peuvent être repérés par rapport aux points de vue, par rapport aux régions d'orientation ou dans un repère de coordonnées. QUALNAV développe alors des cartes métriques et topologiques basées respectivement sur les points de vues et les régions d'orientation.

Le robot NX généralise l'approche de TOUR en manipulant des données continues [KUIP&88^a][KUIP&88^b]. Le problème central est donc de définir les lieux et les chemins qui permettent à un réseau d'être plaqué sur un environnement continu. A la différence de TOUR et de QUALNAV (qui sont des modèles qui fonctionnent sur simulateurs), NX est un véritable robot doté de capteurs ultrasoniques qui lui donnent la distance de l'obstacle le plus proche (NX se déplace dans un bâtiment). Les déplacements sont enregistrés en notant l'ensemble des points singuliers choisis par NX (critère de "distinctivité" où une mesure atteint un extremum).

Une fois les lieux et les chemins définis, NX construit une carte topologique et une carte métrique comparables à celles générées par TOUR.

Dans ces trois travaux, Kuipers accorde une place importante au traitement des erreurs. Il est convaincu qu'il est impossible de construire une carte précise dite métrique, c'est-à-dire comportant de nombreuses mesures. Il pense qu'une architecture mettant en évidence différentes couches d'informations spatiales peut être une réponse à ces erreurs, *la couche topologique constituant la structure de base.*

SPAM et les travaux de McDermott

McDermott est un des auteurs les plus prolifiques dans le domaine de la recherche en intelligence artificielle. Il a travaillé entre autres sur le raisonnement spatial et les inférences spatiales [MDER80] [MDER&84].

Dans [MDER80], il propose d'utiliser des cartes floues dans lesquelles les objets sont approchés par des cadres de références. C'est l'idée souvent exploitée dans les systèmes d'information géographique et les bases de données spatiales de la boîte enveloppante d'un polygone ou d'un volume.

Développé par McDermott et Davis [MDER&84], le système SPAM s'intéresse au tracé de routes dans un territoire incertain. Les solutions que ces auteurs apportent aux problèmes de l'imprécision et du flou ne sont pas présentées ici.

SPAM manipule des composants propositionnels ("propositional components") et une carte floue ("fuzzy map"). Les premiers expriment des faits tels que:

(partie bras15 corps23)

Mc Dermott les qualifie de "topologiques". La carte floue contient des faits appelés métriques. Deux exemples sont donnés :

(> (distance illinois new-york) (3000 km))

(entre illinois new-york californie)

Si le générateur de route utilise des techniques de planification mises au point pour d'autres activités que le raisonnement spatial, il décompose son travail en tâches selon la nature des informations qu'elles exigent.

Dans un premier temps, le système génère le vecteur qu'il doit parcourir, puis un chemin topologique est recherché en tenant compte des discontinuités introduites par les obstacles. Le système passe donc du modèle de la "fuzzy map" au modèle des "propositional components". Sur chacun de ces modèles, il possède des outils d'inférences particuliers.

3D-FORM

Le système 3D-FORM (Frame-based Object Recognition and Modeling) est destiné à la représentation et au raisonnement sur les objets tri-dimensionnels en particulier dans le contexte de l'interprétation de scènes [WALK88].

La représentation des objets retenue est très géométrique. Elle utilise un modèle de schéma avec des attributs qui correspondent aux propriétés géométriques des objets simples ou aux relations géométriques qu'ils entretiennent entre eux. Les objets composites sont construits en renseignant des attributs comme forme ou composants. Les objets sont donc décrits dans leur structure même.

Ce système est un système à base de connaissances. En particulier, les propriétés de certains attributs géométriques sont calculées par attachement procédural en exploitant les valeurs d'autres attributs. De même, il est possible d'exprimer de façon déclarative des contraintes sur des objets. Le contrôle du raisonnement est assuré de façon déclarative par un attribut qui mesure la qualité et le degré de description de l'objet. Cet attribut permet de choisir les objets les mieux décrits qui seront filtrés de manière prioritaire.

3D-FORM est utilisé entre autre pour l'interprétation de données définies sous la forme de figures "fil-de-fer".

Les systèmes de manipulation d'environnement qui viennent d'être décrits présentent tous (à l'exception du dernier) une classification des descriptions spatiales (métrique, topologique...) liées à des utilisations différentes. Le dernier présente clairement des caractéristiques de systèmes à base de connaissances mais ne manipule qu'une représentation géométrique.

1.3.2. Modélisation symbolique

Charniak et McDermott estiment que le deuxième domaine d'application où le raisonnement spatial se rencontre est celui de la "naive physics" qui est ici assimilée (peut-être à tort ?) à la nébuleuse que constituent physique qualitative, modélisation qualitative, modélisation symbolique... Ces disciplines sont toutes fondées sur l'hypothèse "cognitivist" qui dit que la "cognition peut être définie par la computation de représentation symbolique" [VARE89]. Cette hypothèse est à l'origine du développement des systèmes experts.

Quatre systèmes présentés maintenant manipulent des connaissances spatiales. Ils ne réalisent pas pour autant tous les quatre un raisonnement spatial. En effet bien que manipulant une représentation de l'espace FROB ne met pas en évidence un raisonnement. A l'opposé, un système à base de connaissances comme ERASME, qui effectue un raisonnement prend en compte des connaissances spatiales sans pour autant raisonner sur l'espace. ELFIN et NEUREX raisonnent quant à eux sur l'espace en tant que tel.

FROB

Forbus a développé le système FROB [FORB83] pour mener des raisonnements sur le comportement d'une particule soumise à la pesanteur et

qui peut se déplacer dans un puits. Pour décrire l'espace étudié, FROB utilise deux types de structures : le diagramme métrique ("metric diagram") et le graphe spatial ("space graph").

En ce qui concerne le diagramme métrique, ses éléments sont les points, les segments, les paraboles. Il s'agit ensuite d'identifier l'élément (point, segment, etc.), de repérer des intersections éventuelles ou de distinguer des côtés, des intérieurs ou des extérieurs. La réponse à ces trois types de questions réside toutefois dans la définition des éléments du diagramme métrique.

Le graphe spatial présente les relations entre les éléments du diagramme et les identifications de ces éléments (il semble, à ce sujet, qu'il y ait redondance avec le diagramme). Les relations sont du type "up" ou "left", c'est à dire que les directions verticales et horizontales ont été sélectionnées (... il s'agit d'un problème de chute des corps...) et qu'on classe les vecteurs "positions relatives" dans les quadrants.

Le mouvement de la particule est décrit par la séquence d'actions ("Action Sequence") qui contient la description des états de cette particule entre chaque mouvement. Un état est l'indication d'une position, d'une vitesse et d'une direction. Ces informations sont représentées dans le diagramme métrique ou dans le graphe spatial.

Il est difficile de dire si FROB est un système à base de connaissances et s'il réalise un raisonnement. Les détails d'implantation ne sont pas suffisamment décrits dans les papiers. Cependant, comme les systèmes de représentation d'environnements, FROB exploite une classification des descriptions spatiales en les plaçant dans deux modèles distincts : le graphe spatial et le diagramme métrique.

ERASME

ERASME est un système à base de connaissances qui exploite les connaissances de différents experts pour réaliser le diagnostic d'entretien d'une chaussée routière [ANTO&90]. Pour cela une description en différentes couches de cette chaussée lui est fournie.

Dans son raisonnement, ERASME traite chaque couche en tant que telle. Elle peut certes être décrite par des caractéristiques spatiales (épaisseur par exemple) mais sa structure spatiale n'est pas prise en compte. Une couche n'est jamais *décomposée* par le système. De plus les interactions entre couches ne sont pas prises en compte par un module de raisonnement (elles le sont en revanche par les modèles numériques qui sont parmi les attachements procéduraux d'ERASME). Si ERASME réalise, en tant que système à base de connaissances un raisonnement, il ne raisonne pas sur l'espace (ce serait d'ailleurs inutile ; il n'en a nul besoin pour remplir sa mission).

ERASME a été retenu ici car il constitue un bon exemple de système à base de connaissances manipulant des connaissances spatiales sans pour autant, faute d'une décomposition de l'espace, réaliser un raisonnement spatial. De nombreux autres exemples auraient pu être retenus.

ELFIN

Ce système a été développé par Martin-Clouaire à l'Université Paul Sabatier de Toulouse [MCLO84] [MCLO85]. Il a pour objectif d'aider un géologue à déterminer l'ensemble des chemins qu'a pu suivre un hydrocarbure entre la roche mère (origine) et un piège où il sera éventuellement prélevé.

Les entrées du système sont des résultats de forage ou des mesures à partir desquelles le système extrapole la nature des couches profondes du site étudié. Le système contient une description spatiale du sous-sol qu'il *modifie* et qu'il utilise pour conduire des inférences. Il s'agit donc bien de raisonnement spatial.

Les connaissances spatiales utilisées par ELFIN sont relatives aux juxtapositions de contextes. Pour Martin-Clouaire, les contextes sont des unités homogènes du terrain. Il s'agit très grossièrement de couches composées de formations géologiques. Un contexte contient un certain nombre de faits liés à cette couche y compris des composants de cette couche.

Le système manipule une arborescence qui représente ces différentes formations dans leur organisation. L'arborescence indique les relations de superposition et les juxtapositions à un même niveau.

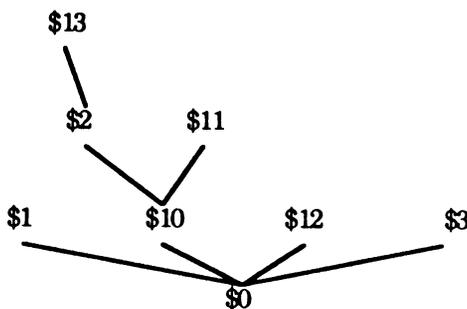


Figure 1.7 : ELFIN travaille sur l'arborescence des contextes. Un contexte correspond approximativement à une couche géologique. Les liens entre contextes correspondent à leur organisation dans l'espace. Le fait (juste au-dessus \$0 \$1) signifie que le contexte \$1 est au-dessus de \$0.

ELFIN réalise des inférences sur l'arborescence des contextes. Ce modèle devant donner finalement le résultat attendu, c'est-à-dire le chemin suivi par les hydrocarbures, le raisonnement spatial dans ELFIN consiste à faire évoluer cette arborescence. Pour cela, le système est capable de décomposer un contexte en deux contextes père et fils ou frères. En effet, un contexte possède une décomposition de la couche géologique qui lui correspond. Si le système estime que cette décomposition doit apparaître dans l'arborescence des contextes, il opère une des deux opérations suivantes:

Exemple 1.12: Création d'un contexte fils ou de deux contextes frères.

Dès que le système repère, dans un contexte, deux objets liés par l'attribut "juste au-dessus", il décompose ce contexte en deux contextes père et fils.

Il ajoute, évidemment, au contexte fils, le fait qui relie ces deux contextes c'est-à-dire "juste au-dessus".

De la même façon, dès que le système repère, dans le centre de gravité courant, deux morceaux d'arbres non consécutifs, il coupe le contexte courant en deux contextes frères.

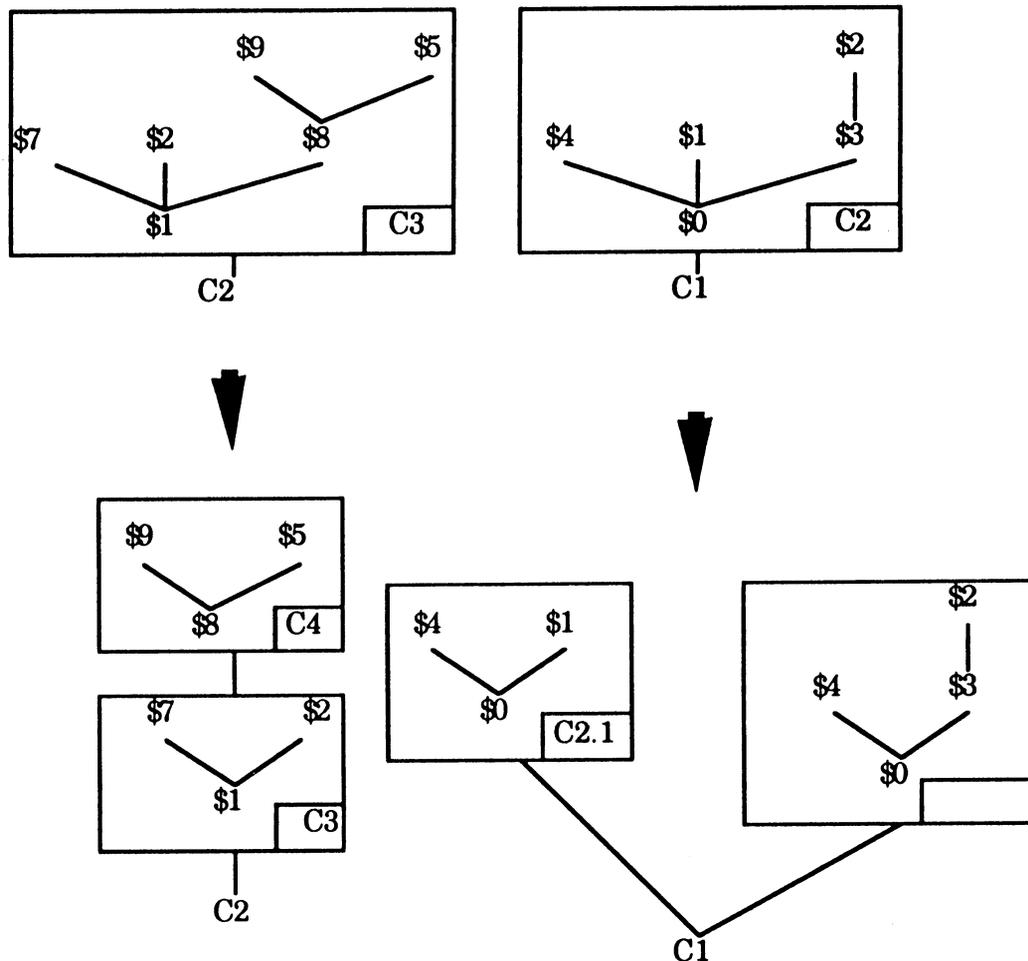


Figure 1.8 : ELFIN peut décomposer un contexte en deux contextes père et fils ou en deux contextes frères.

Le système de Martin-Clouaire est particulièrement intéressant car il ne se contente pas de compléter une représentation de l'espace (comme le fait NEUREX qui va être présenté) mais il modifie la topologie même de l'espace où il travaille.

NEUREX

NEUREX est un système à base de connaissances qui a pour fonction de diagnostiquer des défauts du système nerveux chez l'être humain [XIAN&85]. Xiang et Srihari s'éloignent des systèmes experts en médecine les plus connus tels que MYCIN, car ils utilisent une description de l'anatomie du système nerveux alors que MYCIN travaille en faisant correspondre les symptômes aux maladies.

Pour cela, ils bâtissent une classification sur la distinction analogique-propositionnelle. Pour les auteurs, la représentation analogique correspond à une description géométrique. Elle peut être donnée par des équations définissant des courbes ou des surfaces ou par la division de l'espace en éléments de volumes qui permettent ensuite de définir des courbes, des surfaces ou des objets. Les critiques que font Xiang et Srihari à la représentation analogique concernent l'excès d'information sur des points peu intéressants. En ce sens, elle s'éloigne de l'approche des experts qui ne manipulent que des connaissances pertinentes.

Dans NEUREX, la représentation analogique concerne les sections de la moelle épinière. Chaque zone de la moelle est décrite par un polygone qui approche sa forme.

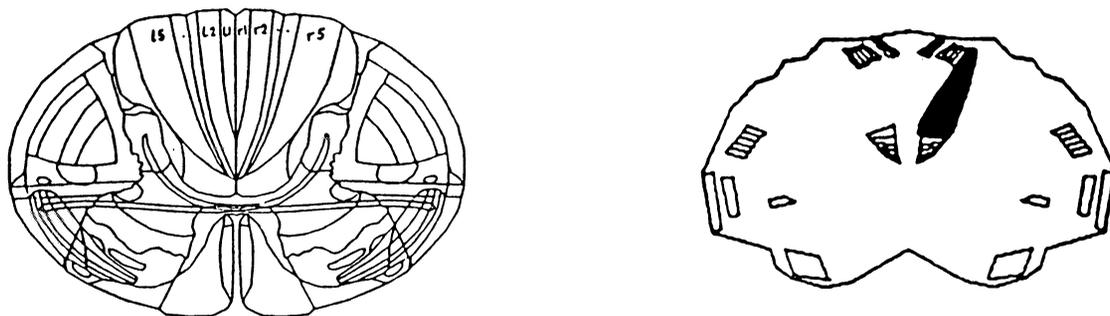


Figure 1.9 : Une section de la moelle épinière et sa représentation dans NEUREX. Pour Xiang et Srihari, il s'agit d'un exemple de connaissance analogique.

Dans cette représentation analogique, les raisonnements sont très spatiaux. NEUREX étudie l'enveloppe convexe des lésions pour rechercher des incohérences éventuelles dans les observations. Ce type de construction exige évidemment une description des polygones par la donnée des coordonnées de leurs sommets.

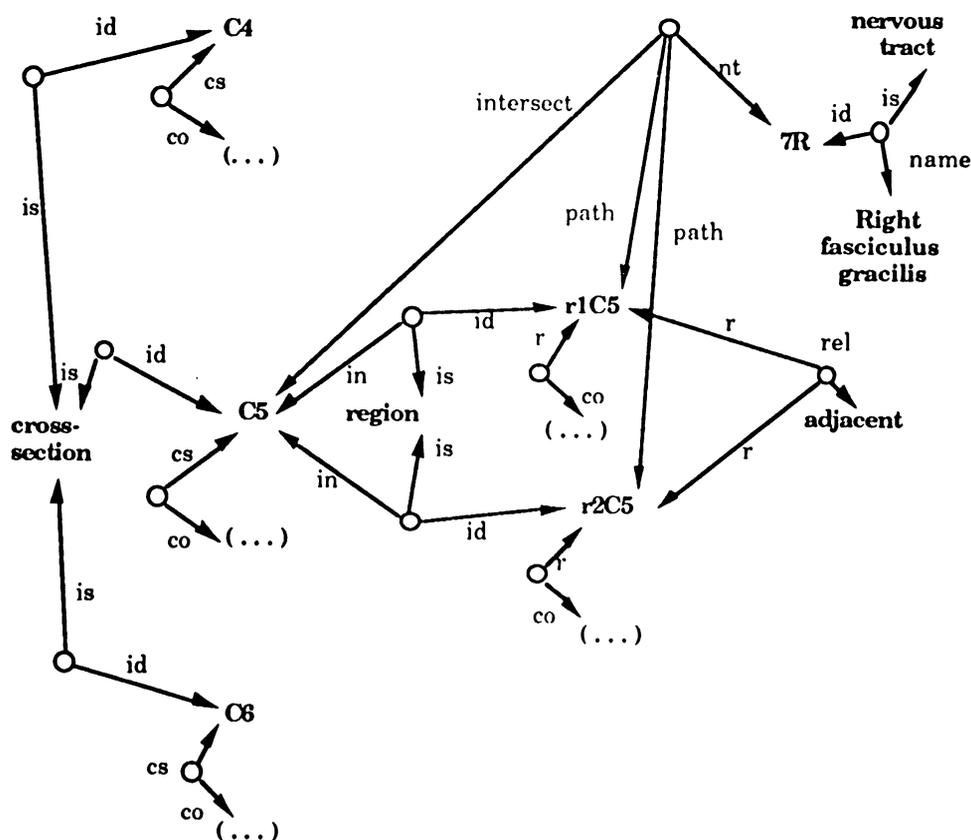


Figure 1.10 : Une partie du réseau sémantique qui représente le réseau des fibres nerveuses de la figure 1.9.

La représentation propositionnelle est une façon d'éviter certains écueils des représentations analogiques. Selon Xiang et Srihari, il s'agit d'une description topologique des entités et des relations entre ces entités. Elle concerne la forme, la position, l'adjacence et les directions. Pour les deux auteurs, elle doit être plus abstraite que la représentation analogique.

Selon les pères de NEUREX, la représentation propositionnelle est malheureusement trop souvent incomplète. La représentation propositionnelle est constituée par la donnée du réseau des fibres nerveuses (Figure 1.10).

Dans NEUREX, les deux types de connaissances communiquent. Des règles ou des calculs permettent de passer de l'une à l'autre.

NEUREX manipule donc une décomposition de l'espace en différentes unités décrites dans des modèles différents selon le type de connaissances spatiales concernées. De plus en tant que système à base de connaissances, NEUREX met bien en œuvre des formes de raisonnement.

1.3.3. Conception

Le dernier type de raisonnement spatial recensé par Charniak et McDermott concerne le domaine de la conception. Trois systèmes à base de connaissances, présentés ici, ont pour objectif de concevoir des systèmes. Il s'agit des systèmes SERF, EXPORT et ANAXAGORE respectivement dédiés à la conception d'outils de tenue de pièce pour machine-outil, de digues portuaires et de satellites spatiaux.

SERF

Le système SERF a été développé par Ingrand [INGR85]. Il s'agit d'un système expert de conception automatique d'outillage de tenue de pièce sur machine-outil. A partir d'une description fonctionnelle de cet outillage, description elle-même issue de la description de la pièce à tenir, le système infère une description géométrique de l'outil de tenue.

SERF est un système à base de règles et entre dans la catégorie des systèmes à base de connaissances et réalise un raisonnement tel que celui-ci a été défini dans la première section.

SERF accepte une description géométrique de la pièce à tenir (en termes de surfaces) et infère une description fonctionnelle de l'outil de tenue. Il exploite donc bien des connaissances spatiales. Il utilise ensuite cette description fonctionnelle pour construire l'outil en assemblant entre eux des éléments standards choisis parmi un catalogue. Ces éléments sont positionnés dans l'espace. L'espace est donc appréhendé par SERF à travers la structure de ces éléments. Il est décomposé en sous-parties.

EXPORT

EXPORT est un système d'aide à la conception d'ouvrages portuaires [MONT90]. De même que ERASME, ce système manipule des connaissances spatiales (qui caractérisent le site à équiper, les ouvrages à construire) mais il ne décompose pas les objets étudiés en différents composants ni ne raisonne sur les articulations dans l'espace entre les différents objets. Pour la même raison, c'est-à-dire faute d'une décomposition de l'espace en sous-parties, il ne réalise pas de raisonnement spatial.

ANAXAGORE

Le système ANAXAGORE a été développé par Trousse dans le cadre de sa thèse sur la coopération entre systèmes à base de connaissances et outils de CAO [TROU89]. Le domaine d'application choisi par Trousse est la conception de satellites spatiaux.

Trousse a étudié entre autres les relations spatiales entre les objets. Les solutions présentées relèvent d'un véritable raisonnement spatial. En effet le satellite est vu comme un ensemble de composants à assembler à l'aide de liaisons qui doivent respecter un certain nombre de propriétés spatiales.

Exemple 1.13 : Une aile de générateur solaire est composée de quatre panneaux. Les panneaux sont reliés entre eux par des dispositifs à distance constante. Il s'agit de savoir comment placer cette aile suffisamment loin de la structure pour ne pas avoir de problèmes d'ombres portées.

Trousse distingue différents types de relations entre les objets. Plus exactement elle définit un continuum entre les relations purement numériques et les relations purement combinatoires. Un inventaire des relations spatiales est notamment dressé. Trousse explique qu'à partir de ces différents types de relations des objets particuliers seront créés (rotule, pivot, etc.). Ces objets sont relatifs à des connaissances structurelles mais plus à des connaissances spatiales.

Les trois systèmes qui viennent d'être exposés sont particulièrement intéressants car ils constituent de rares exemples de ce que l'Intelligence Artificielle, et plus exactement les systèmes à base de connaissances peuvent apporter à l'ingénieur. Les travaux d'Ingrand et de Trousse réalisent bien un raisonnement sur l'espace : ils manipulent des représentations spatiales d'objets physiques, les assemblent les uns aux autres en respectant des contraintes géométriques ou topologiques et offrent ainsi une décomposition de l'espace.

CONCLUSION

Ce premier chapitre avait pour objectif de mettre en évidence, par une courte présentation des systèmes à base de connaissances et par des exemples, le domaine d'étude de cette thèse. Un certain nombre de disciplines et d'applications de l'informatique ont été évoquées. La plupart de ces approches ont été exclues parce qu'elles ne manipulent pas de connaissances spatiales ou parce qu'elles ne les exploitent pas au sein d'un raisonnement. Les autres ont été présentées de façon succincte. Elles seront reprises au fil des chapitres qui suivent et les critères qui ont été choisis pour cette sélection seront formalisés.

De ce court panorama ressortent également quelques impressions. Malgré la diversité des domaines d'application, des idées importantes reviennent à plusieurs reprises : décomposition de l'espace en sous-parties, importance de la topologie, faible part de la géométrie dans de nombreuses formes de raisonnement... Ces impressions ne sont pas suffisantes. Le vocabulaire rencontré au fil des exemples est souvent flou, contradictoire d'un auteur à l'autre (la signification de *topologique* ou *métrique* est particulièrement fluctuante). De plus des définitions aussi élémentaires que celle d'espace ou de connaissances spatiales sont absentes de la littérature. Enfin, la nature du

raisonnement spatial dans les systèmes à base de connaissances reste peu détaillée même si elle a été entrevue dans quelques rares exemples.

Les deux chapitres qui viennent tentent précisément de formaliser toutes ces notions. Le chapitre 2 traite du raisonnement spatial en général. Il formalise ou définit différentes notions : connaissances spatiales, hiérarchie de pertinence, formes de raisonnement spatial. Le chapitre 3 considèrera le raisonnement spatial dans les systèmes à base de connaissances.

2. RAISONNEMENT SPATIAL

Le chapitre précédent a fait clairement apparaître qu'il y avait, dans la littérature, un manque de formalisation à propos des notions de connaissances spatiales et de raisonnement spatial. De plus, les exemples de systèmes présentés mettent en œuvre des mécanismes de raisonnement, explicites pour les systèmes à base de connaissances, implicites pour les autres qu'il est intéressant de regrouper.

Ce chapitre tente donc de donner un cadre plus rigoureux au raisonnement spatial tout en montrant que les exemples du chapitre précédent s'y intègrent bien. Ils sont donc repris à l'occasion pour étayer chacune des idées et propositions de ce chapitre.

Les systèmes à base de connaissances permettent de discerner facilement les connaissances présentes dans la base. Cette caractéristique incite à définir de façon précise la notion de connaissances spatiales. Des structures mathématiques sont utilisées à cette fin et les différents types de connaissances spatiales comparés. Une définition du raisonnement spatial est alors proposée. Trois aspects du raisonnement spatial sont ensuite particulièrement étudiés : la construction de topologies, les inférences entre différents types de connaissances spatiales et la localisation d'un objet spatial dans une partition de l'espace en sous-ensembles connexes. Ces formes de raisonnement ne sont pas exclusives : la construction de topologie consiste ainsi souvent à passer d'un type de connaissances à un autre.

2.1. CONNAISSANCES SPATIALES

Les connaissances spatiales, éléments du discours d'un individu donné, doivent leur signification à l'espace dans lequel elles sont plongées c'est-à-dire, d'un point de vue mathématique, un ensemble muni de structure particulière. Par exemple, dire que deux objets sont orthogonaux suppose qu'ils sont inclus dans un espace euclidien. De même dire qu'un objet est séparé dans l'espace en deux parties exige que cet objet soit plongé dans un espace topologique.

Les classifications des connaissances spatiales sont nombreuses. Le besoin de distinguer différentes structures de connaissances spatiales est ressenti par de nombreux psychologues [PIAG&47], [LYNC60] et constructeurs de systèmes (à base de connaissances ou non) qui traitent de l'espace. Il en est ainsi de Kuipers [KUIP78], Forbus [FORB83], McDermott [MDER&84] ou Xiang [XIAN&85] et des concepteurs de systèmes d'information géographique ou de bases de données géographiques [DAVI88] [ESRI90] [EGEN90]. Dans leurs systèmes, la distinction entre différents types de connaissances spatiales permet la construction de modèles différents de représentation de connaissances spatiales. Cette classification est exploitée pour atteindre deux objectifs qui peuvent, ou doivent, se confondre : la finesse de la modélisation et l'efficacité des systèmes.

Par ailleurs, repérer ces structures est nécessaire pour donner une *définition* précise des connaissances spatiales, définition absente de la littérature.

2.1.1. Une définition et une classification

Toute connaissance spatiale suppose un ou plusieurs types de structure mathématique pour l'espace.

Exemple 2.1 :

“ Le bâtiment B de l'IMAG touche le bâtiment C”,

“ Grenoble est à 100 km de Lyon par le train”,

“ La Tour Eiffel est dans l'axe du palais de Chaillot et de l'Ecole Militaire”,

“ La particule est au point A de coordonnées (25,4; 96,2; 75,6)”,

“ Le cours Jean-Jaurès coupe l'avenue Alsace-Lorraine à angle droit”

sont, en tant que telles, des connaissances spatiales ; elles font cependant référence à des structures d'espace différentes.

La première ne renseigne que sur la juxtaposition de deux bâtiments sans préjuger de leur position (a priori on ne sait pas où est situé l'IMAG), leur taille, leur forme ou leur orientation. La seconde donne une indication de distance suivant un moyen de locomotion: à vol d'oiseau ou en voiture la distance serait différente ; on ne sait rien ni de Lyon ni de Grenoble. La troisième ne fournit d'indication qu'en ce qui concerne l'alignement de monuments parisiens. L'avant-dernière suppose connu un repère cartésien, dans lequel ses coordonnées suffisent à définir le point A. Enfin la dernière exige qu'il soit possible d'estimer un angle. Les structures sous-tendues par ces quatre connaissances seront développées successivement: il s'agit respectivement des structures topologiques, métriques, vectorielles et euclidiennes.

Un ensemble quelconque ne peut être considéré comme un espace. Il doit être pour cela doté d'une structure. Les mathématiciens ne considèrent pas qu'un ensemble pourvu d'une simple *partition* mérite le qualificatif d'espace ; une *algèbre* telle qu'un *groupe*, un *anneau*, non plus. Ils ont par ailleurs défini un grand nombre de structures d'ensemble qu'ils ont baptisées du nom d'espace. Néanmoins seules les quatre structures qui sont présentées par la suite permettent de formaliser des notions qui correspondent aux représentations de l'espace que nous manipulons “tous les jours”.

Les notions d'espace qui correspondent à notre espace quotidien, proviennent soit de l'analyse, soit de l'algèbre. Il est possible de distinguer les espaces topologiques, les espaces métriques, les espaces vectoriels et les espaces euclidiens. Les espaces mathématiques qui n'ont qu'un lointain rapport avec l'espace dans lequel nous vivons... ou dans lequel nous imaginons vivre ne sont donc pas repris. Les définitions précises se trouvent en annexe.

Ensembles et éléments

Les structures qui vont être mentionnées sont toutes basées sur la théorie des ensembles. Les connaissances spatiales concernent toutes en effet des propriétés d'ensembles ou d'éléments. Pour qu'elles aient un sens dans un certain discours, il est nécessaire que ces ensembles ou ces éléments en tant que tels soient définis sans ambiguïtés. Dans de nombreux cas, ces définitions sont une donnée de départ du raisonnement ; il arrive que la définition soit une des premières étapes ; enfin il se peut que cette définition constitue le but du raisonnement.

Espaces topologiques

Les espaces topologiques sont les espaces les plus généraux de l'analyse. Ils sont à la base des définitions les plus importantes de celle-ci. Un espace

topologique est un ensemble doté d'une partie (la topologie) de sa partition vérifiant un certain nombre d'axiomes (cf. annexe). Les éléments de cette topologie sont les *ouverts*. Les complémentaires des ouverts sont les fermés ; ils caractérisent eux aussi la topologie. Les ouverts (respectivement les fermés) constituent une décomposition de l'espace. Sur cette topologie, on va définir des propriétés.

La notion la plus importante des espaces topologiques est la notion de *voisinage* d'un point. A partir du voisinage, les notions de *frontière*, de *limite* et de *continuité* sont mises en évidence. Ce sont les formalismes de continuité et de connexité qui sont les plus utilisés dans les connaissances spatiales. Grâce aux concepts de frontière et de connexité, il est possible également de définir des ensembles connexes "troués" (dont la frontière n'est pas connexe). Un espace topologique ne connaît pas les notions de distance, de direction, d'angle et bien sûr d'orthogonalité. Il n'est caractérisé que par ses éléments et par la topologie qui doit être construite.

Espaces métriques

Un espace métrique est un ensemble muni d'une *distance*, c'est-à-dire d'une application de son produit croisé dans l'ensemble des réels positifs. Cette application vérifie des axiomes. La notion de distance permet la construction d'une topologie. Un espace métrique est donc toujours un espace topologique.

Les espaces métriques permettent de définir des distances entre objets. Le concept de base est la "boule" ensemble des points à une distance donnée d'un point. La distance peut être la distance euclidienne tri-dimensionnelle ou bi-dimensionnelle ou correspondre à une métrique différente (cf. 2.5. Localisation). Comme pour les espaces topologiques, il est nécessaire de construire l'ensemble support de cette structure métrique et de définir ces éléments. En revanche, la topologie associée à cet espace métrique est définie toute seule : il s'agit de l'ensemble des boules ouvertes.

Espaces vectoriels

Les espaces vectoriels permettent de manipuler les notions de *direction*, de *sens* et de *projection*. Les espaces vectoriels (en tout cas ceux de dimension finie) permettent de manipuler des *bases*, ce qui a deux conséquences : il est possible tout d'abord définir un ordre sur une direction en utilisant l'ordre des réels ; il est aussi possible de définir l'ensemble support de la structure de façon implicite. Tout point de l'espace vectoriel est défini par la donnée de ses coordonnées dans un repère cartésien. Il y a donc identité entre les espaces vectoriels et les produits \mathbb{R}^2 , \mathbb{R}^3 ou \mathbb{R}^n . C'est une différence avec les espaces topologiques et métriques qui doivent être définis de façon explicite.

Les espaces vectoriels normés sont des espaces vectoriels munis d'une norme et, par conséquent, métriques. Ils ne sont pas mentionnés plus avant ici car ils n'introduisent aucun concept nouveau dans le contexte présent. Aucune distinction n'est entretenue ici entre espace vectoriel et espace affine, c'est-à-dire entre espace de vecteurs et espace de points. Les structures sont en effet équivalentes.

Espaces euclidiens

Par rapport à la structure vectorielle, la structure euclidienne introduit le produit scalaire. A partir du produit scalaire, les notions de *norme*, d'*angle* et d'*orthogonalité* sont introduites. L'espace euclidien correspond au monde de

tous les jours. Comme l'espace vectoriel dont il est un cas particulier, il peut être équipé d'un repère ce qui rend inutile, là encore, une définition des objets.

Comparaisons

Les espaces qui viennent d'être mentionnés ne sont pas sans rapports. Les espaces métriques sont des espaces topologiques ainsi que les espaces euclidiens qui sont aussi, simultanément, des espaces vectoriels. Les espaces euclidiens regroupent donc l'ensemble des propriétés spatiales. Ces différentes structures d'espace peuvent donc être organisées suivant un ordre partiel qui est celui de la généralité.

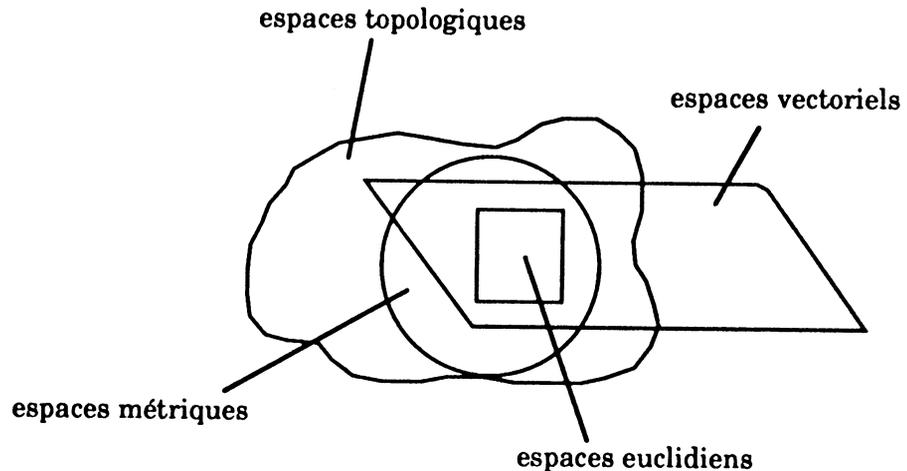


Figure 2.1 : Les ensembles des espaces définis plus haut. Les inclusions des espaces métriques dans les espaces topologiques signifient que tout espace métrique peut être automatiquement muni d'une topologie et peut donc être considéré comme un espace topologique. La forme des ensembles symbolise les structures ; la distance permet de définir les boules (d'où le cercle); les espaces vectoriels les directions (le parallélogramme), les espaces euclidiens l'orthogonalité (le rectangle). Les espaces topologiques sont représentés par une "patate" simplement connexe.

Des connaissances structurées

A partir des définitions de ces espaces, il est possible de proposer une classification des connaissances. Une connaissance sera qualifiée de topologique (respectivement métrique, vectorielle ou euclidienne) si l'espace minimum (c'est-à-dire le plus général) nécessaire pour lui donner un sens est un espace topologique (respectivement métrique, vectoriel ou euclidien).

Exemple 2.2 : Les connaissances de l'exemple 2.1 sont maintenant reprises :

- "Le bâtiment B de l'IMAG touche le bâtiment C".

Pour avoir une signification, cette connaissance exige au minimum que les bâtiments B et C soient définis dans un espace topologique. Il s'agit d'une connaissance topologique.

- "Grenoble est à 100 km de Lyon par le train". Cette connaissance n'a de sens que si la notion de distance est définie. Il s'agit d'une connaissance métrique.

- "La Tour Eiffel est dans l'axe du palais de Chaillot et de l'Ecole Militaire". Cette connaissance nécessite une structure vectorielle d'espace.

- "La particule est au point A de coordonnées (25,4; 96,2; 75,6)". Ici il est nécessaire d'avoir défini un repère, c'est-à-dire une base. C'est encore un espace vectoriel qui

est requis. La plupart du temps, cette base est orthonormée. Il faut donc doter cet espace vectoriel d'un produit scalaire et en faire un espace euclidien.

• "Le cours Jean-Jaurès coupe l'avenue Alsace-Lorraine à angle droit". Cette dernière connaissance exige la notion de mesure d'angle. L'espace dans lequel on place le cours Jean-Jaurès et l'avenue Alsace-Lorraine est un espace euclidien.

Quelle classification ?

Les auteurs des travaux présentés dans le premier chapitre exploitent comme cela a déjà été dit des classifications de connaissances spatiales. Il en va de même pour les chercheurs en psychologie cognitive. Ces différentes classifications ne sont pas toujours cohérentes les unes avec les autres. Il apparaît cependant que celle qui vient d'être proposée les prend toutes en compte.

Tout être humain est capable de raisonner dans l'espace. L'un des objectifs de base de l'informatique étant de reproduire le comportement humain dans le domaine du traitement de l'information, il est naturel d'étudier les caractéristiques du raisonnement spatial chez l'être humain.

Un certain nombre d'auteurs l'ont fait. Ils sont le plus souvent psychologues ou, pour les plus modernes, psychologues "cognitivistes". Il ne s'agit pas de reprendre leurs études qui dépassent très largement le cadre de cette thèse. Toutefois les résultats qu'ils ont obtenus et les observations qu'ils ont pu faire correspondent aux remarques qui précèdent et s'intègrent bien dans la classification proposée.

Piaget et Inhelder [PIAG&47] ont réalisé, dans les années quarante, un grand nombre d'expériences sur les rapports de l'enfant à l'espace. Leurs principales observations concernent la classification des connaissances spatiales telle qu'elle apparaît dans le développement de l'enfant.

Ils sont les premiers à avoir utilisé une classification des représentations et des approches de l'espace. Ils distinguent les espaces topologiques, projectifs et euclidiens. Contrairement à la plupart des autres auteurs, ils ont visiblement à l'esprit, de façon très claire et très précise, les notions mathématiques associées à ces espaces. Pour des psychologues, ils semblent bien plus mathématiciens et rigoureux que de nombreux informaticiens anglo-saxons... Ils distinguent les espaces topologiques, projectifs et euclidiens. L'espace projectif est apparemment l'ancien nom de l'espace vectoriel. L'espace métrique n'est pas abordé dans [PIAG&47]. Il faut préciser, enfin, que la taille et surtout la richesse des ouvrages de Piaget en interdit l'exploitation complète dans le cadre de cette étude. Il y a fort à parier que la lecture attentive de ces ouvrages donnera de nombreuses idées aux informaticiens qui s'intéressent à l'espace.

Il a déjà été précisé que, dans ARC-INFO, la base de données spatiales était séparée en une base de données numérisées et une base de données topologiques (cf. figure 1.4 dans le premier chapitre). La première contient la définition des lignes, polygones et points qui ont été numérisés. Il s'agit des coordonnées des points et des sommets dans le repère d'un espace euclidien ou vectoriel. Après construction d'une topologie (par exemple avec les commandes BUILD et CLEAN de l'exemple 1.3) la base de données topologiques contient des informations topologiques (arcs frontières pour un polygone) mais aussi métriques (longueur d'un arc), vectorielles (polygone à droite et à gauche pour

un arc) ou euclidiennes (surface d'un polygone). Il est facile de constater que le qualificatif de topologique est abusif.

David exploite les notions topologiques dans son modèle SPATIAREL [DAVI88]. Il définit ainsi un domaine au sens du modèle relationnel qui est en fait la topologie d'un espace métrique c'est-à-dire l'ensemble des ouverts de cet espace métrique (i.e. les boules ouvertes). Il définit ensuite sur ce domaine les opérations classiques qui sont des opérations topologiques (union d'ouverts avec incorporation de l'éventuelle frontière commune), des prédicats qui sont aussi de nature topologique. Des mesures sont aussi définies sur la superficie d'un polygone et la distance entre deux polygones. C'est d'ailleurs ce qu'exige la nature métrique de l'espace que manipule David (elle ne devrait d'ailleurs pas suffire, en toute rigueur, pour définir la mesure de superficie).

Dans son modèle TOUR, Kuipers repère un certain nombre de types de connaissances. Ses cartes topologiques contiennent effectivement des informations topologiques. Il s'agit du graphe des vues et de la carte des régions (cf. 1.3.1.1). En revanche, sa carte métrique doit être plongée dans un espace euclidien pour qu'elle prenne tous ses sens. En effet, l'angle des rotations, les coordonnées des positions dans un repère orthonormé n'ont de sens que dans un espace euclidien.

Pour SPAM, McDermott et Davis [MDER84] qualifient de topologiques les composants propositionnels bien qu'ils soient simplement de nature ensembliste et qu'ils n'expriment que des notions de compositions (par exemple (partie bras15 corps23)). Quand aux faits métriques de la carte floue, les exemples qui sont donnés correspondent soit à des vrais faits métriques (par exemple (> (distance illinois new-york) (3000 km))) soit à des connaissances de nature vectorielle ((entre illinois new-york californie)).

Que contiennent réellement les deux structures utilisées par Forbus [FORB83] pour la construction de FROB, le diagramme métrique et le graphe spatial ? Pour le premier, ses éléments (les points, les segments, les paraboles) sont définis dans un espace euclidien muni d'un repère cartésien. Cependant, les questions auxquelles doit répondre ce modèle ont des réponses dans un espace qui peut être alors muni d'une seule topologie. Il s'agit, en effet, pour Forbus, d'*identifier* l'élément (point, segment, etc.), de repérer des *intersections* éventuelles ou de *distinguer* des côtés, des intérieurs ou des extérieurs. La réponse à ces trois types de questions réside toutefois dans la définition, euclidienne elle, des éléments du diagramme métrique. Le graphe spatial présente des relations entre les éléments du diagramme. Les relations sont du type "up" ou "left", c'est-à-dire que la structure sous-jacente est vectorielle. Le système FROB définit donc les objets manipulés dans un espace euclidien mais range ses connaissances dans des structures topologiques ou vectorielles.

Xiang et Srihari [XIAN&85] distinguent de leur côté des connaissances analogiques et des connaissances propositionnelles. Les premières reviennent à décrire les objets de façon géométrique en donnant leur forme et leur position (cf. figure 1.9 dans le premier chapitre). Ces connaissances n'ont donc de sens que dans un espace euclidien. Les connaissances propositionnelles renseignent "topologiquement" sur les objets et les relations entre ces objets (cf.

figure 1.10). En fait, les propositions portent sur l'adjacence, la connectivité (notion effectivement topologique) mais aussi sur la direction (cette fois-ci, il s'agit de connaissances vectorielles).

Trousse définit dans [TROU89] un "spectre continu" entre des relations spatiales purement numériques d'une part et des relations purement combinatoires d'autre part. Les premières expriment des positions relatives d'entités les unes par rapport aux autres. Les secondes "spécifient l'adjacence entre les entités". Entre ces deux types, il en existe un grand nombre comme "au-dessus", "en-dessous".

Ce critère de distinction numérique/non numérique n'est plus exploité ensuite. Trousse dresse alors un inventaire des relations entre entités à partir des classes suivantes :

- distance : il s'agit des relations du type être-distant-de, être-près-de. Les connaissances associées à ces relations sont typiquement métriques.
- adjacence : les relations comme être-en-contact, être-contre correspondent à des connaissances topologiques.
- position, orientation : ce peut être être-à-gauche, être-en-dessous ou être-devant. La structure associée est vectorielle.
- "spatial": Trousse donne comme exemple être-entre O_1 O_2 O_3 . Il est possible de formaliser ces connaissances dans un espace vectoriel.
- "vue": L'exemple donné est n'avoir-rien-dans-son-champ-de-vue. Il s'agit là encore d'une connaissance vectorielle.

Laurini insiste aussi sur l'importance du raisonnement topologique, utilisé lorsque l'espace peut être représenté par un graphe, et du raisonnement géométrique où "toute la sémantique du monde 2D ou 3D est nécessaire" [LAUR90].

La définition et la classification proposées ici intègrent donc bien les différences introduites par les différents auteurs. Elles font une juste place à la topologie et offrent un cadre unique aux différentes classifications.

2.1.2. Connaissances "pictorielles" ?

De nombreux auteurs entretiennent une distinction entre des connaissances spatiales qui seraient symboliques et d'autres qui seraient "pictorielles". Cette distinction semble trouver une partie de son origine dans la distinction entre "vecteurs" et "rasters" en informatique graphique.

Les psychologues cognitivistes entretiennent également cette distinction. Kosslyn [KOSS83] est le plus souvent cité pour opposer "depiction" et "description". La depiction est la donnée d'images ("pictures"), c'est-à-dire l'agencement de points. Une image correspond à des objets. La depiction de Kosslyn constitue la structure de définition de ces objets ; elle contient toutes les informations à leur sujet mais ne peut les exprimer. La description, en revanche, fournit des informations sur ces objets en termes de relations, de classes. Elle correspond aussi à la structure de définition. Elle peut être bâtie sur les données de la depiction.

Dans ce document il semble inutile d'entretenir cette distinction. En effet deux cas de figure se présentent :

- La description en cause permet de relier les pixels d'une image à un objet. Dans ce cas, la description contient une définition de cet objet (ou plus exactement de la projection de cet objet sur le plan de l'image). Il s'agit d'une description spatiale. La couleur ou le niveau de gris des pixels (il s'agit d'une image) transporte des informations sur cet objet.

Il faut préciser que cette description en terme de pixels ne présente aucune différence de nature avec d'autres descriptions.

Exemple 2.3 : Les triangles utilisés dans le système ELSA (cf. chapitre 4) permettent de définir des objets particuliers appelés petits panneaux. Un petit panneau est défini comme la réunion d'un certain nombre de triangles. Ces triangles sont irréguliers.

Il est aussi possible de définir ces petits panneaux comme une réunion de pixels. Dans les deux cas, à partir du moment où les triangles d'une part et les pixels d'autre part sont bien définis les descriptions sont de même nature. Le choix de l'une ou de l'autre solution trouve ses origines ailleurs que dans la nature des connaissances spatiales.

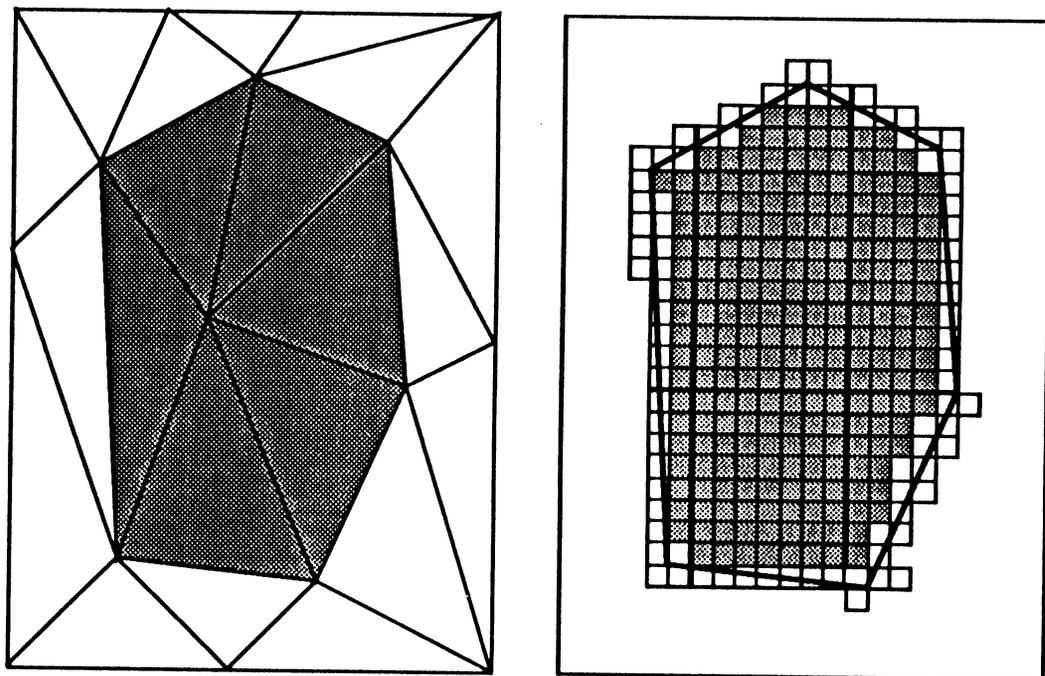


Figure 2.2 : Deux descriptions possibles d'un polygone. Sur la figure de gauche, le polygone est défini comme la réunion de triangles irréguliers, sur celle de droite, par la réunion de carrés qui peuvent être considérés comme des pixels. Ces deux descriptions sont de même nature.

- Dans le deuxième cas, il n'y a pas de lien entre le tableau de pixels qui constitue l'image et des objets définis par ailleurs dans le système. L'image ne contient pas de véritables connaissances spatiales.

Cette séparation a de plus un inconvénient : elle masque le véritable problème de la définition des objets spatiaux. En effet la plupart des capteurs susceptibles d'être utilisés (caméras, "scanners"...) fournissent des images à base de pixels. Dans le même temps, les connaissances les plus pertinentes portent plutôt sur des descriptions symboliques. Il s'agit donc de construire ces descriptions symboliques à partir des images. Pour cela, il est nécessaire de définir des objets et de les lier à des ensembles de pixels. L'opposition

symbolique-pictorielle perd de son importance. Seul demeure le passage d'un type de connaissance à un autre et celui de la construction et de la définition d'objets et de leur extension spatiale. Ces deux problèmes sont traités par la suite.

2.1.3. Une échelle de pertinence

Les différents types de connaissances spatiales ne sont pas tous équivalents. En effet, bien que le monde dans lequel nous vivons soit un espace euclidien (en première approximation), les connaissances euclidiennes, c'est-à-dire les connaissances qui pour prendre leur pleine signification ont besoin d'un espace euclidien, ne sont pas les plus utilisées ni celles qui sont manipulées avec la plus grande rigueur.

Les travaux de Piaget et Inhelder [PIAG47] et ceux de Lynch [LYNC60] ont mis en évidence le rôle important joué par les connaissances topologiques dans l'apprentissage des notions spatiales chez l'enfant ou dans l'apprentissage d'un nouvel environnement spatial chez l'adulte.

Piaget et Inhelder constatent que l'enfant assimile ces notions dans des ordres particuliers qu'ils ont tenté de définir.

La première partie de leur ouvrage est consacrée aux rapports topologiques élémentaires chez l'enfant que ce soit dans le domaine de la perception tactile ou dans celui du dessin. Les notions topologiques sont les premières à être assimilées par l'enfant.

Dans un deuxième temps, l'espace projectif est étudié avec, en particulier, la notion de droite et de perspective.

"Le passage de l'espace projectif à l'espace euclidien" est le titre de la dernière partie de l'ouvrage. Il traite des déformations, des systèmes de coordonnées et de l'apparition de la notion de plan.

Exemple 2.4 : Les enfants examinés par Piaget et Inhelder [PIAG&47] sont capables de reconnaître les surfaces à trous (propriété topologique) avant de faire la distinction entre le cercle et l'ellipse ou entre le rectangle et l'ellipse.

Piaget et Inhelder chez l'enfant, Lynch chez l'adulte [LYNC60] ont constaté que le nombre d'erreurs commises à propos de connaissances euclidiennes (valeur d'angle par exemple) est bien plus grand que celui commis à propos des connaissances métriques (estimation de distance) ou vectorielles (alignement). Les connaissances topologiques (continuité et connexité) sont très rarement sujettes à erreurs.

Exemple 2.5 : Kuipers cite à plusieurs reprises les travaux de Lynch [LYNC60] qui a observé, au cours des interrogatoires qu'il a eus avec les habitants de trois villes différentes, que les erreurs étaient le plus souvent métriques et rarement topologiques.

Exemple 2.6 : "La Tour Eiffel est dans l'axe du palais de Chaillot et de l'Ecole Militaire"

Cette connaissance vectorielle est, en effet, bien connue et très utilisée par de nombreux promeneurs parisiens. Pourtant, aucun de ceux-ci ne connaît les coordonnées de ces monuments dans quelque repère que ce soit et même à cent mètres près (connaissance euclidienne) pas plus qu'il ne sait la distance entre ces monuments (connaissance métrique).

Kuipers de son côté a montré que, dans des systèmes de représentation de l'environnement (cf. chapitre 1), les connaissances topologiques sont bien moins sujettes à erreurs que les connaissances métriques, vectorielles et euclidiennes. En particulier, il estime que l'utilisation d'approches purement métriques (c'est-à-dire, pour Kuipers, faisant appel à des mesures) n'est pas satisfaisante pour représenter un "large-scale space" impossible à saisir d'un seul point de vue. Il pense qu'il est nécessaire de disposer d'une structure de base constituée d'un modèle topologique sur lequel viennent reposer les autres cartes ("layers") définies au chapitre 1 [KUIP&88a].

Egenhoffer, Frank et Jackson [EGEN&90] justifient quant à eux leur recherche d'une théorie topologique pour les systèmes d'information géographique par les lacunes et les défauts des descriptions bâties sur des systèmes de coordonnées. Il s'agit par exemple de la différence de nature entre l'espace réellement continu et l'ensemble des flottants, que peut manipuler un ordinateur et qui n'est pas continu. Il peut s'agir aussi des problèmes de changement d'échelles qui ne respectent pas la topologie d'une polygonalisation (un point à l'intérieur d'un polygone en sort lors d'un changement d'échelle) ou du calcul d'intersection de droites.

Exemple 2.7 : Egenhoffer, Frank et Jackson [EGEN&90] rappellent que, malheureusement trop souvent, le point d'intersection de deux droites tel qu'il est calculé par tel ou tel autre système d'information géographique n'est pas sur une des deux droites. Ceci est le plus souvent dû aux erreurs d'arrondis introduites par le système informatique.

Ces auteurs exploitent des notions issues de la topologie combinatoire. Les fermés sur lesquels ils basent la topologie sont les *simplexes*. Un simplexe d'un espace de dimension d est défini par $d+1$ simplexes de dimension $n-1$. Les simplexes permettent une décomposition de l'espace assez simple sous la forme d'un ensemble de fermés. Par exemple, le plan peut être représenté par des 2-simplexes qui sont des triangles. Il est facile de définir des opérations sur les simplexes qui leur font correspondre leur frontière ou qui permettent d'insérer de nouveaux simplexes en créant la topologie correspondante.

Les points de vue qui viennent d'être exprimés se ramènent tous à la constatation suivante : la quantification est source d'erreurs et d'imprécisions. Les erreurs et les imprécisions sont en effet faciles à commettre dans un contexte métrique, vectoriel ou euclidien et beaucoup moins dans un contexte topologique car l'importance du numérique dans les premiers types de connaissances spatiales pose le problème de l'imprécision de la mesure et, peut être plus encore, de l'enregistrement de cette mesure.

Exemple 2.8 : A partir des coordonnées suivantes:

"A (0; 0; 0,5),

B (1; 2; 0,75),

C (2; 4; 1),"

on peut conclure facilement que:

"les trois points A, B, C sont alignés". Mais les connaissances comprises dans les deux expressions entre guillemets sont loin d'être équivalentes. Si la position des points désire être connue, la première est nécessaire, tandis que, plus couramment, la seconde est directement utilisable et facile à mémoriser.

D'autre part, comme l'espace physique est un espace euclidien, il peut être tentant de décrire l'ensemble des objets manipulés dans un espace euclidien. Mais dans de nombreux cas, cela reviendrait à vouloir enfermer la description spatiale dans un carcan bien étroit et exigerait des quantités d'informations exorbitantes!

Par exemple, le plan d'un circuit électrique est souvent beaucoup plus clair représenté de façon symbolique qu'observé dans sa réalité. La structure utilisée pour cette représentation est un simple espace topologique chargé de contenir les informations de connexité des conducteurs, c'est-à-dire les liaisons électriques entre composants.

De façon générale, l'espace euclidien ne peut rendre compte de certaines structures utilisées pourtant très souvent dans de nombreuses descriptions sauf, bien sûr, à les décrire points à points.

Exemple 2.9: Chacun sait qu'en matière de chemins de fer la distance séparant deux villes n'a souvent qu'un lointain rapport avec la distance euclidienne. On peut ainsi définir, en mathématique, une "distance-SNCF" qui caricature l'aspect très centralisé de notre réseau (cf. annexe). Là encore, la structure d'espace euclidien ne fournit pas une bonne description de la France ferroviaire et surtout des distances réellement parcourues par le voyageur (qui paie pourtant aux kilomètres parcourus).

Selon les applications et la nature des connaissances à représenter, il faut donc choisir un type d'espace plutôt qu'un autre.

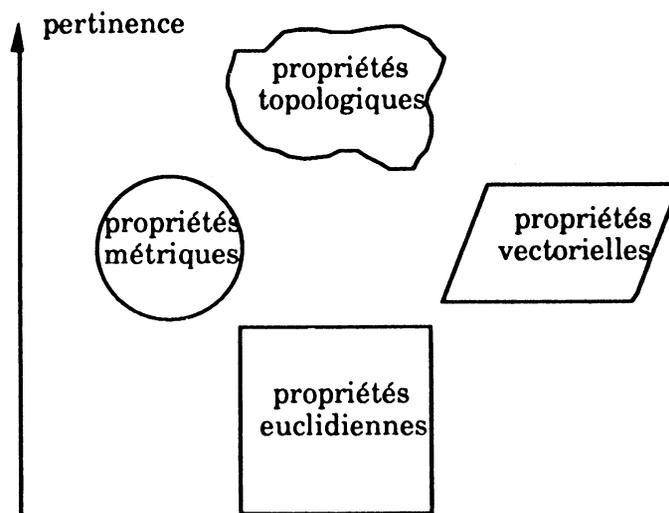


Figure 2.3 : Classement des différentes propriétés. Toutes les propriétés sont rassemblées sous la structure minimale qui permet leur définition. Ainsi les propriétés topologiques (continuité, connexité) existent, certes, dans les structures euclidiennes mais se contentent, pour exister, des structures topologiques beaucoup moins riches. Les propriétés topologiques sont ainsi beaucoup plus utilisées que les propriétés euclidiennes ou métriques. Néanmoins, c'est souvent à travers leurs caractéristiques euclidiennes que certains objets sont acquis par le système à base de connaissances.

Par ailleurs, l'étude d'un expert au travail met en évidence la primauté des propriétés topologiques sur les propriétés métriques et vectorielles, les propriétés euclidiennes n'étant que très rarement explicitement mentionnées. Le chapitre 4 qui traite du système ELSA montre en particulier que dans le but de communiquer avec l'expert spécialiste en avalanches, le système à base de

connaissances ELSA doit posséder des éléments de l'univers du discours de l'expert. En matière spatiale, ces éléments privilégient les structures topologiques.

Il devient donc légitime de représenter les structures spatiales et les connaissances qui leur sont liées suivant un axe de pertinence (cf. figure 2.3).

A priori, compte tenu des exemples issus des travaux antérieurs et du développement du système ELSA, cette pertinence peut être définie comme construite à partir de trois critères :

- précision et qualité des connaissances
- fréquence d'utilisation des connaissances
- importance des connaissances.

2.2. UNE DEFINITION

Les différents auteurs qui mentionnent le raisonnement spatial dans les systèmes informatiques omettent tous d'en donner une définition (cf notamment le paragraphe sur le raisonnement spatial dans [CHAR&85] où aucune définition n'est donnée). Elle apparaît pourtant nécessaire.

Un système réalise un raisonnement spatial lorsqu'il contient des connaissances spatiales relatives à différents objets constituant une décomposition de l'espace, aux relations qu'ils entretiennent entre eux et qu'il réalise des inférences à partir de ces connaissances ou en les enrichissant.

Exemple 2.10 : Le raisonnement suivant (et lui seul):

"La pente a une valeur de 30°, il y a donc risque d'avalanche"

manipule une connaissance spatiale (le fait que la pente soit de 30°). Ce raisonnement ne peut pas être, néanmoins, qualifié de spatial car il n'exploite aucune décomposition de l'espace, en l'occurrence du site avalancheux. Celui-ci est assimilé à une simple pente. Le raisonnement ne prend pas en compte la structure spatiale de ce site.

2.2.1. A propos de cette définition

Les *connaissances spatiales relatives à un objet* consistent à donner des propriétés de cet objet qui n'ont de sens que parce que cet objet est plongé dans une structure d'espace. Elle se situe donc dans la description de la structure de l'objet étudié, dans la description des composants de cet objet, voire dans la description des relations entre ces composants. Elle consiste à décrire les composants et les liaisons entre ces composants dans un ensemble doté d'une certaine structure et nommé, dès lors, espace.

Raisonnement sur un objet spatial n'est pas raisonnement sur l'espace. En effet, pour pouvoir affirmer qu'il y a raisonnement sur l'espace, cet espace et sa *structure* doivent être présents sous la forme de *plusieurs* objets de cet espace. Les descriptions spatiales de ces objets constituent un *modèle* spatial de l'espace. C'est sur ce modèle que le système réalise son raisonnement.

A la variété des descriptions spatiales des composants du modèle, répond la richesse dans le type de structure d'espace qui a été présenté auparavant. Le chapitre 3 développera ce que cette définition implique pour un système à base de connaissances qui raisonne dans l'espace.

Le raisonnement spatial proprement dit est alors la phase qui consiste à mener des inférences à partir de ces connaissances liées à la description spatiale mais aussi à inférer d'autres connaissances spatiales. Il faut remarquer que les connaissances utilisées ou inférées sont rarement exclusivement spatiales.

En effet, *il n'y a pas de raisonnement spatial sans contexte d'utilisation* [LAUM88]. La dimension de cet espace (mais dans certaines structures ce terme n'a pas de sens : espaces métriques ou topologiques par exemple) varie en fonction de l'utilisation mais la nature elle-même des faits inférés dépend de l'application.

La définition qui vient d'être donnée s'inspire de la notion de raisonnement spatial telle que le bon sens la considère.

Il convient de préciser que la définition exposée ci-dessus est surtout destinée au raisonnement spatial tel qu'il apparaît dans les systèmes à base de connaissances (cf. chapitre 3). Il est donc inutile de vouloir lui faire comprendre les problèmes de raisonnement géométrique utilisés en robotique ou ailleurs sauf à contrôler des robots par des systèmes à base de connaissances.

2.2.2. Applications aux systèmes à base de connaissances

Les chercheurs qui travaillent sur les systèmes à base de connaissances, classiquement appelés systèmes experts, distinguent deux grandes familles de systèmes selon la nature des connaissances qu'ils y introduisent. Cette distinction est reprise ici car, comme cela va être montré, elle coïncide, au niveau des systèmes à base de connaissances, à la distinction, introduite dans la définition précédente, entre le raisonnement dans l'espace et un raisonnement sur des propriétés spatiales.

Deux types de connaissances

Les systèmes à base de connaissances apparaissent aujourd'hui comme séparés en deux catégories distinctes.

- Des systèmes manipulent des connaissances dites superficielles qui ne relient pas les symptômes (ou les observations) et les diagnostics (ou les conclusions) à la structure de l'objet de l'étude (un patient, une machine ou tout autre dispositif), qui apparaît alors comme une entité dans laquelle aucun composant n'est repéré. La plupart des premiers systèmes experts de diagnostic relevaient de cette catégorie.

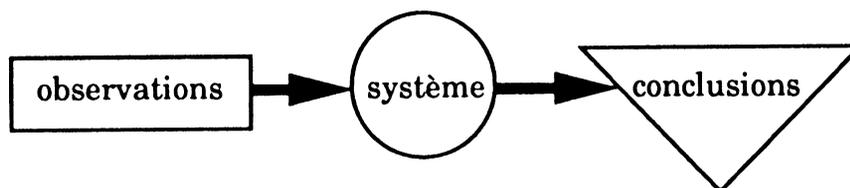


Figure 2.4 : Dans le cas de la connaissance superficielle, le système considère le sujet étudié comme une boîte noire. Il est simplement capable de relier des observations ou symptômes à des conclusions, c'est-à-dire à la nature de la maladie ou du défaut.

- D'autres systèmes utilisent les liens qui existent entre observations et conclusions d'une part et structure de l'objet de l'étude et fonctionnement de cette structure d'autre part. Ces systèmes disposent d'une représentation du patient ou "modèle" [DAVI84] [XIAN&85].

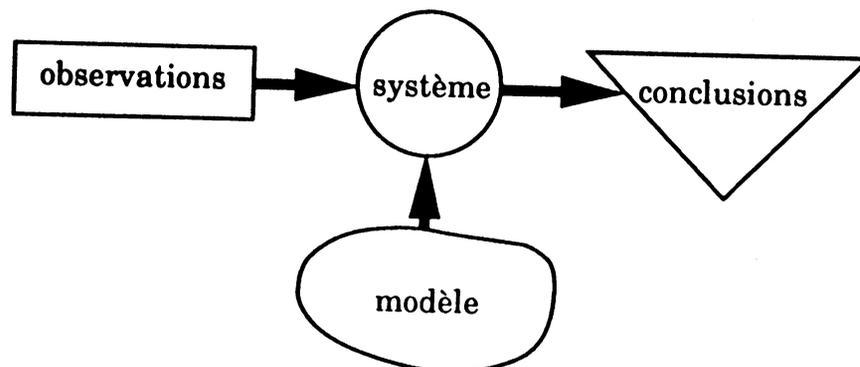


Figure 2.5 : Dans un système à modèle, il existe une décomposition du sujet et une connaissance des propriétés internes qui sont utilisées en fonction des observations pour tirer des conclusions.

La *connaissance superficielle* n'englobe que les liaisons entre les causes et les effets. Le système étudié apparaît ainsi comme une *boîte noire* à un type d'entrée, les causes, et un type de sortie, les effets. La connaissance superficielle exclut toute connaissance sur le fonctionnement interne de la boîte noire puisque sa composition ou sa structure sont inconnues. Cette connaissance superficielle peut être le fruit d'expériences ou d'habitudes. Il peut aussi s'agir d'un raccourci, résultat d'un raisonnement plus long qui exigerait des connaissances plus profondes sur le domaine, sans cependant utiliser des faits inconnus.

A l'opposé, la connaissance profonde exige la présence d'un modèle. Les systèmes à modèles manipulent une description du système qu'ils étudient. Pour cela, une décomposition de ce système en composants est nécessaire. Il faut ensuite décrire chacun de ces composants puis leurs fonctionnements internes et les relations qu'ils ont entre eux. Selon les auteurs, le vocabulaire n'est pas toujours le même (cf. une discussion des différences dans [BUIS87]). Dans ce document, il a été choisi de distinguer, au sein du modèle, les connaissances relatives à la *structure* de l'objet étudié, celles qui concernent chaque *composant* individuel et enfin celles qui donnent une description des *interactions* entre ces composants.

Cette section n'a pas pour objectif de prendre parti entre ces deux approches ni même d'étudier dans quel cas, en général, il est préférable d'utiliser l'une ou l'autre. Un exposé des différents points de vue présents dans la littérature peut être trouvé dans [BUIS87].

En revanche, pour ce qui concerne le raisonnement spatial dans les systèmes à base de connaissances, l'alternative ne se présente pas.

Le raisonnement dans l'espace exige un modèle de l'espace

La définition du raisonnement spatial indique qu'il est nécessaire de manipuler plusieurs objets spatiaux et les relations spatiales qu'ils entretiennent pour pouvoir parler de raisonnement spatial.

Cette condition revient à exiger un modèle de l'espace. Les systèmes à base de connaissances qui réalisent un raisonnement spatial sont donc des systèmes à modèles car ils manipulent des connaissances profondes.

Exemple 2.11 : Dans l'exemple 2.10, le raisonnement consiste à relier une valeur de pente à un risque d'avalanche. Un système à base de connaissances qui met en œuvre un tel raisonnement ne possède pas de modèle de l'objet étudié c'est-à-dire du site. Il ne manipule que des connaissances superficielles et ne raisonne pas dans l'espace. Un système qui possède et raisonne sur une décomposition du site en entités différentes et reliées spatialement entre elles, manipule un modèle spatial du site et réalise bien un raisonnement spatial.

Ces connaissances sont par ailleurs spatiales. La structure de l'espace étudié (ou plutôt du sous-espace) peut être fournie par une topologie, ou des fragments de frontière (qui seront complétés en une topologie). Cette topologie peut être un support pour l'énumération des composants.

Ces composants sont ensuite décrits par des connaissances de différents types (connexité éventuelle, frontières, diamètre, surface, inclinaison). Les interactions entre les composants sont représentées elles aussi par des connaissances de nature différente (alignement, superposition, distance, angle).

Les systèmes à base de connaissances retenus, dans le chapitre 1, comme faisant du raisonnement spatial, en particulier ELFIN [MCLO84] et NEUREX [XIANG&85] possèdent bien un modèle spatial des objets étudiés : le sous-sol et les sections de moelle épinière. Les systèmes SERF [INGR87] et ANAXAGORE [TROU89] manipulent également des modèles. En revanche, ERASME [ANTO&90] et EXPORT [MONT90], bien que raisonnant sur des objets spatiaux ne raisonnent pas sur l'espace, car celui-ci n'est pas présent en tant que tel dans ces systèmes : les relations spatiales entre les objets ne sont pas explicitées dans la base de connaissances.

Le raisonnement spatial étant défini, il convient d'en étudier les formes les plus intéressantes et les plus fréquentes. Trois grands types de raisonnements ont été identifiés. Le premier consiste à construire une décomposition de l'espace sous la forme des fermés minimaux d'une topologie. Il s'agit en fait de construire cette topologie. Le deuxième type consiste à passer de connaissances données peu pertinentes à des connaissances plus pertinentes. La troisième forme de raisonnement, très souvent implantée dans les méthodes d'accès des bases de données géographiques, consiste à localiser un objet dans une topologie.

2.3. DESCRIPTION DE L'ESPACE ET CREATION DE TOPOLOGIE

Dans de nombreuses applications, une décomposition est nécessaire. La plupart du temps cette décomposition peut être vue comme la partition de l'espace en fermés d'une topologie. Le problème de la décomposition de l'espace pose d'abord celui de la détermination des connaissances données.

2.3.1. Connaissances "données" = connaissances euclidiennes ?

Les connaissances introduites par l'utilisateur sont qualifiées de connaissances "données" et non pas de connaissances initiales. Le chapitre suivant reviendra sur cette distinction.

A l'occasion de la présentation des connaissances pictorielles, il a déjà été précisé que les informations dont l'acquisition est la plus facile à automatiser sont le plus souvent de type pictorial du fait de la nature des capteurs les plus courants. Ces informations de type pictorial peuvent être traduites en connaissances euclidiennes. De plus, même à travers les étapes de numérisation, ce sont en général des coordonnées de points qui sont saisies, donc, là encore, des connaissances en partie euclidiennes. L'acquisition de connaissances euclidiennes est la plus facile et, il faut le reconnaître, la plus exhaustive.

En effet, il est facile d'inférer des propriétés topologiques à partir de propriétés euclidiennes alors qu'il est souvent impossible de réaliser l'opération inverse. De plus à la différence de nombreux domaines d'applications des systèmes à base de connaissances, le raisonnement spatial ne peut déterminer à l'avance les connaissances qui lui seront nécessaires.

Exemple 2.12: En diagnostic médical ou technique, les connaissances nécessaires sont, la plupart du temps, relativement peu nombreuses et surtout déterminées à l'avance.

En revanche, dans le contexte de l'analyse de sites avalanches, les connaissances importantes seront dégagées au cours du raisonnement. Par conséquent, la donnée du terrain (c'est-à-dire de l'espace) doit être faite avec les données les plus brutes et de la façon la plus neutre. En effet, on ne sait pas a priori si tel accident de terrain va jouer un rôle important ou si la distance entre deux arêtes doit être connue du système. Plus précisément, il est nécessaire de fournir un *modèle numérique de terrain*. Le système doit ensuite aller lire sur cette représentation les informations qui l'intéressent comme l'expert lit avec son regard les informations sur le terrain.

Le raisonnement spatial exige donc une description exhaustive de l'espace. Cette description doit être la plus neutre possible. Elle joue le rôle du terrain face au regard de l'utilisateur. Raisonner dans l'espace est très difficile sans une représentation structurée de cet espace. Très fréquemment, les connaissances spatiales à partir desquelles le raisonnement sera conduit seront donc euclidiennes.

Malheureusement des connaissances purement euclidiennes ne suffisent pas à donner une décomposition de l'espace qui permette toutes les formes de raisonnement spatial. En particulier il est souvent nécessaire de travailler sur une topologie de l'espace. Or celle-ci n'est pas toujours une connaissance "donnée" obtenue directement à partir des capteurs. Il faut créer cette topologie.

2.3.2. Création de topologie

Il est nécessaire de connaître les objets dont les propriétés spatiales vont intervenir dans le raisonnement. Il s'agit d'ensembles ou d'éléments de ces ensembles. Ces objets doivent parfois être construits au sein du raisonnement. En général, ils sont bâtis sur une topologie. Tout raisonnement s'appuiera sur une décomposition de cet espace. La topologie utilisée caractérisera la finesse de ce raisonnement. En d'autres termes, il est impossible de découper un ouvert.

En mathématiques, les topologies les plus classiques ont la propriété d'être infinies. En d'autres termes, il est possible de considérer une infinité de voisinages pour un point, c'est-à-dire d'ensemble aussi petits soient-ils qui

entourent ce point. La topologie la plus utilisée sur une droite est celle des segments, c'est-à-dire que la topologie contient l'ensemble des segments ouverts (sans les bornes) de cette droite. Cet ensemble est infini. De même, l'espace plan est souvent muni de la topologie des boules ouvertes pour une métrique donnée. Il est par exemple possible de considérer que l'être humain manipule, dans sa vie de tous les jours une topologie de boules bâties sur la distance euclidienne. Les voisinages d'un point sont alors l'ensemble des boules centrées sur ce point.

L'idée retenue ici est de travailler sur une topologie finie. Il n'y aura plus pour un point qu'un nombre fini de voisinages. Cette topologie peut être décrite en donnant en particulier l'ensemble des fermés. Il est préférable, pour limiter la taille de l'ensemble à décrire, de donner l'ensemble des fermés minimaux c'est-à-dire les fermés qui ne sont strictement inclus dans aucun autre fermé. Il s'agit en fait d'une décomposition de l'espace en sous-parties connexes (i.e. en seul morceau), qui contiennent leur frontière, dont les intérieurs (la sous-partie moins sa frontière) sont disjoints et dont la réunion recouvre l'ensemble de l'espace concerné.

De façon moins formelle, l'intérêt d'une telle approche est d'offrir un cadre rigoureux au raisonnement topologique (continuité, connexité, adjacence...) sans pour autant manipuler un trop grand nombre d'objets.

Lors de la construction d'une topologie de ce type, quatre situations peuvent être rencontrées :

Création par "complétion"

La nouvelle topologie est créée par "complétion". La connaissance donnée ne contient pas de topologie mais des fragments de topologie ou des morceaux de frontières. Il s'agit alors de construire une des topologies dans laquelle les frontières incorporent ces éléments donnés et incomplets.

Une partie seulement des fermés est définie, les autres ne sont donnés que par des fragments de leur frontière.

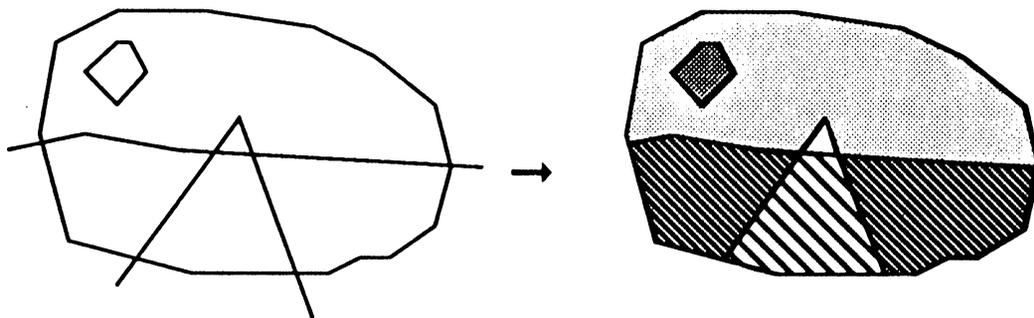


Figure 2.6 : Créer une topologie par "complétion" consiste à créer les fermés minimaux qui s'appuient sur des morceaux de frontières au sein d'un espace donné. Le dessin de gauche représente des lignes telles qu'elles ont pu être obtenues par numérisation. Il ne s'agit pas encore d'une topologie. Pour obtenir celle-ci, il est nécessaire de repérer les différentes sous-parties du dessin de droite et de les définir par leur frontière.

Exemple 2.13 : Suite au travail de numérisation d'une carte, un système d'information géographique ne connaît que des fragments de frontière et des points intérieurs. De plus des fragments de frontières peuvent se croiser. Le système doit construire une topologie. Pour cela, il détermine les arcs élémentaires et génère à partir d'eux les fermés de la topologie. Le système ARC-INFO réalise ainsi ces actions au lancement des commandes CLEAN et BUILD.

Exemple 2.14 : Le "large-scale space" de Kuipers ne peut être observé, par définition, d'un seul point de vue. De chaque point de vue, il est possible d'observer une fraction de l'espace et d'en enregistrer en particulier la topologie sous la forme d'une décomposition de l'espace observé de ce point de vue. Lors de la prise en compte de l'espace dans son ensemble, il est nécessaire de construire une topologie complète, c'est-à-dire une décomposition de l'ensemble de l'espace. Celle-ci contient les fermés qui se trouvaient exclusivement à l'intérieur d'une fraction de l'espace. Sur les bords de chaque fraction, il est nécessaire de créer les fermés à partir des fermés propres à chaque fraction.

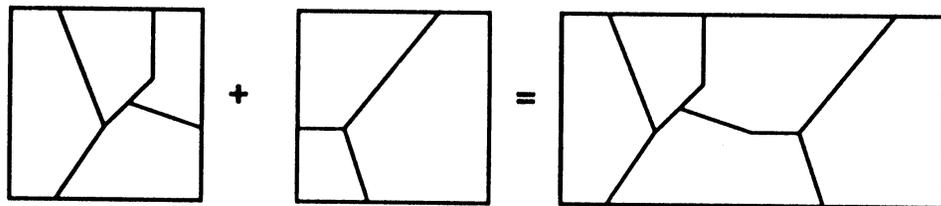


Figure 2.7 : Un "large-scale space" a été observé de deux points de vue. A chaque fois une topologie a été construite. Lors de la "fusion" des points de vues, il faut créer une topologie de l'ensemble de l'espace. Pour cela, les topologies de chaque point de vue sont utilisées sur la totalité de l'espace. Leur réunion n'est pas une bonne topologie, il faut donc la compléter.

Exemple 2.15 : Martin-Clouaire fait réaliser à ELFIN des constructions de topologie dans un graphe [MCLO84]. En effet, ELFIN bâtit au cours de son raisonnement le chemin suivi par les hydrocarbures de la roche mère vers le piège où ils seront prélevés. Ce chemin se présente (cf. chapitre 1) sous la forme de l'arborescence des contextes. La représentation de l'arborescence que manipule ELFIN évolue au cours du raisonnement. Cette arborescence est en fait bâtie à partir des relations topologiques d'adjacence. ELFIN complète donc sa topologie du sous-sol par "completions" successives.

Création par croisement

Dans ce cas, la nouvelle topologie est obtenue par croisement de deux ou plusieurs topologies (il est toujours possible de se ramener au cas de deux topologies). Les nouveaux fermés minimaux sont constitués des intersections non vides des fermés minimaux de chaque topologie.

Exemple 2.16 : Une des capacités les plus intéressantes des systèmes d'information géographique est la possibilité de croiser des couvertures. Il s'agit d'abord de construire une topologie qui contienne les deux topologies initiales. Puis les caractéristiques sont distribuées sur la nouvelle topologie. Dans le modèle SPATIAREL de David [DAVI88], l'opération de produit de deux cartographies s'apparente également à un croisement de topologies tel qu'il vient d'être défini.

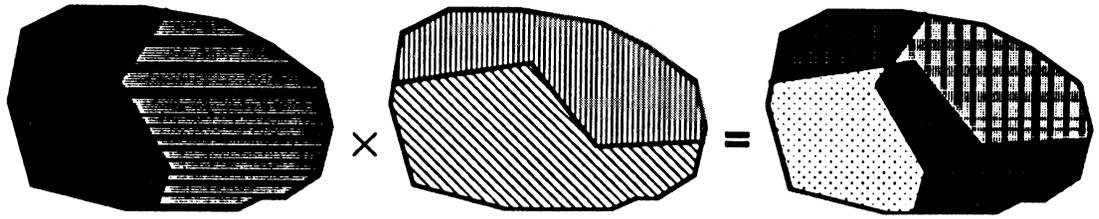


Figure 2.8 : Créer une topologie par croisement revient à considérer les intersections des fermés des deux topologies données. Il s'agit de repérer les fermés obtenus par l'intersection de fermés provenant des deux topologies.

Création ex-nihilo

Il s'agit de créer une topologie d'un domaine donné, à partir éventuellement de points donnés, mais sans aucun fragment de frontière.

Exemple 2.17 : La triangulation de Delaunay est une triangulation bâtie sur un semis de points au sein d'un polygone. Il s'agit en fait de relier ces points pour obtenir des triangles globalement "les plus équilatéraux possibles", c'est-à-dire dont les côtés ont des longueurs les plus proches possibles. Le résultat est une topologie dans le polygone. Les frontières obtenues ne figurent pas parmi les données, même sous forme de tronçons.

Création par appauvrissement de topologie

Dans ce cas, il faut créer une topologie par l'aggrégation des fermés d'une topologie déjà existante. L'agglomération se fait selon des critères qui dépendent de l'application courante.

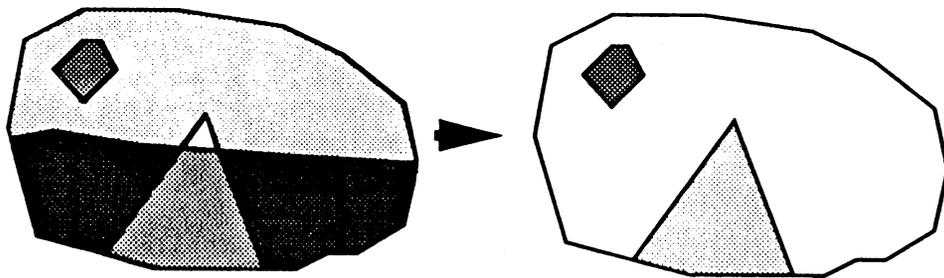


Figure 2.9 : Appauvrir une topologie consiste à réunir des fermés de la topologie donnée. La topologie obtenue est plus pauvre car ses fermés sont plus grands que ceux de la topologie de départ.

Exemple 2.18 : Dans le système ELSA, les *petits panneaux* sont obtenus en assemblant les triangles qui ont les mêmes propriétés ou les même caractéristiques en respectant de plus des critères de convexité (cf. chapitre 4). Les petits panneaux ainsi obtenus forment une topologie dont ils sont les fermés minimaux. Cette topologie sera celle utilisée par ELSA pour réaliser son analyse de sites avalancheux.

Exemple 2.19 : Des systèmes d'information géographique sont capables de vectoriser un tableau de pixels. En d'autres termes, ils peuvent générer les frontières de zones de pixels à valeur homogène c'est-à-dire une nouvelle topologie. De son côté le tableau de pixels peut être considéré comme une topologie dont les fermés sont les pixels.

Exemple 2.20 : Scholl et Voisard présentent dans [SCHO&89] quelques primitives pour la manipulation de cartes thématiques. Parmi ces primitives, certaines correspondent à des opérations de création de topologies. Par exemple, l'opération de fusion s'apparente à un appauvrissement de topologie. Des frontières, respectées jusqu'à présent, sont supprimées.

2.4. D'UNE CONNAISSANCE A L'AUTRE

Raisonnement dans l'espace, ce n'est pas seulement créer des topologies. Quand les objets à analyser ou à prendre en compte ont été définis (à partir de l'inférence d'une topologie par exemple), le raisonnement dans l'espace va consister à inférer d'autres propriétés spatiales qui ne figuraient pas dans l'ensemble des connaissances données ni dans une topologie créée.

Ces inférences qui permettent de passer d'un ensemble de connaissances spatiales à un autre ont deux applications différentes. La première trouve son origine dans les différences de pertinences entre connaissances spatiales, la seconde dans la possibilité qu'a l'utilisateur de représenter, à pertinence égale, un objet de différentes façons.

2.4.1. D'un type de connaissance à un autre

Les connaissances les plus pertinentes ne sont que rarement données. Il a, en effet, été montré que les connaissances données sont plutôt euclidiennes (c'est le cas des modèles numériques de terrain, des résultats de numérisation, des connaissances fournies par la plupart des capteurs...) et par conséquent nombreuses mais peu pertinentes. Le raisonnement spatial prend alors l'aspect décrit par la figure 2.10. Les connaissances données peuvent être considérées comme peu pertinentes vis-à-vis de l'application. En revanche, des connaissances intermédiaires ou terminales que construit le système le sont, elles, beaucoup plus.

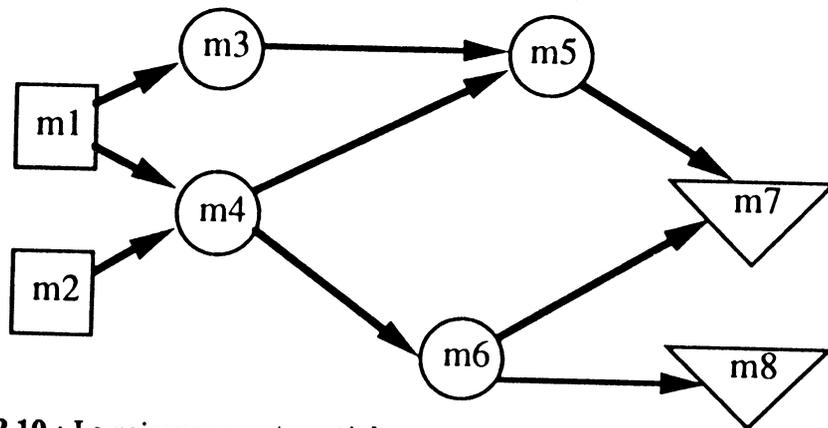


Figure 2.10 : Le raisonnement spatial consiste à inférer des connaissances à partir de connaissances de départ (les carrés). Le plus souvent, ces connaissances sont peu pertinentes. Elles sont utilisées par le système à base de connaissances pour inférer des connaissances intermédiaires (les cercles) qui correspondent à une partie des connaissances de la base (par exemple les connaissances d'un expert). Les connaissances terminales (les triangles) sont les plus pertinentes.

Dans les systèmes d'information géographique, les entités spatiales sont définies dans des phases de numérisation. Il s'agit le plus souvent d'arcs, de points ou de polygones. Afin d'améliorer le fonctionnement du système, des

informations topologiques complémentaires sont calculées (cf figure 1.4). Une redéfinition des arcs est assurée en déterminant l'ensemble des intersections.

- Pour les arcs, sont calculés en général les points de départ et d'arrivée, les polygones à droite et à gauche et les arcs prolongés ou prolongeants.

- Pour les polygones, sont déterminés les différents arcs frontières, en précisant dans quel sens ces arcs sont parcourus. Souvent les surfaces des polygones sont également calculées.

Il s'agit là encore de passer d'un niveau de description assez pauvre, essentiellement euclidien, à un niveau topologique (les limites et les voisins) et vectoriel (les sens de parcours). Parce que ces niveaux sont plus pertinents, le système tire avantage à les utiliser. Il est plus efficace.

Ce passage des connaissances spatiales peu pertinentes vers des connaissances spatiales pertinentes est immédiat chez l'expert. Les connaissances topologiques sont utilisées en permanence, contrairement aux connaissances euclidiennes, simples "réservoirs de données".

Exemple 2.21 : Un système tel que NEUREX [XIAN&85] réalise le même type de raisonnement avec les différents modèles qualifiés d'analogique (qui est plutôt euclidien) et de propositionnel (plutôt topologique). A partir d'une description euclidienne de sections de la moelle épinière ce système construit des relations topologiques. Elles seules peuvent être exploitées par les connaissances de l'expert. Celui-ci n'a pas une image précise (au sens métrique ou euclidien) des sections de la moelle mais simplement une description des relations d'adjacence. Ce sont ces relations d'adjacence qui sont utilisées pour le diagnostic médical proprement dit.

Exemple 2.22 : Le système ELSA qui sera décrit en détail dans le chapitre 4 dispose, à un stade de son raisonnement, d'entités spatiales considérées comme pertinentes pour l'analyse de sites avalancheux : les petits panneaux. Ces entités ne sont données à ce stade du raisonnement que par une description euclidienne en terme de coordonnées des sommets. Or ce ne sont pas les connaissances les plus utilisées dans l'analyse de sites avalancheux. Le système ELSA doit donc inférer ces propriétés pertinentes : adjacence entre petits panneaux, pentes et expositions, organisation par rapport aux lignes de plus grande pente.

Cette prépondérance des connaissances pertinentes doit donc non seulement donner son sens au raisonnement spatial mais aussi diriger les méthodes d'inférences ; en d'autres termes, les connaissances topologiques, puisque utilisées très souvent, doivent bénéficier de certaines priorités. Ce point sera détaillé dans le chapitre suivant.

2.4.2. D'une représentation à une autre

Le même objet spatial peut être décrit de différentes façons sans qu'il soit toujours possible d'accorder plus de pertinence à une description qu'à une autre.

Exemple 2.23 : Un point dans le plan peut être défini par ses coordonnées polaires dans un repère ou par ses coordonnées cartésiennes dans un autre repère. Il est aussi possible d'être en présence de deux repères polaires ou de deux repères cartésiens, voire d'un plus grand nombre de repères. Selon les raisonnements à mener, un repère peut apparaître comme plus intéressant qu'un autre ou bien encore moins intéressant. Si l'utilisateur désire exploiter aussi bien une description qu'une autre, il doit les manipuler toutes les deux.

Un système comme ARC-INFO permet par exemple de manipuler plusieurs dizaines de systèmes de projection [ESRI89].

Exemple 2.24 : Un polygone peut être approché de différentes façons : par le cercle circonscrit, par le rectangle enveloppant ou encore par l'ellipse enveloppante. Chacune de ces approximations peut être intéressantes dans un raisonnement donné de géométrie approximative telle que celle présentée dans [MANO&86] ou dans la section suivante.

Dans ce cas, il est rare de disposer de chacune de ces descriptions au départ du raisonnement. Il faut alors en inférer une à partir des autres. Par exemple, il est utile de passer d'une représentation polaire à une représentation cartésienne.

2.5. LOCALISATION D'UN OBJET DANS UNE TOPOLOGIE

Il s'agit d'un autre exemple de raisonnement spatial important et très courant. Il s'agit, face à une décomposition d'un espace en connexes fermés d'une topologie, de localiser un point ou un objet dans le ou les connexes qui le contiennent. La description du point ou de l'objet peut être faite de différentes manières. Les raisonnements qui vont être exploités font donc appel eux aussi à des connaissances de natures différentes.

2.5.1. Le problème

La localisation d'un objet dans un espace consiste le plus souvent à déterminer dans quelle(s) partie(s) de l'espace il est situé. Ce problème fait l'objet de nombreuses recherches. C'est le cas en particulier en géométrie algorithmique ou dans les systèmes de traitement de l'information géographique pour les méthodes d'accès. Il ne s'agit pas ici de proposer de nouveaux algorithmes qui dépassent largement le cadre de cette étude mais de mettre en évidence différents types de raisonnements liés aux connaissances utilisées.

Une bonne formalisation de ce problème revient à considérer une partition en fermés connexes minimaux d'une topologie de l'espace et à déterminer sur quels fermés est situé l'objet.

Un exemple est retenu ici pour illustrer cette section. Il s'agit de savoir si une localité donnée se trouve en Basse-Normandie ou non. Pour mettre en évidence les différentes formes de raisonnement possibles, différentes localités vont être étudiées. *La France est supposée décomposée dans la partition de ses régions qui sont disjointes.*

L'explication donnée à chaque réponse sera considérée comme caractéristique du raisonnement suivi.

2.5.2. Raisonnement ensembliste

“Nantes, capitale de la Bretagne, est-elle en Basse-Normandie ?”

La réponse est évidemment non car les régions sont disjointes et qu'une même localité ne peut se trouver dans deux régions.

Le raisonnement tenu ici consiste à exploiter des connaissances ensemblistes. Celles-ci peuvent être plus complexes que celles de l'exemple et être basées sur une décomposition hiérarchique de l'espace.

Ainsi, dans le domaine des bases de données spatiales, ces informations ensemblistes sont très utilisées pour l'optimisation de requêtes. Les

“quadrees” [SAME89] sont utilisés dans ce contexte ainsi que la plupart des index spatiaux.

Exemple 2.25: Si la question, qui portait sur Nantes, porte désormais sur Quiberon et qu’il est connu que le canton de Quiberon (sur lequel, évidemment, se trouve Quiberon) est en Bretagne, le même type de raisonnement ensembliste peut être exploité.

Il est possible de multiplier les niveaux dans la hiérarchie d’inclusion. Le génie administratif français nous offre en effet une bien belle structure arborescente : arrondissements, communes, cantons, départements, régions, sans compter les découpages qui ne s’y superposent pas, hérités qu’ils sont des anciennes provinces ou des hasards de l’histoire (académies, régions militaires, régions judiciaires...).

Pour pouvoir tenir ces raisonnements ensemblistes, il est nécessaire de posséder les connaissances sur l’appartenance de la localité à un ensemble lié d’une façon ou d’une autre à la Basse-Normandie.

Une fois cette appartenance établie, le raisonnement ensembliste ne peut plus être considéré comme spatial. De la même façon, si la construction d’un quadtree exige des formes de raisonnement spatial (du type de celles qui suivent), son exploitation ne relève pas du raisonnement spatial.

2.5.3. Raisonnement topologique sur les frontières

“Caen est en Basse-Normandie; la Seine est une frontière de la Basse-Normandie ; pour aller de Caen au Havre, il faut franchir une et une seule fois la Seine ; Le Havre est-il en Basse-Normandie ?”

Il est clair que la réponse est non. Il n’y a pas de chemin du Caen au Havre qui soit complètement en Basse-Normandie.

C’est le même type de raisonnement qui est utilisé pour déterminer si un point est à l’intérieur d’un polygone en étudiant la parité du nombre d’intersections entre les limites du polygone et une demi-droite ayant pour origine le point étudié.

2.5.4. Raisonnement métrique

“Noyarey est-il en Basse-Normandie ? Il y a environ 800 km entre Noyarey et Caen, la capitale de la Basse-Normandie ; le diamètre d’une région ne dépasse jamais 400 km”.

Peu de personnes savent que Noyarey se trouve en Rhône-Alpes. Cette connaissance n’est pas utilisée. Le raisonnement ensembliste précédent ne peut être exploité. En revanche il est clair que Noyarey est trop *loin* de Caen pour être en Basse-Normandie.

Ici le raisonnement tenu est de nature métrique. C’est en fait l’inégalité triangulaire qui est utilisée (cf. annexe).

Le raisonnement métrique n’exige pas nécessairement l’utilisation de la distance euclidienne. L’être humain l’évalue certes facilement mais l’ordinateur beaucoup moins. Il est possible d’utiliser d’autres distances plus faciles à calculer mais pour lesquelles, par définition, l’inégalité triangulaire reste vraie. La distance d_2 définie en annexe permet de donner une explication métrique à la technique de localisation basée sur les carrés ou rectangles “enveloppants”. Cette technique est très utilisée en informatique pour éliminer facilement un grand nombre de cas manifestes de non appartenance d’un point à un polygone.

2.5.5. Raisonnement vectoriel

“Paris se trouve, le long de la Seine, à l’aval de Fontainebleau et à l’amont de la Basse-Normandie. Fontainebleau est-il en Basse-Normandie ?”

La Seine permet d’ordonner, sur un axe, Fontainebleau, Paris et la Basse-Normandie. Fontainebleau ne peut donc être situé en Basse-Normandie.

Le raisonnement tenu ici est vectoriel. Il consiste à placer un certain nombre d’objets sur un axe (ici la Seine) et à utiliser l’ordre sur cet axe pour éliminer la ville de Fontainebleau.

2.5.6. Raisonnement géométrique euclidien

“L’abbaye de Jumièges se trouve dans un méandre de la Seine en Normandie ; est-elle en Basse-Normandie ?”

Cette fois-ci, il est impossible de savoir si Jumièges est en rive droite (Haute-Normandie) ou en rive gauche (Basse-Normandie). La complexité des méandres l’interdit. Il faut alors avoir recours à un positionnement exact de Jumièges sur une carte ou dans un repère et déterminer sa position par rapport à la Seine. Cette requête qui vise à *établir une connaissance topologique* nécessite des calculs dans le repère euclidien. Le raisonnement est donc géométrique dans un premier temps, il est ensuite topologique.

Les différents exemples de raisonnements de localisation qui viennent d’être présentés correspondent à la nature des connaissances données pour chaque problème. Selon la nature de ces connaissances, il est judicieux d’utiliser un raisonnement ensembliste, vectoriel, métrique ou topologique. Ici encore, la distinction entre connaissances spatiales se traduit par des différences de raisonnement. Il apparaît aussi qu’il ne peut y avoir de méthode unique pour localiser un objet dans une topologie.

CONCLUSION

Ce chapitre suggère, à propos du raisonnement spatial, des définitions et des regroupements qui prennent en compte les nombreux exemples et applications du premier chapitre. La définition des connaissances spatiales proposée dans ce chapitre est basée sur des structures mathématiques. Elle met bien en évidence une classification des connaissances spatiales ainsi qu’une hiérarchie de pertinence qui distingue dans l’ordre croissant connaissances euclidiennes, vectorielles, métriques et topologiques. Le raisonnement spatial respecte cette classification et cette hiérarchie, que ce soit pour construire une représentation d’un espace, enrichir une description de l’espace ou localiser des objets dans une topologie.

Ces caractéristiques du raisonnement spatial en général prennent un sens particulier lorsque ce raisonnement est assuré par un système à base de connaissances, car celui-ci permet alors une exploitation efficace de ces caractéristiques en particulier pour l’aspect dynamique. C’est ce que va tenter de montrer le chapitre suivant.

3. DYNAMIQUE DES BASES DE CONNAISSANCES SPATIALES

Le chapitre précédent a dégagé certaines caractéristiques du raisonnement spatial en général. Il convient maintenant d'étudier quels avantages il y a à implanter un tel raisonnement dans un système à *base de connaissances spatiales*.

L'étude de la dynamique des bases de connaissances spatiales est le point central de ce chapitre. La dynamique est d'abord celle des connaissances qui sont introduites dans la base et de leur *acquisition*. Une formalisation basée sur la notion de connaissances données est introduite ainsi qu'une typologie des systèmes à base de connaissances fondée sur cette notion. La dynamique provient aussi des inférences réalisées sur les connaissances spatiales et de l'enregistrement de ces inférences qui est souvent nécessaire. L'attachement procédural est une des formes d'inférence les plus utilisées. Enfin, cet enregistrement exige l'utilisation d'un système de *maintien de la vérité* (TMS). Sur une base de connaissances spatiales, l'utilisation de TMS d'un type particulier permet d'exploiter un parallèle entre la localité dans le graphe de dépendances et la localité spatiale.

3.1. ACQUISITION DES CONNAISSANCES

Un système à base de connaissances produit des raisonnements. Il crée de la connaissance à partir d'autres connaissances. Celles-ci n'ont pas toujours été inférées. Elles ont pu être introduites dans la base. Ce problème est étudié dans ce chapitre. Il convient donc dans un premier temps de définir ces connaissances "données".

3.1.1. Connaissances données et connaissances inférées

Dans cette sous-section la distinction entre connaissances inférées par le système à base de connaissances et connaissances fournies à ce système est précisée. Dans la sous-section qui suit celle-ci, cette distinction permettra d'évoquer le problème des inférences et de l'acquisition.

Les connaissances d'une base comprennent les connaissances introduites par l'utilisateur ou le monde extérieur et les connaissances inférées par le moteur d'inférence.

Dans l'ensemble des connaissances présentes dans une base (effectivement ou potentiellement) à un instant donné, une fonction est définie qui relie une connaissance aux connaissances qu'elle a directement permis d'inférer (cette fonction est à valeur dans l'ensemble des parties de l'ensemble des connaissances).

Cette fonction est appelée fonction de justification. Par définition le domaine de cette fonction est l'ensemble des *connaissances inférées*. Le complémentaire de cet ensemble est l'ensemble des *connaissances données*. Les connaissances données sont donc celles qui n'ont pas été inférées. Elles ont été introduites dans la base au cours d'une phase d'acquisition. Parmi les

connaissances données, il y a les connaissances propres au système de gestion de la base de connaissances, les connaissances propres au système de l'application et enfin celles propres à la session.

Le terme de connaissances données a été préféré à celui de connaissances initiales car ce dernier n'est pas adapté au cas où des connaissances sont introduites par l'utilisateur *au cours du raisonnement*.

Les connaissances données peuvent être implicites ou explicites. Par définition, les *connaissances implicites* sont celles qui figurent dans la base en tant qu'éléments du graphe d'une fonction. Ceci signifie que si une fonction existe en tant que telle dans la base de connaissances, l'ensemble de son graphe est compris dans cette base de connaissances. Ces connaissances implicites seront traitées dans la section suivante.

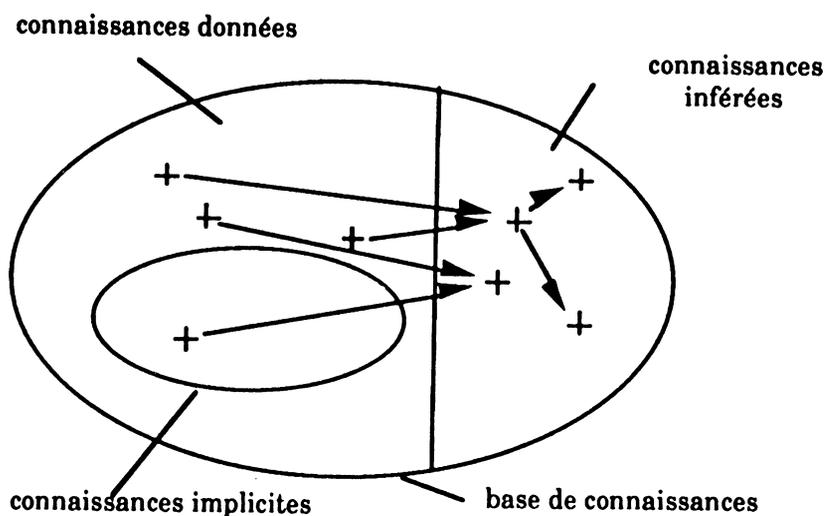


Figure 3.1 : Les connaissances d'une base. Les connaissances données ont toutes été introduites par l'utilisateur. Il peut s'agir de connaissances explicitement entrées ou de connaissances implicitement entrées (graphe des fonctions). La fonction de justification est représentée sur la figure par les flèches. L'union des éléments du co-domaine de cette fonction est l'ensemble des connaissances inférées.

Exemple 3.1 : Euzenat a implanté un réseau de justification dans SHIRKA [EUZE87]. Ce réseau de justification est équivalent au graphe de la fonction de justification présentée ci-dessus.

Cette définition véhicule un certain arbitraire. Il est par exemple clair que l'ensemble des inférences possibles est "implicitement" dans la base. La définition proposée ici correspond à une approche PROLOG dans laquelle il est possible, d'une part, de considérer l'ensemble des clauses présentes dans la base à un instant donné et, d'autre part, de construire des prédicats pré-définis qui correspondent aux fonctions. Ce problème sera repris dans la prochaine section de ce chapitre.

Cette section examinera également l'attitude à adopter vis-à-vis des connaissances inférées. En particulier, est-il nécessaire d'en enregistrer les résultats.

3.1.2. Acquisition et inférence

La tradition en informatique conduit le plus souvent à traiter de façon séquentielle les phases d'initialisation et d'acquisition d'une part et de calcul

d'autre part. Cette tradition vient de la volonté d'automatiser les calculs. Les données étant fournies, le programme se débrouille tout seul. Ceci est justifié pour les programmes "en batch" ou pour les programmes dont le déroulement est trop long pour que l'utilisateur reste devant la machine. Cette nécessité disparaît quand l'utilisateur peut rester à côté de la machine ou quand le calcul se déroule alors que de nouvelles informations arrivent.

Les mêmes situations se rencontrent dans la plupart des systèmes à base de connaissances mais pas dans tous. Il est assez utile de placer un système à base de connaissances dans des situations différentes pour saisir les différentes formes de la dynamique de l'acquisition.

Les connaissances données peuvent avoir été entrées dans le système avant toute inférence dans une phase d'initialisation. Elles peuvent être entrées après les premières inférences. Un système à base de connaissances chargé d'analyser une situation bien définie peut avoir une phase d'acquisition très distincte de la phase d'inférence. Si la situation est susceptible d'évolution et que le système doit maintenir son analyse en permanence, ou si les connaissances ne peuvent être acquises en une fois, les phases d'acquisition et d'inférence doivent être intimement liées.

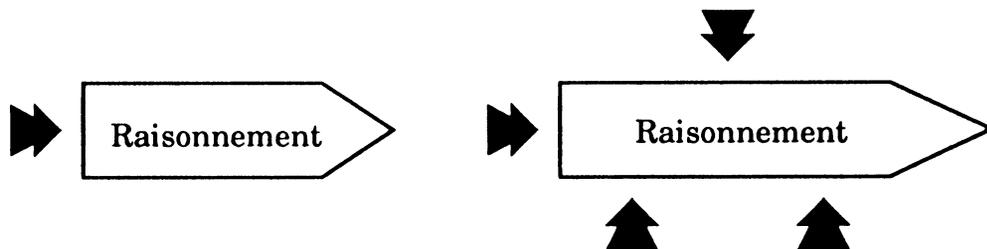


Figure 3.2 : L'acquisition de connaissances données est représentée par les flèches noires. Elle peut être distincte dans le temps du raisonnement, c'est-à-dire de la phase où s'exécutent les inférences. Elle peut y être mêlée lorsque les connaissances initiales sont susceptibles d'évolution ou lorsqu'elles ne peuvent être fournies en une fois.

Exemple 3.2 : Un système de diagnostic médical se voit décrire un ensemble de symptômes en une fois. Il réalise un diagnostic puis attend une autre session qui commencera par une phase d'acquisition.

Exemple 3.3: S'il doit gérer une chaîne de production, des capteurs fournissent au système des connaissances le renseignant sur la situation de cette chaîne. A chaque nouvelle acquisition il réagit ou doit pouvoir réagir. C'est le cas des systèmes "temps réel".

Exemple 3.4 : Un système à base de connaissances contient la représentation de l'environnement d'un robot mobile. Au gré de ses déplacements, les connaissances sur l'environnement se complètent par acquisition grâce à des capteurs. A chaque instant le robot doit pouvoir disposer d'une représentation de cet environnement la plus exacte possible.

Mais mélanger dans le temps les phases d'acquisition de connaissances données et celles de raisonnement posent le problème du maintien de la vérité dans la base de connaissances. En d'autres termes, que deviennent les connaissances inférées à l'aide de connaissances qui viennent d'être modifiées ou qui ont été supprimées ? Cette question sera étudiée dans la dernière section de ce chapitre.

3.2. INFERENCE SPATIALES

Certains aspects des inférences dans une base de connaissances spatiales sont maintenant passés en revue. Ils concernent notamment les aspects calculatoires et les appels à des fonctions. L'importance de l'attachement procédural justifie la première sous-section, qui tente de donner un cadre à l'attachement procédural dans les systèmes à base de connaissances. Le choix des inférences et des représentations est ensuite étudié avant que l'intérêt de l'enregistrement des inférences ne soit présenté. Celui-ci apparaît nécessaire dans de nombreux cas. Il peut être dirigé par la pertinence des connaissances.

3.2.1. Attachements procéduraux

Connaissances implicites et fonctions externes

La plupart des systèmes de gestion de bases de connaissances permettent l'utilisation de connaissances implicites. Celles-ci ne sont pas inférées mais n'ont pas été non plus introduites par l'utilisateur. Le moteur d'inférence est simplement capable d'y accéder.

Typiquement, les fonctions sont en général introduites sous forme de connaissance implicite ; leur graphe est en effet trop volumineux pour être entré dans la base de connaissances en tant que tel par l'utilisateur.

S'il s'agit d'une fonction arithmétique classique on pourra considérer qu'elle a été acquise lors de la construction du système de gestion de base de connaissances.

Exemple 3.5 : L'unité de connaissance de l'exemple 1.1 fait référence à la division :

"Pour obtenir la moyenne des salaires au laboratoire ARTEMIS, il faut faire la division de la masse salariale par le nombre de salariés"

La fonction division est donc présente dans toute base de connaissances qui désirera exploiter cette connaissance. Son graphe s'y trouve de façon implicite. Ceci signifie que la base de connaissances contient implicitement toutes les unités de connaissances du type :

"La division de x par y est z".

L'ensemble de ces unités est infini. C'est pourquoi il ne figure que de façon implicite dans la base de connaissances.

Dans PROLOG le prédicat `is` indique un appel à une fonction externe. Comme il l'a déjà été signalé plus haut, ce prédicat signale au moteur d'inférence qu'il ne doit pas se contenter d'unification mais bien de génération de valeur.

La situation est en fait identique à celle où la base de connaissances contiendrait l'ensemble (infini!) des graphes des fonctions prédéfinies telle que `f` et où l'appel à ces fonctions se ferait par des prédicats du type :

$f(_x, _y, _z)$
au lieu de :
`is(_z, f(_x, _y))`

Exemple 3.6: L'addition dans PROLOG se fait par l'appel au prédicat `is` :

`is(_z, +(_x, _y))` signifie par exemple que `_z` est la somme de `_x` et de `_y`. Si l'interprète doit effacer ce but, il le fait en vérifiant que les valeurs des variables vérifient l'addition. Il peut éventuellement calculer la valeur de `_z` si les variables

`_x` et `_y` sont instanciées mais ne peut calculer la valeur de `_x` (respectivement de `_y`) même si les valeurs de `_z` et de `_y` (respectivement de `_x`) sont connues.

L'exemple précédent montre toutefois que la situation n'est pas tout-à-fait identique dans les deux cas, car l'unification dans PROLOG peut se faire indifféremment sur `_x`, `_y` ou `_z` dans `f(_x, _y, _z)` alors qu'elle ne se fait que sur `_z` dans `is(_z, f(_x, _y))`.

Comme il l'a déjà été signalé, le graphe entier de la fonction correspondante est supposé contenu dans la base de connaissances. C'est d'ailleurs ainsi que les attachements procéduraux sont traités dans SHIRKA. Chaque appel à un attachement procédural conduit à la création d'un objet qui est en fait un élément du graphe de la fonction appelée ; et dans SHIRKA la connaissance est contenue dans la base d'objets.

Exemple 3.7: Les attachements procéduraux dans SHIRKA sont installés comme des facettes `ssib-exec` permettant le calcul de valeurs d'attributs ou des facettes `sa-verifier` contrôlant certaines contraintes [RECH&89]. Derrière la facette est décrite une classe, spécialisation de la classe `méthode`. Cette classe possède en attributs le nom d'une fonction LISP et des arguments : plusieurs peuvent être des arguments d'entrée, un seul est argument de sortie.

A l'appel de l'attachement procédural, une instance de cette classe est créée incomplète. Le moteur d'inférence la complète grâce à la fonction qu'elle contient et rend la valeur. En d'autres termes, la base de connaissances contient de façon implicite le graphe entier de cette fonction.

Si la connaissance implicite correspond à une fonction représentant une connaissance du domaine de l'application, on pourra considérer qu'elle a été entrée lors de la construction de l'application.

La qualité de l'explication est inversement proportionnelle à la complexité des fonctions. Il va de soi, en effet, que l'explication ne peut porter sur le résultat d'un appel à une fonction. Les connaissances implicites ne sont pas par définition explicables comme d'ailleurs toutes les connaissances données. Seul peut les expliquer l'utilisateur qui les a rentrées dans la base de connaissances. Ceci n'est pas étonnant : prétendre expliquer le graphe de toute fonction externe au système est naturellement impossible.

Calculs géométrique

Les connaissances implicites jouent un rôle très important dans la manipulation de connaissances spatiales puisque de nombreux calculs géométriques ou inférences spatiales sont réalisés par des appels à des fonctions.

Ces fonctions peuvent être surtout des fonctions de calculs numériques mais aussi des fonctions implantant des algorithmes non numériques ou partiellement numériques.

Il a été expliqué dans le chapitre précédent que les connaissances nécessaires à de nombreuses formes de raisonnement spatial devaient être exhaustives. Très souvent, il s'agit de la donnée des coordonnées de points dans un repère euclidien. A partir de ces coordonnées un certain nombre de connaissances plus pertinentes sont inférées. La nature des connaissances de départ exige de nombreux calculs géométriques. Ces calculs sont implantés à

l'aide d'appel à des fonctions c'est-à-dire par attachement procédural. Les autres mécanismes d'inférences ne peuvent être utilisés.

Exemple 3.8 : Lors d'une construction de topologie, par exemple à partir de fragment de frontières ou par croisement de topologies, il est souvent nécessaire de réaliser le calcul des intersections des éléments de frontières. Parce que ces éléments sont définis la plupart du temps par des lignes polygonales repérées par leur sommet, la détermination des intersections ne peut se faire que par un calcul géométrique. Ce calcul est implanté dans la base par un attachement procédural. Si les frontières sont définies par l'ensemble des points qui les constituent, ensemble donné par extension, une inférence du type filtrage permet de déterminer les intersections. Malheureusement ce cas est trop rare.

Exemple 3.9 : Un système à base de connaissances se voit fournir la description d'un triangle sous la forme des coordonnées de ses sommets. Pour déterminer certaines caractéristiques de ce triangle (surface, pente, centre, diamètre), le système utilise des fonctions de calcul géométrique. Ces fonctions correspondent à des attachements procéduraux.

Exemple 3.10 : La localisation d'objets dans une topologie peut également exiger l'utilisation d'attachements procéduraux. Dès que l'objet est défini par ses coordonnées il est souvent nécessaire, en vue de réaliser ensuite un raisonnement même topologique, de lancer des calculs géométriques. Ainsi, dans la dernière section du chapitre précédent, les formes de raisonnement métrique ou topologique exigeaient des connaissances sur les distances ou sur les ponts entre Caen et Le Havre. En vue d'exploiter un tel raisonnement, un système à base de connaissances qui se verrait donné une définition géométrique de ces objets devrait réaliser des opérations de calcul géométrique à l'aide d'attachements procéduraux.

L'exploitation de l'attachement procédural ne se limite pas aux inférences, il peut aussi être utilisé pour des contraintes. Pour peu que celles-ci portent également sur des propriétés géométriques, l'utilisation de l'attachement procédural est nécessaire.

3.2.2. Choix de la représentation et de l'inférence

Une des caractéristiques les plus intéressantes des systèmes à base de connaissances consiste à offrir à l'utilisateur la possibilité de représenter un objet de différentes façons mais aussi d'inférer une même connaissance de différentes manières. La représentation et l'inférence de connaissances spatiales offrent, dans le cadre du raisonnement spatial, de nombreux exemples de l'intérêt d'une telle souplesse.

Il y a tout d'abord la possibilité de choisir les méthodes d'inférences ou de représentation. Le second chapitre a présenté l'intérêt qu'il y avait à distinguer différents types de connaissances spatiales en vue d'un raisonnement bien adapté. Les raisonnements de localisation en particulier exploitent des mécanismes de raisonnement qui dépendent des connaissances disponibles dans la base. Il est ainsi envisageable d'installer les différentes techniques de localisation présentées et d'utiliser la plus pertinente compte tenu des connaissances disponibles. C'est d'ailleurs l'idée à la base de la géométrie approximative dans l'article de Manola sur le projet PROBE [MANO&86] (cf. exemple 2.24).

Exemple 3.11 : Un objet spatial bi-dimensionnel, défini par un contour polygonal, peut ainsi être représenté par différentes approximations. Les deux plus simples

sont le cercle circonscrit (défini par son centre et son diamètre) et le rectangle enveloppant (par rapport à une paire d'axes). Une approximation elliptique permet d'affiner la représentation. Il est aussi possible de représenter cet objet comme ARC-INFO en donnant la liste des arcs qui constituent sa frontière ou encore la liste de ses voisins s'il constitue avec eux une topologie de l'espace. Les systèmes à base de connaissances permettent facilement de représenter le même objet de différentes façons.

Des requêtes en localisation, en inférence de propriétés utiliseront alors l'une ou l'autre des représentations. Cette approche permet d'utiliser, à chaque fois, la méthode d'inférence optimale.

Exemple 3.12: Dans SHIRKA, un point du plan peut être représenté de deux façons, en coordonnées polaires ou cartésiennes :

```
( point
  sorte-de           = objet ;
  coordonnee-x $un reel ;
  coordonnee-y $un reel ;
  angle             $un reel
                  $intervalle [0. 360.] ;
  rayon            $un reel
                  $intervalle [0. +inf])
```

L'idée sous-jacente exploitée ici est d'utiliser les systèmes à base de connaissances comme des bases de méthodes d'inférences spatiales. Cette notion de base de méthodes [ROUS88] sera développée dans le chapitre suivant relatif à ELSA pour les méthodes des environnements de résolution de problèmes. De même qu'un système à base de connaissances peut choisir la meilleure méthode de résolution d'un système linéaire en fonction de ce système, de même il peut choisir la meilleure technique d'inférence spatiale.

3.2.3. Enregistrement des résultats d'inférences spatiales

Les systèmes à base de connaissances n'enregistrent pas tous, a priori, les résultats des inférences qu'ils ont réalisées. L'intérêt qu'il y a à utiliser un système d'enregistrement dépend fortement de l'application mise en œuvre. Plus exactement cet intérêt varie en fonction des caractéristiques du graphe de la fonction de justification.

Intérêt de l'enregistrement

En particulier, dans les inférences spatiales, il arrive très souvent que les connaissances intermédiaires inférées au cours de la réponse à une requête soient aussi utilisées pour donner une réponse à une autre requête.

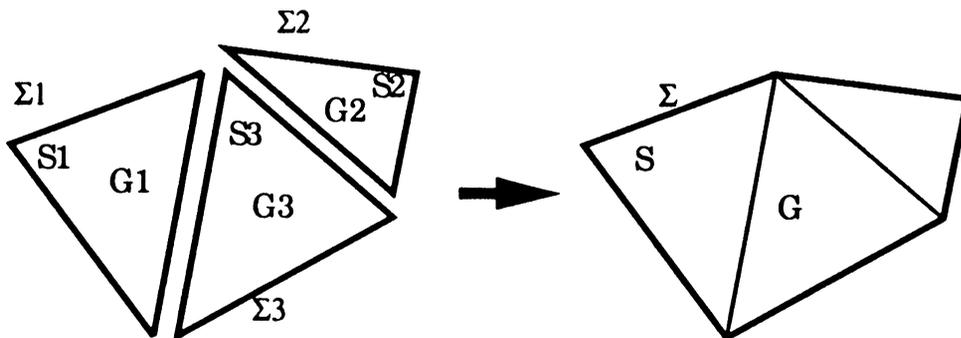


Figure 3.3 et exemple 3.13 : Un polygone est défini comme la réunion de trois triangles. Pour connaître la superficie du polygone, il faut connaître la superficie

de chaque triangle. Si celle-ci n'est pas connue, il faut la calculer. Pour connaître le centre de gravité du polygone, il faut bien sûr connaître les centres de gravité des trois triangles mais aussi, pour appliquer la formule du barycentre, connaître la superficie de chaque triangle. La requête au système à base de connaissances qui porte sur le centre utilise des connaissances qu'utilise aussi la requête qui porte sur la surface. Cet exemple est loin d'être un cas particulier.

Il arrive même qu'une connaissance intermédiaire doive être utilisée plusieurs fois au cours d'une réponse à une seule requête. Le chapitre 5 donne des exemples chiffrés de cette situation rencontrée dans le système ELSA (exemples 5.1 et 5.2).

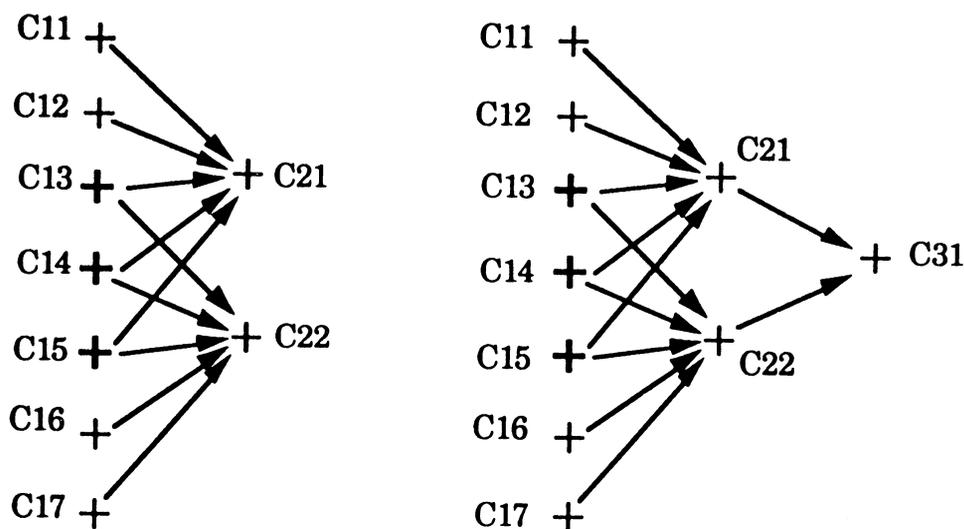


Figure 3.4 : Deux exemples de raisonnement dans un graphe de justification. Les connaissances sont toujours représentées par des croix. Sur le dessin de gauche, les connaissances en gras C13, C14 et C15 sont utilisées par les deux inférences qui conduisent à C21 et C22 (ces inférences correspondent à deux requêtes). Il s'agit d'un cas comparable à celui de la figure précédente. Si C13, C14 et C15 sont des connaissances données, c'est sans importance ; s'il s'agit de connaissances inférées, il est intéressant de les enregistrer après leur première inférence. Le dessin de droite donne un exemple où, pour l'inférence d'une seule connaissance (C31, donc suite à une seule requête), les connaissances en gras sont utilisées deux fois. Là encore si C13, C14 et C15 sont des connaissances inférées, il est très intéressant de les enregistrer après leur première inférence.

Dès lors, l'enregistrement des connaissances intermédiaires (ou "caching") devient intéressant. Pour le raisonnement spatial, il est même nécessaire si l'utilisateur souhaite éviter de réinférer des milliers de fois la même connaissance (cf. chapitre 5). Cet enregistrement n'est pas nécessaire uniquement pour des raisons quantitatives. En effet, son absence conduit à une explosion du nombre d'inférences qui *interdit* l'utilisation du système. En effet, les temps de réponse et la croissance de la base rendent impossible l'exécution normale des inférences. Le quantitatif devient qualitatif. Le système ELSA ne peut pas fonctionner sans enregistrement.

Exploitation des différences de pertinence

Le chapitre précédent a mis en évidence des différences de pertinence entre les types de connaissances spatiales. Cette hiérarchie correspond aussi bien aux critères de fréquence d'utilisation, de qualité de l'information que de

l'importance dans le raisonnement. Dans cette sous-section la fréquence d'utilisation sera le critère de pertinence le plus utilisé.

Les connaissances topologiques peuvent ainsi être considérées comme plus dignes d'intérêt que des connaissances métriques, vectorielles ou surtout euclidiennes. En d'autres termes, il est naturel d'enregistrer les connaissances les plus pertinentes car leurs nombreuses ré-inférences feraient perdre du temps. Il est regrettable en revanche d'enregistrer le résultat d'une inférence qui n'est lancée qu'une fois.

Il est alors possible, grâce à la hiérarchie établie parmi les connaissances spatiales de construire une économie cognitive. Celle-ci consiste à arbitrer entre le gain en temps et le gain en place c'est à dire à réaliser une "allocation des ressources" la plus optimale.

Les connaissances données figurent naturellement dans la base de connaissances. En effet ne pas les enregistrer après la phase de saisie reviendrait à les réclamer à l'utilisateur à chaque fois que c'est nécessaire.

En revanche, pour les connaissances spatiales inférées, il est envisageable de n'enregistrer (et naturellement de ne maintenir) que les connaissances les plus pertinentes. La pertinence la plus intéressante à exploiter ici est bien sûr celle de la fréquence. Une connaissance rarement utilisée ne doit pas encombrer la base de connaissances. A l'opposé, l'inférence d'une connaissance très souvent utilisée ne doit pas accaparer le moteur d'inférence à chaque fois qu'elle est réclamée.

Il faut préciser, pour clore cette sous-section, que le TMS présenté ci-dessus n'est pas tout à fait adapté à une prise en compte des différences de pertinence. Il faut en effet déclarer au moteur d'inférence quels sont les prédicats à enregistrer et ceux qu'il est inutile d'enregistrer. En revanche les notions de niveaux d'inférence et de modifications ne sont pas nécessaires pour exploiter ces différences de pertinence.

Enfin, l'idée proposée ici consiste à gérer l'enregistrement *a priori*, uniquement d'après la nature de la connaissance puisqu'il existe un lien entre cette nature et la fréquence d'utilisation. Il serait naturel de prendre en compte aussi le temps d'inférence de la connaissance pour la gestion de l'enregistrement. En effet, une connaissance longue à inférer "mérite" plus d'être enregistrée qu'une connaissance dont l'inférence est peu coûteuse en temps. Malheureusement, une telle gestion ne peut se faire qu'*a posteriori*. Elle est donc moins intéressante que celle basée sur la pertinence.

3.3. MAINTIEN DE LA VERITE

L'enregistrement des connaissances intermédiaires exige un module de maintien de la vérité. Si ce module est absent, des incohérences risquent d'apparaître dans la base de connaissances.

L'expression anglaise "Truth Maintenance System" est retenue ici sous la forme de son acronyme "TMS" inventé par Doyle [DOYL79]. En résumé, un système de maintien de la vérité s'assure (en les supprimant éventuellement) que les connaissances inférées par un système de raisonnement restent vraies après des modifications faites sur la base (cf. figure 3.5).

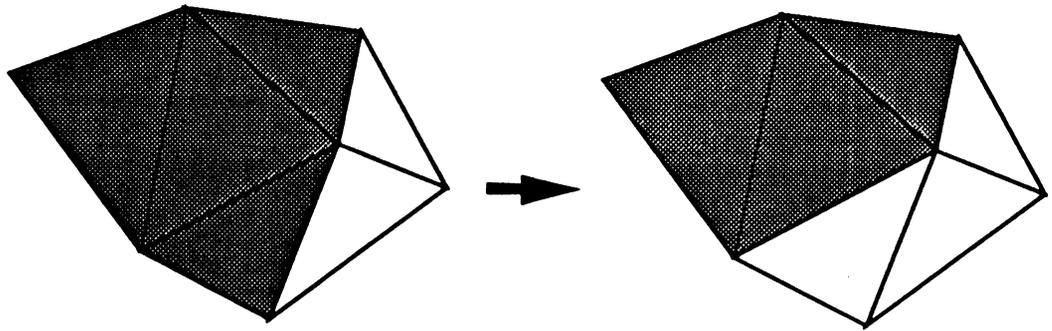


Figure 3.5 : Deux polygones ont été définis par les triangles qui les composent. Des inférences comparables à celles de l'exemple ont été lancées. Les résultats ont été enregistrés. Si un polygone se voit prélevé un triangle au profit de l'autre, les inférences déjà réalisées doivent être invalidées. En fait, seules les inférences relatives aux polygones sont à invalider. Celles qui concernaient les triangles restent valables.

Cette sous-section est, dans une très large mesure, inspirée des travaux réalisés par Euzenat sur les systèmes de maintien de la vérité dans les "systèmes de raisonnement". L'auteur de ces lignes est entièrement responsable toutefois du "formalisme" succinct proposé ici qui offre un cadre aux considérations sur la localité qui ferment ce chapitre 3. Précisons enfin que ce formalisme ne peut que constituer une base de départ pour le développement d'un système offrant ces caractéristiques de localités.

3.3.1. Un "formalisme"

Le formalisme présenté ici est celui de la logique du premier ordre, plus particulièrement restreinte à la manipulation des seules clauses de Horn. C'est le modèle de PROLOG dont il sera fait usage à plusieurs reprises. Ce formalisme est nécessaire car il permet d'illustrer des propriétés du raisonnement spatial. De plus, le TMS qu'il définit possède des propriétés que d'autres n'ont pas, même si le maintien sur les seules valeurs - seul retenu ici, comme expliqué plus loin - est très restrictif et très pauvre.

Dans ce formalisme, toute unité de connaissance est représentée par une clause. Une clause sans second membre représente une connaissance factuelle ; avec un second membre elle correspond à une connaissance d'inférence ou règle. Seul est étudié ici le problème de la maintenance sur les connaissances factuelles. Les connaissances d'inférence ne sont pas dynamiques. Cette importante hypothèse serait trop restrictive dans une approche globale des systèmes à base de connaissances. Néanmoins, dans le cadre du raisonnement spatial, elle ne présente pas d'inconvénients majeurs car les aspects dynamiques ne concernent que les faits.

Par ailleurs, le formalisme présenté ici est incomplet car il ne traite pas la négation. Celle-ci présente un intérêt certain dans le cadre du raisonnement spatial mais son traitement est délicat et prendrait trop de place ici.

Enfin la dernière hypothèse consiste à poser qu'il n'y a pas de cycle dans le réseau de dépendance, c'est-à-dire qu'aucune connaissance ne dépend d'une autre connaissance dont elle peut permettre l'inférence.

Une fois les règles écrites dans la base, un certain nombre de clauses factuelles sont données au système c'est-à-dire introduites dans la base. L'utilisateur émet alors un certain nombre de requêtes sous la forme d'un prédicat partiellement (ou totalement instancié) :

$\text{predp}(_b1, _b2, \dots, _bp)$ où les $(_bj)_{jp}$ sont des constantes ou des variables.

Pour répondre à une requête de ce type, l'interprète va générer d'autres requêtes auxquelles il tentera de répondre. A chaque instant de la durée d'exploitation de la base de connaissances, il existe donc un certain nombre de requêtes : certaines sont des requêtes utilisateur, d'autres des requêtes générées par l'interprète. L'idée de base est de maintenir dans la base l'ensemble complet des réponses aux requêtes déjà formulées (qu'il s'agisse de requêtes utilisateur ou de requêtes générées par l'interprète).

Le formalisme du TMS proposé consiste, pour chacune de ces requêtes, à enregistrer les éléments suivants (attention ! "enregistrer" ne signifie pas ici que ces enregistrements figurent dans la base de connaissances, mais plutôt dans une autre base ou dans plusieurs autres bases) :

- (1) la requête elle-même $\text{predp}(_b1, _b2, \dots, _bp)$;
- (2) les liens $[\text{predp}(b'1, b'2, \dots, b'p), \text{pred}(a1, a2, \dots, an), r]$ où $\text{predp}(b'1, b'2, \dots, b'p)$ est une clause factuelle totalement instanciée qui a été directement inférée en utilisant la clause $\text{pred}(a1, a2, \dots, an)$ dans la cadre de l'application de la règle r pour la réponse à la requête $\text{predp}(_b1, _b2, \dots, _bp)$. Les $(ai)_{in}$ et les $(b'j)_{jp}$ sont des constantes mais les $(_bj)_{jp}$ sont des variables ou des constantes auxquelles doivent s'unifier les $(b'j)_{jp}$.
- (3) les liens $[\text{predp}(_b1, _b2, \dots, _bp), \text{pred}(_a1, _a2, \dots, _an), r]$ où $\text{predp}(_b1, _b2, \dots, _bp)$ est une requête pour la résolution de laquelle une unification a été réalisée au niveau du prédicat $\text{pred}(_a1, _a2, \dots, _an)$ dans le cadre de l'application de la règle r . Les $(_ai)_{in}$ sont des constantes ou des variables ; et les $(_bj)_{jp}$ sont des constantes ou des variables.

Fonctionnement

A chaque requête de l'utilisateur ou de l'interprète, ce dernier consulte la liste des enregistrements du type (1). Si la requête s'y trouve, ou bien si elle s'unifie à une requête déjà existante, il se contente de lire, dans la base de connaissances, les clauses factuelles qui peuvent s'unifier à la requête sans jamais rien résoudre. Si la requête ne s'y trouve pas et si elle ne s'unifie pas à une requête déjà existante, il doit relancer la résolution de manière classique. De plus, dans ce cas, si une requête de la base (1) s'unifie à cette nouvelle requête, il convient de ne pas répéter artificiellement les réponses correspondant à cette ancienne requête. Pour cela, les enregistrements de type (2) sont utilisés conjointement à un indexage des clauses présentes dans la base (une répétition de clause peut être licite).

A chaque suppression de clause totalement instanciée telle que $\text{pred}(a1, a2, \dots, an)$, les enregistrements du type (2) sont utilisés pour invalider les clauses telles que $\text{predp}(b'1, b'2, \dots, b'p)$. Il faut alors ensuite invalider les clauses qui dépendaient de ces dernières à travers d'autres enregistrements de type (2).

A chaque ajout de clause totalement instanciée telle que $\text{pred}(a1, a2, \dots, an)$, les enregistrements du type (3) comme $[\text{predp}(_b1, _b2, \dots, _bp), \text{pred}(_a1, _a2, \dots, _an), r]$ indiquent, si $(a1, a2, \dots, an)$ peut s'unifier avec $(_a1, _a2, \dots, _an)$, que les réponses à la

requête `predp(_b1, _b2, ..., _bp)` ne sont plus suffisantes et qu'il faut relancer l'inférence pour cette requête à l'aide de la règle *r*. Là encore, il convient de ne pas répéter artificiellement les réponses qui avaient déjà été produites. Pour cela, la solution qui consiste à exploiter les enregistrements (2) peut être utilisée. Par ailleurs, les clauses éventuellement ajoutées donneront, à leur tour, lieu à un traitement comparable à celui qui vient d'être évoqué.

L'ajout ou la suppression de clauses sont ainsi suivi de phases de propagation d'invalidation ou d'inférence.

Exemple

Soient les clauses

```
(r1) pred(_x, _y, _z) :-
    pred1(_x, _y),
    pred2(_y, _z),
    pred3(_x, _y, _t),
    pred4(_t, _z).
(r2) pred(_x, _y, _z) :-
    pred1(_x, _y),
    pred2(_y, _z),
    pred5(_z).
(r3) pred(_x, _y, _z) :-
    pred1(_x, _y),
    pred2(_y, _z).
```

La base de faits contient initialement les clauses factuelles suivantes (connaissances données) :

```
pred1(x0, y0)
pred2(y0, z10)
pred2(y0, z20)
pred3(x0, y0, t10)
pred3(x0, y0, t21)
pred3(x0, y0, t22)
pred4(t10, z10)
pred4(t21, z20)
pred4(t22, z20)
```

En réponse à la requête `pred(x0, y0, _z)`, le moteur d'inférence a rendu

```
_z = z10 (avec _t = T10)
_z = z20 (avec _t = T21)
_z = z20 (avec _t = T22)
_z = z10
_z = z20
```

La base de connaissances est alors enrichie des clauses :

```
pred(x0, y0, z10)1
pred(x0, y0, z20)1
pred(x0, y0, z20)2
pred(x0, y0, z10)2
pred(x0, y0, z20)3
```

Certaines clauses sont identiques mais le moteur d'inférence est capable de les distinguer. Comme il l'a été précisé plus haut, elles sont indexées. Il faut donc stocker :

- (1a) pred(X0, Y0, _z)

- (2a) [pred(X0, Y0, Z10)1, pred1(X0, Y0), r1]
- (2b) [pred(X0, Y0, Z10)1, pred2(Y0, Z10), r1]
- (2c) [pred(X0, Y0, Z10)1, pred3(X0, Y0, T10), r1]
- (2d) [pred(X0, Y0, Z10)1, pred4(T10, Z10), r1]
- (2e) [pred(X0, Y0, Z20)1, pred1(X0, Y0), r1]
- (2f) [pred(X0, Y0, Z20)1, pred2(Y0, Z20), r1]
- (2g) [pred(X0, Y0, Z20)1, pred3(X0, Y0, T21), r1]
- (2h) [pred(X0, Y0, Z20)1, pred4(T21, Z20), r1]
- (2i) [pred(X0, Y0, Z20)2, pred1(X0, Y0), r1]
- (2j) [pred(X0, Y0, Z20)2, pred2(Y0, Z20), r1]
- (2k) [pred(X0, Y0, Z20)2, pred3(X0, Y0, T22), r1]
- (2l) [pred(X0, Y0, Z20)2, pred4(T22, Z20), r1]
- (2m) [pred(X0, Y0, Z10)2, pred1(X0, Y0), r3]
- (2n) [pred(X0, Y0, Z10)2, pred2(Y0, Z10), r3]
- (2o) [pred(X0, Y0, Z20)3, pred1(X0, Y0), r3]
- (2p) [pred(X0, Y0, Z20)3, pred2(Y0, Z20), r3]

- (3a) [pred(X0, Y0, _z), pred1(X0, Y0), r1]
- (3b) [pred(X0, Y0, _z), pred2(Y0, _z), r1]
- (3c) [pred(X0, Y0, _z), pred3(X0, Y0, _t), r1]
- (3d) [pred(X0, Y0, _z), pred4(T10, _z), r1]
- (3e) [pred(X0, Y0, _z), pred1(X0, Y0), r2]
- (3f) [pred(X0, Y0, _z), pred2(Y0, _z), r2]
- (3g) [pred(X0, Y0, _z), pred5(_z), r2]
- (3h) [pred(X0, Y0, _z), pred1(X0, Y0), r3]
- (3i) [pred(X0, Y0, _z), pred2(Y0, _z), r3]

Un sous-produit de ces enregistrements est naturellement le graphe de la fonction de justification qui n'en est qu'une représentation graphique partielle.

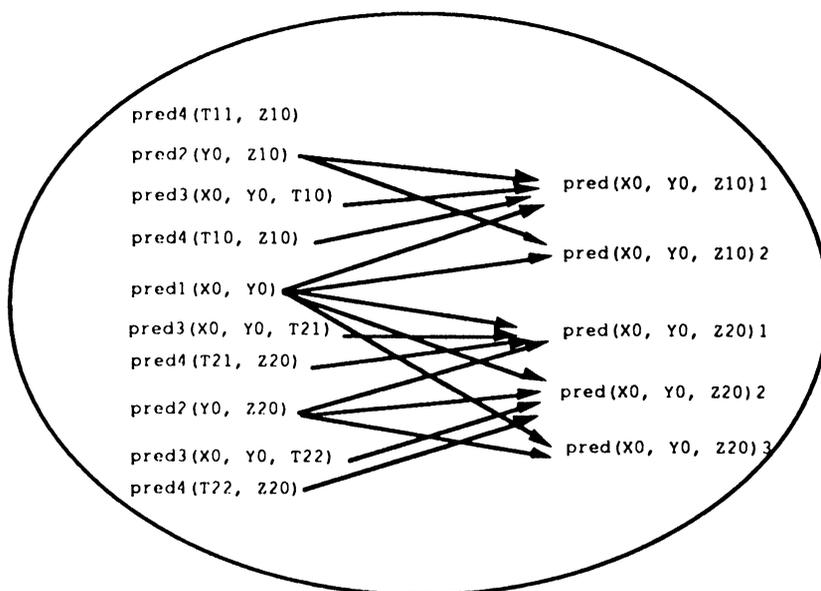


Figure 3.6 : La fonction de justification pour les inférences qui viennent d'être présentées.

Si la requête suivante est émise :

pred(x0, y0, z10),

le TMS repère qu'il s'agit d'un cas particulier de la précédente ; il consulte alors la base de connaissances et répond deux fois vrai.

Si la requête est en revanche :

pred(x0, _y, _z),

le TMS ne trouve pas de requête semblable déjà émise, mais, au contraire remarque que la première requête s'unifie à cette dernière. Il lance alors l'inférence et obtient des réponses qui sont celles obtenues suite à la première requête. La présence des enregistrements (2) lui indique qu'il ne doit pas enregistrer ces réponses dans la base de connaissances ni répéter ces enregistrements de type (2) mais remplacer les enregistrements de type (1) et (3) par les enregistrements suivants :

(1a)	pred(x0, _y, _z)
(3a)	[pred(x0, _y, _z), pred1(x0, _y), r1]
(3b)	[pred(x0, _y, _z), pred2(_y, _z), r1]
(3c)	[pred(x0, _y, _z), pred3(x0, _y, _t), r1]
(3d)	[pred(x0, _y, _z), pred4(T10, _z), r1]
(3e)	[pred(x0, _y, _z), pred1(x0, _y), r2]
(3f)	[pred(x0, _y, _z), pred2(_y, _z), r2]
(3g)	[pred(x0, _y, _z), pred5(_z), r2]
(3h)	[pred(x0, _y, _z), pred1(x0, _y), r3]
(3i)	[pred(x0, _y, _z), pred2(_y, _z), r3]

Le TMS peut naturellement être utilisé pour prendre en compte des suppressions de connaissances :

- Si le prédicat $\text{pred1}(x0, y0)$ vient à être supprimé, le TMS doit détruire les prédicats suivants $\text{pred}(x0, y0, z10)_1$, $\text{pred}(x0, y0, z10)_2$, $\text{pred}(x0, y0, z20)_1$, $\text{pred}(x0, y0, z20)_2$ et $\text{pred}(x0, y0, z20)_3$ à cause des enregistrements (2a), (2e), (2i), (2m) et (2o).

- De même, si le prédicat $\text{pred2}(y0, z10)$ (respectivement $\text{pred2}(y0, z20)$) est supprimé le TMS doit supprimer les prédicats $\text{pred}(x0, y0, z10)_1$ et $\text{pred}(x0, y0, z10)_2$ (respectivement les prédicats $\text{pred}(x0, y0, z20)_1$, $\text{pred}(x0, y0, z20)_2$ et $\text{pred}(x0, y0, z20)_3$) à cause des enregistrements (2b) et (2n) (respectivement des enregistrements (2f), (2j) et (2p)).

- Si le prédicat $\text{pred3}(x0, y0, T22)$ est supprimé, le TMS doit supprimer le prédicat $\text{pred}(x0, y0, z20)_2$ à cause de l'enregistrement (2k).

- Si le prédicat $\text{pred4}(T21, z20)$ est supprimé, le TMS doit supprimer le prédicat $\text{pred}(x0, y0, z20)_1$ à cause de l'enregistrement (2h).

- Si le prédicat $\text{pred4}(T11, z10)$ est supprimé, le TMS ne doit rien supprimer.

Le TMS réagit également à l'ajout de prédicats. Seuls sont utilisés dans ce cas les enregistrements du troisième type.

Si le prédicat $\text{pred2}(Y0, Z30)$ est rajouté dans la base de connaissances, les enregistrements (3b), (3f) et (3i) indiquent au moteur d'inférence qu'il ne doit plus se contenter des prédicats $\text{pred}(X0, Y0, Z10)_1$, $\text{pred}(X0, Y0, Z10)_2$, $\text{pred}(X0, Y0, Z20)_1$, $\text{pred}(X0, Y0, Z20)_2$ et $\text{pred}(X0, Y0, Z20)_3$, comme réponses à la requête $\text{pred}(X0, Y0, _z)$ mais qu'il doit relancer l'inférence correspondante, en tenant compte, cette fois-ci, de l'existence de $\text{pred2}(Y0, Z30)$. La règle $r3$ donnera une réponse supplémentaire : $\text{pred}(X0, Y0, Z30)$. Comme aucun prédicat tel que $\text{pred3}(X0, Y0, T30)$ n'est présent dans la base, l'inférence par la règle $r1$ conduira au même résultat. Il en va de même pour la règle $r2$ puisqu'aucun prédicat tel que $\text{pred5}(Z30)$ n'est présent dans la base.

Un TMS peut être implanté sur différents types de systèmes de représentation de la connaissance. Il peut être de différents niveaux. Dans [EUZE87], Euzenat présente l'implantation de 3 niveaux de TMS sur un système de représentation de connaissances par objets : SHIRKA. L'aspect le plus étudié ici est celui de la maintenance sur les valeurs.

De plus, le TMS mentionné ici est en fait couplé de façon intime au moteur d'inférence ce qui l'éloigne des TMS de la littérature, généralement très distincts du moteur d'inférence.

Bien que le modèle proposé ici soit très proche de PROLOG, peu d'implantations de TMS sur PROLOG existent actuellement. L'une d'entre elles, décrite dans [APT&87] par Apt et Pugin, réalise les spécifications exposées ci-dessus. Les hypothèses restrictives retenues ici sont d'ailleurs inutiles car Apt et Pugin manipulent des cycles à partir du moment où ils ne contiennent pas de négations. De plus, Apt et Pugin donnent un cadre théorique à leurs proposition celui des *bases de données stratifiées*.

En revanche, l'implantation présentée dans [FINI88] n'introduit le TMS que par une perversion de PROLOG, c'est-à-dire par l'introduction du chaînage avant. Cette solution est révélatrice d'une idée très répandue qui est qu'un TMS n'a de sens qu'en chaînage avant. Le point de vue retenu ici et illustré par le modèle décrit ci-dessus est celui du chaînage arrière avec enregistrement des résultats. Il semble que la perversion de PROLOG soit ainsi beaucoup plus faible.

Il faut quand même ajouter que si l'utilisation de l'assert ne pose pas de problème particulier au TMS, elle constitue elle aussi un écart important par rapport au calcul des prédicats.

3.3.2. Modification \neq suppression + ajout

Le TMS proposé par Euzenat propage les invalidités sans relancer le calcul des nouvelles valeurs. Seules existent les notions de suppression de valeurs et d'ajout de valeurs. C'est aussi ce que réalise le TMS présenté ci-dessus, qui, en plus, puisqu'il est lié au moteur d'inférence, relance les inférences.

L'idée proposée ici est d'introduire les notions de *modification* de valeurs et de *niveau d'inférence*.

Une modification est un couple $(\text{pred}(X_i), \text{pred}(X_j))$. Elle est déclarée au moteur d'inférence qui va réagir en exploitant la notion de niveau d'inférence.

Un interprète PROLOG travaille à différents niveaux. Pour établir un prédicat (la requête, de niveau 0) il cherche à établir des prédicats (de niveau 1) et pour cela il tente d'établir la validité d'autres prédicats (de niveau 2), etc.

Les enregistrements décrits dans le formalisme précédent permettent de définir la notion de juxtaposition dans l'inférence. Par exemple, l'enregistrement (2a) indique que le prédicat $\text{pred}(x_0, y_0, z_{10})$ dépend de $\text{pred}_1(x_0, y_0)$ et qu'il lui est connexe. De même, l'enregistrement (2b) indique que le prédicat $\text{pred}(x_0, y_0, z_{10})$ dépend de $\text{pred}_2(y_0, z_{10})$ et qu'il lui est connexe.

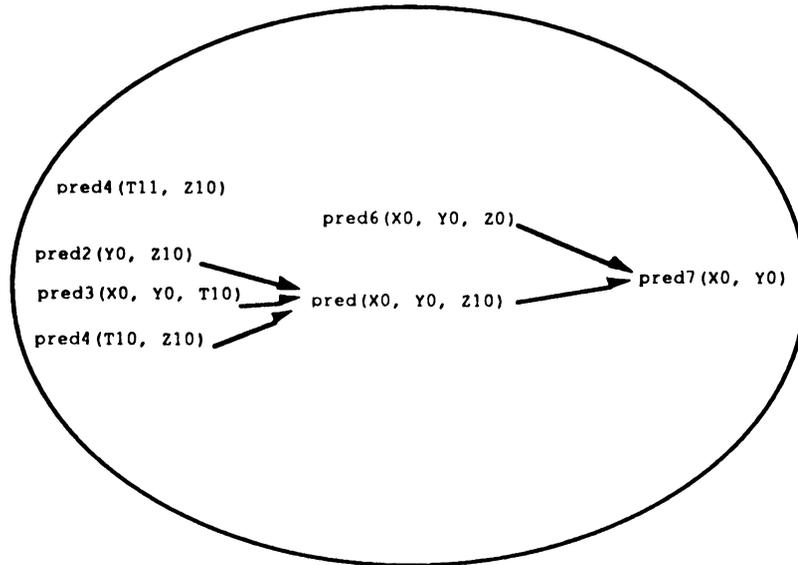


Figure 3.7 : Le graphe de la fonction de dépendance avec deux niveaux d'inférence.

Une nouvelle règle est installée dans la base de connaissance :

```
pred7(_x, _y) :-
    pred1(_x, _y, _z),
    pred6(_x, _y, _z).
```

Pour inférer $\text{pred}_7(x_0, y_0)$ de niveau 0, le moteur d'inférence utilise $\text{pred}_6(x_0, y_0, z_0)$ et établit $\text{pred}(x_0, y_0, z_{10})$ qui sont de niveau 1. Pour établir $\text{pred}(x_0, y_0, z_{10})$, il utilise $\text{pred}_2(y_0, z_{10})$, $\text{pred}_3(x_0, y_0, t_{10})$ et $\text{pred}_4(t_{10}, z_{10})$ qui sont de niveau 2.

Une modification est définie, de l'extérieur, comme la suppression d'une clause instanciée - et donnée - et l'ajout d'un prédicat de même identifiant mais avec d'autres constantes pour arguments. Mais, à la différence d'un TMS classique, le TMS dit "à modification" fait la distinction entre la suppression-ajout et la modification. Pour cela, il propage les invalidités au niveau d'inférence immédiatement inférieur puis s'arrête. Il relance ensuite les inférences liées aux ajouts en restant au même niveau puis s'arrête. Il fait alors le bilan des prédicats disparus et des prédicats apparus et ne passe au second niveau que pour les prédicats disparus et non remplacés par un prédicat identique ou pour les prédicats entièrement nouveaux.

Si la modification est [$\text{pred}_3(x_0, y_0, t_{10})$, $\text{pred}_3(x_0, y_0, t_{11})$], le moteur d'inférence invalide $\text{pred}(x_0, y_0, z_{10})$ puis relance l'inférence de ce prédicat avec la base suivante :

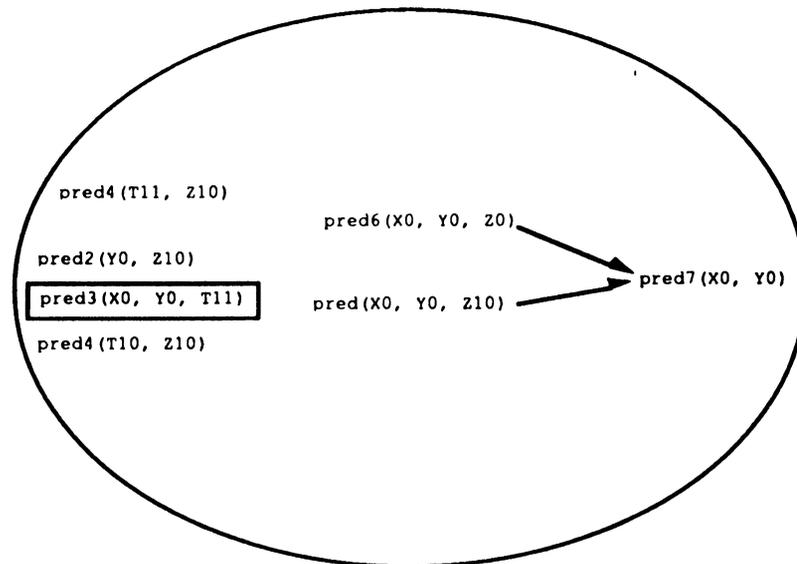


Figure 3.8: Une modification apparaît dans le graphe de dépendances.

Il parvient de nouveau à établir $\text{pred}(x_0, y_0, z_{10})$ en utilisant d'autres prédicats. Le niveau 1 étant inchangé, le moteur d'inférence stoppe la propagation des invalidités ; la base reste cohérente. La propagation de l'invalidité reste localisée dans le graphe de dépendances aux connaissances directement concernées.

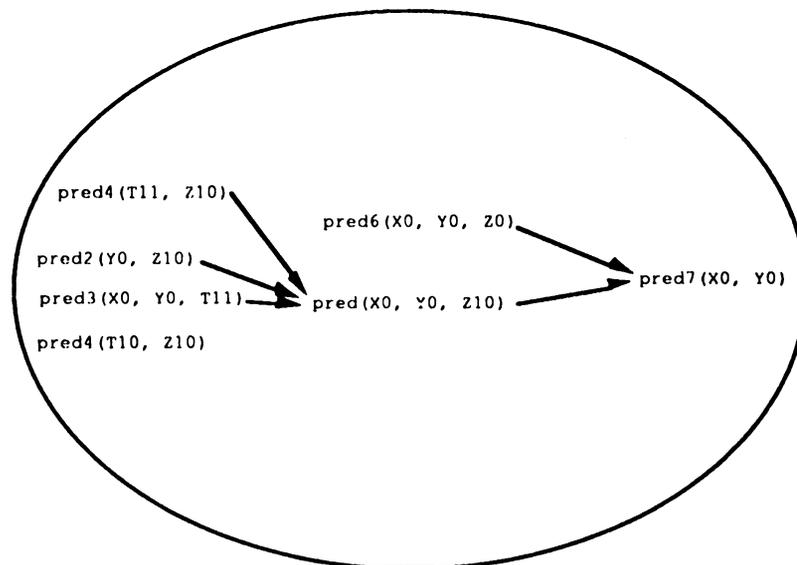


Figure 3.9: La propagation de la modification est restée très locale.

Alors que les systèmes de maintien de la vérité contiennent une boucle simple de propagation, le TMS à modification présenté ici suit la règle : propager-reinférer-comparer. Le TMS classique invalide toutes les connaissances qui se trouvent à l'aval de la connaissance supprimée ou ajoutée ; le TMS à modification n'invalide que les connaissances concernées effectivement par la modification.

Le TMS proposé par Apt et Pugin [APT&87] offre le même type de capacité à réagir aux modifications. Les niveaux d'inférence présentés ici de façon

intuitive sont pris en compte dans leur théorie des programmes logiques stratifiés. Apt et Pugin proposent même une notion plus générale que la modification : la *transaction* capable de traiter un ensemble d'ajouts et de suppressions.

3.3.3. Exploitation de la localité

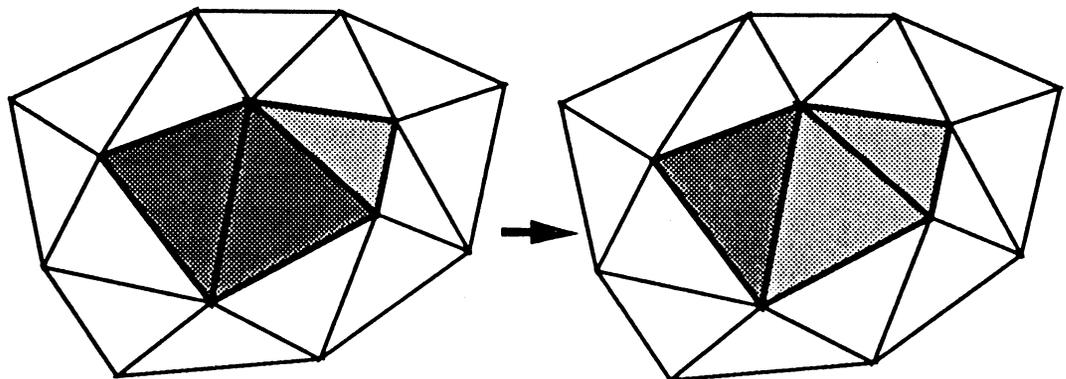
La localité est une notion éminemment spatiale qui fait référence aux notions de voisinage (notion topologique) et de proximité (notion métrique). Lors de raisonnements spatiaux relatifs à une partie bien définie de l'espace, il semble naturel de ne pas tenir compte du reste de l'espace.

Il est possible, avec un système à base de connaissances doté d'un TMS classique d'exploiter cette localité. La propriété utilisée ici est la suivante.

Il existe un parallélisme entre la localité dans l'espace et la localité dans le graphe de dépendances d'un système de maintien de la vérité.

Localité et TMS classique

Cette localité dans le graphe de dépendances s'exprime tout d'abord par des relations de justifiants à justifié ou plus exactement par la clôture transitive de ces relations.



Exemple 3.14 et figure 3.10 : Au sein d'une triangulation de l'espace, deux polygones ont été définis par la donnée des triangles qu'ils contiennent. Les inférences de l'exemple ont été réalisées sur ces deux polygones. Si, comme dans l'exemple, l'un de ces polygones perd un triangle au profit de l'autre le TMS doit invalider les inférences enregistrées qui portaient sur ces deux polygones. Les inférences qui ont pu être réalisées sur des polygones définis sur d'autres triangles ne seront pas, elles, invalidées. L'invalidation reste donc locale.

Localité et TMS à modifications

Mais il est encore plus intéressant d'exploiter ces notions de localité lors d'une modification. Il faut pour cela que le TMS soit tel qu'il a été présenté en 3.4.2. Ce type de TMS permet de rendre compte de l'affaiblissement spatial des effets d'une modification comparable à l'affaiblissement d'un signal qui se propage dans l'espace.

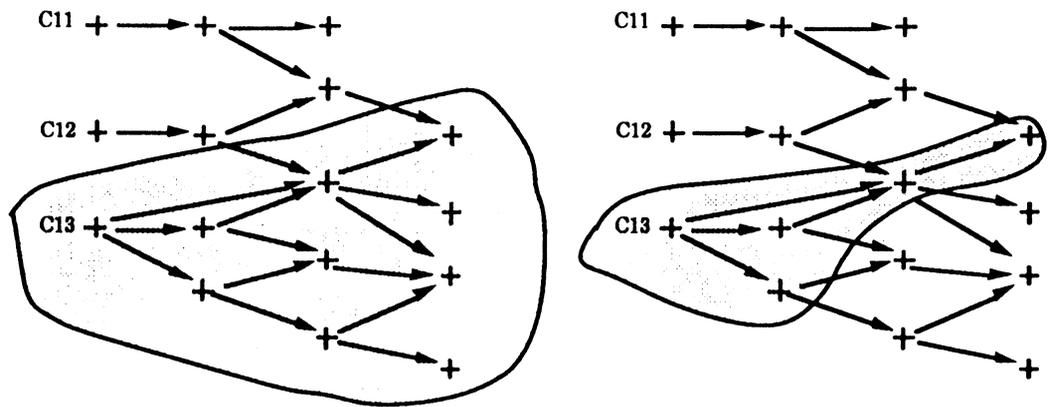


Figure 3.11 : Pour un même ensemble d'inférences, deux TMS invalident des ensembles de connaissances différents. Sur la figure de gauche, le TMS est classique. Les connaissances qui ne sont pas invalidées ne dépendent en aucune façon de C13 (par exemple les connaissances des autres polygones ou celles de tous les triangles dans l'exemple). En revanche, tout le "cône d'influence" de C13 doit être invalidé, c'est à dire toutes les connaissances "à l'aval" de C13. Sur la figure de droite, la cohérence de la base est garantie par un TMS "à modifications". Celui-ci fonctionne par niveaux et ne propage la modification que sur les connaissances qui sont effectivement remises en cause par la modification (et non pas celles qui sont susceptibles de l'être, comme c'est le cas dans le TMS classique). Celles qui dépendaient de la connaissance modifiée mais qui, avec la nouvelle connaissance, peuvent être réinférées telles quelles ne sont pas invalidées. Le nombre de connaissances invalidées est ainsi beaucoup plus faible. Le "cône d'influence" est ainsi limité dans son extension.

Davis [DAVI88] estime devoir éliminer les approches de maintenance de la vérité basées sur les graphes de dépendances pour les cartes cognitives associées à des "large-scale spaces". Il pense que les inférences spatiales dépendent, la plupart du temps, de trop d'informations pour que la complexité du graphe de dépendances le rende utilisable. Cette critique est fondée. C'est précisément pour éviter la remise en cause de toute une partie du graphe que la notion de "modification" a été introduite. De plus il faut préciser que la complexité du graphe, c'est-à-dire ici l'ouverture des "cônes d'influence", dépend grandement de la modélisation du raisonnement qui a été retenue.

Les inférences qui autorisent une telle exploitation sont celles qui permettent à une connaissance intervenant dans cette inférence d'être modifiée sans que le résultat de l'inférence ne soit changé. Les inférences géométriques qui utilisent des connaissances numériques en fournissent de bons exemples. Il n'est pas sûr que cette capacité puisse être exploitée avec des raisonnements plus symboliques.

Ces capacités peuvent être exploitées dans les trois grands types de raisonnement spatial présentés dans le chapitre précédent : la construction de topologie, l'inférence de propriétés spatiales et la localisation d'objets.

Dynamique des constructions spatiales

Dans le chapitre précédent, une forme importante de raisonnement spatial a été mise en évidence : celle qui consiste à construire une topologie à partir de connaissances spatiales données de nature différente (fragments de frontières, croisement de topologies...).

Les topologies ainsi créées dépendent naturellement de ces connaissances données. Si celles-ci viennent à être modifiées, les topologies inférées doivent être remises en cause. L'utilisation d'un TMS classique permet déjà de n'invalider que les connaissances qui dépendent des connaissances modifiées. Malheureusement, en matière de création de topologie, les "cônes d'influence" du graphe de la fonction de justification sont parfois très ouverts.

Exemple 3.15 : Lors d'une création de topologie du type croisement de topologies, il est nécessaire de repérer les intersections de frontières. Ces intersections définissent ensuite des tronçons de frontières qui vont être utilisés pour la suite de la création de topologie. La modification d'une frontière peut conduire à un bouleversement complet de ces intersections et, partant, à une modification radicale de la topologie. Le graphe de justification possède donc bien des cônes d'influence particulièrement ouverts.

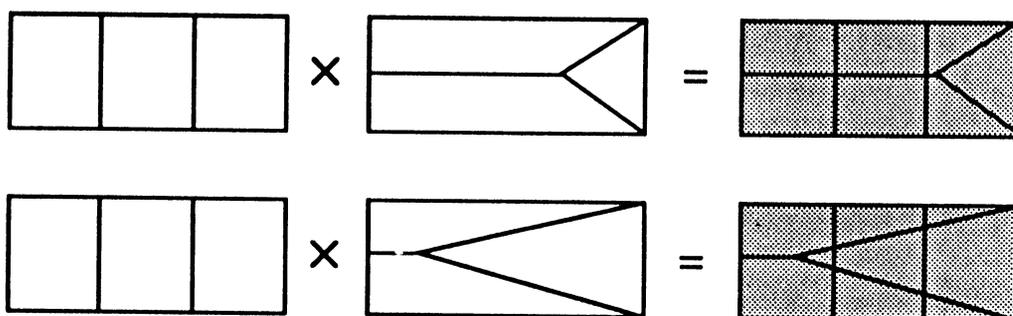


Figure 3.12 : Deux créations de topologies par croisement dans deux cas différents. La simple modification d'une frontière peut concerner l'ensemble de la topologie. En effet, les intersections de frontières sont complètement différentes dans les deux situations. Les cônes d'influence sont donc très ouverts. Le TMS classique doit donc invalider de nombreuses connaissances à chaque modification. L'ensemble du rectangle de travail est concerné (il est en grisé).

Un TMS à modifications permet de passer outre ce problème. Il faut pour cela, cependant, que le raisonnement implanté dans le système à base de connaissances soit bien décomposé. Cette décomposition doit surtout présenter la particularité de permettre au système à base de connaissances d'exploiter au mieux ses capacités de maintien de la vérité et en particulier la notion de localité.

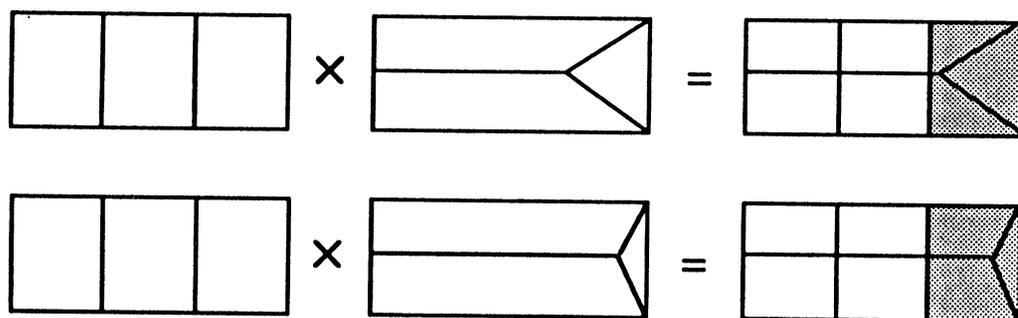


Figure 3.13 : Deux créations de topologies par croisement dans deux cas différents. Le TMS à modification se voit annoncer une modification entre les deux cas. Il commence à propager les invalidations. Comme il constate que les intersections de frontières ne sont pas modifiées dans les deux tiers gauches du rectangle de travail il n'y propage pas de modifications. Seules les parties de droite en grisé sont concernées. Il va de soi que ce gain sur l'ensemble des connaissances à invalider

est bien plus grand dans le cas d'une topologie plus étendue (l'exemple donne trois fermés concernés sur sept ; le rapport peut être beaucoup plus favorable).

En résumé, le graphe de justification d'un raisonnement de création de topologie possède des cônes d'influence très ouverts c'est-à-dire que les connaissances spatiales ont "des effets" sur les connaissances inférées même à distance. En revanche, ces effets s'atténuent très rapidement dans le graphe de dépendances. Un TMS à modification est donc particulièrement intéressant. Les chapitres 5 et 6 donneront des exemples d'application où ce gain de temps est précieux.

Dynamique des passages d'un type de connaissances à un autre

Ici les objets spatiaux sur lesquels les propriétés spatiales sont inférées sont supposés donnés dans *une topologie qui n'est pas modifiée*. Les cônes d'influence sont plus refermés. Les connaissances spatiales ont une influence à distance plus limitée. Le TMS à modifications n'est pas aussi important.

Exemple 3.16 : Les exemples 3.12 et 3.13 relatifs aux inférences sur les triangles traitent tous de ce type d'inférence. La topologie de base (celle des triangles) n'y est pas modifiée. Le TMS classique permet déjà d'exploiter la localité.

CONCLUSION

Ce troisième chapitre a permis, à l'occasion d'un retour sur les systèmes à base de connaissances d'étudier différents aspects de la dynamique des bases de connaissances spatiales.

De telles bases sont le lieu de nombreuses inférences spatiales conduites à partir de connaissances données. Les systèmes à base de connaissances permettent une utilisation importante du calcul géométrique à travers l'attachement procédural. La complexité de certaines de ces inférences, leur grand nombre et la nature du *graphe de dépendances* exige un enregistrement des résultats de ces inférences.

Cet enregistrement ainsi que l'imbrication possible des phases d'introduction de connaissances données et des phases de raisonnement rendent nécessaire l'utilisation d'un TMS. Le raisonnement spatial offre de ce point de vue un bon terrain d'application des TMS car il existe un parallélisme entre la localité spatiale et la localité dans le graphe de dépendances. En particulier, les TMS manipulant la notion de modification permettent d'exploiter des propriétés d'atténuation spatiale des influences dans ce graphe. Une telle capacité permet de diminuer de façon très importante les temps de réaction suite à des modifications des connaissances données. Cette propriété est essentielle dans des systèmes interactifs.

Les considérations qui viennent d'être exposées dans les chapitres 2 et 3 trouvent leur origine dans les réflexions conduites à propos d'une application particulière : le développement du système ELSA. Le chapitre 4 le décrit en détail dans son contexte et sa méthodologie. Le chapitre 5 présentera plus en détail le traitement du raisonnement spatial qui y est réalisé. Le chapitre 6 traitera d'autres applications du raisonnement spatial dans les systèmes à base de connaissances.

4. ELSA ET L'ANALYSE DE SITES AVALANCHEUX

Ce chapitre contient une description du système ELSA (Etude et Limites de Sites Avalancheux) à travers le contexte d'application et la méthodologie de son développement. Il n'explicite pas, en revanche, le traitement du raisonnement spatial.

Il constitue également une illustration de ce que peut être le développement d'un environnement de résolution de problèmes dédié à une application particulière, en l'occurrence l'analyse de sites avalanches. Cette approche peut concerner d'autres risques naturels ou d'autres domaines d'application. Un système identique est ainsi actuellement en développement pour la lutte contre les congères : le système CLER (Congères : Localisation et Evaluation du Risque) [GOAR88] [MOUT90]. L'auteur de ces lignes est également le responsable de ce développement.

Bien que l'analyse de sites avalanches ne constitue pas le thème de cette thèse, il est intéressant de commencer par une description détaillée du contexte dans lequel ELSA a été développé. Ensuite, après un court aperçu sur les systèmes à base de connaissances liés aux risques naturels et sur les environnements de résolution de problèmes, les objectifs assignés au système ELSA sont présentés. La modélisation du raisonnement spatial qui a été retenue sera présentée dans le chapitre suivant en insistant sur les inférences de propriétés spatiales et sur le problème de la construction d'une représentation de terrain.

4.1. ANALYSE DE SITES AVALANCHEUX

L'objectif est ici de présenter le contexte dans lequel ELSA a été développé. Après une présentation succincte des avalanches de neige, la problématique de l'analyse de sites avalanches est exposée ainsi qu'une méthodologie d'analyse développée au cours du présent travail. Les outils de cette analyse, et en particulier les techniques de modélisation, sont ensuite décrits. Enfin, les évolutions récentes et les besoins qu'elles font apparaître sont présentés.

4.1.1. Avalanches

Une avalanche est un phénomène naturel qui consiste en l'écoulement d'une masse de neige à une vitesse significative (supérieure à 1 km/h) sur une dénivelée qui dépasse la centaine de mètres. Par sa nature même, une avalanche peut exercer des efforts mécaniques sur des biens ou des personnes qui se trouvent dans l'écoulement. Ces efforts éventuels constituent une menace. Là est le *risque* d'avalanche.

Ce risque existe depuis toujours en montagne et il était assumé autrefois par les populations, le plus souvent par un choix judicieux des zones d'habitation et de passage ainsi que des périodes de leur fréquentation, plus rarement par la construction d'ouvrages de protection et naturellement par un certain fatalisme.

L'apparition d'activités nouvelles exigeantes en espace, en sécurité et en régularité conduit l'être humain à disputer aux avalanches des parties du territoire qu'il leur abandonnait autrefois le temps d'un hiver.

Cette lutte passe naturellement par une connaissance du phénomène et de ses origines. De ce point de vue l'avalanche se distingue des autres risques naturels par la nature et la diversité des phénomènes en jeu.

Ces différents phénomènes se déroulent sur des parties distinctes du site avalancheux. La plupart des sites sont considérés comme étant composés de trois parties : une zone d'accumulation, une zone d'écoulement et une zone de dépôt (cf. figure 4.1).

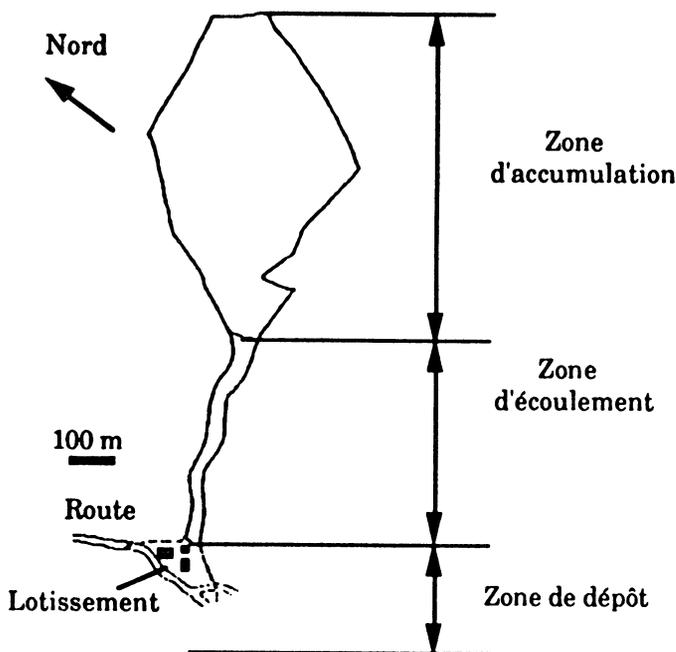


Figure 4.1 : Le vallon de Ciagé à Tende dans les Alpes Maritimes. Ce site avalancheux a été étudié à cause d'un groupe de maison construites dans la zone de dépôt. Les trois zones sont clairement séparées les unes des autres.

Sur chacune de ces zones, les phénomènes impliqués dans l'apparition d'une avalanche sont différents (cf. figure 4.2).

La formation et l'évolution du manteau neigeux dépendent dans une très large mesure des conditions météorologiques et des directions et de la force du vent car le transport de neige par le vent joue un rôle fondamental.

Les montagnes modifient l'écoulement horizontal de l'air. Les pentes et les crêtes créent ce qu'on appelle des effets orographiques sur le champ des vitesses du vent : accélération au vent d'une arête, détente sous le vent. Or il se trouve que la capacité d'un écoulement d'air à transporter des particules de neige est très nettement relié à sa vitesse. Par conséquent, les accumulations de neige, les plaques à vent et les corniches consécutives à un épisode météorologique dépendent de la position des arêtes principales par rapport au vent dominant durant cet épisode.

Une fois en place, le manteau neigeux évolue : les précipitations, les abrasions par le vent, la fonte, la sublimation, le tassement, le regel, tous ces phénomènes font du manteau neigeux un milieu en perpétuel changement jusqu'à sa disparition totale sous forme d'eau, de vapeur, de glace ou... d'avalanche.

A aucun moment, le manteau neigeux n'est homogène. Au cours de sa mise en place puis des différentes métamorphoses qu'il subit, des couches aux

caractéristiques physiques distinctes apparaissent. Leur stabilité est très variable. Elle dépend de la qualité de la neige (densité, type de cristal, etc). Des plans de glissement virtuels s'installent ainsi à travers le manteau neigeux. Ils vont permettre le déclenchement d'une avalanche.

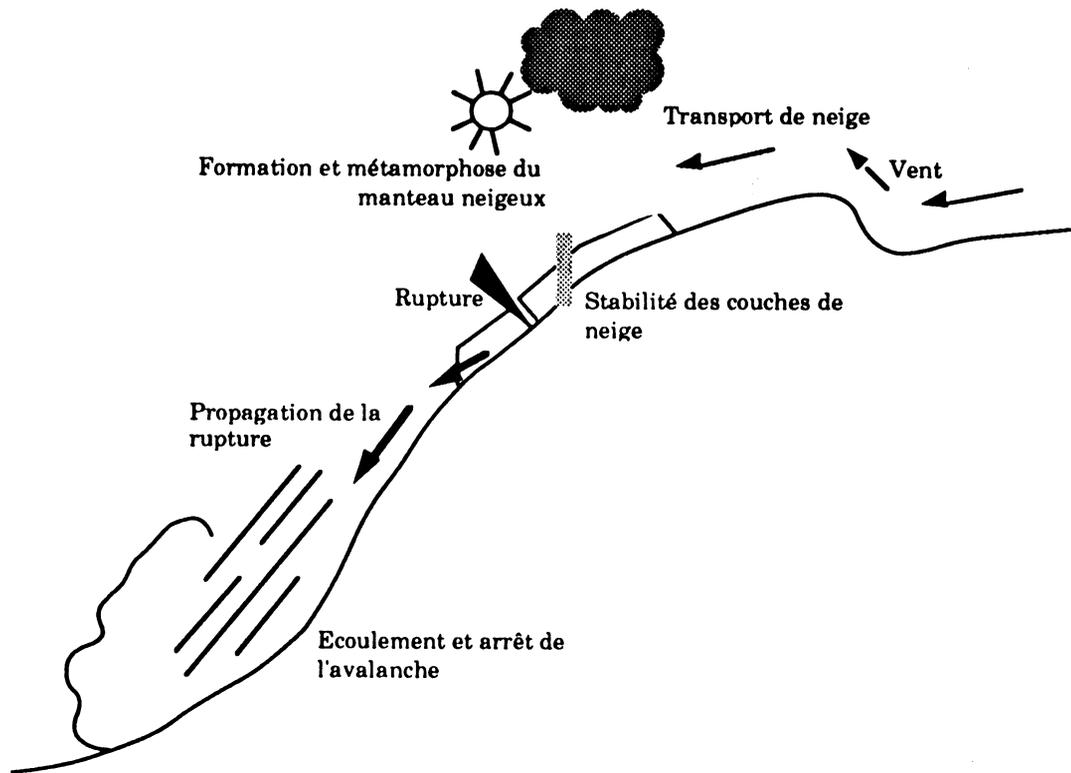


Figure 4.2 : Quelques uns des phénomènes qui conduisent à l'apparition d'une avalanche.

Celui-ci trouve son origine dans une intervention soudaine (skieur, animal, chute de pierres ou de séracs, explosion) ou progressive (réchauffement, tassement). Dans tous les cas, il s'agit d'une rupture dans le manteau neigeux.

Quand une avalanche est déclenchée, l'équilibre dans le manteau neigeux disparaît dans une partie au moins de la zone de départ. La propagation de cette rupture sur la zone de départ devient alors le phénomène important qui va décider de la taille de l'avalanche. La répartition du manteau neigeux sur la zone de départ, sa stabilité en chaque point, la présence d'ancrages et de ruptures de pentes vont en effet décider de ce qui doit partir et de ce qui doit rester en place.

Ensuite, commence la phase d'écoulement. Elle comprend des phénomènes d'accélération et de ralentissement, des phénomènes de reprise de la neige en place, des changements de type d'écoulement et enfin l'arrêt de l'avalanche. Au cours de l'écoulement, l'interaction avec des individus ou des équipements en place va occasionner les éventuels accidents et dégâts. L'ancrage de ces équipements, leur comportement mécanique deviennent finalement les phénomènes déterminants.

Les zones d'accumulation, d'écoulement et d'arrêt sont en fait définies par rapport aux phénomènes qui s'y déroulent. La zone d'accumulation est le lieu de constitution, d'évolution et de rupture du manteau neigeux. Sur la zone d'écoulement, l'avalanche est dans une phase dynamique ; les phénomènes statiques sont négligeables. Sur la zone d'arrêt, l'avalanche entre dans une phase de décélération et de dépôt. La délimitation de ces zones exige de fait une analyse déjà fine du site. Dans la suite de ce document, les sites seront définis par une enveloppe maximale et la limite zone d'écoulement - zone d'arrêt n'interviendra plus.

Les phénomènes qui viennent d'être décrits se déroulent tous sur un site donné et c'est l'objectif de l'analyse de sites que de les étudier sur un terrain réel.

4.1.2. Analyser des sites

En effet, contrairement aux prévisionnistes du risque d'avalanche dans le temps [COLE&89], les spécialistes de l'analyse de sites s'intéressent aux avalanches qui menacent un équipement particulier. L'objectif n'est pas d'annoncer un risque, à chaque instant, à l'échelle d'un massif, mais de protéger, de manière permanente, un équipement donné.

Problématique de l'analyse de sites avalancheux

Au cours de l'étude préliminaire d'un projet d'équipement en montagne, parmi les nombreuses études de faisabilité qui sont réalisées, l'une d'entre elles concerne le risque d'avalanche [BUIS&89a].

Les questions auxquelles il faut répondre sont : Cet équipement est-il soumis à un risque d'avalanche ? Si c'est le cas, que faire pour le protéger ?

Les réponses sont toujours inscrites dans les caractéristiques du site de l'équipement : massif, météorologie locale, végétation, topographie, géologie... Analyser un site, c'est y lire ces réponses.

Depuis bientôt quinze ans, en France, des spécialistes ont développé une expertise qui leur permet de réaliser de telles analyses. Ces experts sont peu nombreux, assurément moins d'une dizaine. Leur expertise est d'abord bâtie sur une méthodologie d'analyse qui est suivie quelque soit l'équipement à protéger, qu'il s'agisse d'un bâtiment d'habitation à construire ou d'un pylône de remontée mécanique à renforcer.

Méthodologie de l'expertise du risque d'avalanche

Cette méthodologie d'analyse n'avait, jusqu'à présent, jamais fait l'objet de tentative de formalisation. Un effort de formalisation a débuté en 1986 avec une étude de faisabilité pour un système expert d'analyse de sites avalancheux [AUDF86].

Aujourd'hui quatre étapes apparaissent comme essentielles au cours d'une analyse de sites avalancheux [BUIS&89a].

- Une *étude locale du site* permet de repérer les caractéristiques essentielles du site et de déterminer les conditions nivo-météorologiques auxquelles il est soumis. C'est aussi l'occasion d'identifier les traces des avalanches du passé et de suggérer un ou plusieurs fonctionnements avalancheux de ce site qui concernent toutefois l'équipement.

- L'expert doit ensuite imaginer les phénomènes avalancheux sous leurs différents aspects : scénario de déclenchement, déroulement de l'avalanche, types d'écoulement, dynamique, étalement et arrêt. Il s'agit de prévoir le déclenchement et le déroulement des *avalanches majeures*.

- Il faut ensuite choisir une *stratégie de protection* adaptée à l'équipement menacé. Il peut en effet s'agir de s'éloigner de la menace, y faire face en renforçant l'équipement, supprimer l'avalanche ou encore la modifier dans sa taille, son parcours ou sa fréquence.

- Enfin, à partir de la stratégie de protection choisie, l'expert doit sélectionner un dispositif de protection parmi ceux disponibles ou en combinant des dispositifs élémentaires. Il doit également dimensionner ces dispositifs.

La méthodologie qui vient d'être présentée n'est en aucune façon une procédure. Une analyse de site n'est pas en effet une succession linéaire d'opérations. L'expert est susceptible à tout instant de revenir sur des observations, des choix ou des interprétations au fur et à mesure que des impossibilités techniques apparaissent.

L'étape essentielle reste cependant la seconde. Plus l'image de l'avalanche majeure est précise, meilleur est le choix de la stratégie de protection, plus fin le dimensionnement. Cette capacité à imaginer le fonctionnement de l'avalanche doit aussi être exploitée dans une phase de vérification de la solution choisie. L'expert doit analyser à nouveau le site une fois qu'il est protégé.

Pour imaginer ces avalanches majeures sous tous leurs aspects, l'expert dispose d'un certain nombre d'outils. La plupart peuvent être considérés comme des modèles de phénomènes en jeu.

4.1.3. Modéliser pour protéger

Quel type d'avalanche peut-il apparaître ? Dans quelles conditions ? Suite à quel type de scénario nivo-météorologique ? Quelles sont les caractéristiques de cette avalanche, sa taille, sa vitesse, son emprise, la répartition des pressions, etc ? Pour répondre à ces questions l'ingénieur peut utiliser différentes méthodes d'analyse. A l'exception de la première, il s'agit de modèles qui ne peuvent offrir que des *images* du phénomène réel.

Observation et mesures d'avalanches sur le terrain

L'expert peut bien sûr faire des mesures et des observations d'avalanches sur le site à protéger. Malheureusement, outre les dangers et les difficultés affrontés, il n'en tirera que des informations sur les avalanches qui se sont produites en sa présence sur le site. Sur les avalanches passées ou à venir il ne sait rien. De plus, les avalanches sont difficiles à observer ; elles n'attendent pas le spectateur. Enfin, le nombre de sites à analyser interdit leur équipement systématique en instruments de mesure.

Ce travail de mesure et d'observation existe cependant [BRUG&85] [NORE&86b]. Mais il relève de la recherche et ne concerne qu'un nombre réduit de sites expérimentaux. L'objectif n'est plus alors la protection de ces sites mais une meilleure compréhension du phénomène avalanche. Les paramètres dont la mesure est tentée sont actuellement les vitesses d'écoulement, les pressions développées, les densités et les volumes.

L'expert en est donc réduit à l'exploitation de modèles qui, à partir de données faciles à obtenir, fournissent des images de l'avalanche ou des phénomènes impliqués dans les avalanches.

Modélisation physique

L'expert peut tenter des simulations physiques de ces phénomènes. Compte tenu de la taille de ces phénomènes, il ne peut s'agir que de modèles réduits.

La prise en compte des effets orographiques dans l'écoulement du vent peut se faire à l'aide de tels modèles réduits en remplaçant par exemple l'air par de l'eau [MENA90]. La construction d'une maquette est cependant nécessaire, ce qui est assez coûteux en temps ou en moyens.

L'utilisation de soufflerie pour modéliser le transport de neige par le vent est également à l'étude. La neige est alors remplacée par de la sciure ou du sable. Toutefois l'étude du transport de neige sur des maquettes de zones d'accumulation de sites avalancheux reste un objectif encore très éloigné.

L'écoulement d'avalanches de neige poudreuse, c'est-à-dire l'écoulement d'un aérosol de neige est également modélisé à une échelle réduite dans des bassins remplis d'eau. Les phénomènes observés sont en fait des courants de gravité (courants de densité et courants de turbidité) [BEGH&83]. Mais représenter une topographie réelle sur une maquette est une opération lourde et coûteuse. De plus, les conditions de similitude ne sont pas respectées et les résultats doivent être interprétés avec de grandes précautions. Toutefois l'étude de ces courants de gravité est une approche très intéressante dans une optique de recherche, soit qu'elle donne des résultats qualitatifs sur des géométries simples, soit qu'elle permette le calage de modèles numériques qui pourront ensuite être appliqués aux véritables avalanches de poudreuse.

Modélisations numériques

La modélisation numérique est en fait une triple modélisation⁴. Dans un premier temps, un ensemble d'équations différentielles ou d'équations aux dérivées partielles est écrit qui modélise les phénomènes physiques. Ces équations peuvent être conservatives ou empiriques ; elles peuvent aussi contenir des simplifications. Un schéma numérique est ensuite choisi ; il s'agit du deuxième modèle qui rend compte du comportement des variables et des fonctions des équations. Enfin un dernier modèle, informatique celui-là, prend en charge le schéma numérique. Il arrive que l'ensemble d'équations admettent une solution analytique ou que le recours à l'informatique soit inutile pour la résolution du schéma numérique. Il s'agira néanmoins ici toujours de modélisation numérique.

De nombreux phénomènes parmi ceux impliqués dans les avalanches font l'objet de tentative de modélisation symbolique.

Le comportement du vent sur un relief a été modélisé par de nombreux programmes [TESC&76] [GUIL&90]. Il faut préciser cependant qu'il n'existe pas actuellement de modèles de vent fonctionnant à une échelle telle qu'elle permette leur utilisation dans l'analyse de sites avalancheux.

⁴ Cette présentation de la modélisation numérique est due à Jean-Paul VILA qui l'a formulée de façon verbale très clairement devant l'auteur de ces lignes.

Pour ce qui concerne le transport de neige par le vent les modèles actuellement en développement [CLAP89] restent très éloignés des préoccupations des ingénieurs. Il s'agit plutôt d'outils de recherche destinés à approfondir la connaissance du phénomène.

Sur l'évolution du manteau neigeux, les efforts importants accomplis par le Centre d'Etude de la Neige dans le cadre du programme CROCCUS donnent des résultats très intéressants. Actuellement, le modèle disponible permet de modéliser les métamorphoses de la neige dans un manteau neigeux réel avec une très grande fidélité [COLE&89]. Pour la stabilité, en revanche, il n'existe pas de modèle numérique satisfaisant de même que pour l'apparition de la rupture et la propagation de celle-ci.

C'est en fait pour l'écoulement de l'avalanche elle même que les travaux en modélisation numérique sont les plus anciens. Voellmy [VOEL55] a été l'initiateur de cette série de nombreux travaux. Un certain nombre d'entre eux ne constituent d'ailleurs que des variations sur le thème de Voellmy [CHEN&79] [MCLU&84b]. Les autres, plus spécifiques, sont destinés aux avalanches de neige dense [VILA84], de neige poudreuse [BEGH&83] ou de neige quelconque [NORE&86a] [DENT&80].

Dans tous les cas une description de la topographie est nécessaire ainsi que la fourniture de conditions initiales. Par exemple, le programme AVAL [VILA84] accepte en entrée une description de la géométrie sous la forme de tronçons à section constantes (figure 4.3) et une condition initiale sous la forme d'un hydrogramme en début de zone d'écoulement (figure 4.4).

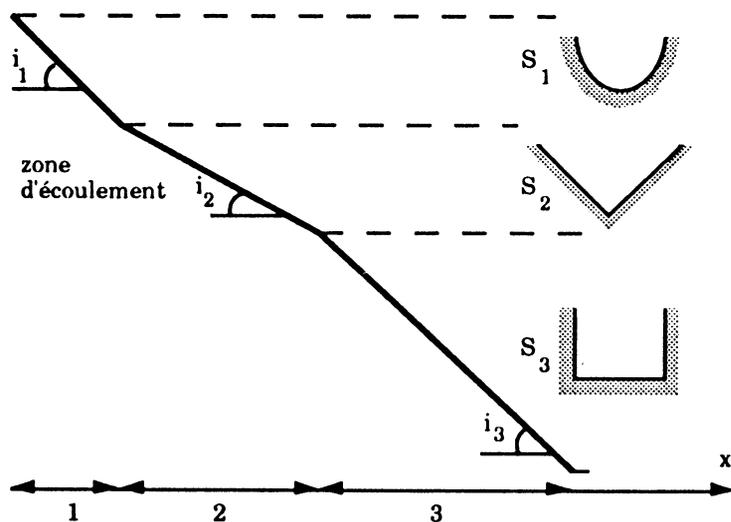


Figure 4.3 : Description de la géométrie de la zone d'écoulement dans le programme AVAL. Le programme fait un calcul mono-dimensionnel mais exige une description dont la dimension peut être estimée à 2,5. Elle consiste en la donnée de tronçons à pente constante (i_1 , i_2 et i_3 sur les tronçons 1, 2 et 3) et à section également constante (S_1 , S_2 et S_3).



Figure 4.4 : Hydrogramme en début de zone d'écoulement. Le débit Q en fonction du temps t , constitue la condition initiale dont le programme AVAL a besoin pour lancer le calcul de l'écoulement.

Ces systèmes ne sont pas sans défauts. Leur utilisation reste délicate : calage, description des géométries, choix de paramètres, choix des conditions initiales ou des conditions aux limites sont souvent problématiques et exigent de l'utilisateur une expérience certaine. Cependant, dans les années qui viennent, des systèmes opérationnels devraient être capables de modéliser l'écoulement d'une masse de neige d'un type donné.

Ces méthodes numériques ont toutes pour ambition d'exploiter une interprétation du phénomène. La plupart par exemple exploite la conservation de la masse.

Méthodes statistiques

D'autres méthodes, plus empiriques, ont été développées. Elles sont basées sur des approches statistiques. Le but n'est plus alors de comprendre en détail le phénomène, mais de corréliser entre eux quelques paramètres grâce à l'exploitation des données provenant d'un grand nombre de sites. Le phénomène est vu alors comme une boîte noire dont seuls sortent les différents paramètres.

La formation de congères a ainsi été étudiée par une approche statistique [TABL75]. Elle permet une prise en compte du transport de neige par le vent, mais ne concerne actuellement que des reliefs de type plateaux et à une échelle qui interdit, là encore, son utilisation pour l'analyse de sites avalancheux.

Enfin, ces approches statistiques ont été appliquées aux avalanches par les chercheurs du NGI (Norwegian Geotechnical Institute) [LIED&80] [BAKK&83] [MCLU&84b]. Il s'agit de relier la distance d'arrêt maximale des avalanches sur un site donné à quelques caractéristiques géométriques du profil en travers de ce site (figure 4.5).

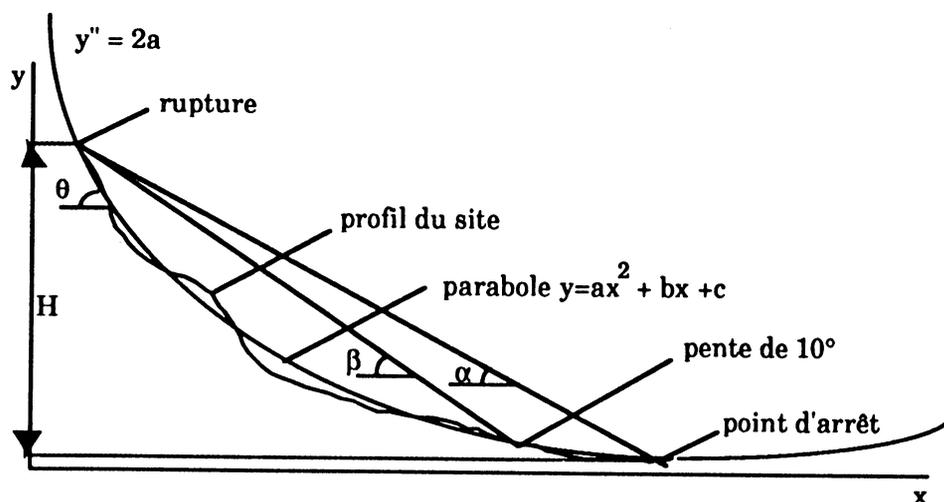


Figure 4.5 : La méthode statistique développée par [LIED&80] exige des paramètres topographiques du site avalancheux : θ pente moyenne dans la zone de départ, point où la pente fait 10° (plus exactement l'angle β entre l'horizontale et la droite qui passe par ce point et le sommet du couloir), H et y'' qui dépendent de la parabole $y = ax^2 + bx + c$ qui approche le mieux le profil en long du site (en fait $y'' = 2a$ et H est la hauteur de la parabole). La distance d'arrêt maximum (elle s'obtient en fait par l'angle α défini comme l'angle β) a été corrélée à ces paramètres par une analyse statistique sur un grand nombre de sites avalancheux connus. Pour l'analyse d'un nouveau couloir, la mesure de ces paramètres donne une estimation empirique de la distance d'arrêt sous la forme de l'angle α en fonction de θ , β , H et y'' .

Ces techniques ont été testées également sur des sites américains et suisses [MART86], ainsi que sur des sites français [BERT86].

Récapitulatif

Ces différents modèles sont pour la plupart encore en développement. Les modèles "opérationnels" demeurent délicats à utiliser. L'ingénieur doit très bien les connaître pour les exploiter. En particulier, des connaissances en analyse numérique ou une certaine expertise dans ce domaine restent nécessaires.

De plus, ces modèles ne sont pas compatibles entre eux. Ils tournent dans des services différents, sur des machines hétérogènes. Leurs fichiers d'entrées-sorties n'ont pas les mêmes formats. En d'autres termes, l'expert exploite relativement peu ces différents outils. Il exploite surtout ses propres connaissances en *imaginant* les phénomènes qu'il doit étudier. Son expérience est à l'origine de l'essentiel des analyses.

Imagination de l'expert et modélisation symbolique

En effet, un expert est capable, face à un site inconnu, d'en imaginer son fonctionnement. Répartition du manteau neigeux, présence de suraccumulations, de plaques à vent, de corniches, plans de glissement dans le manteau neigeux, zones de ruptures, trajectoires d'avalanches de neige dense ou poudreuse... Il parvient à estimer sans trop d'erreur la magnitude et la localisation de ces phénomènes. Pour cela, il exploite les connaissances de nature empirique qu'il a accumulées au cours de sa carrière.

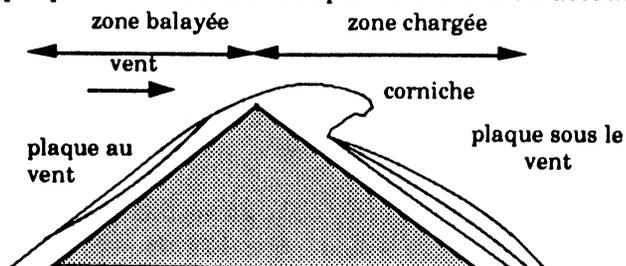
Quatre exemples sont développés qui mettent en évidence les connaissances manipulées par l'expert.

Exemple 4.1 : Face à un site l'expert doit, entre autres, délimiter les contours du site. Pour cela, il faut définir la surface de terrain susceptible d'alimenter une avalanche menaçant le point à protéger. Ce travail exige de pouvoir répondre à une question du type :

"Cette partie de terrain est-elle au dessus de celle-là ?"

Pour répondre à cette question il faut faire appel à des notions de bon sens telle que ligne de plus grande pente, ligne de partage des eaux qui ne peuvent être définies mathématiquement sans perdre une partie du sens qu'elles ont auprès d'un expert donné. Des techniques procédurales classiques existent déjà [MARK&83] [DUFO88] mais elles utilisent une décomposition du terrain en unités régulières appelées mailles et se contentent de tracer les limites de sous-bassins.

Exemple 4.2 et figure 4.6 : L'analyse de la zone d'accumulation passe par une évaluation du manteau neigeux en épaisseur et en qualité sur cette zone. L'expert doit prendre en compte l'action du vent et du soleil, évidemment soumise à la topographie. La position de l'unité de terrain considéré par rapport à une arête fera qu'une partie de terrain sera balayée par le vent, pourvue d'une corniche ou d'une plaque à vent ou tout simplement d'une suraccumulation.



Là encore, les notions de distance, d'orientation, d'exposition, évidentes pour l'expert ou le simple montagnard, doivent être appréhendées.

Exemple 4.3 : Selon la nature du sol ou des sous-couches, une couche de neige peut être plus ou moins stable. La stabilité d'un manteau neigeux peut être évaluée par un spécialiste sans qu'il ait recours à aucun modèle de calcul. Il exploite pour cela son expérience et les observations qu'il a pu faire dans le passé.

Exemple 4.4 : Une fois une rupture apparue dans le manteau neigeux, se pose le problème de la propagation de cette rupture. La pente, la répartition du manteau neigeux, certaines frontières naturelles peuvent être exploitées par des experts pour déterminer la zone qui finalement est déclenchée.

L'hypothèse fondamentale faite ici est que les connaissances empiriques utilisées par l'expert dans ces phases d'estimation et d'imagination du phénomène peuvent être modélisées à l'aide de symboles. En d'autres termes, l'intelligence de l'expert peut être modélisée par le calcul de représentations symboliques. La formulation choisie ici est en fait celle retenue par Varela pour définir l'hypothèse cognitive qui est à la base de nombreux travaux d'intelligence artificielle [VARE89].

4.1.4. Evolutions nécessaires

Les analyses de sites avalanches soulèvent aujourd'hui trois problèmes. Tout d'abord, le nombre des analyses augmente sans cesse suite à des changements dans les mentalités : l'avalanche fait désormais partie des phénomènes que les responsables n'hésitent pas à prendre en compte dans les études préparatoires de certains projets. Ensuite, le niveau des enjeux qui correspondent aux équipements à protéger exige des analyses de plus en plus fines et de plus en plus sûres. Enfin, les projets d'équipements actuels se rapprochent de plus en plus de l'avalanche car les surfaces les plus éloignées ont été équipées les premières.

Les analyses sont donc appelées à être de plus en plus nombreuses et de plus en plus complexes.

Face à cette évolution, les experts ressentent deux besoins très précis.

Le premier concerne la *rationalisation des analyses*. Il ne s'agit plus désormais de savoir si l'avalanche va frapper un bâtiment, mais de savoir quels efforts elle va exercer sur lui. Les caractéristiques de l'avalanche doivent être accessibles : densité et vitesse. De même, il ne s'agit pas de savoir qu'une route est menacée en un endroit mais de déterminer si, à un instant donné, lors d'un épisode nivo-météorologique, la route doit être interdite au passage. L'analyse de site se transforme petit à petit en une gestion temporelle du risque sur un site.

Dès lors, la part réservée à la subjectivité doit être réduite et l'appel à des techniques de calcul développé. L'imagination d'un expert ne peut en effet lui fournir les valeurs des densités ou des vitesses d'un écoulement. Enfin, les erreurs d'estimation, souvent à l'origine de mauvaises analyses doivent être limitées par l'automatisation.

Le second besoin concerne plus directement les experts. Il a déjà été dit qu'ils étaient trop peu nombreux pour répondre à toutes les demandes d'analyse. Ils sont aussi appelés à changer d'activité, à partir à la retraite. C'est alors le problème de la *perpétuation de l'expertise* qui se pose et, à travers lui, celui de la formation de nouveaux experts.

Il devient dès lors naturel d'essayer de construire un système informatique qui disposerait de l'ensemble des techniques de modélisation et les intégrerait au sein d'une méthodologie unique. Il pourrait alors aider un expert à réaliser des analyses de sites et, dans le même temps, fournir un outil de formation intéressant.

Le développement du système ELSA vise cet objectif.

Mais comment celui-ci doit-il être bâti ? Quels en sont les principes ? Pour définir clairement le travail à réaliser, et afin de mettre en évidence les fonctionnalités du système, l'approche qui a été choisie doit être présentée succinctement.

4.2. APPROCHE CHOISIE

L'approche retenue pour le développement du système ELSA est présentée ici. Dans un premier temps, les similitudes avec des systèmes experts déjà existants dans le domaine des sciences de la terre ou des risques naturels sont exposées. Dans un deuxième temps, le concept d'environnement de résolution de problèmes est décrit à travers les besoins auxquels il essaie de répondre et les solutions informatiques qui lui sont adaptées. ELSA adhère tout à fait à ce concept d'environnement de résolution de problèmes. Il lui doit son outil de représentation de connaissances : SHIRKA. Dans un troisième temps les objectifs du système ELSA sont repris, en particulier l'aspect base de méthodes. Enfin l'importance de la communication entre ELSA et les experts est illustrée à travers la modélisation des objets du problème, l'interface graphique et la représentation de l'espace. Ce dernier point fait l'objet des deux dernières parties de ce chapitre sur ELSA qui traitent du raisonnement spatial dans ce système.

4.2.1. Sciences de la terre, risques naturels et systèmes à base de connaissances

Les nombreuses disciplines ou techniques qu'on rassemble sous le nom générique de sciences de la terre (géologie, géotechnique, vulcanologie, hydrologie, nivologie, etc.) et dont beaucoup se préoccupent de risques naturels (inondations fluviales, laves torrentielles, avalanches, glissements de terrain) font d'ores et déjà largement appel à l'informatique.

En effet, leur fonction principale étant le diagnostic (de risques, de potentialités, d'efforts...), elles utilisent depuis longtemps des dispositifs de simulation numérique ou de traitement statistique. Cependant, leur condition de sciences naturelles les oblige à appliquer ces codes ou ces méthodes dans des conditions les plus *naturelles* possibles c'est-à-dire sur des surfaces aux formes complexes, avec des conditions initiales ou aux limites difficiles à définir.

Dès lors, la détermination du type de modèle à utiliser, des conditions initiales, des domaines de calcul et des conditions aux limites apparaît comme un problème crucial.

Il en va de même pour la construction de maillages ; la topographie ou la géologie de la terre se prêtent mal à une réduction à des formes géométriques simples. L'automatisation des tâches de maillage ou d'évaluation des conditions initiales ou des conditions aux limites est rarement réalisée et, même quand elle l'est, elle laisse souvent le modèle hors de portée des non-numériciens, pour des problèmes de calage ou de convergence par exemple.

Les modèles existants sont donc difficiles à choisir et à appliquer. L'introduction d'heuristiques pour le choix des modèles ou le traitement symbolique de certaines données, en vue de leur utilisation par ces modèles, permettent la résolution de ces problèmes par la machine. L'utilisation de systèmes à base de connaissances autorise l'introduction d'heuristique et l'intégration de différents modèles.

Par ailleurs, face à l'absence de modèles dédiés à certains phénomènes, l'essor actuel des systèmes à base de connaissances prouve qu'une modélisation purement symbolique ou logique et n'ayant plus rien à voir avec la simulation numérique classique peut être riche d'enseignement dans ces mêmes sciences de la terre.

Des activités jusque là réservées aux experts, et par conséquent peu formalisées, peuvent être désormais menées par la machine. Mais là encore, la difficulté réside dans la nature du sujet étudié ; il ne se décompose pas facilement en composants simples : la nature n'est pas géomètre!

Tout ceci conduit à un développement important des systèmes à base de connaissances dédiés aux sciences de la terre. En géologie, les systèmes PROSPECTOR [DUDA&79] ou SADIG [PUVI&89] sont dédiés à la prospection minière ; les systèmes LITHO [BONN&82], ELFIN [MCL84] à la prospection pétrolière. Pour ce qui concerne les risques naturels les systèmes MEPRA [GRAN&84] [LEFE&85] et NEVAI [MASS87] traitent d'avalanches, GEM [WIL86] d'incendies de forêt. Rosenthal et al proposent de traiter des problèmes de vulcanologie [ROS85] et Ke des glissements de terrain [KE87].

ELSA s'inscrit dans cette approche.

Toutefois, à l'exception d'ELFIN, déjà mentionné dans les chapitres précédents, ces systèmes à base de connaissances, y compris ceux qui ont traité le problème des avalanches [LEFE&85] [MASS87], n'ont pas pris en compte les caractéristiques spatiales du phénomène, qui jouent pourtant un rôle fondamental. MEPRA s'intéresse plutôt à l'évolution du risque dans le temps à l'échelle d'un massif montagneux. Il a été montré que la situation est toute différente quand il s'agit de protéger un site donné.

Par ailleurs, l'importance de l'espace dans les sciences de la terre et dans la maîtrise des risques naturels est illustrée par de nombreuses autres utilisations de l'informatique. Il peut s'agir d'applications à forte composante "base de données" comme le programme de la division Nivologie du CEMAGREF de rénovation et de numérisation des CLPA (Cartes de Localisation Probables d'Avalanches) sur le système d'information géographique ARC-INFO [MORI&89], ou alors d'application plus orientées vers le calcul de zones dangereuses. C'est le cas pour TERMOS-HAZARD qui détermine, par application automatique de la méthode statistique [LIED&80] et en exploitant un modèle numérique de terrain, les zones soumises à un risque d'avalanche [TOPP86].

Un des principes de base d'ELSA est précisément de choisir une approche "système expert", ou plus exactement "système à base de connaissances" en prenant pleinement en compte l'aspect spatial du phénomène.

4.2.2. Environnements de résolution de problèmes

Un environnement de résolution de problèmes est un système à base de connaissances chargé de fournir à un spécialiste d'un domaine particulier

l'ensemble des outils qu'il est susceptible d'utiliser en les intégrant dans une méthodologie.

Le concept d'environnement de résolution de problèmes est à distinguer de celui de résolveur de problèmes popularisé dans les années soixante par GPS (General Problem Solver). A la différence de celui-ci, un environnement de résolution de problèmes a pour fonction d'aider un spécialiste d'un domaine à résoudre un type de problèmes particulier. Les ambitions sont donc beaucoup plus modestes et les préoccupations plus appliquées.

Il est possible de distinguer deux grandes familles d'environnements de résolution de problèmes : ceux consacrés à l'ingénierie et les environnements de résolution de problèmes scientifiques. Les environnements de résolution de problèmes d'ingénierie permettent à un ingénieur de traiter un problème technique particulier. Les outils qui s'y trouvent sont des outils de modélisation, d'aide à la décision, d'estimation de coûts. Le domaine d'application est présent partout. Le meilleur exemple, et le plus avancé sans doute, est le système ERASME développé par le CETE d'Aix-en-Provence pour la Direction des Routes au Ministère des Transports [ANTO&90]. Il a pour fonction d'aider un responsable des services de l'équipement à suivre et entretenir un réseau routier.

Les environnements de résolution de problèmes scientifiques sont destinés à traiter une famille de problèmes. Les solutions présentées par Rousseau [ROUS88] concernent le poste de travail du biométricien. Le système EVE [CHAI&89] s'adresse au numéricien et lui propose une aide dans le calcul des solutions de systèmes d'équations aux dérivées partielles.

Le thème des environnements de résolution de problèmes est très lié à celui du couplage numérique-symbolique en modélisation ainsi qu'à celui de l'intégration des connaissances provenant de différents experts.

Couplage numérique-symbolique

Ce thème a été particulièrement développé dans la série de conférences organisées par Kowalik [KOWA86] [KOWA88]. L'idée de base est que les techniques développées en intelligence artificielle peuvent apporter des améliorations à l'usage des programmes numériques. En général ce couplage est vu comme la possibilité pour un programme d'intelligence artificielle d'appeler et de contrôler des codes numériques.

Abelson *et al* [ABEL89] dresse un inventaire assez complet de ce que le symbolique peut apporter au numérique en calcul scientifique : préparation et contrôle de simulation numérique, interprétation qualitative de résultats issus de telles simulations, formulation symbolique d'algorithmes, génération automatique de programme.

Le système MEPRA [GIRA87], qui utilise les connaissances d'experts pour estimer la stabilité d'un manteau neigeux donné, est par exemple destiné à fonctionner avec le système CROCCUS qui simule l'évolution dans le temps d'un manteau neigeux. Les sorties de CROCCUS sont les entrées de MEPRA. La partie 4.2.3 donne des exemples de ce qui peut être fait dans ELSA.

Intégration de connaissances

L'intégration de connaissances d'origines diverses et de méthodes d'inférence différentes fait partie des thèmes de recherche importants en

intelligence artificielle. Les études sur la coopération entre connaissances provenant d'experts différents sont à l'origine des études sur les "blackboards" et du développement du système MEPAR devenu depuis SMECI (Système Multi Experts pour la Conception en Ingénierie) [HARE&85] [ILOG].

L'installation de méthodes d'inférence différentes pose le problème du choix, par le système, de la méthode la mieux adaptée. Une solution possible est présentée dans [ROUS88] qui consiste à exploiter un treillis de tâches en choisissant, par classification sur les données du problème, la méthode la mieux adaptée.

La partie 4.2.3 présente à ce propos les objectifs d'ELSA.

SHIRKA

Le système SHIRKA est le produit de la participation de l'équipe "Base de connaissances" du laboratoire ARTEMIS au projet de l'INRIA EDORA (Equations Différentielles Ordinaires Récurentes Appliquées). En particulier, la prise en compte des méthodes sous forme d'objets via l'attachement procédural a été pensé dans une optique base de méthodes [RECH86]. Les maquettes produites dans le cadre d'EDORA l'ont été en utilisant SHIRKA [EDOR87] [PAVE&86].

Il faut préciser que SHIRKA est actuellement utilisé pour le développement de deux environnements de résolution de problèmes : EVE [CHAI&89] et SCAI [PONC90]. Le premier a déjà été mentionné ici et traite de la résolution de systèmes d'équations aux dérivées partielles. Le second est destiné à aider les utilisateurs d'une bibliothèque de calcul scientifique à choisir les méthodes et à planifier leur exécution.

Le nombre des modèles et méthodes susceptibles d'être utilisés (cf 4.1.3. Modéliser pour protéger), leur complémentarité et les difficultés de leur utilisation conduisaient naturellement à considérer ELSA comme un environnement de résolution de problèmes.

Le choix du système de représentation de connaissances SHIRKA en vue du développement d'ELSA trouve ici sa raison principale.

4.3. ELSA : UN ENVIRONNEMENT DE RESOLUTION DE PROBLEMES

L'objectif d'ELSA est de formaliser l'analyse de site, de conserver les expériences des différents experts et de les intégrer dans un système utilisable par un ingénieur chargé de la protection des sites, sous la forme d'un environnement de résolution de problèmes. Le but ultime consiste à offrir à l'ingénieur un outil d'aide à la décision pour le choix des techniques de protection.

Ce qui suit illustre, dans le cas particulier d'ELSA et des avalanches, l'application des idées présentées ci-dessus et qui concernent toutes l'intégration d'un grand nombre de méthodes.

4.3.1. Base de méthodes

A la diversité des phénomènes répond déjà la diversité des méthodes de simulation qui ressort de la sous-section "Modéliser pour protéger". Cette diversité doit à son tour se traduire au sein du système ELSA par une diversité des méthodes qui seront à la disposition de l'expert.

Il ne s'agit pas ici de comparer les différentes méthodes disponibles en qualité ou en validité. Leur manipulation est souvent difficile. Leur intégration dans un système unique devrait faciliter leur utilisation ; c'est simplement cet aspect des choses qui est discuté ici.

Dans ELSA, les différentes méthodes coopèrent de différentes façons.

- Tout d'abord, elles constituent une base de méthodes unique disponible au cours de la même session et sur la même représentation de connaissances.

La première conséquence de ceci est que les appels au système d'exploitation sont réduits au minimum, puisque l'utilisateur reste dans le même processus : celui de l'environnement de résolution de problèmes. A l'opposé, quand chaque méthode correspond à un programme différent, l'utilisateur doit les appeler depuis le système d'exploitation sous leur nom et avec les options éventuelles.

- Deuxièmement, dans une base de méthodes les paramètres et données des différents programmes sont les mêmes. Cela signifie que ELSA a une représentation du site avalancheux et des conditions de déclenchement qui peuvent être utilisées par toutes les sortes de modèles.

Exemple 4.5 : Un exemple est donné ici avec trois méthodes utilisées dans l'analyse de sites avalancheux. Elles sont complètement différentes les unes des autres aussi bien du point de vue de leur validité que de celui de leur fondement théorique.

La première est la méthode statistique décrite en 4.1.3. Elle calcule une distance d'arrêt empirique à partir de paramètres topographiques du site avalancheux. La description du terrain est bi-dimensionnelle.

La deuxième méthode correspond à une simulation symbolique qui sera présentée par la suite. Elle exige en fait une description 3-D de la zone d'accumulation basée sur une décomposition triangulaire et sur la définition des arêtes principales qui jouent un rôle important dans le transport de neige par le vent. Ce terrain imaginaire est supposé approcher le terrain réel.

La dernière méthode, AVA, décrit également en 4.1.3, simule le flot d'une avalanche de neige dense. La description du terrain qui lui est nécessaire peut être qualifiée, en dimension, de 2,5 (cf. figure 4.3).

Ces trois méthodes exigent donc toutes une description du terrain. Celle-ci peut être 2D, 2,5D ou 3D. Avec une description complète 3D de tout le site avalancheux, un système tel qu'ELSA est capable de construire les descriptions 2D et 2,5D. En fait des routines de calcul géométrique très simples, telles que des intersections de lignes ou des calculs de surface, permettent de calculer les paramètres de chacune de ces descriptions. L'utilisateur doit simplement désigner les sections en long sur lesquelles il désire voir appliquer les méthodes statistiques et numériques.

4.3.2. Méthodes concurrentes

La disponibilité de différentes méthodes dans un même système est également intéressante quand ces méthodes sont concurrentes.

Pour un type de problème, différentes méthodes peuvent être utilisées selon la situation. Par exemple, la Division Nivologie du CEMAGREF a développé des programmes pour simuler les avalanches de neige poudreuse [BEGH&83] ou de neige dense [VILA84]. Ces deux programmes peuvent être utilisés comme des méthodes concurrentes pour étudier l'écoulement de l'avalanche. Elles exigent toutes les deux une description du terrain, mais ces descriptions

peuvent être construites par l'environnement de résolution de problèmes à partir d'une unique représentation de terrain.

Dans un système tel qu'ELSA, cette capacité de l'utilisateur à choisir ou à mettre en évidence la méthode courante est très importante. L'utilisateur doit toujours savoir exactement quelles sont les conditions de la simulation. Il en est responsable. En effet, l'objectif ultime d'un environnement de résolution de problèmes d'ingénierie n'est pas de se substituer au spécialiste mais de l'aider. ELSA n'est pas autorisé à prendre une décision. Seul l'utilisateur a ce droit.

4.3.3. Méthodes complémentaires

Les résultats d'une méthode de modélisation peuvent en alimenter d'autres. Deux exemples sont donnés.

Un modèle statique de la stabilité de la neige est utilisé par un programme pour la simulation symbolique du déclenchement sur une zone de départ.

Plusieurs méthodes peuvent être utilisées pour calculer la stabilité du manteau neigeux. Parmi elles, ELSA en exploite une très simple (simpliste ?) basée sur une interprétation de mécanique des sols. g étant l'accélération de la pesanteur, pour un angle de pente i , une densité d , une cohésion c et un angle de frottement φ , la hauteur critique h_{crit} est :

$$h_{crit} = \frac{c \cos \varphi}{d g \sin (i - \varphi) \cos i}$$

Le modèle symbolique connaît la pente moyenne de chaque petit panneau. Parce qu'il est capable de prendre en compte le transport de neige par le vent, il connaît aussi les hauteurs de neige. Dès qu'il a besoin de la stabilité sur un petit panneau, il utilise ce modèle numérique, obtient la hauteur critique et infère l'instabilité par comparaison avec la hauteur réelle. En fait, ceci est écrit avec SHIRKA en utilisant une méthode qui appelle une fonction Le-Lisp qui calcule la stabilité grâce à la formule ci-dessus. Par conséquent, ce modèle symbolique peut déterminer le petit panneau qui joue le rôle de décrocheur. Cette connaissance est ensuite exploitée pour réaliser une simulation symbolique du déclenchement.

Afin d'utiliser le programme AVAL d'écoulement de neige dense, un hydrogramme, c'est-à-dire l'expression du débit en fonction du temps au sommet de la zone d'écoulement, est nécessaire (figure 4.4).

Le résultat de la simulation symbolique peut permettre de construire cet hydrogramme. Pour chaque petit panneau déclenché, un chemin jusqu'au sommet de la zone d'écoulement est calculé et un temps de déclenchement déterminé. Ensuite, en considérant chaque petit panneau comme une plaque rigide, un modèle développé par Voellmy [VOEL55] donne le temps mis pour atteindre le point de contact des zones d'accumulation et d'écoulement et la vitesse de la plaque en ces points.

Il est ainsi possible d'obtenir un hydrogramme propre à chaque petit panneau. Les différents hydrogrammes sont ensuite ajoutés pour obtenir l'hydrogramme final.

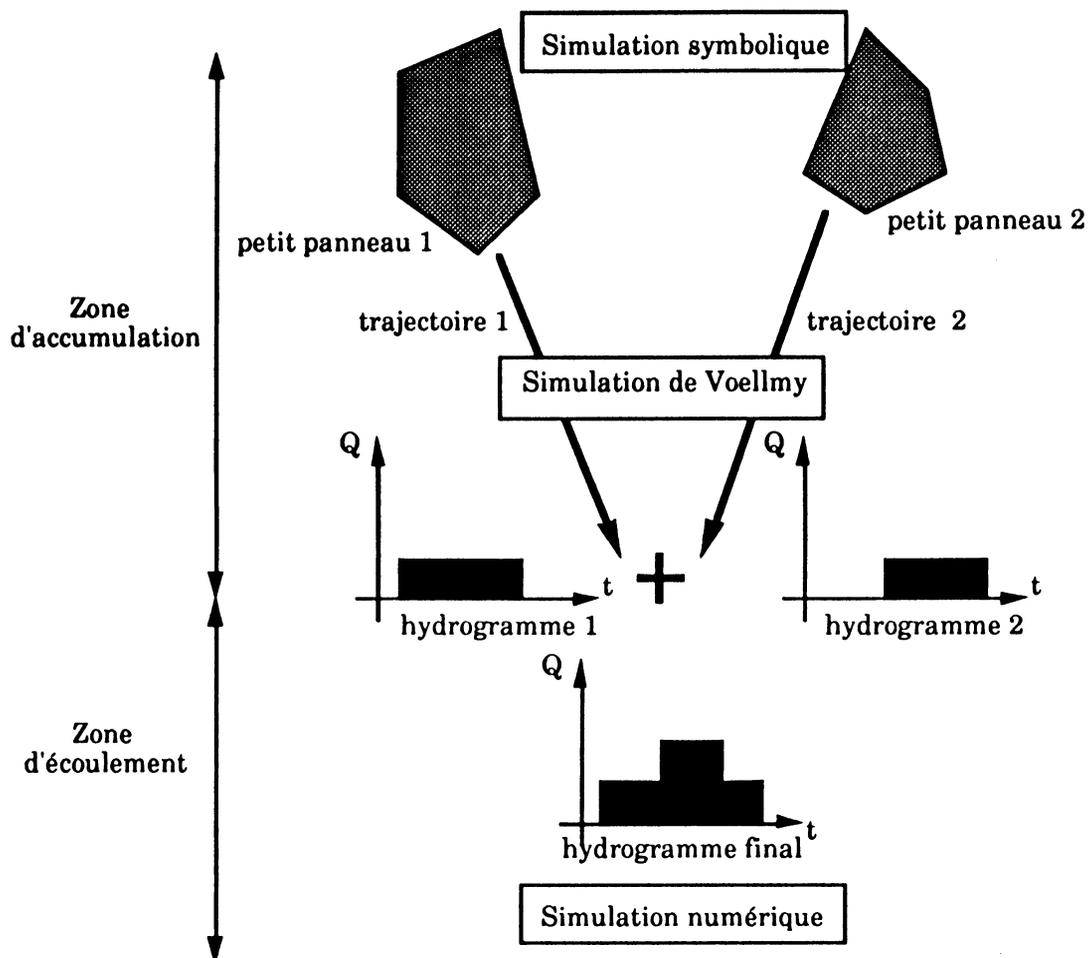


Figure 4.7 : Cette figure montre comment les résultats de la simulation symbolique peuvent être utilisés par le modèle de Voellmy pour fournir au programme AVAL un hydrogramme.

Comme il l'a déjà été écrit, le volume de l'avalanche est un paramètre important dans un programme comme AVAL. Pour l'illustrer, nous présentons deux analyses de la zone de départ correspondant à deux directions de vents différentes. Les masses des avalanches déclenchées varient dans un rapport de 1 à 3. Ceci sur le même site.

4.3.4. Contrôle de programme numérique

Les modèles numériques sont construits sous l'hypothèse que la précision du schéma numérique choisi pour obtenir la solution est suffisante.

Pour utiliser le programme AVAL de simulation des avalanches de neige dense, une décomposition du couloir est nécessaire. Des discontinuités de pente peuvent créer des instabilités numériques dans le calcul de l'écoulement. Dans AVAL, la précision et la stabilité du schéma numérique dépendent de la taille des mailles autour des irrégularités. Si la taille est trop grande, la solution obtenue semble numériquement correcte mais n'est pourtant pas physiquement significative. Un spécialiste est tout à fait capable, à l'aide de règles simples, de choisir une taille de maille bien adaptée.

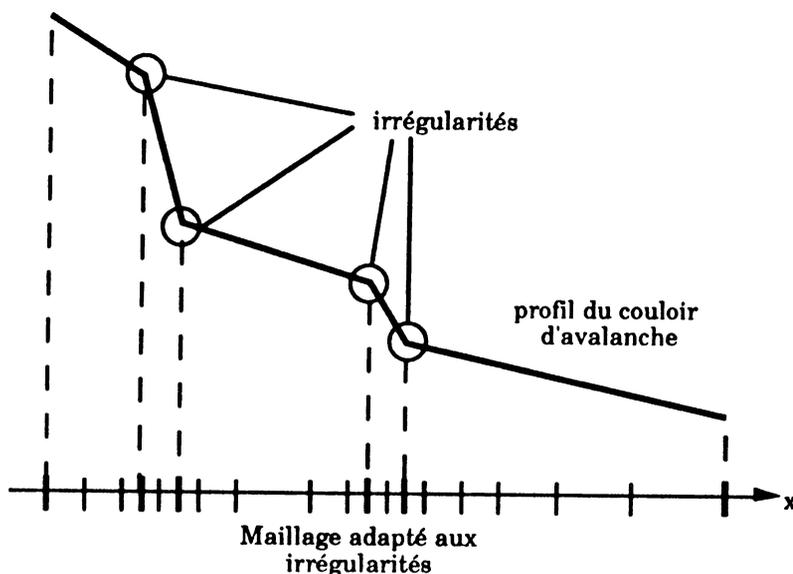


Figure 4.8 : La taille de maille doit être adaptée à la pente et aux irrégularités de celle-ci le long du couloir d'écoulement. Autour des irrégularités, ces mailles doivent être plus fines.

Le choix de la meilleure taille de maille⁵ peut être fait automatiquement par un système qui possède une description du terrain comparable à celle d'ELSA. Un module de raisonnement peut ainsi contrôler un programme numérique.

Cette description des objectifs d'ELSA met bien en évidence les problèmes rencontrés dans le développement d'un environnement de résolution de problèmes d'ingénierie adapté à un phénomène ayant une dimension spatiale. Cette approche peut être exploitée dans d'autres domaines. Le problème des congères et du transport de neige par le vent est par exemple très proche.

L'importance du rôle des experts, que ce soit pour fournir les modèles symboliques, exploiter des schémas numériques (il s'agit alors de l'expert numéricien), récupérer les résultats d'un modèle ou fournir des données à un autre modèle, exige une communication parfaite entre ELSA et ces experts.

4.4. COMMUNIQUER AVEC DES EXPERTS

Les objectifs d'ELSA définis, sa conception est donc maintenant examinée sous l'angle de la communication avec les experts. Leur étant destiné, ELSA a été construit avec leurs connaissances, leur vocabulaire, leurs images des phénomènes. Les objets de l'analyse sont donc ceux que manipulent les experts. L'interface utilisateur doit représenter ces mêmes objets sous la forme manipulée habituellement par les experts. Enfin, l'importance de l'espace conduit à une communauté des représentations de l'espace chez l'expert et dans ELSA.

4.4.1. Objets de l'analyse

Lors d'une analyse, un expert manipule un certain nombre d'entités qu'il a créées pour modéliser le problème. Ces objets doivent également exister dans ELSA. Il peut s'agir d'objets physiques ou temporels. Quelques exemples sont détaillés maintenant. Certains avaient été formalisés depuis longtemps, d'autres l'ont été (et le sont encore...) depuis le développement d'ELSA. A

⁵ qui aille...

chaque fois une partie de leur représentation sous forme d'objets SHIRKA est donnée.

L'objectif principal d'ELSA est l'aide à l'analyse d'un site. Celui-ci doit donc être présent dans la base de connaissances. Il l'est sous la forme d'une instance de la classe site :

```
{site
  sorte-de          = surface;
  commune           $un chaîne;
  departement       $un chaîne
                   $domaine "04" "05" "06" "38" "63" "64" "65"
                   "73" "74";

  massif            $un chaîne;
  altmax            $un reel;
  altmin            $un reel;
  limites           $liste-de point;
  antecedents       $un antecedents;
  scenario          $un scenario ;
  petits-panneaux  $liste-de petit-panneau;
  triangles         $liste-de triangle;
  aretes            $liste-de arete;
  declenchement     $liste-de depart }
```

Face à un site avalancheux, et plus exactement face à une zone d'accumulation, l'expert tente toujours de déterminer une structure, des frontières, des lignes caractéristiques. Ces lignes déterminent des entités appelées *petits panneaux* et qui, considérées comme homogènes vis-à-vis des critères de l'analyse (pente, exposition, végétation, géologie...) constituent l'unité de terrain sur laquelle vont s'appliquer les connaissances nivologiques de l'expert.

Les petits panneaux seront plus détaillés dans l'inférence de leur propriétés spatiales et dans leur construction au fil du chapitre suivant. Il s'agit en effet d'objets physiques sur lesquels le raisonnement spatial va intervenir.

Une avalanche naît de la rencontre entre un site et un scénario de déclenchement. Un scénario est constitué d'une situation initiale qui correspond à une répartition du manteau neigeux sur le site en début de scénario (une condition initiale) et une succession d'événements dont le dernier est toujours critique s'il s'agit d'un scénario de déclenchement. Les événements peuvent être progressifs (événements météo) ou soudains (passage de skieurs, d'engins ou d'animaux, chutes de pierres ou de séracs, explosion).

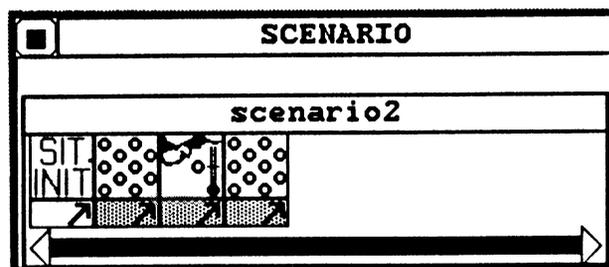


Figure 4.9 : Un exemple de scénario de déclenchement. La première case correspond à la situation initiale (SIT INIT), les cases suivantes aux différents événements qui constituent l'historique du scénario.

A titre d'exemple, les classes `scenario` et `evenement` sont représentées par les schémas suivants (pour la lisibilité, ils sont présentés de façon incomplète) :

```
{scenario
  sorte-de           = objet;
  situation-initiale $un situation;
  historique         $liste-de evenement;
  evenement-critique $un evenement-critique;
  decrocheur        $un petit-panneau;
  situation-pre-critique $un situation }

{evenement
  sorte-de           = objet;
  description        $un chaine;
  duree             $un duree;
  force-vent        $un entier
                  $domaine 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15;
  dir-descript      $un chaine
                  $domaine "nord" "est" "sud" "ouest" "nord-est"
                          "sud-est" "sud-ouest" "nord-ouest";
  direction-vent    $un vecteur;
  temperature       $un entier;
  precipitation-eq  $un entier}
```

A chaque instant du scénario, un petit panneau possède un profil stratigraphique de neige. Ce profil possède une ou plusieurs couches de neige considérées comme homogènes.

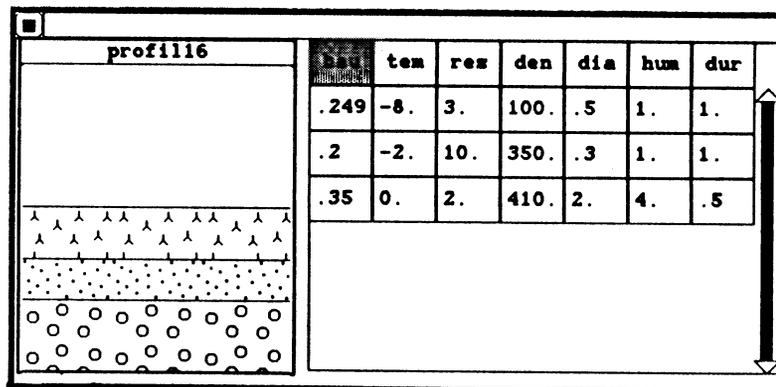


Figure 4.10 : Un exemple de profil stratigraphique

```
{profil
  sorte-de           = objet;
  couches            $liste-de couche;
  couches-faibles   $liste-de couche;
  plan-glissement   $un couche;
  masse-mob         $un reel)

{couche
  sorte-de           = objet;
  hauteur           $un reel;
  type              $un qualite;
  densite           $un reel;
  temperature       $un reel;
  humidite          $un reel;
  cohesion          $un reel;
  frottement        $un reel;
  resistance        $un reel;
  diamètre          $un reel )
```

Actuellement ELSA compte une trentaine de classes d'objets.

4.4.2. Interface utilisateur

Le système SHIRKA possède une interface utilisateur qui permet d'interroger la base de connaissances ou de manipuler des objets et qui est munie d'un langage de commande. La complexité des liens entre objets, la syntaxe, la formation des utilisateurs (il s'agit d'ingénieurs spécialistes d'avalanches et non pas d'informaticiens), ainsi que la nature essentiellement spatiale ou graphique des résultats, ont conduit au développement d'une interface graphique la plus conviviale possible.

L'ensemble de l'interface a été écrite à l'aide du logiciel AIDA de la société ILOG [ILOG2]. AIDA est un environnement de développement d'interfaces graphiques pour des applications écrites en Le-Lisp. L'existence d'AIDA et sa facilité d'utilisation et de portage (cf.4.5.1. Historique) ont joué un rôle dans le choix de l'ensemble SHIRKA-Le-Lisp pour le développement d'ELSA.

La méthodologie d'analyse des experts, présentée en 4.1, a été adaptée et modifiée. Elle est présente sous la forme de la barre des menus en haut de l'écran :

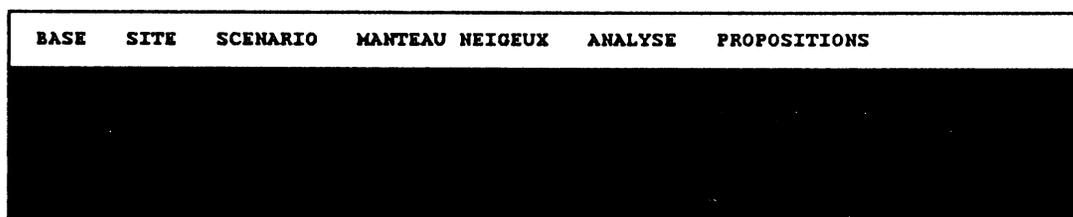


Figure 4.11 : La barre des menus du système ELSA.

Cette barre des menus respecte l'ordre des opérations que doit suivre l'expert lors d'une analyse (la première phase présentée en 4.1.2 n'est pas prise en compte). Ceci ne signifie pas qu'il faille les consulter dans cet ordre. Le fonctionnement en chaînage arrière du moteur d'inférence de SHIRKA permet à l'utilisateur de n'interroger que le dernier menu. Pour répondre, le moteur d'inférence va examiner des questions que l'utilisateur aurait pu poser en intervenant sur les menus précédents (il peut d'ailleurs le faire ensuite et détailler ainsi les étapes du raisonnement d'ELSA).

La barre des menus comprend six menus correspondant à une famille importante de tâches.

Les menus SITE, SCENARIO, MANTEAU NEIGEUX, ANALYSE et PROPOSITIONS offrent tous deux types d'articles. Les premiers sont relatifs aux objets de l'analyse de sites avalancheux. Les seconds sont relatifs aux méthodes utilisées pour donner des valeurs aux attributs de ces objets. La méthode peut être sélectionnée par le système ELSA grâce à une heuristique ; elle peut être sélectionnée par l'utilisateur ; elle peut être aussi neutralisée par le choix par l'utilisateur d'une valeur déterminée jusque là par une méthode. Pour chaque problème, le système ELSA présente une liste de méthodes dont les méthodes "décision utilisateur" et "requête à l'utilisateur" et indique la méthode utilisée pour obtenir les valeurs présentes à cet instant dans la base de connaissances.

Par exemple le menu ANALYSE est fait de deux listes. La première contient les items d'accès aux valeurs. Les items tels que VITESSES, DEBITS, HAUTEUR

fournissent ainsi les caractéristiques de l'écoulement. La seconde liste autorise l'utilisateur à accéder aux méthodes. En choisissant l'item `ECOULEMENT`, il peut consulter la liste des méthodes disponibles dans ELSA et en particulier la méthode courante active. Il peut aussi passer à une nouvelle méthode.

Le menu `BASE` propose les actions sur les bases de connaissances du système ELSA. Il s'agit de la gestion des bases associées à un site (créer une nouvelle base, ouvrir une base existante, fermer une base, enregistrer l'état d'une base ...), du contrôle des impressions et de l'abandon d'ELSA.

Le menu `SITE` offre à l'utilisateur l'ensemble des opérations lui permettant d'interroger le système ELSA sur le site à l'exclusion de ce qui dépend du scénario (les menus `SCENARIO`, `MANTEAU NEIGEUX`, `ANALYSE` se chargent de ces informations). Il s'agit des superficies des petits panneaux, de leur végétation, pente, exposition, de la forme des arêtes ... Pour ce qui concerne les méthodes d'inférence, il est possible de choisir différentes méthodes pour évaluer l'efficacité d'une arête.

Le menu `SCENARIO` propose à l'utilisateur de choisir, d'examiner ou de construire un nouveau scénario d'étude. Les méthodes de calcul de vent, d'ensoleillement, de transport de neige, de détermination de la qualité de la neige et de calcul des fréquences sont accessibles dans ce menu.

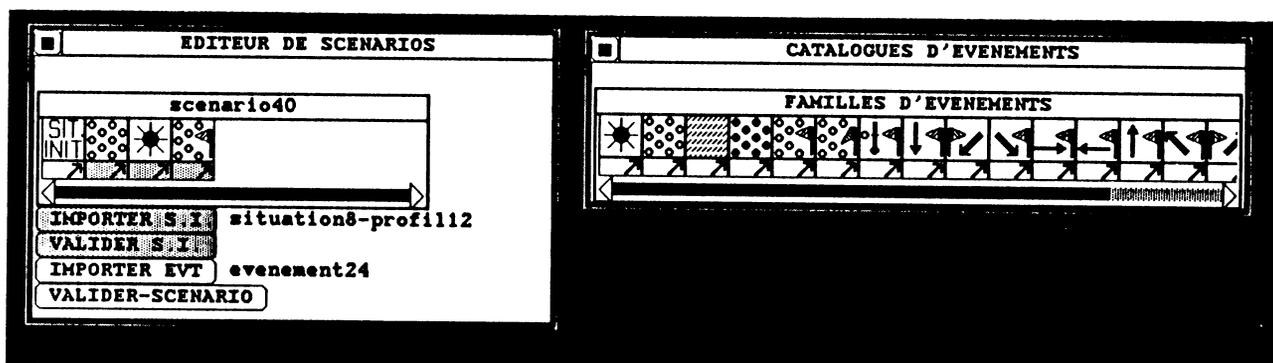


Figure 4.12 : Une session de création de scénario. Après avoir sélectionné une situation initiale parmi un catalogue existant et avoir validé ce choix, l'utilisateur sélectionne une succession d'événements qui vont constituer l'historique de ce scénario.

Le menu `MANTEAU NEIGEUX` permet à l'utilisateur de connaître les informations relatives à l'évolution et à la stabilité des profils de chaque petit panneau. A travers ce menu, il peut aussi accéder aux méthodes de construction de profils et de calcul des stabilités (cf. figure 4.13).

Le menu `ANALYSE` offre à l'utilisateur la possibilité d'étudier l'avalanche proprement dite. Ses caractéristiques (déclenchement, volume, vitesses, débits...) y sont présentées ainsi que les méthodes qui peuvent être utilisées pour les inférences.

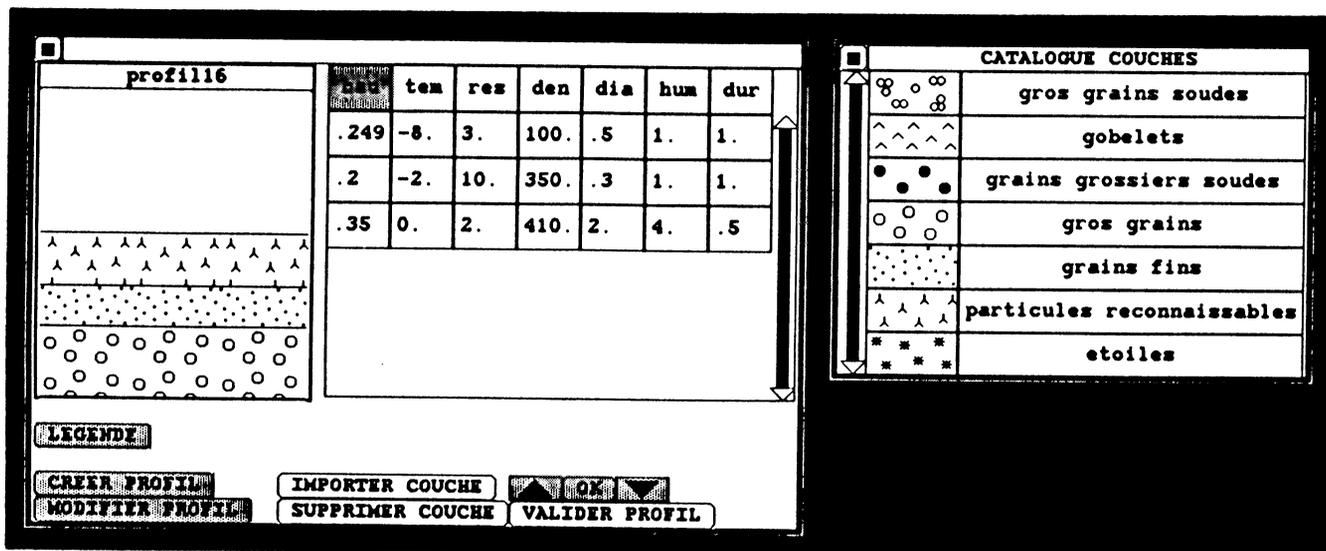


Figure 4.13 : Une session de construction de profil stratigraphique de neige. L'utilisateur choisit un type de qualité de neige pour chaque couche. ELSA lui fournit une hauteur et des caractéristiques physiques par défaut. L'utilisateur peut les modifier.

Le menu PROPOSITIONS présente à l'utilisateur les solutions de protection inférées par le système ELSA et les méthodes d'estimation de la valeur de l'équipement à protéger et des coûts de la protection. L'absence actuelle de développement sur ce menu a conduit à ne plus le présenter dans la suite de ce document.

L'interface proprement dite comprend deux zones. Une zone cartographique avec carte et légende actives où sont représentées toutes les informations ayant une existence spatiale et une zone à éditeurs où apparaissent les différents éditeurs (de scénarios, d'événements, de profils, de couches...). La zone cartographique permet à l'utilisateur de désigner les objets définis dans l'espace au système ELSA et réciproquement.

CONCLUSION

L'analyse de sites avalanches est une activité en pleine évolution qui, du fait de la complexité du phénomène étudié, fait appel à de nombreuses techniques d'analyse et de modélisation. Pour cette raison, il a été décidé de construire un environnement de résolution de problèmes avec une architecture de système à base de connaissances : le système ELSA.

Un environnement de résolution de problèmes est un système hautement interactif. Il doit communiquer avec ses utilisateurs, ici des experts. Cette communication se fait à travers les objets manipulés par ELSA qui doivent correspondre à ceux reconnus par les experts. La communication passe aussi par une interface utilisateur bien adaptée. Mais, dans le cas d'ELSA, la place prépondérante jouée par le raisonnement spatial oblige le système à partager une même représentation du terrain avec les spécialistes. Le chapitre suivant étudie précisément les caractéristiques de la base de connaissances spatiales du système ELSA.

5. ELSA : UNE BASE DE CONNAISSANCES SPATIALES

La communication la plus importante que le système ELSA doit entretenir avec l'expert porte sur le terrain et les propriétés spatiales du site. En effet, l'importance de la modélisation symbolique dans l'analyse de sites avalanches, c'est-à-dire de l'exploitation des connaissances de l'expert, exige une mise en commun de la représentation de l'espace. Des exemples ont été donnés qui mettent en évidence l'importance de notions comme la superposition de deux parties de terrain suivant la ligne de plus grande pente (cf. exemple 4.1), la position relative d'une partie de terrain et d'une arête pour une direction de vent donnée (cf. exemple 4.2), les positions relatives de deux parties de terrain (cf. exemple 4.4). Il a aussi été précisé que l'unité de terrain avec laquelle travaille l'expert s'appelle le petit panneau.

ELSA doit donc manipuler cette entité petit panneau et doit être capable de "comprendre" les différentes notions exploitées par l'expert. Le système à base de connaissances doit contenir des unités de l'univers du discours de l'expert (cf. définition du premier chapitre) ; il doit aussi pouvoir raisonner sur ces connaissances qui sont spatiales. C'est ici que deviennent nécessaires des capacités de raisonnement spatial.

Les deux premières sections de ce chapitre traitent de deux aspects de ce raisonnement spatial dans ELSA. La première développe les techniques utilisées pour inférer des propriétés spatiales pertinentes pour les petits panneaux à partir de leur définition et comment ces connaissances spatiales pertinentes sont utilisées pour l'analyse de sites. La deuxième section traite de la construction de ces petits panneaux. La logique eût voulu que ces deux thèmes soient étudiés dans un ordre différent mais la complexité des inférences et des problèmes est plus grande pour le deuxième thème. De plus l'utilité du second ne se trouve que dans le premier. Enfin, la dernière section illustre les problèmes de dynamique de la base de connaissances et en particulier les problèmes de performance, d'enregistrement, de maintien de la vérité et de nombre d'objets. Des résultats chiffrés sont donnés. Cette dernière section concerne surtout les connaissances spatiales mais elle donne aussi des résultats sur des analyses complètes.

5.1. INFÉRER DES PROPRIÉTÉS SPATIALES PERTINENTES

Les connaissances d'un expert s'appliquent sur les entités spatiales qu'il a créées et qu'il manipule. Plus exactement, il utilise ou complète des propriétés de ces entités. Il a déjà été dit que ces entités devaient être connues du système ELSA. Celui-ci reçoit donc les définitions de ces objets spatiaux, les petits panneaux par exemple (la partie 4.4 détaille le problème de la création de ces objets). Cependant, les propriétés de ces objets que l'expert utilise ne sont pas celles qui sont directement données à ELSA. Celui-ci doit donc inférer des propriétés plus pertinentes qui pourront, elles, être exploitées par les connaissances du spécialiste.

5.1.1. Des connaissances données aux connaissances pertinentes

Les connaissances spatiales manipulées par le système ELSA sont rassemblées dans des modèles. Ces modèles constituent une interprétation de l'objet du raisonnement qu'est la zone d'accumulation. Certains de ces modèles sont donnés à ELSA, d'autres, les plus pertinents pour l'analyse, sont inférés.

Connaissances données

Il s'agit des modèles constitués par les triangles, les petits panneaux et les arêtes.

Le modèle de base est celui des triangles dont la réunion constitue une approximation de la zone d'accumulation que les experts jugent satisfaisante. Chaque triangle est caractérisé par la donnée de ses trois sommets qui sont définis dans un repère cartésien, translaté du repère de la projection Lambert qui est à la source des cartes topographiques en France. La structure de définition est donc euclidienne, ce qui n'est pas étonnant : ces entités n'ont d'existence que dans l'espace de tous les jours qui demeure un espace euclidien dans la plupart des modèles de la physique...

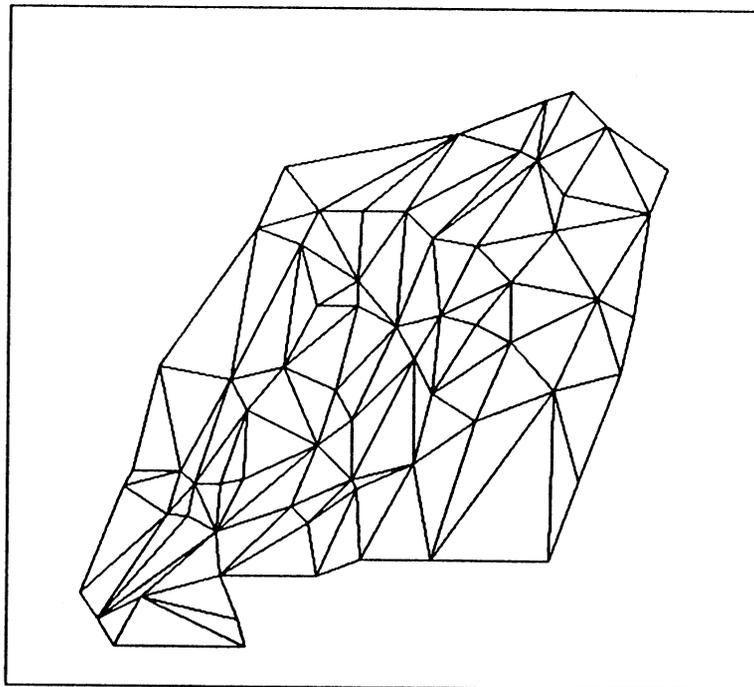


Figure 5.1: Une décomposition en triangles du site de Tende.

Toutes les entités manipulées par la maquette sont définies à partir de ces triangles. L'ensemble des triangles constituent une topologie de la zone

d'accumulation (en fait une topologie de fermés). Les composants des autres modèles seront donc les intersections ou les réunions de ces triangles ou de leur complémentaires.

Dans ELSA, les triangles sont choisis de façon telle que leurs côtés correspondent à des limites "naturelles" (fossés, arêtes, végétation, état de surface, exposition, rupture de pente, ligne de partage des eaux "significatives", etc). Cette condition va permettre la donnée du modèle des petits panneaux sous une forme simple.

Les petits panneaux sont des réunions de triangles réalisées de façon à avoir une végétation et un état de surface homogènes sur un petit panneau. Les arêtes, crêtes et les lignes de ruptures de pente sont par ailleurs exclues de l'intérieur d'un petit panneau. Le modèle des petits panneaux constitue pour ELSA, dans la phase d'analyse, une topologie *minimale* ; ELSA ne peut *couper* un petit panneau au cours de ses inférences. Aussi, en prévision de la prise en compte du vent, les distances aux arêtes ont été incorporées dans les définitions des triangles et des petits panneaux.

Un petit panneau est donné comme la réunion de certains triangles et affecté d'une rugosité correspondant à la végétation qui le couvre et à son état de surface ; il est, de plus, connexe. Les différents petits panneaux sont disjoints.

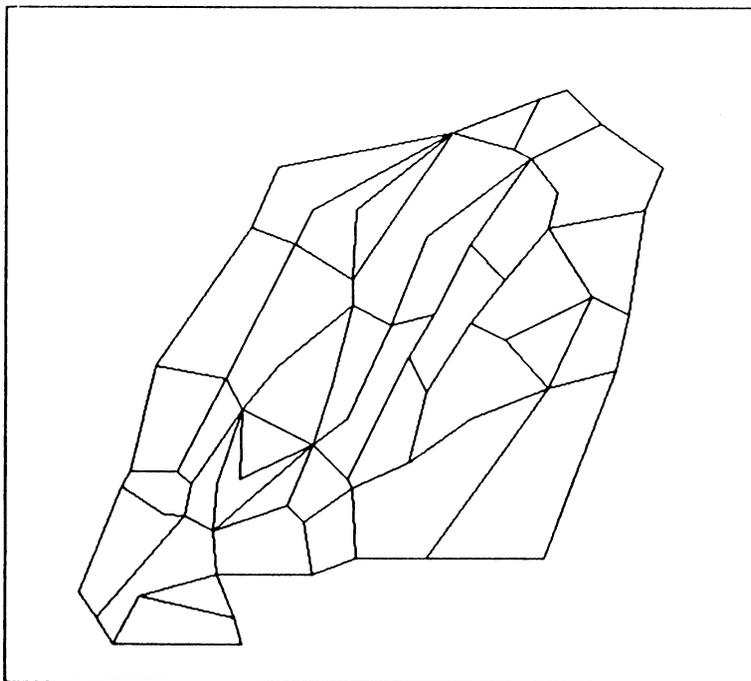


Figure 5.2 : Une décomposition en petits panneaux sur le site de Tende

Le troisième et dernier modèle donné à ELSA est celui des crêtes et des arêtes qui sont définies comme des listes de points rangés dans l'ordre dans lequel on les rencontre en suivant l'arête (cf.figure 5.3 ; le sens de parcours n'a aucune importance). Chaque arête se voit affectée d'une efficacité qui dépend

de sa forme, de sa position dans le site et, pour les arêtes limitrophes, du terrain qui les borde du côté extérieur. Là encore, ce modèle ne contient pas véritablement de connaissances spatiales explicites.

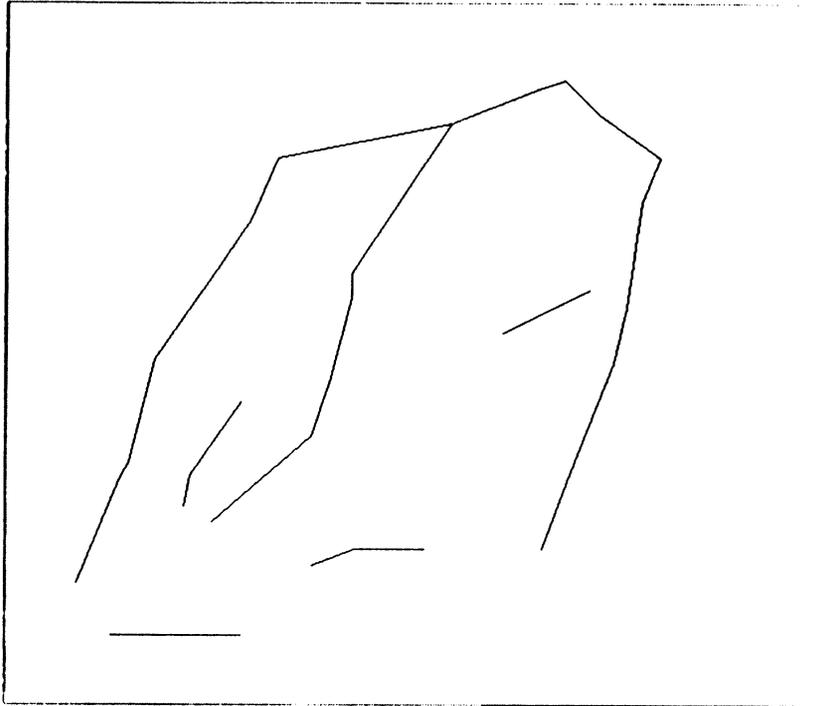


Figure 5.3 : Les arêtes sur le site de Tende

La section suivante reviendra sur la construction de ces modèles donnés.

Connaissances inférées

A partir de ces trois modèles de connaissances données, ELSA constitue des modèles de connaissances pertinentes qui lui permettront d'exploiter les connaissances de l'expert. Ces trois modèles, qui ne possèdent qu'une information relativement brute qui ne peut être utilisée telle quelle par les experts, se verront donc délaissés au profit de modèles plus pertinents aux yeux du nivologue.

Dans un premier temps, le système calcule, à partir du modèle numérique de terrain, certaines caractéristiques des triangles : superficie, vecteur normal, centre de gravité, pente, diamètre bi-dimensionnel et côtés du triangle (cf. figure 5.4). Ceci constitue un enrichissement du modèle euclidien initial. Les nouvelles connaissances sont encore euclidiennes sauf l'avant-dernière qui est *métrique* et la dernière qui est *topologique*. Le ou les triangles situés sous le triangle étudié sont également repérés grâce à une détermination de la ligne de plus grande pente. Il s'agit cette fois-ci de connaissance *vectorielle*.

ELSA exécute le même type de raisonnement pour déterminer le centre de gravité et le diamètre bi-dimensionnel des arêtes. Cette dernière information est là encore métrique.

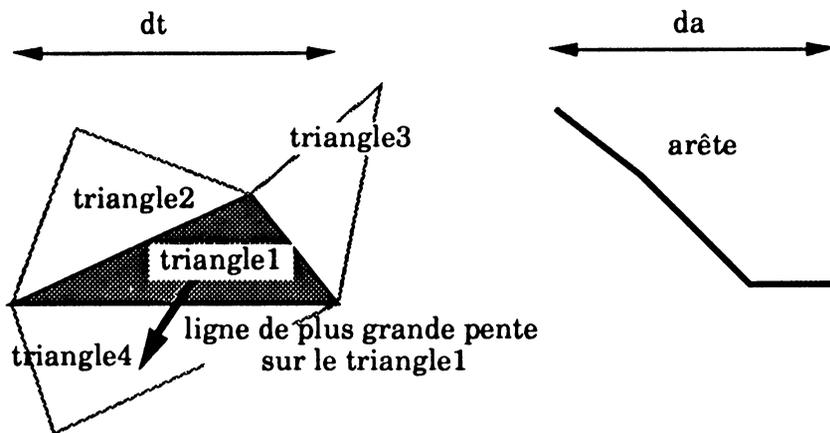


Figure 5.4 : Les triangles et les arêtes sont définis par la donnée des coordonnées de leurs sommets. Ces connaissances vectorielles et euclidiennes sont ensuite transformées en connaissances topologiques de contiguïté (le triangle1 est à côté des triangles 2, 3 et 4), métriques (le diamètre bi-dimensionnel vaut dt pour le triangle et da pour l'arête) ou vectorielles (le triangle1 se déverse sur le triangle4 suivant la ligne locale de plus grande pente).

En exploitant les connaissances obtenues pour les triangles et les arêtes, ELSA infère des propriétés nouvelles sur les petits panneaux. Le centre de gravité, la superficie, la pente, l'altitude moyenne, toutes connaissances *euclidiennes*, sont calculées.

Les petits panneaux qui se trouvent être arrosés par le petit panneau étudié sont également déterminés. Cette connaissance est *vectorielle*.

Utilisant ensuite les résultats sur les diamètres, ELSA détermine les petits panneaux qui se trouvent sous l'influence d'arêtes. A ce stade, seules interviennent les connaissances métriques. La distance utilisée n'est d'ailleurs pas la distance tri-dimensionnelle, mais la distance horizontale qui, selon les experts, rend mieux compte des phénomènes de dépôt et de balayage de la neige par le vent au voisinage d'arêtes. Le modèle euclidien permet aussi d'évaluer l'angle d'incidence du vent sur l'arête qui est un paramètre important pour estimer le rôle du vent (cf. figure 5.5).

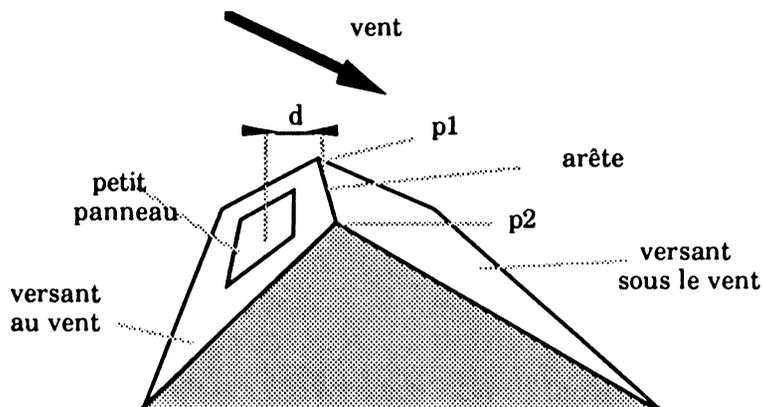


Figure 5.5 : Pour chaque arête, le système repère les petits panneaux qui sont dans sa zone d'influence. Il attribue à chaque petit panneau retenu le tronçon d'arête (segment $[p1, p2]$) qui le concerne, évalue l'incidence du vent sur ce tronçon, la distance horizontale d entre le petit panneau et l'arête. Le système distingue aussi les panneaux au vent et les panneaux sous le vent.

Des considérations, purement vectorielles elles, offrent ensuite la possibilité de distinguer les petits panneaux au vent des petits panneaux sous le vent. Ce modèle *vectoriel* est bâti, lui aussi, à partir du modèle euclidien.

Le modèle de juxtaposition entre petits panneaux est obtenu en exploitant le modèle topologique des triangles. Il est qualifié de *topologique* car la notion de connexité est une notion typiquement topologique. Le modèle des frontières est établi, en calculant les longueurs des frontières des petits panneaux juxtaposés. Il s'agit d'un modèle topologique et métrique. Le modèle topologique des petits panneaux est alors inféré de façon triviale. Il faut remarquer ici que la facilité avec laquelle ces inférences sont conduites vient de l'élimination des connaissances non pertinentes dans ce problème de connexité. La nature purement topologique du modèle est capitale.

Le système peut alors fournir la simulation du phénomène provoqué par le décrochement du petit panneau le moins stable (cf. 5.1.3. Utilisation des connaissances pertinentes).

5.1.2. Représentations et inférences spatiales dans ELSA

Les représentations centrées objet du style de SHIRKA semblent bien adaptées pour représenter les caractéristiques d'objets spatiaux. Le système ELSA utilise désormais cette représentation de connaissance.

Des objets

Quelques exemples d'inférence sont donnés ici. Ils ne sont pas discutés en détail, en particulier pour ce qui concerne les limitations du système SHIRKA tel qu'il existe actuellement. Cet aspect du problème sera évoqué dans le dernier chapitre.

Voici, à titre d'exemple, la classe `petit-panneau` et une de ses instances, `pp1` :

```
{petit-panneau
sorte-de = surface ;
superficie $un reel ;
c-gravite $un point ;
altitude $un reel ;
diametre2D $un reel ;
diametre3D $un reel ;
pente-% $un reel ;
contient $liste-de triangle ;
p-frontieres $liste-de petit-panneau;
a-cote $liste-de petit-panneau;
frontieres $liste-de frontiere ;
aretes-proches $liste-de arete ;
arrose $un petit-panneau }

{pp1
est-un = petit-panneau ;
superficie = 6850. ;
c-gravite = %point-552 ;
altitude = 1075. ;
diametre2D = 115. ;
diametre3D = 138. ;
pente-% = 68 ;
contient = tr2 tr93 ;
p-frontieres = po4 po6 po5 po1;
a-cote = pp2 pp3 ;
frontieres = %firt-589 %firt-590;
aretes-proches = ar1 ar 3 ar4 ar5 ;
arrose = pp3 }
```

Seuls sont indiqués les attributs spatiaux. Les valeurs de ces attributs sont soit de type simple (réel par exemple), soit de type objets (il peut alors s'agir d'objets initiaux acquis par la base ou d'objets construits). Le type de l'attribut `frontieres` met en évidence une lacune du modèle centré objet. Il est créé pour décrire la frontière entre le petit-panneau courant et ses voisins. Il est en effet impossible de représenter des associations entre trois objets (les deux petits-

panneaux et leur frontière commune) sans créer une classe spéciale. La classe présentée correspond au cas où un petit panneau n'en arrose qu'un.

Pour passer d'une représentation initiale, très souvent euclidienne, à une représentation plus pertinente, par exemple topologique, il faut pouvoir réaliser de très nombreuses inférences basées sur le filtrage et l'attachement procédural.

L'exemple d'instance présenté ci-dessus a, en effet, été inféré à partir de l'instance initiale suivante :

```
{pp1
  est-un      = petit-panneau ;
  contient   = tr2 tr93 }
```

Pour cela le moteur d'inférence de SHIRKA a exploité les facettes présentes dans le schéma de classe suivant :

```
{petit-panneau
  sorte-de    = surface;
  lui-meme    $var-nom p;
  superficie  $un reel
    $sib-exec {m-addition-reelle
      l-sup $var<- sup-tr ;
      res   $var-> superficie } ;
  c-gravite   $un point
    $sib-exec {m-pp-c-gravite
      l-sup      $var<- sup-tr ;
      l-c        $var<- c-tr
      c-g        $var-> c-gravite }
    $var-nom c-pp;
  altitude    $un reel
    $sib-exec {m-pp-moyenne-reelle
      l-sup      $var<- sup-tr ;
      l-val      $var<- alt-tr ;
      res        $var-> altitude } ;
  diametre2D  $un reel
    $sib-exec {m-pp-diametre2D
      l-c        $var<- c-tr ;
      c-g        $var<- c-gravity ;
      l-d2       $var<- d2-tr ;
      d2        $var-> diametre2D } ;
  pente-%     $un reel
    $sib-exec {m-pp-moyenne-reelle
      l-sup      $var<- sup-tr ;
      l-val      $var<- p-tr ;
      res        $var-> pente-% } ;
  contient    $liste-de {triangle
    c-gravite   $var-nom c-tr ;
    pente-%    $var-nom p-tr ;
    altitude    $var-nom alt-tr ;
    superficie  $var-nom sup-tr ;
    diametre2D  $var-nom d2-tr ;
    segment1    $var-nom s1 ;
    segment2    $var-nom s2 ;
    segment3    $var-nom s3 ;
    au-dessus   $var-nom tr-dessous }
  $var-nom tr-pp;
```

```

points-brd $liste-de point
           $var-nom p-front
           $sib-exec {m-pp-points-brd
                     l-s1      $var<- s1 ;
                     l-s2      $var<- s2 ;
                     l-s3      $var<- s3 ;
                     $var-> points-brd } ;
a-cote     $liste-de petit-panneau
           $sib-filtre
           {petit-panneau
            lui-meme   $var-> a-cote ;
            points-brd
            $a-verifier
            {p-pp-a-cote
             l-p1 $var-liste<- p-front ;
             l-p2 $var-liste<- points-brd }} ;
frontieres $liste-de frontiere
           $sib-exec {m-pp-frontieres
                     l-pp $var-liste<- a-cote ;
                     pp   $var<- p ;
                     l-p  $var-liste<- points-brd ;
                     l-front $var-> borders } ;
aretes-pro $liste-de arete
           $sib-filtre
           {arete
            lui-meme $var-> aretes-pro ;
            centre
            $a-verifier
            {p-pp-aretes-pro
             cpp   $var<- c-pp ;
             carete $var<- centre }} ;
au-dessus $liste-de petit-panneau
           $sib-filtre
           {petit-panneau
            lui-meme   $var-> au-dessus ;
            contient
            $a-verifier
            {p-pp-au-dessus
             l-tr-sous $var<-      tr-dessus ;
             l-tr-pp  $var-list<- tr-pp ;
             l-tr-cont $var-list<- contient }}}

```

Filtrage

Ce type d'inférence permet la détermination des valeurs d'un grand nombre d'attributs. Ainsi l'attribut `a-cote` de la classe `petit-panneau` peut-il être évalué par filtrage en utilisant l'attribut `p-frontieres`. Un `petit-panneau p1` est à côté d'un `petit-panneau p2` si leurs attributs `p-frontieres` ont au moins deux points distincts en commun, sommets d'un même triangle.

Il faut préciser que la régularité des relations peut être traitée par filtrage. La relation de juxtaposition entre triangles est par exemple symétrique. SHIRKA peut exploiter cette propriété de la façon suivante :

```

{triangle
 sorte-de = surface ;
 lui-meme $var-nom t ;
 a-cote   $un triangle
 $sib-filtre {triangle
              lui-meme   $var-> a-cote ;
              a-cote     $var<- t}}

```

Malheureusement, ce type de solution conduit à des bouclages qui interdisent son utilisation.

L'inférence des arêtes proches se fait aussi par filtrage sur les arêtes en calculant à chaque fois la distance avec le petit-panneau courant. Ce calcul se fait alors par attachement procédural.

La détermination du (ou des) petit-panneau (x) arrosé (s) par un petit panneau se fait aussi par filtrage.

Attachement procédural

Dans le cas des petits panneaux (cf la description du schéma de classe ci-dessus), comme dans celui des triangles, l'attachement procédural est très utilisé.

Ce mécanisme d'inférence est utilisé pour l'exploitation par calcul des connaissances euclidiennes. La superficie d'un petit-panneau, sa pente, ses diamètres (bi- ou tri-dimensionnels) ou le petit-panneau qu'il arrose se calculent grâce à des fonctions LISP liées aux méthodes.

Chaque appel à un attachement procédural conduit à la création d'un objet qui est en fait un élément du graphe de la fonction appelée ; et dans SHIRKA la connaissance est contenue dans la base d'objets.

5.1.3. Utilisation des connaissances pertinentes

Les connaissances spatiales pertinentes inférées, il devient possible de les exploiter à l'aide des connaissances des experts.

Prise en compte du vent

A l'aide des trois types de connaissances spatiales décrites au début de 4.3.1, le système ELSA estime, par un traitement symbolique, pour chaque petit panneau, sa capacité à être chargé ou balayé par le vent. Cette capacité se traduit par l'affectation d'un coefficient éolien. Ce coefficient éolien est défini, pour chaque petit panneau et pour chaque épisode de chute de neige, par le rapport entre la hauteur de neige sur le petit panneau et la hauteur de neige moyenne sur la zone d'accumulation.

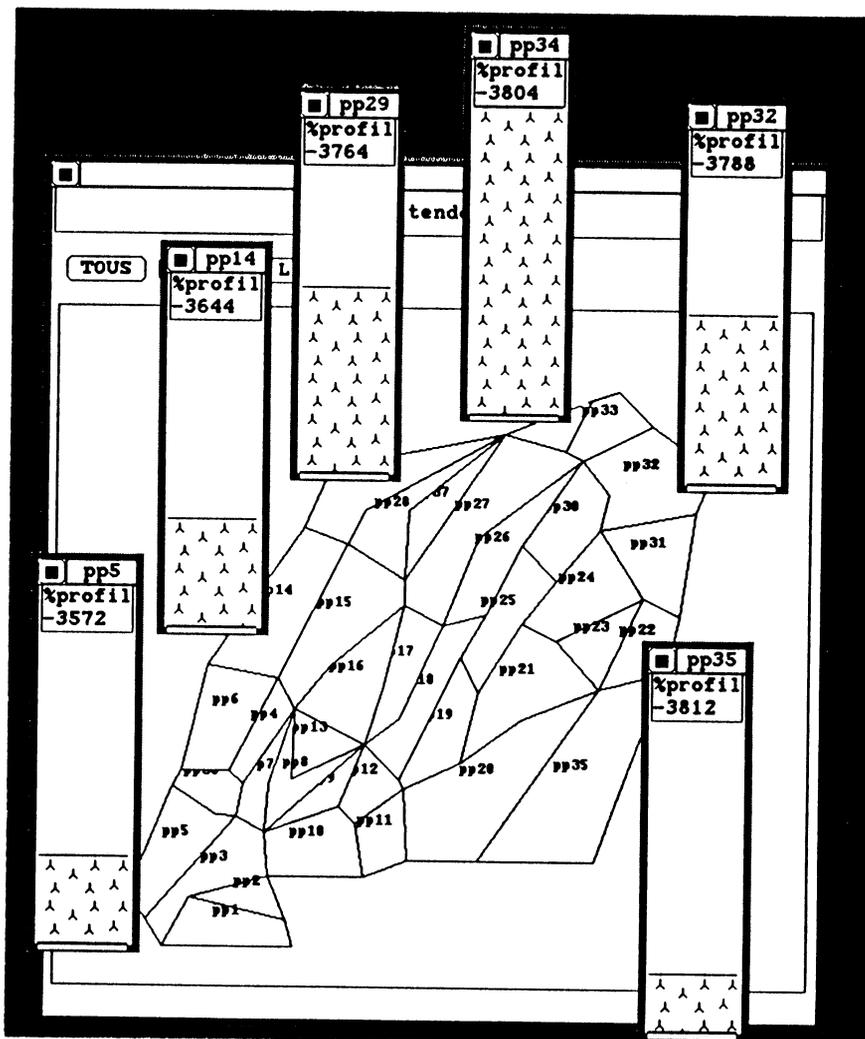
Ce coefficient éolien est obtenu à partir de quatre coefficients qui représentent chacun un paramètre important de l'analyse : *garête*, *gorientation*, *ginfluence* et *galtitude* qui correspondent respectivement à la forme de l'arête la plus proche, à l'incidence du vent sur cette arête, à la distance ou à la juxtaposition entre l'arête et le petit panneau et à l'altitude relative du petit panneau par rapport à la zone d'accumulation. Ces coefficients sont calculés en tenant compte de l'asymétrie entre le versant au vent et le versant sous le vent. Ils sont ensuite utilisés à l'aide d'une fonction linéaire dont les coefficients dépendent de l'efficacité du vent dans le transport de neige.

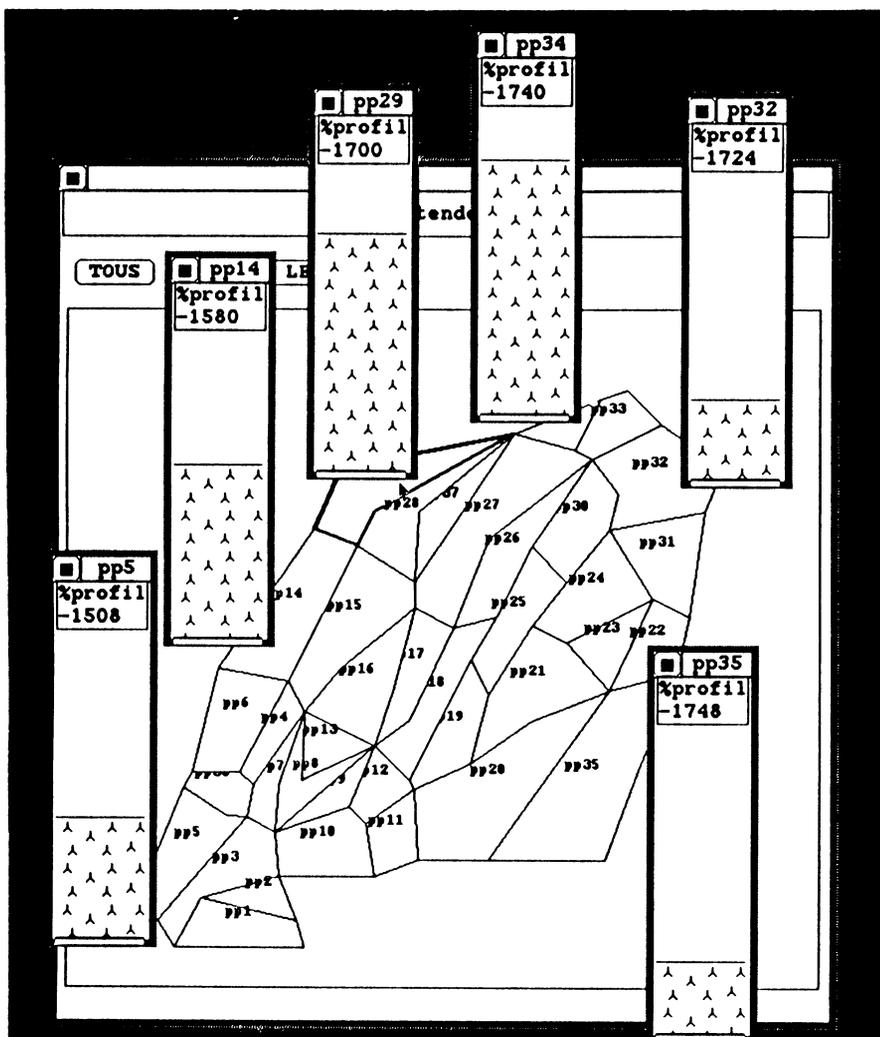
Le calcul de ces différents coefficients et, à travers eux, du coefficient éolien est le fruit d'une formalisation des connaissances empiriques des experts. Il ne peut être réalisé qu'à l'aide des connaissances pertinentes inférées par le système ELSA à partir des connaissances données.

Le système est alors capable d'obtenir, pour une précipitation donnée, la répartition en hauteur de neige du manteau neigeux sur la zone d'accumulation (cf. figure 5.6). Pouvant inférer, par ailleurs, la pente de chaque petit panneau, et connaissant sa végétation et son état de surface, il

peut déterminer la précipitation critique à partir de laquelle un petit panneau décroche spontanément et de façon nette (il faut éliminer les phénomènes de purge locale qui ne dépassent pas les limites d'un petit panneau et qui sont traités en augmentant la rugosité du petit panneau concerné).

Figure 5.6 (ci-dessous): Quelques profils inférés par ELSA pour deux scénarios qui ne diffèrent que par la seule direction du vent. Ces scénarios ne sont constitués que d'un seul événement (une chute de neige à saturation) et d'une situation initiale très simple (pas de neige sur le site). ELSA prend en compte le transport de neige par le vent. Dans le premier cas, le vent vient du nord ; dans le second de l'ouest. Les petits panneaux sous le vent des arêtes sont plus chargés que ceux qui sont au vent. Seules les conséquences quantitatives du transport sont intégrées c'est-à-dire que si les profils ont des hauteurs différentes, la neige est la même partout.





Propagation de la rupture

A ce stade, le raisonnement spatial a permis de déterminer les conditions minimales dans lesquelles la zone d'accumulation "commence à bouger". Cette précipitation critique, dite précipitation d'étude, permet de déterminer un modèle des hauteurs et des instabilités des petits panneaux. Il faut maintenant déterminer la partie de cette zone qui part et comment elle part ; ce phénomène donnera-t-il naissance à une avalanche et, si oui, quelle masse de neige mettra-t-elle en œuvre au sommet de la zone d'écoulement ? Le raisonnement spatial va permettre d'apporter des éléments de réponse.

Le déclenchement d'un petit panneau par un autre dépend de leur juxtaposition, de la longueur de leur frontière, et de l'incidence du panneau déclencheur sur le déclenché (plus simplement, de l'existence d'une ligne de plus grande pente définie sur le déclencheur et débouchant sur le déclenché).

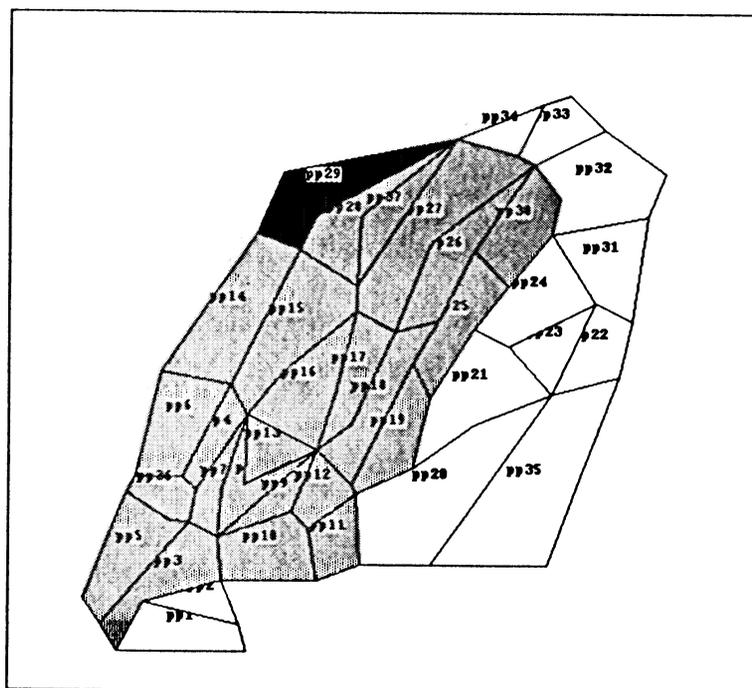
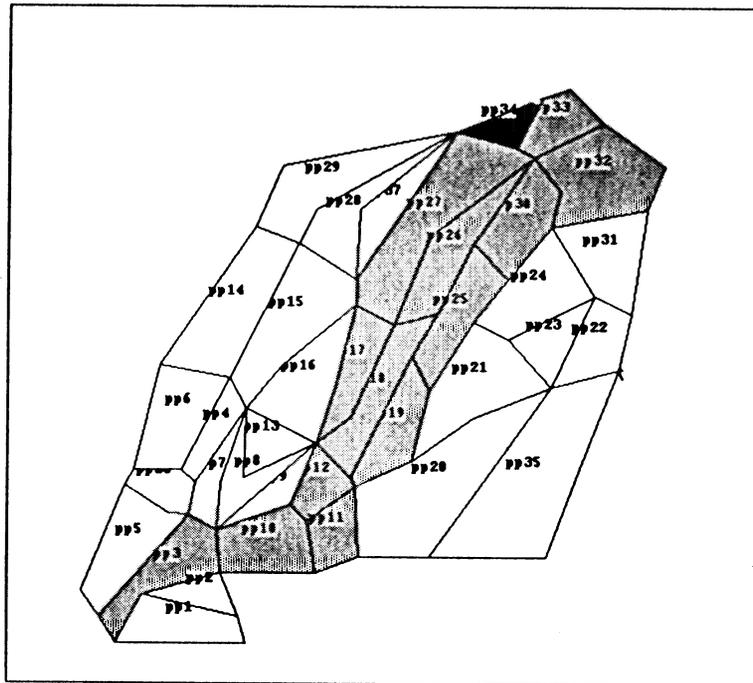
Le déclenchement est traité à l'aide d'un algorithme inclus dans un attachement procédural. Cet algorithme exploite les relations de juxtaposition et d'arrosage des petits panneaux ainsi que la stabilité et la masse de neige mobilisable de chaque petit panneau. Tout panneau mis en mouvement exerce des sollicitations sur ces voisins. Cette sollicitation dépend de la position

relative des deux petits panneaux. Un petit panneau n'agit pas de la même façon sur un petit panneau qu'il arrose ou qui lui est simplement juxtaposé.

De manière identique, selon la nature de la sollicitation, la masse de neige arrivante sera comparée à la masse en place, ou alors, la longueur de frontière sollicitante sera comparée au périmètre du petit panneau.

Ici encore une analyse rapide du déclenchement exige la présence dans la base des connaissances pertinentes qui ont été inférées par ELSA.

Figure 5.7 (ci dessous): Les déclenchements inférés par ELSA pour les scénarios de l'exemple précédent.



Changement d'hypothèses

Le TMS est utilisé pour modifier le scénario sans avoir besoin de recharger la description du site. En effet, un simple changement de valeur pour l'attribut scénario de l'instance `tende de site`, conduit à l'invalidation des valeurs d'attributs correspondant au déclenchement. Une nouvelle analyse peut être demandée. Le TMS permet de ne réinférer que les connaissances qui avaient été invalidées. C'est ainsi que les deux scénarios de la figure 5.6 sont analysés l'un derrière l'autre avec une invalidation des connaissances après le changement de scénario.

Il est clair que la gestion des hypothèses est entièrement à la charge de l'utilisateur qui doit compléter la base pour lui permettre de raisonner (par exemple, fournir un scénario). Le système ne peut gérer plusieurs hypothèses ou contextes hypothétiques, c'est-à-dire qu'il ne peut posséder simultanément, dans sa base, les résultats d'analyses correspondant à des scénarios différents. Un autre type de TMS devrait alors être utilisé, plus proche des ATMS (Assumptions-based Truth Maintenance System) [CORD87] ou du CP-TMS (Contexts Propagation Truth Maintenance System) [EUZE90].

5.2. CONSTRUIRE UNE DECOMPOSITION DU TERRAIN

Ce qui précède concernait l'inférence des propriétés spatiales de trois familles d'objets : les triangles, les petits panneaux et les arêtes. Ils étaient supposés définis par ailleurs. C'est en effet ainsi que le système ELSA a été développé. Pourtant, la décomposition du terrain est une des étapes les plus importantes de l'analyse de sites parce que toute l'analyse en dépend. Il eût donc été regrettable de ne pas l'intégrer dans le système ELSA. De plus, il a déjà été montré (chapitre 2) que la construction d'une topologie est une des formes les plus intéressantes de raisonnement spatial. Il est donc logique d'y consacrer une partie du chapitre qui traite d'ELSA et du raisonnement spatial en particulier.

5.2.1. Données disponibles et résultats désirés

L'objectif recherché dans cette construction est l'obtention d'une décomposition d'une zone d'accumulation en petits panneaux tels qu'ils ont pu être définis dans ce qui précède. L'existence des triangles n'est en effet intéressante que par le support topologique et géométrique qu'ils offrent pour l'inférence et le calcul de la plupart des propriétés spatiales des petits panneaux. Ce choix sera d'ailleurs étudié plus en détail dans ce qui suit.

Informations disponibles

Il faut tout d'abord présenter les documents ou informations de base qui sont disponibles. Il s'agit des descriptions de la topographie et de la végétation. Les plus courantes et les plus précises sont des cartes de courbes de niveau. Il y a également les cartes de végétation et les cartes d'arêtes.

Certains de ces documents existent sur support papier, d'autres sur support informatique. En France, l'IGN est engagé dans une politique de construction de bases de données topographiques et géographiques qui offriront l'ensemble des informations disponibles sous une forme numérisée [IGN90]. D'autres pays ont déjà effectué la numérisation complète des courbes de niveaux de leur territoire. En Norvège, par exemple, cette opération a permis le

développement du système TERMOS et son application aux risques naturels [TOPP86].

En l'absence de documents déjà numérisés, une numérisation est possible. Contrairement à une opinion très répandue, cette opération ne constitue pas une tâche importante. Avec un matériel bien adapté (table à numériser et logiciel de saisie) et du personnel expérimenté, l'ensemble des données de base nécessaire à l'analyse d'un site de taille moyenne peut être saisi en une demi-journée de travail.

D'autres cartes pourraient être utilisées qui correspondent au produit d'une partie de l'analyse, mais la volonté d'automatiser et de formaliser cette opération de construction incite à travailler avec les données les plus "objectives" possibles : courbes de niveau, contours de végétation, dessins d'arêtes et de thalwegs.

Description retenue

Comme il l'a déjà été signalé, la décomposition de la zone d'accumulation en petits panneaux est nécessaire pour permettre au système ELSA et à l'expert de communiquer à propos du terrain.

Le problème qui s'est posé lors du développement du système ELSA concernait le choix de la structure sur laquelle les petits panneaux seraient définis. En effet le système ELSA doit avoir accès aux propriétés spatiales de ces petits panneaux. Ceux-ci doivent donc être définis dans l'espace. Il a été choisi de les définir à partir d'objets géométriques simples : des triangles. Par trois points définis dans l'espace, ne passe en effet qu'un triangle. De plus les calculs géométriques sur les triangles sont assez simples. Ils constituent une topologie de fermés qui est à la base de toute la représentation du terrain. Ceci conduit, par ailleurs, à des fermés assez simples. En effet les triangles sont les simplexes en dimension 2, qui est la dimension de la surface de terrain, même si elle est définie dans un espace de dimension 3 (cf. [EGEN&90]). Les inférences topologiques seront donc faciles à réaliser.

Une alternative s'est alors présentée. La décomposition en triangles doit-elle être bâtie sur un découpage régulier ou sur un découpage irrégulier ?

Si chacun des détails du terrain doit être représenté dans une grille régulière (qu'il s'agisse d'ailleurs d'une maille triangulaire ou d'une maille carrée, hexagonale, etc.), la taille des mailles de cette grille doit être réduite afin d'approcher au mieux les formes du terrain naturel. Malheureusement, il est des zones où le terrain est peu complexe. De nombreuses mailles sont alors répétées identiques à elles-mêmes. L'utilisation des "quadtree" [SAME90] est une réponse à cette diminution de la taille des mailles. Le problème des formes demeure cependant car les quadtrees ont une forme fixe.

L'utilisation de triangles irréguliers est une bonne réponse aux problèmes de taille et de forme. Elle permet d'approcher les lignes significatives sans conduire à une explosion du nombre de triangles. Cette approche a été finalement retenue. D'autres applications manipulant le terrain naturel ont choisi également cette solution. C'est le cas de la synthèse d'image pour les simulateurs de vol ou des modèles de terrain.

5.2.2. Différentes solutions

Différentes solutions se présentent selon la façon dont l'automatisation est traitée.

La première solution consiste à construire cette représentation en petits panneaux à la main. Cette solution a pour avantage de laisser à l'opérateur la maîtrise, donc la responsabilité, de tâches importantes. Cependant, la construction d'un système à base de connaissances ayant pour objectif une augmentation de la rapidité et de la qualité des analyses, cette phase d'acquisition manuelle constituerait un goulot d'étranglement insupportable sauf dans une phase d'ébauche du système.

La deuxième consiste à automatiser tout le processus à l'aide de programmes procéduraux qui se voient fournir des données disponibles. Il n'y a pas moyen, alors, d'accéder à des explications sur les résultats et on ne peut modifier des valeurs sans relancer toute la procédure.

La troisième consiste à construire un premier système à base de connaissances dont les connaissances données sont les données des programmes procéduraux précédents. Une partie des connaissances inférées sont alors fournies au deuxième système à base de connaissances en tant que connaissances acquises. Il est ainsi possible d'expliquer l'existence d'un petit panneau ou d'une frontière ou de supprimer certains de ces éléments. Néanmoins l'existence de deux systèmes conduit à des problèmes d'interface.

La quatrième, enfin, consiste à ne construire qu'un seul système à base de connaissances possédant, parmi ses connaissances données, celles du premier système à base de connaissances. Les problèmes d'interface disparaissent, la construction des petits panneaux se fait parallèlement à l'analyse et les capacités d'explication et de modification peuvent être utilisées systématiquement.

5.2.3. Construction manuelle

La construction manuelle fut la première adoptée au cours du développement du système ELSA. En particulier, toutes les représentations du site de Tende qui figurent jusqu'ici dans ce rapport sont issues d'un travail réalisé à la main par un expert.

Il va de soi que cette solution laisse une grosse part du travail à l'utilisateur. Si les experts les plus expérimentés sont tout à fait capables d'assumer ce travail, un utilisateur néophyte rencontrera des difficultés dans la construction de ces petits panneaux.

Cette solution a été abandonnée pour des raisons d'automatisation des tâches. Cependant, les résultats qu'elle donnait restent l'objectif à atteindre. La solution qui va être présentée dans la sous-section suivante ne l'atteint que partiellement, ce qui justifie la tentative présentée dans la sous-section 5.2.5.

Il est toutefois intéressant d'étudier de plus près les raisonnements suivis par l'expert dans cette phase de délimitation des petits panneaux.

5.2.4. Construction procédurale

Afin de construire un système opérationnel, un outil de construction de représentation du terrain a été développé. En effet, cette représentation joue un rôle capital dans l'ensemble du projet. Il s'agit, à partir des courbes de niveau (figure 5.8), des arêtes (figure 5.9), des thalwegs (figure 5.10) et de la végétation (figure 5.11), de créer une triangulation et les petits panneaux. L'approche informatique choisie est celle d'une programmation procédurale classique en Le-Lisp.

Procédure de construction

Il ne s'agit pas de décrire les algorithmes utilisés mais d'esquisser l'approche qui a été retenue pour la construction des petits panneaux. Cette procédure n'a aucune prétention. Les compétences nécessaires à la réalisation d'un programme efficace et optimisé dépassent le cadre de cette thèse et celles de son auteur et rejoignent celles de la recherche opérationnelle [TALO87].

Plutôt que de décrire la procédure de construction, il a paru préférable de la détailler dans les légendes des figures 5.8 à 5.14.

Des essais ont été faits sur le site de Tende sur lequel le projet ELSA a débuté (cf figures 5.8 à 5.14) mais aussi sur le site de Drayre qui constitue le site d'essai principal (cf figure 5.15 et paragraphe suivant).

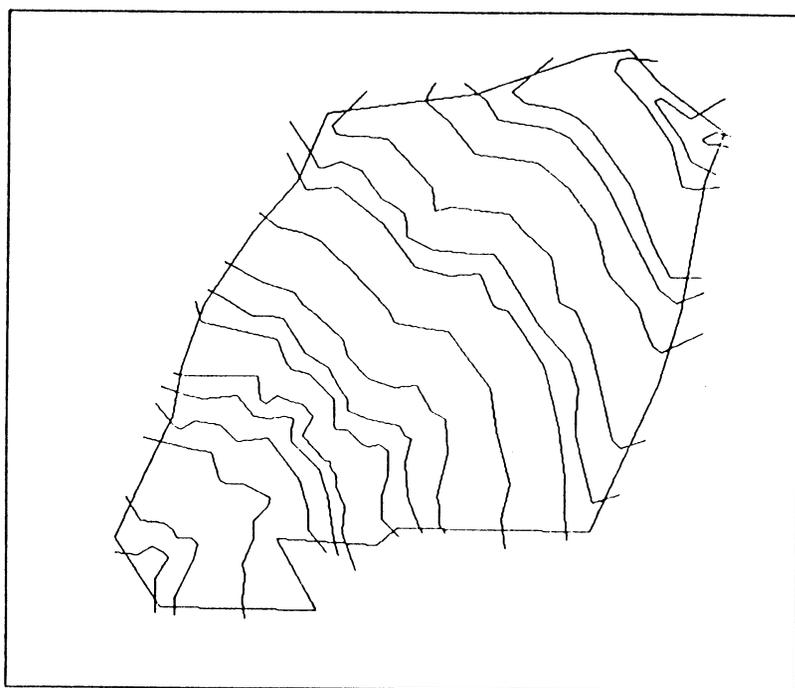


Figure 5.8 : Courbes de niveau numérisées pour la construction de la représentation du site de Tende. Les limites dépassent le bassin versant. Comme ce sera le cas également pour les arêtes, les thalwegs et les zones de végétation, les points hors limites sont éliminés. Il s'agit de l'opération communément appelée "clipping".

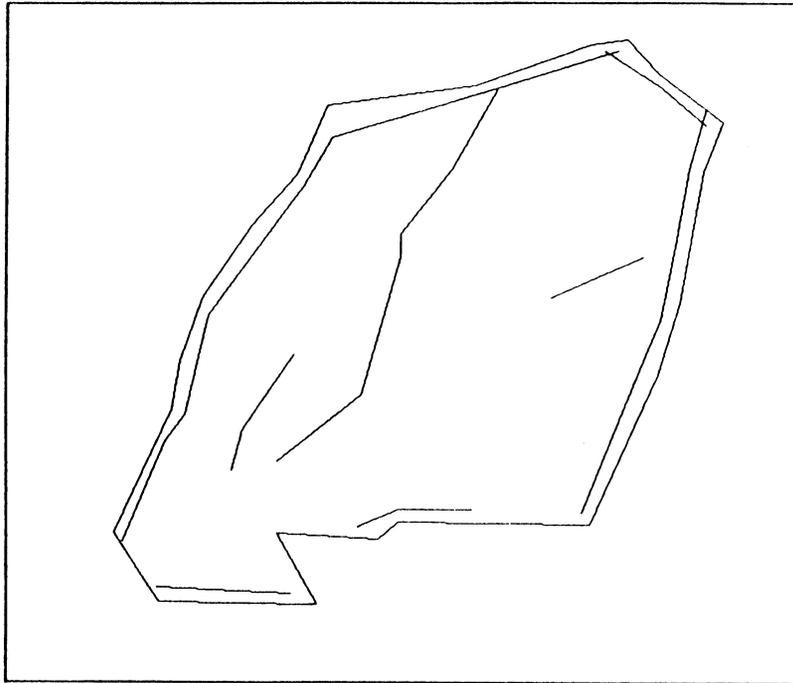


Figure 5.9 : Arêtes du site de Tende telles qu'elles sont numérisées.

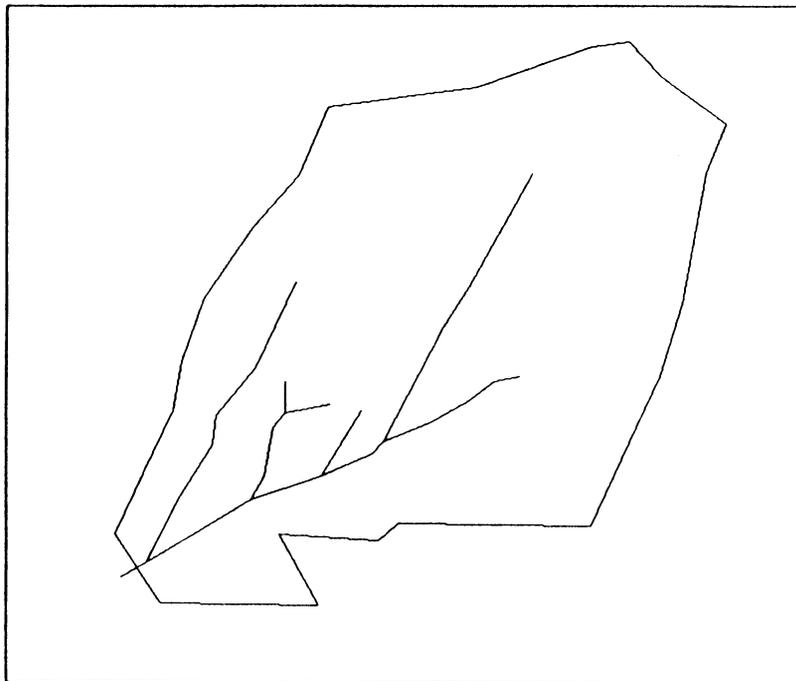


Figure 5.10 : Thalwegs du site de Tende tels qu'ils sont numérisés.

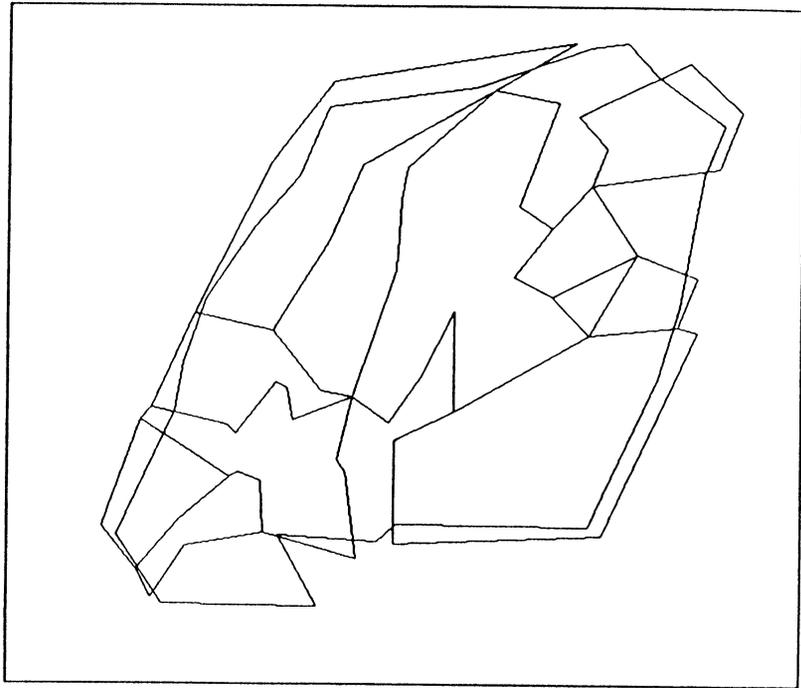


Figure 5.11 : Zones de végétation sur le site de Tende. Les intersections des thalwegs, arêtes et contours de végétation (appelées lignes de terrain) avec les courbes sont déterminées. Désormais les courbes de niveau et les lignes de terrain ne seront plus considérées qu'à travers les lignes polygonales basées sur ces intersections. A cet instant de la construction, les intersections très proches sont confondues.

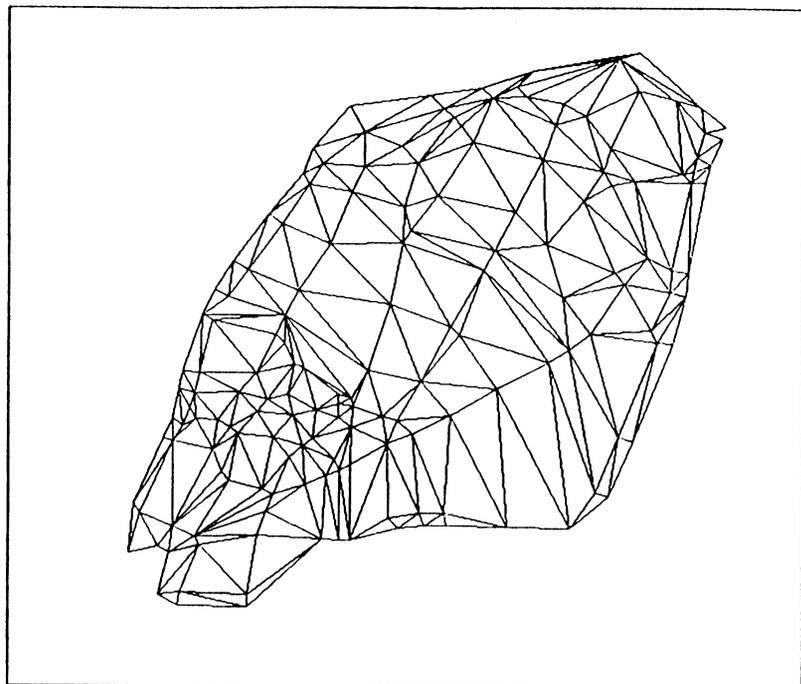


Figure 5.12 : Une triangulation sur la zone d'accumulation du site de Tende. La triangulation s'effectue entre deux courbes de niveau. Un triangle s'appuie sur des courbes de niveau ou sur des lignes de terrain. Les triangles sont ensuite renseignés quant à leur végétation.

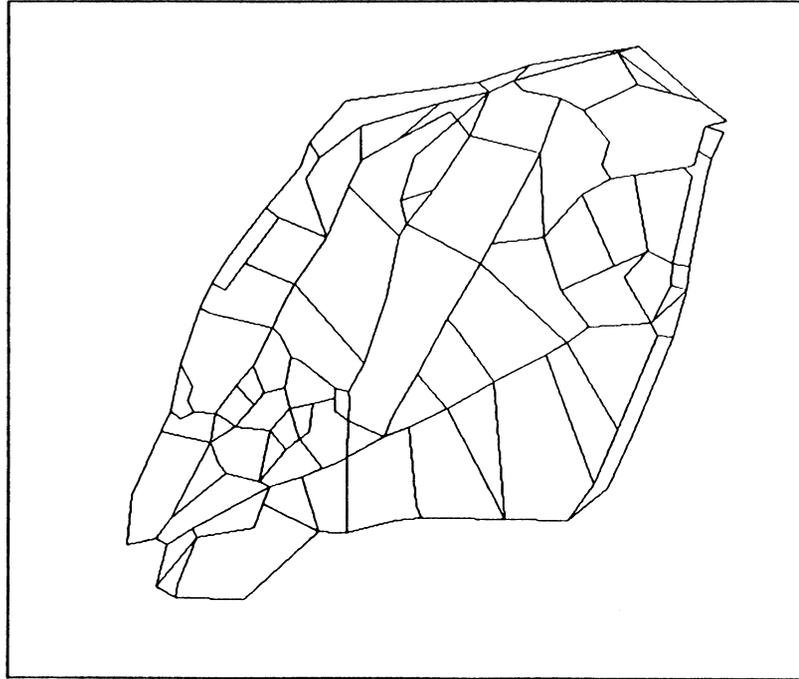


Figure 5.13: Une décomposition en petits panneaux sur le site de Tende. Les petits panneaux sont construits en deux phases. La première consiste à créer des entités très homogènes (en pente et orientation) et qui ne soient pas traversées par des arêtes ou des thalwegs. Dans une deuxième phase, le programme tente une incorporation des entités les plus réduites en surface en respectant les lignes de terrain importantes ainsi que des critères de géométrie (convexité globale des petits panneaux).

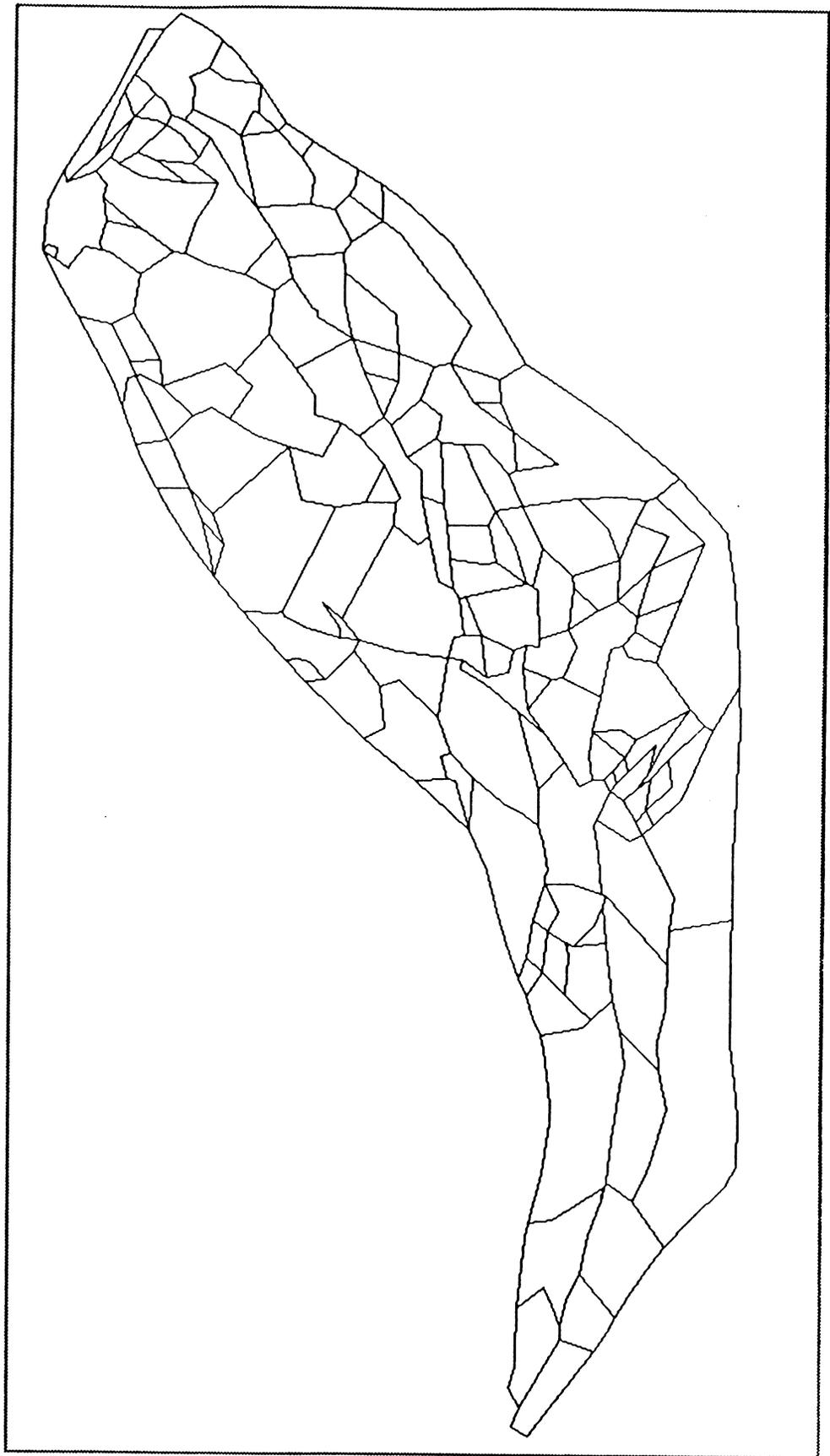


Figure 5.14 : Une décomposition en petits panneaux sur le site de Drayre (commune de Vaujany dans l'Isère).

Par rapport à une construction faite à la main par l'expert, le nombre de petits panneaux est plus élevé. En effet, pour prendre en compte tous les cas, les critères doivent être assez serrés et ils conduisent à distinguer des zones que l'expert confond. Il est possible de réduire ce nombre mais c'est alors au détriment de la qualité de la décomposition : les petits panneaux sont alors moins homogènes ou leur forme de plus en plus allongée. La décomposition est beaucoup moins pertinente.

5.2.5. Construction déclarative

Il s'agit de l'objectif final mais des capacités particulières sont nécessaires (voir le dernier chapitre). Il ne s'agit pas de décrire ici, dans le détail, la procédure de construction de la représentation de terrain à l'aide d'un système à base de connaissances mais de mettre en évidence l'intérêt qu'une telle solution présenterait.

Modélisation du raisonnement

Le raisonnement qui aboutit à la construction des triangles puis des petits panneaux peut être décomposé. D'une part, il s'agit de construire les triangles, d'autre part d'assembler les petits panneaux.

Le premier raisonnement s'apparente au premier type de construction de topologie (cf chapitre 2). Il s'agit de trouver les fermés d'une topologie obtenue par intersection de topologies déjà existantes (végétation, sols, courbes de niveaux) ou produites par la complétion de fragments de topologie (réseaux partiels de thalwegs et d'arêtes). Il faut ensuite générer un maillage triangulaire en introduisant éventuellement des points nouveaux.

Le deuxième raisonnement consiste, à partir des triangles qui viennent d'être construits, à créer les petits panneaux. La nature des frontières intervient ainsi que la valeur de certaines propriétés sur les triangles. Il s'agit alors d'appauvrir une topologie.

Ces deux phases correspondent donc à trois opérations de création de topologie telles qu'elles ont pu être décrites dans le chapitre 2.

Intérêt de la construction déclarative

Il est facile de s'apercevoir qu'une analyse de site par ELSA telle qu'elle a été décrite auparavant repose essentiellement sur la décomposition du terrain en petits panneaux. Cette décomposition dépend bien entendu des lignes caractéristiques du terrain. L'importance de ces lignes peut varier dans le temps ou selon les experts. Et pourtant, l'utilisateur d'ELSA ne peut intervenir sur elles sauf à les modifier et à relancer la construction puis l'analyse.

L'incorporation du raisonnement qui conduit à construire les petits panneaux au sein du système à base de connaissances permettrait à l'expert d'intervenir sur la décomposition en petits panneaux. Approfondir un thalweg, modifier le contour de zones de végétation, modifier des mouvements de terrain, "couper ou planter des arbres" sont des opérations que l'expert désire simuler pour étudier leur influence sur les avalanches d'un site de la même façon qu'il change les conditions météorologiques dans lesquelles il désire étudier le site. Le plus souvent, ces interventions conduisent à modifier les contours des petits panneaux et, partant, à remettre en cause l'opération de construction des ces petits panneaux. Si cette opération est réalisée au sein du système à base de connaissances, le TMS assure une gestion de la dynamique

qui prend en compte, comme cela a été expliqué au chapitre 3, la localité de l'invalidation. Celle-ci permet des gains de temps considérables et autorise l'expert à rester devant la machine pendant cette remise à jour. Cette fonctionnalité est très importante dans le contexte d'un environnement de résolution de problèmes.

5.3. DYNAMIQUE DE LA BASE DE CONNAISSANCES

Le chapitre 3 a évoqué différents aspects de la dynamique des bases de connaissances spatiales. Cette section montre l'importance qu'ils ont dans une application réelle.

5.3.1. Enregistrement

La taille de la base de connaissances et, surtout, la fréquence d'utilisation de certaines connaissances exigent une forme d'enregistrement. En effet, certaines connaissances sont assez longues à inférer ou sont utilisées très souvent au cours d'inférences complexes.

Exemple 5.1 : Le centre de gravité d'une arête est nécessaire pour savoir si elle a de l'influence sur chaque petit panneau. Il en va de même pour les petits panneaux qui se trouvent à côté d'un autre petit panneau. L'inférence des arêtes proches d'un petit panneau se faisant par filtrage des arêtes, à chaque fois, le centre de gravité est recalculé. Ceci se traduit par des temps de calcul importants et par la création de nombreux objets (à chaque fois un nouveau centre est construit ainsi que de nouvelles instances des attachements procéduraux).

Exemple 5.2 : Compte tenu des méthodes d'inférence présentées ci-dessus, des temps de calcul pris par le système ELSA, en Le-Lisp *interprété* sur un SUN 3-110 (12 Mo de mémoire centrale, configuration Le-Lisp à 6,3 Mo) pour réaliser certaines inférences sans faire d'enregistrement sont présentés⁶. Le jeu d'inférences exécutées est le suivant :

```
Shirka : val? pp25 a-cote
Shirka : val? pp25 a-cote
Shirka : val? pp23 a-cote
Shirka : val? pp23 a-cote
Shirka : val? pp21 a-cote
Shirka : val? pp1 a-cote
Shirka : val? pp1 a-cote
```

Le temps de calcul CPU estimé est de 3970 s, dont 695 s pour la première requête. Après un nouveau jeu d'inférence :

```
Shirka : val? pp34 aretes-proches
Shirka : val? pp34 aretes-proches
Shirka : val? pp33 aretes-proches
Shirka : val? pp32 aretes-proches
Shirka : val? pp1 aretes-proches
Shirka : val? pp2 aretes-proches
```

qui exige 273 s, la base de connaissances prend l'aspect suivant (figure 5.15) :

⁶ En Le-Lisp compilé, les temps de réponse sont nettement plus courts...

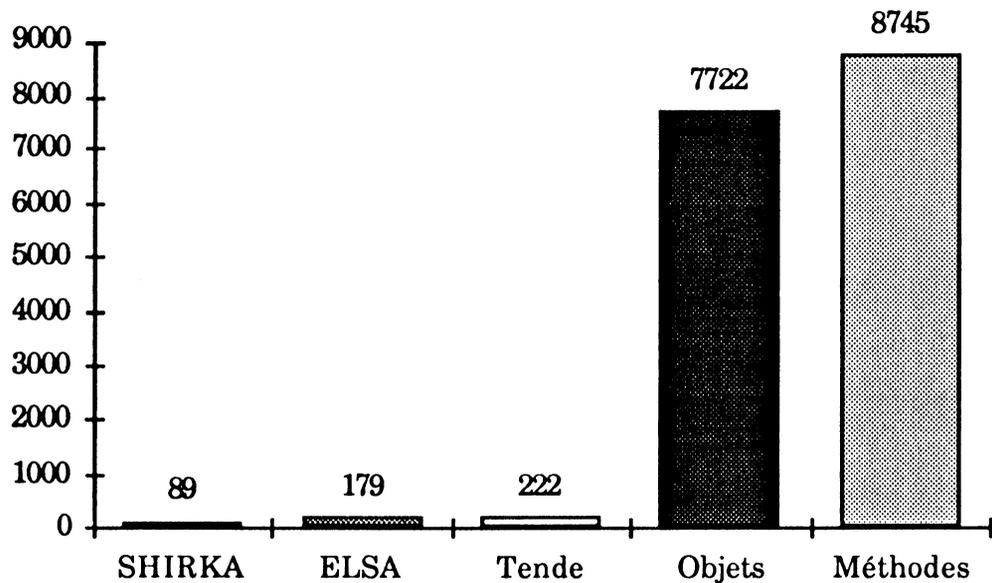


Figure 5.15 : Les objets dans la base après le jeu d'inférences. Les trois premières colonnes correspondent aux objets propres respectivement à SHIRKA, ELSA et ceux qui correspondent à une description initiale du site de Tende. La colonne Objets contient les objets créés lors de ces inférences. La dernière colonne correspond aux instances des attachements procéduraux.

Il faut préciser que la répétition de certaines requêtes ne fait que mettre en évidence la nécessité absolue d'enregistrer les résultats. L'utilisateur est en effet susceptible de reposer une question. De plus, la fréquence d'utilisation d'un grand nombre de résultats intermédiaires pose le problème de l'enregistrement.

Sur une base de connaissances de taille réduite, ceci ne pose pas réellement de problèmes. Malheureusement, comme le montrent les exemples, ELSA possède une base de connaissances qui devient très vite volumineuse et ceci en grande partie à cause de l'absence d'enregistrement. ELSA fournit un bon exemple de ce qui était présenté dans le chapitre 3 à propos de l'enregistrement (cf. 3.2.3)

Cet enregistrement, ainsi apparu nécessaire, est assuré par les fonctions de "caching" ou d'enregistrement du TMS de SHIRKA [EUZE87]. Toute valeur d'attribut inférée est enregistrée ainsi que la justification de l'inférence correspondante. Si une nouvelle requête concerne le même attribut, la valeur enregistrée est restituée immédiatement.

Exemple 5.3 : Avec l'enregistrement, les temps de calcul sont diminués de façon importante. Le premier jeu d'inférences demande 268 s (au lieu de 3970 s) et le deuxième 38 s (au lieu de 273 s). La première inférence ne demande plus que 226 s. La base de connaissances voit sa taille demeurer "raisonnable" ; ainsi, après le deuxième jeu d'inférences, elle a l'aspect suivant :

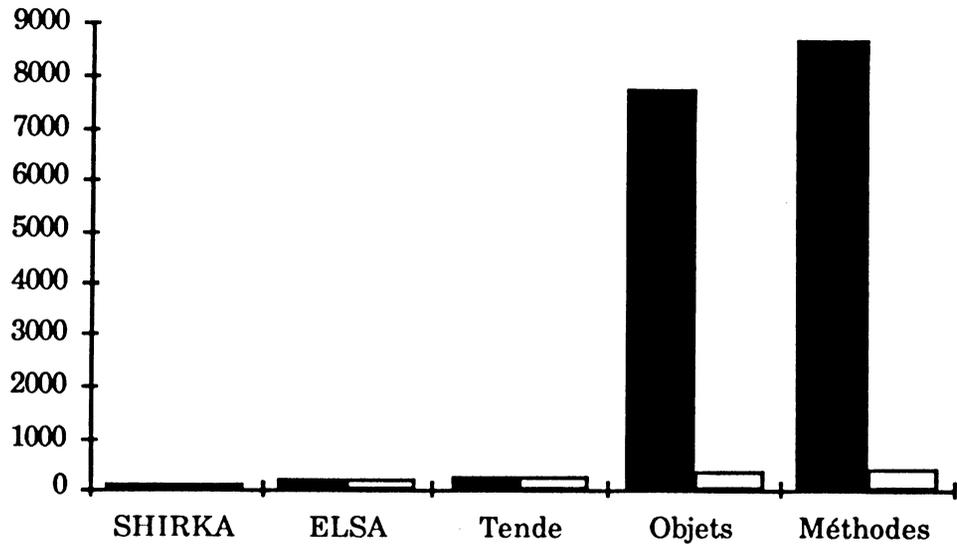


Figure 5.16: en noir le cas précédent, en blanc le cas avec enregistrement.

5.3.2. Maintien de la vérité

Cependant, si le gain en temps de réponse est important, la contrepartie de cet enregistrement est la croissance de la base de connaissances. En effet, le TMS de SHIRKA crée de nombreux objet instances de la classe justification. Cette croissance demeure cependant très modeste en comparaison de l'explosion de la base à laquelle conduit l'absence d'enregistrement.

Exemple 5.4 : L'utilisation du TMS conduit à la création d'un grand nombre d'objets non significatifs et à des temps de calcul légèrement accrus. L'augmentation des temps de calcul est liée à la construction des justifications. Elle reste faible : 319 s (au lieu de 268 s) pour le premier jeu d'inférences, 44 s au lieu de 38 s pour le second.

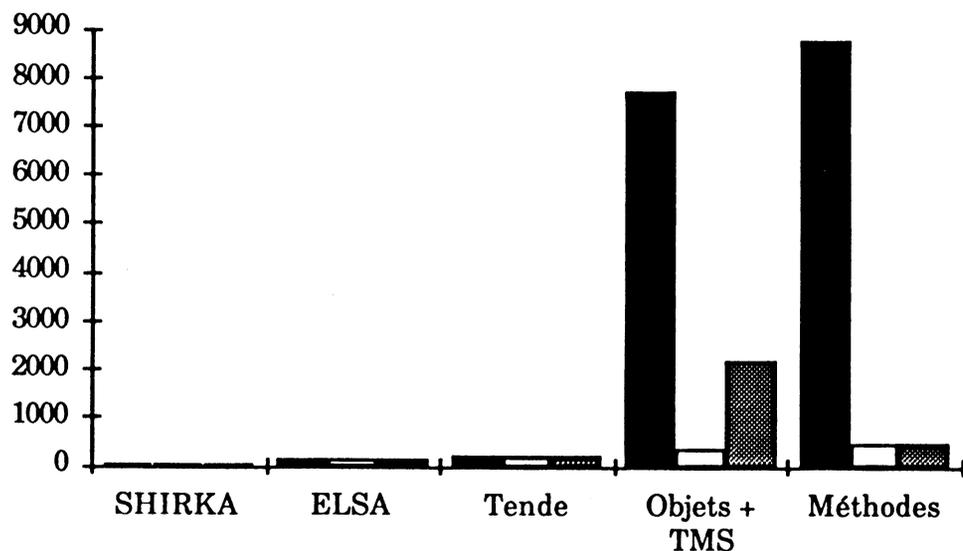


Figure 5.17 : Elle indique les objets qui ont été créés par le TMS pour les inférences des deux exemples précédents. En noir le rappel sans enregistrement, en blanc avec un enregistrement simple et en gris l'enregistrement avec TMS.

5.3.3. Evolution de la base au cours d'une analyse

Cette sous-section ne concerne plus les connaissances spatiales mais les connaissances liées à l'analyse de site proprement dite. Au cours d'une session d'ELSA, le nombre d'objets devient vite important. A coté des objets physiques (une centaine de triangles, une quarantaine de petits panneaux pour le site de Tende, une vingtaine de scénarios disponibles, autant d'événements, une quarantaine de profils de neige...) existent des objets relations (petits panneaux ventés, enneigés) et les objets instances des méthodes appelées.

Pour une session, par exemple au cours d'une analyse, si les objets propres à SHIRKA et au système ELSA sont moins d'une centaine, la simple description spatiale complète d'un site comme celui de Tende et de quelques scénarios exige près de 900 objets (sont exclues ici les instances des méthodes qui ont permis l'inférence des propriétés spatiales). Après une analyse, 201 nouveaux objets ont été créés ainsi que 322 instances de méthodes et ceci toutes inférences spatiales exclues. De plus le TMS est à l'origine d'un grand nombre d'objets (figure 5.18).

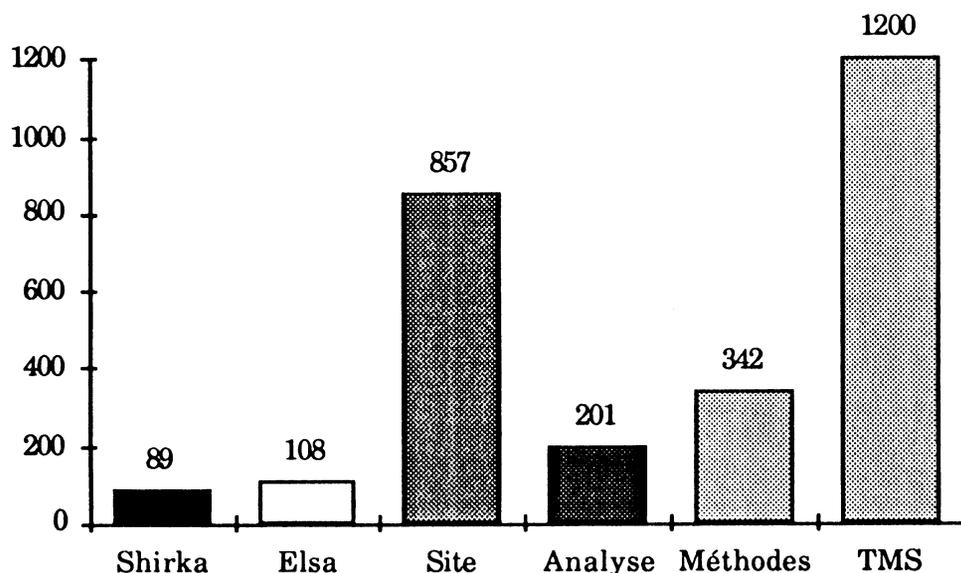


Figure 5.18 : Les objets dans la base de connaissances après une première analyse. La dernière colonne correspond aux objets créés par le TMS.

Après un changement de scénario, une nouvelle analyse est réalisée (celle dont les résultats sont présentés sur la deuxième carte de la figure 5.7). Le nombre des objets a augmenté, en particulier celui des objets créés par le TMS (figure 5.19). Cela dit, le TMS a permis de ne réinférer que les connaissances qui avaient été invalidées.

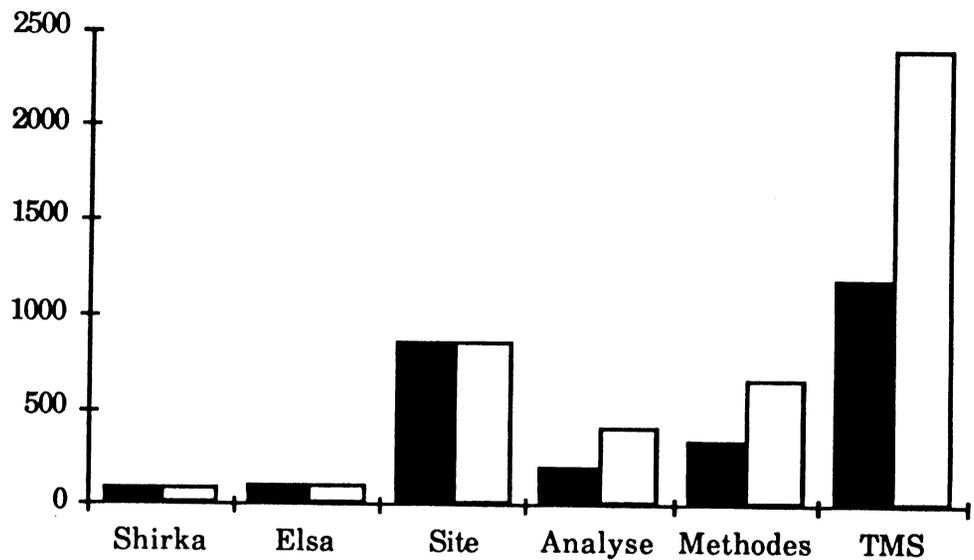


Figure 5.19 : La base de connaissances après une deuxième analyse (en noir la situation après la première inférence ; en blanc après la seconde).

5.4. SITUATION ACTUELLE DU PROJET ELSA ET PERSPECTIVES

Cette courte sous-section a pour but de présenter les aspects matériels du système ELSA ainsi que son avenir.

5.4.1. Historique

Le projet ELSA a vu sa première implantation réalisée en PROLOG sur Macintosh XL (alias Lisa) de la société APPLE. Celle-ci a été réalisée entre février et mai 1987. Une partie des problèmes présentés ci-dessus ont alors été rencontrés : explosion des objets physiques dans la base, traitement impropre de l'enregistrement, absence d'environnement de développement riche et confortable.

Après un portage (toujours en PROLOG) sur une machine SUN 3/110, il a été décidé de réécrire le système en SHIRKA sur cette même machine. Les raisons sont multiples : intérêt pour une représentation centrée objet naturellement adaptée aux objets spatiaux, prise en compte facile de méthodes d'inférence différentes, présence d'un TMS, richesse et portabilité de l'environnement Le-Lisp... Le langage est naturellement Le-Lisp et le logiciel AIDA a été utilisé pour l'interface graphique. La réécriture en SHIRKA (sans TMS) a été réalisée au printemps 1988 et a été suivie des premiers développements de l'interface graphique. Une adaptation sur SHIRKA-TMS a été conduite à l'automne 1988.

Le développement du constructeur procédural de représentation de terrain a commencé avec l'année 1989 et a été mené de front avec celui de l'interface graphique pour se terminer à l'automne 1989.

Des essais de construction et d'analyse sur d'autres sites ont alors commencé et se sont poursuivis par le portage d'ELSA au CEMAGREF (cf ci-dessous).

Le développement a été entièrement réalisé par l'auteur à l'exception de la deuxième partie de l'interface graphique réalisée par des stagiaires en DESS.

L'ensemble des programmes en Le-Lisp représente environ 810 ko (dont 585 ko pour ELSA et 225 ko pour le module de construction). Les bases d'objets SHIRKA représentent 46 ko (sans les bases associées à des sites, beaucoup plus volumineuses).

5.4.2. Etat actuel

Le système ELSA a été jusqu'à présent développé au sein du laboratoire ARTEMIS sur des machines SUN. Or il se trouve que le CEMAGREF est équipé de machines DEC (Digital Equipment Corporation). Un portage a été réalisé qui permet désormais d'utiliser le système ELSA dans les locaux du CEMAGREF.

Ce portage était relativement délicat car les systèmes d'exploitation sont différents (UNIX "SUN-OS" chez SUN et VMS chez DEC). Néanmoins, la portabilité du langage Le-Lisp et l'utilisation, dans les deux cas, du système graphique standard X-Windows (appelé Dec-Windows chez DEC) ont facilité les choses.

5.4.3. Perspectives

L'exploitation du système ELSA va débiter au CEMAGREF avec la fin de cette thèse. Dans un premier temps elle se fera sur des sites de référence puis, petit à petit, sur des sites sur lesquels des opérations de conseil seront en cours. A cette occasion la base de connaissances devra être complétée.

Trois domaines en particulier présentent des insuffisances :

- Les types de scénarios à traiter restent trop limités. Il est nécessaire de pouvoir étudier un nombre plus important de scénarios en prenant en compte l'évolution du manteau neigeux.

- Ceci doit conduire à une prise en compte de profils plus complexes. Les précipitations, le transport de neige par le vent créent des strates au comportement très diversifié. Les modèles de rupture et de propagation doivent en tenir compte.

- Le couplage avec des modèles numériques d'écoulement doit être assuré.

La reprise du système ELSA doit permettre de réduire les temps de réponse et d'éliminer le plus grand nombre de comportements anormaux. Une réécriture de certains modules devra certainement être assurée.

La validation des outils de simulation symbolique ainsi que celle de la méthodologie sera conduite dans le cadre d'opérations de conseil réelles d'une part et d'opérations de validation pure sur des sites de référence (dont le site de Drayre à Vaujany).

Le développement à plus long terme d'ELSA concernera l'incorporation des opérations de construction de la représentation de terrain dans le système à base de connaissances ainsi que le couplage avec le système CROCCUS de modélisation des évolutions du manteau neigeux [COLE&89].

CONCLUSION

Le développement du système ELSA a montré que les idées développées dans les chapitres 2 et 3 pouvaient être utilisées sur une application réelle. En particulier, les formes de raisonnement spatial mises en évidence au chapitre 2 jouent ou joueront un rôle très important dans ELSA. C'est d'ores et déjà le cas pour le raisonnement qui consiste à enrichir des représentations de l'espace peu pertinentes. Ce sera le cas pour le raisonnement qui consiste à construire une topologie d'un espace donné. Les aspects dynamiques sont également illustrés à travers l'utilisation du TMS et le rôle essentiel qu'il joue : sans lui l'enregistrement de connaissances intermédiaires est illicite et, sans cet enregistrement, les temps de réponse deviennent prohibitifs. Le TMS offre également une plus grande souplesse puisqu'il permet à l'utilisateur d'émettre des hypothèses et de revenir sur elles.

ELSA partage des caractéristiques avec d'autres applications. En d'autres termes, ce qui est fait dans ELSA pour raisonner sur l'espace à l'aide d'un système à base de connaissances peut être reproduit dans d'autres applications. Le chapitre suivant traite de ces applications.

Par ailleurs, l'expérience du développement d'ELSA à l'aide de SHIRKA, a également mis en évidence des faiblesses du modèle et de son implantation. Le chapitre 7 reviendra sur ces observations.

6. INTERET DES BASES DE CONNAISSANCES SPATIALES

Dans les chapitres 2 et 3, de nombreux exemples ont été fournis sans préciser outre mesure le cadre de l'application. Le chapitre précédent a traité en détail d'une application : l'analyse de sites avalancheux. L'idée sous-jacente qui justifie cette présente thèse est que des idées dégagées pour cette application peuvent être utilisés dans d'autres contextes.

Trois domaines d'applications, déjà évoqués dans le premier chapitre, sont présentés maintenant qui mettent en évidence l'intérêt que peut présenter le raisonnement spatial dans les systèmes à base de connaissances. Les détails relatifs à la représentation des connaissances et à l'implantation dans les machines ne sont pas donnés ici. Ils seront regroupés dans le dernier chapitre qui traitera de l'utilisation des représentations centrées objet.

Sont traités successivement, la modélisation symbolique, l'utilisation de représentations de l'espace dans les environnements de résolution de problèmes et les bases de connaissances géographiques. L'apport d'un système à base de connaissances spatiales pour la modélisation symbolique ou les environnements de résolution de problème a déjà été évoqué dans le chapitre précédent relatif au système ELSA. Ce système réalise en effet des actions de modélisation symbolique et possède par ailleurs les caractéristiques d'un environnement de résolution de problèmes. Il s'agit ici de généraliser les remarques qui ont déjà été faites. L'absence de réalisations sur ces thèmes autres que ceux déjà décrits pour ELSA explique le faible volume du présent chapitre.

6.1. MODELISATION SYMBOLIQUE

Le terme de modélisation symbolique regroupe ici l'ensemble des techniques informatiques qui modélisent des phénomènes par le calcul symbolique et non pas par du calcul numérique. La modélisation symbolique apparaît de plus en plus comme un outil intéressant pour traiter les situations que la physique ou la mécanique d'une part, et l'analyse numérique d'autre part ne peuvent maîtriser. Il peut en effet arriver, en modélisation numérique, que ce soient les phases de modélisation, de résolution ou d'informatisation, présentés au chapitre 4, qui constituent l'obstacle insurmontable.

La modélisation symbolique entretient de nombreux liens avec d'autres activités de recherche en intelligence artificielle : modélisation qualitative, physique naïve, raisonnement à partir des connaissances de base, raisonnement sur les connaissances profondes... L'étude et la nature de ces liaisons ne sont pas étudiées ici. Cette analyse dépasse largement le cadre de cette thèse. De nombreux travaux ont été consacrés à ces recherches [DAVIS84] [COHN87].

Les connaissances utilisées en modélisation symbolique proviennent la plupart du temps des connaissances de spécialistes ou des connaissances de base et de bon sens. La machine doit alors reproduire les raisonnements que tiennent les spécialistes du domaine. Les systèmes à base de connaissances constituent une des solutions pour traiter ce problème [CLAN86].

Dès que le phénomène a une extension spatiale, le raisonnement sur le phénomène peut être qualifié de spatial. C'est le cas dans deux systèmes introduits dans le chapitre 1, ELFIN et NEUREX, et dans le système ELSA détaillé dans le chapitre précédent. L'approche bases de connaissances présente alors un intérêt à deux niveaux : celui de la construction de la représentation de l'espace et celui de l'inférence des connaissances pertinentes pour la modélisation symbolique.

6.1.1. Construction de la représentation de l'espace

La modélisation symbolique ne traite pas le continu. Comme la modélisation numérique, elle exige une décomposition de l'espace. Cette décomposition, appelée maillage dans le contexte numérique (cf. 4.3. Environnements de résolution de problèmes), doit posséder des caractéristiques bien définies. Ainsi, comme cela a été montré pour ELSA, la décomposition du terrain en petits panneaux doit être parlante pour le spécialiste dont les connaissances vont être utilisées sur cette décomposition. Les systèmes NEUREX et ELFIN introduisent également des entités spatiales qui sont destinées à être manipulées par les connaissances des spécialistes.

Cette décomposition peut être réalisée par un système à base de connaissances. L'intérêt présenté par cette solution concerne la déclarativité et la dynamique. En d'autres termes, cette solution permet, d'une part, à l'utilisateur d'adapter facilement la représentation de l'espace aux connaissances qu'il compte utiliser et, d'autre part, de modifier la description de l'espace qu'il donne au système.

Adapter la représentation de l'espace

Spécifier les caractéristiques de la décomposition de l'espace à l'aide d'un système à base de connaissances permet de revenir plus facilement sur ces caractéristiques. C'est l'idée qui vient d'être évoquée dans la dernière section du chapitre précédent.

La décomposition de l'espace peut non seulement dépendre du spécialiste concerné mais aussi dépendre de la phase de raisonnement dans laquelle se trouve le spécialiste. En particulier, la finesse de la décomposition risque d'augmenter avec la progression de l'analyse.

Exemple 6.1 : Lors de la construction du système ELSA, la prise en compte du transport de neige par le vent a exigé de choisir entre une distance tri- ou bi-dimensionnelle pour tenir compte de l'affaiblissement des effets d'une arête. Selon les spécialistes, c'est une distance ou bien l'autre qu'il faut retenir. Ce choix peut avoir de l'influence sur la décomposition du site.

Exemple 6.2 : Le système ELFIN doit construire "l'arborescence des contextes", c'est-à-dire qu'il doit inférer le chemin suivi par un hydrocarbure à travers le sous-sol, de la roche mère au piège. En début d'analyse, l'arborescence des contextes ne possède qu'un seul nœud. Les différentes phases d'analyse consistent à enrichir cette arborescence. Dans le chapitre 1, il a été montré que cette décomposition des contextes correspond à une décomposition de l'espace de plus en plus fine. L'utilisateur peut rajouter des règles de décomposition et enrichir ainsi la décomposition de l'espace.

Enfin, la décomposition de l'espace s'effectue souvent à l'aide de critères qui sont le produit d'interprétations ou de formalisations de connaissances heuristiques. Ces interprétations ou ces formalisations sont susceptibles

d'évoluer. L'utilisateur bénéficie alors de la déclarativité pour l'écriture des connaissances correspondantes.

Changement dans la description donnée de l'espace

Les informations utilisées pour la décomposition de l'espace sont susceptibles d'évoluer. Sans enregistrement de résultat, cette dynamique des connaissances données ne pose pas de problèmes. Malheureusement, comme cela a déjà été expliqué, des problèmes d'efficacité incitent à enregistrer les résultats. Il faut alors les maintenir. Le chapitre 3 a montré qu'un TMS pouvait parfaitement exploiter les notions de localité face à un raisonnement de construction de topologie.

Si la description donnée de l'espace est modifiée, la décomposition de cet espace est alors mise à jour de façon judicieuse et la modélisation symbolique du phénomène étudié en tient compte.

Exemple 6.3 : Dans le système ELSA, si l'expert décide d'implanter des râteliers paravalanches sur la moitié de la surface d'un petit panneau, ce petit panneau ne peut plus être considéré comme homogène. Il doit par conséquent être coupé en deux. Si la construction de ces petits panneaux a été assurée par un système à base de connaissances doté d'un TMS, cette redéfinition des petits panneaux sera réalisée automatiquement. L'utilisateur peut alors déclarer un changement dans l'état de surface d'une partie de la zone d'accumulation (changement lié à l'installation des râteliers) et demander une nouvelle analyse qui tienne compte de ces râteliers.

6.1.2. Inférence des connaissances spatiales pertinentes

Une fois une décomposition de l'espace obtenue (qu'elle ait été inférée ou non par le système à base de connaissances), l'exploitation des connaissances de spécialistes exige souvent l'accès à des propriétés particulières des composants de l'espace. Ces propriétés peuvent être directement comprises dans cette décomposition ; il arrive aussi souvent que le système doive les inférer.

Exemple 6.4 : Dans ELSA, les propriétés d'adjacence entre petits panneaux ou les propriétés de superposition suivant les lignes de plus grande pente ne sont pas connues directement à partir de la décomposition en petits panneaux. Il faut les inférer.

Dans un système expert à règles qui traite de phénomènes spatiaux, par exemple, les conditions en partie gauche des règles porteront sur les propriétés spatiales. Si celles-ci ne sont pas données, le système devra les inférer.

De même, la propagation de contraintes qui est souvent utilisée en modélisation symbolique, utilise un réseau au travers duquel elle propage des contraintes. Ce réseau peut parfois correspondre à des relations spatiales (adjacence par exemple). Un raisonnement spatial peut être à l'origine de sa construction.

Les avantages qu'il y a à utiliser une architecture de système à base de connaissances pour traiter cette forme de raisonnement spatial sont les mêmes que ceux qui ont été évoqués dans la sous-section précédente : déclarativité et comportement face à la dynamique.

6.2. ENVIRONNEMENTS DE RESOLUTION DE PROBLEMES

Les environnements de résolution de problèmes ont déjà été présentés succinctement dans le chapitre 4 pour expliquer que le système ELSA en était un. Il n'est pas question ici de revenir sur ces environnements mais de montrer que dans deux domaines ils doivent réaliser des formes de raisonnement spatial.

Dans un environnement de résolution de problèmes, les différents outils informatiques susceptibles d'être utilisés par le spécialiste d'un domaine d'activité donné sont intégrés dans un système à base de connaissances. Si les phénomènes en jeu ont une existence spatiale, l'environnement de résolution de problèmes doit posséder une description spatiale des phénomènes et des objets physiques et réaliser des inférences pour compléter ou exploiter cette description.

Une remarque s'impose. Il s'agit ici d'étudier des exemples de raisonnement spatial utilisé dans les environnements de résolution de problèmes en tant que tels. Il ne s'agit pas ici des modules de modélisation symboliques que ces environnements peuvent contenir (par exemple, les outils de modélisation symbolique contenus dans ELSA). Leur cas a déjà été traité dans la première section de ce chapitre.

6.2.1. Partager une base de connaissances spatiales

Un environnement de résolution de problèmes comporte, entre autres, une base de méthodes [ROUS88]. Ces différentes méthodes sont susceptibles d'être appliquées au cours d'une même session et exigent des connaissances spatiales. Si ces connaissances diffèrent selon les méthodes, elles peuvent la plupart du temps être obtenues à partir du même groupe de connaissances.

Exemple 6.5 : Les ingénieurs qui travaillent sur un problème d'avalanches ont à leur disposition plusieurs outils de calcul ou de simulation. Pour les utiliser, en général, les connaissances issues d'une carte au 25 000^e sont suffisantes.

Un environnement de résolution de problèmes qui traite de phénomènes spatiaux peut, de la même façon, posséder une description de l'espace relativement neutre, qui sera exploitée par chaque méthode avec l'inférence de la représentation spatiale la mieux adaptée.

Exemple 6.6 : Une même description d'un site avalancheux peut servir, après inférences complémentaires, à trois outils de l'expert en avalanches : la modélisation symbolique du déclenchement, l'utilisation du programme AVAL et l'application de la méthode statistique.

Exemple 6.7 : Le système CLER, dédié à la localisation et à l'évaluation du risque de congères, actuellement en développement, possède une description du terrain qui peut être utilisée aussi bien par des modules de modélisation symbolique que par des méthodes de calcul statistique.

La section précédente a montré que des systèmes de modélisation symbolique exigeaient une représentation de l'espace adaptée aux connaissances utilisées dans le cadre de cette modélisation. La modélisation symbolique peut participer à un environnement de résolution de problème au même titre que des méthodes de modélisation numérique. Celles-ci ont également besoin d'une décomposition de l'espace : le maillage.

6.2.2. Inférer un maillage

Les éléments de ce maillage doivent avoir un certain nombre de propriétés calculées (surface, éléments adjacents...) qui dépendent de la méthode numérique calculée. Cette opération constitue, par ailleurs, en analyse numérique, un des goulots d'étranglement les plus importants. Il est fréquent de voir développer un maillage particulier pour une application donnée.

Mailleur automatique

Afin d'alléger cette tâche souvent fastidieuse, des bibliothèques de mailleurs et des éditeurs de maillage adaptés à ces bibliothèques ont été développés. C'est cependant le thème du mailleur automatique qui fait l'objet du plus grand nombre de recherches dans le domaine.

Certaines approches consistent à décomposer les domaines, envoyer des commandes aux éditeurs, prendre en charge certaines opérations fastidieuses, recoller les sous-domaines, le tout de façon simplement procédurale.

Trau propose de son côté [TRAU85] d'exploiter une architecture de système à base de connaissances pour commander une bibliothèque de mailleurs issue du club MODULEF. Il base son approche sur la conviction qu'aucun mailleur général ne peut être bâti dans une approche procédurale.

Le système SYMATRAU qu'il a développé permet à l'utilisateur de définir les limites du domaine ainsi que les conditions aux limites (le système est dédié aux calculs de structures en mécanique). L'utilisateur peut également choisir la précision de son calcul en jouant sur la taille des mailles. Ce paramètre a également de l'importance pour la validité du calcul. En effet, les lieux de fortes contraintes doivent correspondre à un maillage assez fin sous peine d'erreurs importantes dans le calcul. L'utilisateur peut déclarer au système les zones de fortes contraintes. Le système affinera le maillage en ces endroits.

Dans d'autres types de phénomènes à modéliser, la finesse du maillage doit être adaptée à certaines caractéristiques du phénomène (cf. exemple dans la figure 4.8). Cette capacité de SYMATRAU est donc très intéressante. Enfin SYMATRAU peut calculer les propriétés des différentes mailles et préparer ainsi les fichiers de données pour le calcul.

Le système EVE, déjà évoqué au chapitre 4, constitue, à la différence de SYMATRAU, un véritable environnement de résolution de problèmes dédié aux systèmes d'équations aux dérivées partielles (analyse du problème, détermination de la nature du problème, existence de solution, choix du schéma numérique, appel à un mailleur, assemblage des matrices, résolution de systèmes linéaires, visualisation des résultats...) [CHAI&89].

EVE est capable d'appeler un mailleur 2D extérieur. En revanche, en une dimension, EVE possède son propre mailleur, écrit certes de façon procédurale, qu'il est capable de rappeler après un premier calcul pour affiner le maillage en certains points, si les résultats n'y sont pas satisfaisants.

L'idée développée maintenant suggère de construire un mailleur complet dans un système à base de connaissances. Construire un maillage revient en effet à réaliser une opération de décomposition de l'espace et de construction de topologie telle qu'elle a été décrite à la fin du chapitre 2. Plus exactement, il s'agit d'une décomposition du troisième type (c'est-à-dire ex-nihilo). En effet, seules les frontières du domaine et les caractéristiques des mailles sont

imposées. Aucun élément de frontière n'est fourni. Là encore, différents avantages apparaissent. Ils concernent la déclarativité de la programmation et la gestion de la dynamique.

Définir et construire un maillage

Utiliser une approche base de connaissances permettrait d'exploiter la déclarativité. Il serait ainsi possible de définir les mailles souhaitées ou plus exactement de poser les contraintes que les mailles doivent respecter. Ces contraintes portent en général sur des critères "locaux" : convexité, taille, forme ou autres (homogénéité de certaines propriétés, comme la pente ou l'exposition par exemple). Dans certains cas, les critères peuvent être topologiques (respect de certaines lignes par les frontières du maillage).

Le système à base de connaissances peut alors inférer les mailles à partir de la déclaration de leurs propriétés.

L'intérêt du déclaratif est de permettre à l'utilisateur de modifier les critères de construction du maillage. Si l'utilisateur désire observer les effets du changement d'un critère de maillage, il peut simplement modifier la définition des mailles qu'il désire obtenir.

Relancer le mailleur ?

L'architecture à base de connaissances permet, comme cela a été montré au chapitre 3, de réagir de façon sélective et locale dans le cas d'un changement des connaissances données. Si le domaine est modifié, le maillage ne sera réinféré que de façon locale.

De la même façon, si les conditions aux limites sont modifiées, l'invalidation du maillage peut n'être que partielle. Enfin, à l'occasion de la description de SYMATRAU, il a été expliqué que la finesse du maillage dépendait localement de certaines caractéristiques du phénomène calculé (contraintes, gradient de vitesse...). Une gestion efficace de la dynamique permettrait au mailleur de travailler en coopération avec le module de calcul. Ce que réalise le logiciel EVE en dimension 1 avec son mailleur procédural pourrait être alors étendu à d'autres dimensions et à d'autres environnements de résolution de problèmes.

6.3. BASES DE CONNAISSANCES GEOGRAPHIQUES

Le troisième domaine d'application des bases de connaissances spatiales qui est abordé ici est relatif à un contexte bien différent a priori de celui de la modélisation symbolique et des environnements de résolution de problèmes. Il s'agit d'utiliser un système à base de connaissances pour le traitement des informations présentes dans les systèmes d'information géographique et dans les bases de données géographiques (cf. chapitre 1), et de tenter d'en faire un usage voisin (partage des informations dans la base, transactions nombreuses, etc.).

Cette utilisation des systèmes à base de connaissances ne présente un intérêt que si elle offre des avantages absents des systèmes d'information géographique traditionnels. Le travail mentionné au chapitre 1 [SMIT&87] entre dans la catégorie de ces applications. Malheureusement, comme cela a été mentionné, à aucun instant, n'est détaillé l'intérêt spécifique qu'il y a à utiliser un système à base de connaissances.

Or précisément, des systèmes à base de connaissances manipulant des connaissances spatiales permettent de dépasser certaines des limitations des

systèmes d'information géographique dans deux domaines : celui de l'évolution et de la dynamique des données et celui de l'introduction, dans le système d'information géographique, de différentes techniques d'inférence.

6.3.1. Dynamique des connaissances données

Le premier intérêt est naturellement le comportement des systèmes à base de connaissances face à la dynamique des données.

Construction de topologie

Lorsque les données ont été numérisées, une représentation topologique est construite sous la forme d'une base de données dites topologiques. Cette construction est réalisée par des programmes procéduraux qui lisent la base de données des définitions et remplissent la base de données topologiques.

Naturellement, si des modifications ou des corrections sont introduites par l'utilisateur après la construction de cette base de données, celle-ci n'est plus à jour. Il faut alors relancer sa construction. Dans les applications classiques des systèmes d'information géographique, ces modifications sont courantes.

Exemple 6.8 : Les connaissances qu'un utilisateur introduit dans un système d'information géographique sont soit soumises à un risque d'erreur, soit susceptibles d'évoluer dans le temps. Dans le premier cas, l'origine est à chercher dans l'opération de numérisation faite très souvent de façon manuelle. Une fois constatée, l'erreur doit être corrigée. Dans le deuxième cas, les phénomènes saisis (répartition de population de végétation, réseaux de distribution, etc.) évoluent dans le temps. Il faut alors réactualiser ces connaissances données et conserver la base cohérente.

Un système à base de connaissances doté d'un TMS permet de résoudre ce problème. Il faut tout d'abord admettre le principe d'une base de connaissances unique comprenant non seulement les informations provenant de la digitalisation (qui sont alors les connaissances initiales), mais aussi les informations topologiques (qui sont inférées). Les méthodes qui permettent de passer des connaissances initiales aux connaissances topologiques sont elles aussi introduites dans la base.

Si par la suite, durant la session, des modifications interviennent dans la définition, le système de maintien de la vérité permet de revenir sur les connaissances topologiques invalides et d'en réinférer de valides.

Une des opérations les plus classiques dans un système d'information géographique consiste à croiser des couvertures. Comme cela a été expliqué au chapitre 2, cette opération consiste le plus souvent à construire une nouvelle topologie. La situation est tout à fait comparable à l'opération de création de topologie après numérisation. Les caractéristiques en dynamique qui viennent d'être évoquées peuvent être aussi exploitées.

Ces propriétés, exposées à la fin du chapitre 3, concernent surtout l'exploitation des notions de localité au cours d'une modification.

Inférences des propriétés spatiales

Un système d'information géographique ne se contente pas de créer des topologies correctes à partir de fragments de frontières ou de deux topologies. De nombreuses propriétés spatiales sont également calculées par les systèmes

d'information géographique (centre, superficie, périmètre...). Là encore, le maintien de la vérité peut permettre d'enregistrer les résultats tout en les garantissant contre d'éventuelles modifications de connaissances initiales.

6.3.2. Inférences et contraintes

Les systèmes d'information géographique sont souvent utilisés pour enregistrer des informations de type spatial. Il s'agit de la distribution dans l'espace plan des valeurs de certaines propriétés ou "couvertures". Cette représentation est en général utilisée à des fins d'aide à la décision en gestion de réseau, aménagement, etc. C'est alors la base de données descriptives du système d'information géographique qui est utilisée.

Il est alors très tentant de vouloir introduire des inférences simples capables de fournir un diagnostic sur une zone représentée. Le concept qui apparaît alors est celui de "système d'information géographique déductif" pour calquer l'expression de "bases de données déductives".

Il est possible d'utiliser les techniques des bases de données déductives pour enregistrer les informations spatiales et mener ensuite un raisonnement en les exploitant. Il est aussi possible de coupler un système d'information géographique à un système à base de connaissances ; les liens existent.

Exemple 6.9 : Le système d'information géographique ARC-INFO peut être doté d'une interface SQL. Par ailleurs, il existe des interfaces entre certains langages symboliques (Le-Lisp par exemple) et SQL. Il est a priori possible de créer un lien entre un gestionnaire de base de connaissances écrit sur ce langage (SHIRKA, SMECI...) et ARC-INFO.

Cependant, une telle approche revient à multiplier les couches. Si en soi, cette multiplication ne pose pas de problème (l'empilement des couches est un des ressorts du génie logiciel), elle devient délicate dans la mesure où chaque couche contient son modèle des objets manipulés. Dans l'exemple, les objets spatiaux sont vus en tant qu'objets ARC-INFO, objets "relationnels" pour SQL et objets SHIRKA (ou objets SMECI).

Une autre solution consiste à utiliser un système à base de connaissances. Il s'agit alors, d'enregistrer dans la base de connaissances spatiales, les connaissances jusqu'alors présentes dans la base de données descriptives du système d'information géographique. Ces connaissances ne sont plus nécessairement spatiales.

Exemple 6.10 : Dans une application relative à la répartition des essences forestières, la base de connaissances contient donc aussi bien la définition des zones homogènes en peuplement que la nature de ces peuplements, leur âge, leur densité moyenne ...

Richesse des expressions

L'intérêt principal d'une telle solution est de disposer alors, dans le système d'information géographique, de la richesse d'expression et de la puissance d'inférence d'un système à base de connaissances.

Exemple 6.11 : Mainguenaud et Portier [MAIN&89] proposent un type particulier de requête pour les systèmes d'information géographique.
"Quelle est la prévision d'extension du feu de forêt localisé à 50 km au nord-ouest de Toulon sachant que le vent souffle force 6 direction sud-sud-est ?"

Il va de soi qu'un système d'information géographique actuel ne peut répondre seul à ce type de requête. En effet, de simples données et requêtes spatiales ne sont pas suffisantes pour modéliser un phénomène aussi complexe qu'un feu de forêt. La seule solution est d'exploiter par un programme extérieur les informations contenues dans le système d'information géographique. Au sein d'un système à base de connaissances unique les fonctionnalités base de données et calcul peuvent être rassemblées.

Tout type de raisonnement complexe prenant en compte, après inférences complémentaires éventuelles, les connaissances spatiales d'un système d'information géographique peut être réalisé dans la base de connaissances géographiques.

Les requêtes

Dans le premier chapitre une remarque a déjà été faite sur la différence de nature entre un langage de consultation de base de connaissances et un langage de manipulation de base de données. Les systèmes d'information géographique appartiennent à la catégorie des systèmes de bases de données et ils possèdent par conséquent un langage de manipulation de données qui permet d'émettre des requêtes très variées et très nombreuses qui ne pouvaient avoir été toutes prévues par le concepteur de la base de données. Il avait été précisé à cette occasion que les systèmes à base de connaissances ne possédaient pas de langage de consultation de la base comparable à un langage de requêtes de base de données. Le chapitre suivant propose, pour les représentations centrées objet une extension du langage de consultation.

CONCLUSION

Ce chapitre a présenté trois domaines d'applications qui peuvent être concernés par les bases de connaissances spatiales. L'exemple d'ELSA prouve que cette solution est intéressante dans deux d'entre eux : la modélisation symbolique et les environnements de résolution de problèmes. Le troisième domaine concerne les bases de connaissances géographiques et n'a pas donné lieu à des implantations plus poussées.

Toutefois, l'expérience de la construction d'ELSA d'une part, et ce qui est entrevu à propos de ces trois domaines d'autre part, mettent en évidence des besoins particuliers auxquels doivent répondre les systèmes de représentations de connaissances. Le chapitre suivant traite précisément de l'adaptation d'un type de représentation à ces besoins ; il s'agit des représentations centrées objet et du système SHIRKA en particulier.

7. REPRESENTATION DES CONNAISSANCES SPATIALES

Les deux chapitres qui précèdent ont été consacrés aux applications des bases de connaissances spatiales. Ce chapitre traite maintenant de la représentation et de la manipulation des connaissances spatiales dans un système centré objet. Il ne prétend donc pas passer en revue l'ensemble des systèmes de représentation de connaissances qu'il est possible d'utiliser pour la manipulation des connaissances spatiales. L'objectif est plutôt d'étudier en détail ce qu'un type de représentation de connaissances peut apporter au raisonnement spatial et ce qu'il faudrait lui donner pour qu'il constitue un bon outil.

Dans [BUIS87], différents formalismes de représentation de connaissances présentés dans la littérature comme manipulant des connaissances spatiales ont été exposés.

Le formalisme retenu ici est celui de SHIRKA. Des exemples d'implantation ont déjà été donnés à l'occasion des chapitre 4 et 5 relatifs au développement du système ELSA. En particulier, des besoins (pour l'enregistrement, pour le maintien de la vérité) et des lacunes (enregistrement sélectif, récupération de la place occupée par les objets inutiles) ont été évoqués.

Ce chapitre envisage successivement différents aspects du problème. Pour certains, des solutions sont assez approfondies, pour d'autres, le développement n'est pas poussé⁷. De même, certains de ces aspects sont très "techniques", d'autres plus importants dans l'optique de la représentation des connaissances. La conclusion de ce chapitre reviendra sur eux.

Tout d'abord, il examine le problème de la représentation des entités à manipuler : ensembles et leurs éléments, associations entre ces entités... Les problèmes liés aux inférences sont ensuite étudiés. Ils sont regroupés autour de deux méthodes importantes dans SHIRKA : le filtrage et l'attachement procédural. La nature du langage d'accès à la base de connaissances est ensuite discutée : langage de consultation ou langage de requête ? Dans une quatrième section, la gestion de la dynamique d'une base de connaissances spatiales SHIRKA est observée. Enfin, la dernière section traite des problèmes liés à la taille de la base de connaissances.

7.1. DES OBJETS A REPRESENTER

Le chapitre 2 a proposé une classification des connaissances spatiales basées sur des définitions mathématiques. Ces définitions sont toutes basées sur un formalisme : celui de la théorie des ensembles. Si ce formalisme est nécessaire à la classification des connaissances spatiales, il est aussi particulièrement utile pour leur représentation.

Cette section examine donc le problème de l'expression des connaissances mathématiques dans un système de représentation centrée objet. Le problème lié aux constructeurs est alors examiné, celui des relations entre entités vient ensuite.

⁷ Comme l'introduction l'indiquait, aucune implantation d'un nouveau moteur d'inférence n'a été réalisée.

7.1.1. Objets

Les connaissances spatiales s'expriment dans des espaces. Comme cela a été dit au chapitre 2, ces espaces sont des ensembles. Les connaissances spatiales concernent des éléments ou des parties de ces ensembles qui constituent une interprétation du monde réel ; ils correspondent à des objets de l'espace.

Dès lors il est tout naturel de chercher à reproduire ces entités dans la base de connaissances. Les représentations centrées objet s'y prêtent naturellement. L'expérience du développement de la première maquette d'ELSA en PROLOG met bien en évidence l'intérêt qu'il y a à regrouper autour d'entités bien définies l'ensemble des connaissances qui les concernent [BUI87a].

Les systèmes à objets (langages orientés objet, bases de données orientées objet, représentations centrées objet...) sont très souvent utilisés pour manipuler des entités spatiales. C'est le cas en interprétation de scènes [WALK88] ou pour la constitution de bases de données géographiques [MANO&86], pour ne citer que deux exemples.

Le chapitre 5 donne de nombreux exemples d'objets SHIRKA représentant des objets spatiaux au sein du système ELSA. La plupart sont liés au terrain. La prise en compte du raisonnement spatial sur un objet conduit souvent à l'utilisation d'attributs a priori peu prévisibles ou peu utilisés. Ils se révèlent pourtant très utiles. Les exemples de raisonnement de localisation du chapitre 2 ont montré l'importance de certaines notions comme le centre, le diamètre ou la boîte englobante d'un objet.

Le formalisme objet est presque systématiquement lié à la notion de spécialisation et d'héritage. SHIRKA ne fait pas exception à la règle. Comme cela a pu être constaté dans le chapitre 4 relatif à ELSA, il n'est pas fait une grande utilisation de ces concepts dans ce système, en tout cas pour les objets spatiaux (il n'en va pas de même pour les objets relatifs aux scénarios météorologiques). Ceci s'explique d'une part par la pauvreté des types spatiaux manipulés (triangles, petits panneaux, site, arêtes), d'autre part par la spécificité d'un grand nombre d'algorithmes de traitement. Pour les mêmes raisons, le multi-héritage, présent dans SHIRKA, n'a pas été exploité.

Ce multi-héritage est pourtant particulièrement important car un objet spatial possède deux interprétations : il est objet spatial et il est objet de l'application. En tant qu'objet spatial il possède, partage ou hérite d'un grand nombre d'attributs et de méthodes d'inférence. En tant qu'objet de l'application il possède, partage ou hérite d'autres attributs ou méthodes d'inférence. Les objets avec lesquels il partage des propriétés ou dont il hérite ne sont pas les mêmes dans un cas et dans l'autre. Il est donc impossible de modéliser cet héritage si une classe ne peut être spécialisation que d'une seule classe. Le multi-héritage est une des solutions.

Elle n'est pas la seule. Une représentation centrée objet acceptant plusieurs *points de vue* constitue une autre réponse particulièrement bien adaptée à cette double interprétation des objets spatiaux [MARI&90].

7.1.2. Ensembles et listes

La notion d'ensemble peut être abordée de deux façons dans un système centré objet :

- ensemble des instances d'une classe
- ensemble des valeurs d'un attribut d'un objet.

La première approche consiste à voir dans la classe la définition par compréhension de l'ensemble de ses instances. Deux ensembles peuvent d'ailleurs être considérés : celui des instances effectivement dans la base et celui de toutes les instances potentielles de la classe.

La deuxième approche est la plus utilisée. L'attribut est alors dit "multivalué". Malheureusement, dans ce contexte, la notion d'ensemble n'est souvent pas distinguée de celle de liste.

Le système SHIRKA propose par exemple deux facettes de typage pour un attribut donné. La première correspond à un attribut monovalué, c'est la facette `$un` ; la seconde à un attribut multivalué, la facette `$liste-de`. La sémantique liée à cette facette s'apparente à la liste de LISP, c'est-à-dire à une famille ordonnée. Cependant, SHIRKA interdit la répétition. De plus, à la différence de la liste de LISP, celle de SHIRKA est constituée d'instances d'une même classe : celle dont le nom suit la facette (ou d'instances des spécialisations de cette classe). La liste de SHIRKA est donc en fait une séquence d'instances d'une même classe.

En mathématiques en général et pour l'expression de connaissances spatiales en particulier, il est courant d'utiliser en tant que telle la notion d'ensemble mais aussi de liste, de tas et de séquence telles qu'elles sont définies dans la figure 7.1.

	ordonnés	non ordonnés
répétition autorisée	liste	tas
répétition interdite	séquence	ensemble

Figure 7.1 : Les différentes organisations d'objets. Les deux critères utilisés sont le respect de l'ordre et la possibilité de répéter un même objet plusieurs fois.

Exemple 7.1 : Les triangles qui décomposent la surface du terrain dans ELSA sont les éléments d'un ensemble. Ils ne sont pas ordonnés et il est absurde de les répéter dans une liste.

Exemple 7.2 : Pour des raisons techniques de passage d'arguments, il est nécessaire de considérer pour un petit panneau les valeurs des superficies des triangles qu'il contient à l'aide d'un attribut `superficies-tr`. Si ces triangles sont donnés sous une forme ordonnée dans un autre attribut, les valeurs des superficies doivent respecter cet ordre. Cependant, il n'est pas exclu que deux triangles aient la même superficie. La structure de liste doit donc être retenue pour typer cet attribut. L'exemple 7.4 fournit un autre exemple d'utilisation de la structure de liste.

Exemple 7.3 : Les petits panneaux d'un site peuvent être organisés en un réseau dont les arcs représentent les lignes de plus grande pente. Pour un petit panneau donné, il est intéressant de considérer les petits panneaux qui sont à l'aval. La structure de tas peut être utilisée à cette occasion : en effet, les petits panneaux à

l'aval d'un petit panneau ne peuvent être ordonnés et un petit panneau peut se trouver, indirectement, à l'aval d'un autre de deux façons différentes.

Exemple 7.4 : Les points qui définissent la ligne polygonale frontière d'un polygone "non troué" (c'est-à-dire à frontière connexe) sont ordonnés. Par ailleurs la répétition d'un point est interdite (sauf si la ligne polygonale doit être fermée). La structure de séquence doit donc être retenue (si la ligne polygonale est fermée, il faudra retenir la structure de liste).

Il est possible, bien entendu, d'exploiter SHIRKA et son type `$liste-de` pour manipuler des ensembles. C'est ce qui est fait dans ELSA.

Exemple 7.5 : Les triangles de l'exemple 7.1 sont donnés comme valeurs de l'attribut `triangles` de l'objet `site`. Cet attribut possède une facette `$liste-de`. En d'autres termes la base de connaissances contient une connaissance sur l'ordre des triangles. Or cette connaissance n'existe pas ; l'ordre des triangles est en effet complètement arbitraire. Les méthodes d'inférences qui exploitent cette séquence ne doivent donc pas utiliser l'ordre.

En revanche l'impossibilité de répéter la même valeur interdit la manipulation en tant que telle des listes et des tas. Il serait possible, dans SHIRKA, de substituer aux séquences les listes. Cette structure est en effet la plus générale. Les méthodes d'inférences (filtrages ou attachements procéduraux) devraient alors être écrits en conséquence.

Exemple 7.6 : Un attribut multivalué de type ensemble ne peut avoir deux fois la même valeur. Si la facette d'inférence fait appel à un attachement procédural, la fonction LISP associée à celui-ci doit vérifier qu'aucune valeur n'est répétée. Le programmeur doit donc inscrire dans son code cette contrainte. Si la facette d'inférence est un appel à un filtre, rien n'interdit le renvoi par ce filtre d'une même valeur plusieurs fois. La garantie de non-répétition doit alors être assurée par un prédicat, c'est-à-dire par un attachement procédural.

Cette solution n'est pas vraiment satisfaisante car elle oblige l'utilisateur à répéter dans le code associé aux méthodes ou aux prédicats des tests sur le type de la structure correspondante. Il semble donc particulièrement important d'offrir à l'utilisateur, pour typer les attributs, les constructeurs intéressants en mathématiques : la liste, le tas, la séquence et l'ensemble. A défaut d'offrir les quatre, l'ensemble et la liste semblent plus particulièrement intéressants.

Les mécanismes de typage et d'inférence peuvent alors traiter de façon spécifique les attributs.

Exemple 7.7 : Selon qu'il se trouve derrière une facette `$liste-de`, `$ensemble-de`, `$tas-de` ou `$sequence-de`, un filtre va être activé de façon différente par le moteur d'inférence. Dans le premier et le troisième cas, le moteur acceptera plusieurs fois la même valeur, pas dans le second et quatrième cas.

Il faut préciser que ce problème du typage du "multivalué" existe dans d'autres représentations de connaissances.

Exemple 7.8 : Dans PROLOG, l'utilisateur dispose de deux possibilités pour exprimer qu'un objet est lié à plusieurs autres objets selon que ceux-ci sont organisés en ensemble ou en liste.

Par exemple, pour exprimer que les identificateurs X0 et Y0 sont en relation avec l'ensemble d'identificateurs {Z10, Z20, Z30}, l'utilisateur écrira dans l'environnement les prédicats suivants :

```

pred(X0, Y0, Z10)
pred(X0, Y0, Z20)
pred(X0, Y0, Z30)

```

En revanche, si l'utilisateur désire exprimer que les identificateurs X0 et Y0 sont en relation avec la liste (Z10 Z20 Z30), il écrira dans l'environnement le prédicat suivant :

```

pred(X0, Y0, ( Z10 Z20 . Z30)).

```

Ce parallèle n'est toutefois pas très bien respecté. L'ordre dans lequel les prédicats sont rentrés dans l'environnement peut avoir de l'importance. De plus, le même prédicat peut être répété plusieurs fois dans l'environnement...

Le langage de commande doit également permettre d'agir sur les valeurs d'un attribut multivalué de façon globale ou individuelle. Modifier l'ensemble des valeurs (ou la liste, le tas, la séquence des valeurs) d'un attribut est différent de modifier une valeur parmi l'ensemble des valeurs (ou la liste, le tas, la séquence des valeurs).

Enfin, alors que SHIRKA n'autorise pas la combinaison des constructeurs (en fait du seul qu'il possède, c'est-à-dire de `$liste-de`), il semble intéressant d'offrir cette possibilité à l'utilisateur. Ce problème sera plus particulièrement détaillé par la suite à propos des variables et des passages d'arguments.

7.1.3. Relations

Les relations entre objets jouent un rôle fondamental dans l'expression des connaissances spatiales. Que celles-ci soient de nature topologique, métrique, vectorielle ou euclidienne, elles correspondent le plus souvent à des associations entre objets.

Les représentations centrées objets offrent un bon support pour implanter des réseaux de relations. En effet, ces représentations permettent d'assimiler les nœuds aux objets et les relations aux attributs.

Exemple 7.9 : Le schéma suivant permet de représenter les attributs d'un petit panneau. Certains de ces attributs correspondent à des propriétés (en général les attributs de type simple), d'autres correspondent aux relations que ce petit panneau entretient avec d'autres objets (contenance, voisinage, proximité, superposition).

```

(petit-panneau
  sorte-de      = surface ;
  superficie    $un reel ;
  c-gravite    $un point ;
  altitude     $un reel ;
  diametre2D   $un reel ;
  pente-%      $un reel ;
  contient     $liste-de triangle ;
  points-brd   $liste-de point ;
  a-cote       $liste-de petit-panneau ;
  frontieres   $liste-de frontiere ;
  aretes-pro   $liste-de arete ;
  au-dessus    $liste-de petit-panneau )

```

Cependant les relations ainsi prises en compte sont des relations binaires c'est-à-dire que leur graphe est une partie du produit de deux ensembles. En d'autres termes, les représentations centrées objet sont bien adaptées aux liens entre un objet et un autre objet ou aux liens entre un objet et les types simples (booléen, entier, reel, chaîne de caractères et symboles dans SHIRKA).

En revanche, les relations ternaires ou plus généralement les relations n-aires ne peuvent être prises en compte aisément dans ces représentations. Il est alors nécessaire d'introduire des objets supplémentaires qui représentent la relation.

Exemple 7.10 : La distance entre deux objets peut s'exprimer à l'aide d'une relation ternaire dont le graphe contient les triplets du type (objet1, objet2, distance). Il est possible de représenter cette connaissance dans SHIRKA à l'aide d'un objet supplémentaire qui comporte trois attributs : les deux objets et la distance.

Exemple 7.11 : Dans ELSA, les relations de position selon la ligne de plus grande pente entre petits panneaux correspondent à des relations ternaires du type suivant :

```
arrose ( p1 , p2 , fraction )
```

qui signifie que la neige qui recouvre p1 s'écoule dans une proportion de fraction sur p2.

Dans ELSA, cette relation est représentée par une instance de la classe suivante :

```
{arrosage-pp
  sorte-de          = objet ;
  pp-arroseur      $un petit-panneau ;
  pp-arrose        $un petit-panneau ;
  partie           $un reel }
```

Mais dans cet exemple, le traitement est réalisé par le concepteur de la base lui-même.

Cette limitation en expressivité est la marque des représentations centrées objet. En effet, dans la mesure où elles organisent la connaissance autour de la notion d'objet, elles favorisent l'expression des connaissances qui concernent particulièrement un objet. Pour les liens entre deux objets, la priorité donnée à l'un des deux permet encore d'utiliser, de façon satisfaisante, le formalisme objet. En revanche, avec trois objets en relation c'est impossible. De ce point de vue les représentations centrées objets s'opposent clairement aux représentations par prédicats du type PROLOG.

Des représentations centrées objet telles que SRL [WRIG83] ou ROSACE [HUET84] proposent d'ailleurs une telle analogie attribut-relation. L'implantation systématique de cette dualité pour les relations binaires paraît excessive. Pour les relations n-aires, elle pourrait être retenue. L'utilisateur pourrait alors spécifier au niveau d'un attribut d'un objet, l'existence d'une relation n-aire entre cet objet et d'autres objets ou types simple. Cette solution allègerait beaucoup l'écriture de la base de connaissances. Les relations sont très fréquentes dans l'expression des connaissances spatiales.

Exemple 7.12 : Pour reprendre l'exemple 7.11, la classe petit-panneau pourrait posséder la définition (partielle) suivante :

```
{petit-panneau
  sorte-de = surface ;
  arrose   $relation3   lui-meme petit-panneau reel }
```

Cette définition indique que l'objet courant entretient une relation ternaire avec un autre petit panneau et un réel.

La classe-relation correspondante, du type de celle décrite en 7.11, pourrait être créée alors de manière automatique.

Le problème des relations est lié à celui des inférences et du filtrage (cf. section suivante).

7.2. DES INFERENCEES EFFICACES

Cette section traite des capacités d'inférence de SHIRKA appliquées au raisonnement spatial. Elle tente également de faire le point sur les mécanismes qu'il serait intéressant d'ajouter à SHIRKA. Elle est organisée autour des deux mécanismes d'inférence les plus utilisés dans ELSA : le filtrage et l'attachement procédural. Le terme "efficaces" n'a un sens que par rapport à la taille de la base de connaissances. Lorsque peu d'inférences sont réalisées, leurs temps d'exécution ont peu d'importance. Tout change lorsque le nombre de ces inférences devient important, comme c'est le cas dans les bases de connaissances spatiales.

7.2.1. Filtrage

Le filtrage a déjà été présenté succinctement dans le premier chapitre. Les problèmes de performance ne seront pas évoqués ici. Ils sont intimement liés à la taille de la base de connaissances et seront mentionnés dans la dernière section de ce chapitre. Deux problèmes sont présentés ici : l'exploitation des propriétés de certains attributs par le filtrage et la possibilité de *filtrer en construisant* des objets.

Attributs et relations

Dans la première section, il a été mentionné que de très nombreux attributs spatiaux représentaient des relations entre objets. Un grand nombre de ces relations possèdent des caractéristiques qu'il est possible d'utiliser en inférence spatiale.

Exemple 7.13 : Si `petit-panneau1` possède comme valeur de son attribut `a-cote`, `petit-panneau2`, alors `petit-panneau2` possède comme valeur de son attribut `a-cote`, `petit-panneau1`. Ceci traduit simplement que la relation `a-cote` est symétrique.

Exemple 7.14 : Si `petit-panneau1` possède comme valeur de son attribut `au-dessus`, `petit-panneau2`, alors `petit-panneau2` possède comme valeur de son attribut `au-dessous`, `petit-panneau1`. Cette fois-ci, c'est le fait que l'inversion du graphe de la relation `au-dessus` donne le graphe de la relation `au-dessous` qui est exploité.

Exemple 7.15: Si `triangle1` possède comme valeur de son attribut `est-dans`, `petit-panneau1` et si `petit-panneau1` possède comme valeur de son attribut `site-pp`, `drayre`, alors `triangle1` possède comme valeur de son attribut `site-tr`, `drayre`. Là, c'est une propriété de composition de relation qui est utilisée : la relation `est-dans` composée avec `site-pp` est égale à la relation `site-tr`. Cette propriété de composition peut être vue comme une forme d'héritage généralisée. Elle est particulièrement adaptée à la prise en compte des relations de composants à composé.

Cette régularité des relations peut être traitée par filtrage. La relation de juxtaposition entre triangles est symétrique. SHIRKA pourrait exploiter cette propriété de la façon suivante :

```

{triangle
  sorte-de      = surface ;
  lui-meme      $var-nom t ;
  a-cote        $un triangle
                $sib-filtre {triangle
                              lui-meme    $var-> a-cote ;
                              a-cote      $var<- t}}

```

Il est facile de constater que cette écriture conduit le moteur d'inférence de SHIRKA dans une boucle dont il ne sortira pas sans mal. Pour le protéger, il faudrait lui *interdire d'inférer au sein du filtrage*, plus exactement, lui interdire d'inférer sur un attribut d'un objet dont l'inférence a déjà été lancée. A supposer que cette précaution fût prise, chaque attribut pourvu de certaines propriétés de régularité ou de composition devrait, dans la classe où il est défini, se voir doter des filtres qui traduisent et exploitent cette propriété.

Il semble plus intéressant de déclarer ces propriétés sous une forme différente directement exploitable par le moteur d'inférence. Ces attributs se verraient alors, à leur définition, déclarer des propriétés de symétrie, de transitivité, d'inversion ou de composition.

Exemple 7.16 : L'attribut `a-cote` de la classe `triangle` pourrait être ainsi déclaré :

```

{triangle
  sorte-de      = surface ;
  a-cote        $un triangle
                $symetrique a-cote }

```

qui signifierait que l'attribut est symétrique. Cette propriété pourrait être exploitée de façon automatique par le moteur d'inférence en générant, par exemple, le filtre correspondant.

Le système SRL offre une telle possibilité [WRIG83].

Filtrage constructif ?

Le filtrage porte actuellement sur les instances d'une classe présentes à un instant donné dans la base de connaissances. L'ensemble d'instances est alors défini par extension. En d'autres termes, un filtre définit une spécialisation de classe et le moteur d'inférence retient l'ensemble des instances de cette classe qui, à l'instant du filtrage, sont présentes effectivement dans la base et qui vérifient les contraintes de la spécialisation. Le filtrage apparaît ainsi comme tout à fait déclaratif et statique.

Il est possible d'imaginer un filtrage constructif qui porterait sur un ensemble d'instances défini par compréhension. Certaines classes de mécanismes d'inférence permettant de générer les instances d'une classe à l'appel d'un filtrage sur ces classes autoriseraient un tel filtrage constructif. Le moteur d'inférence essaierait les instances explicitement présentes dans la base, mais aussi celles qu'il pourrait générer grâce à des mécanismes d'inférence. Celles générées et non retenues seraient éliminées. Ceci permettrait de filtrer des objets facilement constructibles mais trop nombreux pour exister en permanence dans la base.

Cette forme d'inférence particulière qui étend le filtrage est une solution au problème de génération d'une décomposition de l'espace telle qu'elle a été évoquée à la fin du chapitre 3.

Exemple 7.17 : Lors de la génération de la représentation d'un site pour ELSA, une triangulation de la surface est réalisée. Il faut ensuite construire les petits panneaux comme des zones connexes homogènes selon certains critères. Il est possible d'obtenir dans un premier temps l'ensemble des zones connexes puis, dans un deuxième temps, de les filtrer pour ne retenir que les zones homogènes. Il faut cependant s'attendre à obtenir un très grand nombre de zones connexes dont peu se révéleront homogènes. De plus, il est difficile, dans un filtrage classique de chaîner la construction d'instances et le filtrage sur ces instances. Un mécanisme de filtrage constructif permettrait de résoudre le problème.

Par exemple, l'inférence des zones connexes homogènes pourrait se faire de la façon suivante :

```
{site
  triangles      $liste-de triangle ;
  connexes-homogenes
    $sib-filtre-constr
      {construire-connexes
        triangulation      $var-liste<- triangles ;
        composants        $var-liste-> zone-connexe )
      {zone-connexe
        lui-meme           $var-> connexes-homogenes ;
        homogene?         $valeur vrai ) }
```

La facette `$sib-filtre-constr` indique qu'il s'agit d'un filtrage constructif, c'est à dire que le filtre défini en deuxième position doit être utilisé sur les instances courantes de la classe `zone-connexe` mais aussi sur les instances construites par la méthode `construire-connexes`, comparable à un attachement procédural définie en première position. Chaque instance ainsi créée est testée, conservée si le test est positif, rejetée dans le cas contraire.

7.2.2. Attachement procédural

Ce mécanisme d'inférence (facette `$sib-exec`) ou de vérification de contraintes (facette `$a-verifier`) est utilisé pour le calcul de nombreuses connaissances spatiales. Il est assez satisfaisant. Les remarques à faire portent sur la création d'objets et le rendu d'attributs multivalués.

Création d'objets

La seule façon de créer un objet dans la base au cours d'une inférence est d'inclure cette création d'objet dans un attachement procédural. Ceci est tout à fait possible et fonctionne relativement bien. Les inférences spatiales exigent souvent la création de tels objets : la donnée par l'utilisateur d'un triangle ne crée pas de façon explicite le centre de ce triangle. Un attachement procédural pourra en revanche s'en charger. Le seul problème posé est lié à celui de l'augmentation de la taille de la base. Tant qu'un objet créé est utilisé dans la base, la place qu'il occupe n'est pas perdue. En revanche, il peut devenir inutile suite à une modification des connaissances données. La dernière section évoquera la possibilité de récupérer la place occupée en mémoire par les objets devenus inutiles et en particulier dans l'espace des noms.

Attributs multivalués

Un attribut multivalué peut être évalué par un attachement procédural de deux façons : une seule méthode instanciée donne l'ensemble (ou le tas ou la liste ou la séquence) des valeurs ou autant d'instanciations de la même méthode donnent autant de valeurs. Une sémantique claire doit être attribuée aux facettes d'inférence par attachement procédural. En faut-il une seule ou plusieurs ?

7.2.3. Variables et passage d'arguments

Les deux mécanismes d'inférence qui viennent d'être étudiés sont tous deux destinés à être appelés à partir des différentes facettes d'inférences du type `$sib-filtre` ou `$sib-exec`. Les filtres et les méthodes qui suivent ces facettes d'inférence sont écrits avec des références aux valeurs de certains attributs d'objets.

SHIRKA possède un mécanisme de passage d'argument original : la spécialisation locale. Cette solution est assez élégante mais l'emboîtement des spécialisations locales est limité par la complexité des types ainsi créés. SHIRKA ne connaît pas de type plus complexe que le type `$liste-de`.

La possibilité de composer les différents constructeurs définis en 7.1.2 conduit naturellement à des types complexes.

Exemple 7.18: Dans l'objet

```
{petit-panneau
  sorte-de      = surface;
  contient      $liste-de {triangle
                        sommets      $var-nom som-tr } } ,
```

le type de la variable `som-tr` est "liste-de liste-de points" car l'objet `triangle` est défini ainsi :

```
{triangle
  sorte-de      = surface
  sommets      $liste-de point }
```

Une non-limitation des emboîtements implique donc la possibilité de manipuler des types complexes au niveau des méthodes et des filtres. Ceci implique donc, dans un souci d'homogénéité, que cette possibilité soit offerte aux attributs de tous les objets. Les facettes de typage doivent donc pouvoir être composées. Une conséquence de ceci est que les mécanismes d'inférence exploitent de façon correcte ces types composés pour leurs arguments.

7.3. DES REQUETES

Dans le système SHIRKA le langage de commande permet de consulter la base de connaissances. En d'autres termes, il permet de répondre essentiellement à la question :

```
val? <objet> <attribut>
```

Cette commande conduit le moteur d'inférence à consulter la base de connaissances. Si la réponse figure explicitement dans la base, le moteur se contente de la rendre. Dans le cas contraire il tente d'exploiter les mécanismes d'inférence attachés à l'attribut.

La classification est une commande tout à fait différente. Le mécanisme d'inférence utilisé pour y répondre ne figure pas en tant que tel dans la base de connaissances. L'algorithme de classification constitue un exemple de requête.

Jusqu'à présent, dans ce document, l'utilisateur a été considéré comme un seul et unique individu. Il convient ici, désormais, de distinguer deux types d'utilisateur. Le premier type (appelé concepteur [de l'application]) est celui qui conçoit la base de connaissances c'est-à-dire, dans le contexte SHIRKA, qui écrit les classes et les méthodes d'inférence. Le deuxième utilisateur (appelé simplement utilisateur) se contente de fournir des instances et de formuler des

questions sur la base. Les problèmes d'interrogation de la base de connaissances vont provenir des différences qui existent entre les univers du discours respectifs de ces deux utilisateurs.

7.3.1. Consultation ou requête ?

La consultation d'une base de connaissances consiste à poser une question telle que celle présentée ci-dessus, c'est-à-dire une question "prévue" par le concepteur. Si l'utilisateur désire accéder à une connaissance qui ne figure pas dans la base sous la forme de la valeur de l'attribut d'un objet, il ne peut le faire.

Dans un système de bases de données, la situation est toute différente. Dans un système relationnel par exemple, les requêtes ne se limitent pas à la consultation d'une ligne d'une table. Il est ainsi possible de réaliser des requêtes beaucoup plus complexes qui vont utiliser des opérateurs de comparaison, des connectives logiques... [DELO&82].

Exemple 7.19 : Une requête dans le langage SQL a la structure suivante :

```
SELECT X1, X2, ... Xn
FROM R
WHERE F
```

L'expression logique F permet d'ouvrir le champ des requêtes possibles, puisqu'elle peut comporter une combinaison de conditions de sélection infiniment variées et que par conséquent le concepteur de la base de données (c'est-à-dire l'équivalent du concepteur) n'a pu prévoir.

Les systèmes de bases de données géographiques présentés dans le premier chapitre proposent des requêtes spatiales encore plus complexes. L'introduction de types spatiaux permet des opérations particulières qui étendent l'algèbre classique.

Exemple 7.20 : Scholl et Voisard définissent ainsi les types région et carte sur lesquels des opérations nouvelles sont définies [SCHO&89]. Ceci leur permet d'exprimer des requêtes plus riches que celles exprimables à partir de SQL. En particulier, les opérations étant la plupart du temps définies dans l'espace, de véritables requêtes spatiales peuvent être faites comme celles mentionnées dans le premier chapitre. Ici encore l'utilisateur peut poser des questions qui n'ont pas été prévues par le concepteur.

En réalité, la nature du langage d'interrogation de la base de connaissances dépend énormément du type de l'application implantée sur cette base et plus exactement des besoins de l'utilisateur.

Une application du type modélisation symbolique ou environnement de résolution de problème possède une formalisation assez complète et fermée des connaissances utilisables dans ce contexte qui évite à l'utilisateur d'émettre des requêtes complexes. De plus, ces applications sont dotées d'interfaces qui peuvent à l'occasion réaliser des embryons de raisonnement. Le concepteur est sensé avoir prévu tous les besoins de l'utilisateur.

En revanche, une application du type bases de connaissances géographiques doit offrir à l'utilisateur la possibilité de définir des requêtes plus complexes. En effet, le concepteur ne peut avoir prévu l'ensemble des questions que peut se poser l'utilisateur

Exemple 7.21 : Dans un système d'information géographique, plusieurs cartes thématiques (couvertures) peuvent concerner une même unité de territoire. L'utilisateur est susceptible d'émettre des requêtes portant sur plusieurs couvertures en les combinant les unes avec les autres. Il est impossible au concepteur de prévoir un objet ou un attribut particulier pour chacune de ces requêtes. Le langage de consultation de la base de connaissances doit prévoir cette ouverture.

7.3.2. Des filtres ?

Une solution pour augmenter l'expressivité du langage de commande consiste à offrir à l'utilisateur la possibilité de définir en interactif des filtres. Ces filtres ne sont naturellement plus associés à une facette d'inférence.

```
Shirka : filtrer {classe
                attribut1  $condition-1 ;
                attribut2  $condition-2 ;
                ...
                attributn  $condition-n ;
                attribut0  $resultat }
```

Ce formalisme s'apparente à celui de la requête relationnelle :
condition de sélection / condition de projection

L'attribut (ou les attributs) typés \$resultat correspondent à la condition de projection ; les attributs typés \$condition-i correspondent à la condition de sélection.

Exemple 7.22 : L'ensemble des triangles dont le centre est à moins de 200 mètres de petit-panneau1 et qui sont recouverts d'herbe ne présente pas d'intérêt dans une application d'ingénierie telle qu'ELSA. En revanche, le système doit pouvoir répondre à cette requête s'il s'agit d'une application bases de données géographiques.

Cette fonctionnalité qui consiste à étendre à l'infini les possibilités de requêtes correspond parfaitement aux besoins des applications de type bases de connaissances géographiques.

Il faut noter que cette fonctionnalité est aussi utile, hors raisonnement spatial, dans tout système à base de connaissances à caractéristique transactionnelle, c'est-à-dire lorsque le concepteur ne peut prétendre recenser l'ensemble des requêtes que l'utilisateur peut désirer émettre.

7.4. GESTION DE LA DYNAMIQUE

SHIRKA est actuellement doté d'un système de maintien de la vérité spécifié et implanté par Euzenat [EUZE87]. Ce système possède un certain nombre de capacités non évoquées ici essentiellement parce que le raisonnement spatial ne constitue pas un cadre d'application adapté à ces capacités (maintien sur les méthodes et sur les classes). En revanche, d'autres capacités absentes se révèlent intéressantes pour la maîtrise du raisonnement spatial.

7.4.1. Modifications

Le concept de modification tel qu'il a été introduit au chapitre 3 présente un grand intérêt pour l'exploitation de la notion de localité. Le TMS implanté par

Euzenat n'offre pas la possibilité de prendre en compte cette notion de modification. Pour ce faire, il devrait être capable de manipuler également des niveaux d'inférence (qui correspondent aux couches de stratification de Apt et Pugin [APT&87]).

Pour l'utilisateur, il lui suffirait de déclarer en tant que telle la modification au lieu de procéder par suppression-ajout.

```
mod-val <instance> <attribut> <ancienne-valeur> <nouvelle-valeur>
```

Cette commande est, en effet, actuellement interprétée par SHIRKA comme une suppression suivie d'un ajout ce qui revient, comme cela a été prouvé au chapitre 3, à relancer l'inférence de tout ce qui dépend *potentiellement* de la connaissance modifiée, et non pas seulement ce qui en dépend *effectivement*.

7.4.2. Enregistrement sélectif

Le chapitre 2 a montré que les connaissances spatiales n'avaient pas toutes la même pertinence. En particulier, elles ne sont pas toutes utilisées avec la même fréquence.

Il serait intéressant de disposer, au sein du TMS, d'un moyen de discriminer, parmi les connaissances inférées, entre celles qui sont très utilisées et qui méritent, par conséquent, d'être enregistrées et maintenues et celles qui sont moins utilisées et qu'il n'est pas nécessaire d'enregistrer.

Cette discrimination n'est intéressante qu'en présence de très grandes bases de connaissances pour lesquelles l'enregistrement de toutes les inférences conduirait à des problèmes d'occupation mémoire difficiles à régler.

7.4.3. Implantation

Il ne s'agit pas ici de proposer dans le détail une solution d'implantation d'un TMS dans SHIRKA. Ce sujet dépasse largement le cadre de cette thèse. De plus et surtout, ce travail a déjà été réalisé [EUZE87]. Néanmoins, comme cela a déjà été expliqué au chapitre 4, ce TMS est d'ores et déjà utilisé. A l'appui de cette expérience un certain nombre de remarques, relatives à l'implantation, peuvent être faites qui dépassent le cadre d'une simple chasse aux bogues qui n'a pas sa place ici.

Homogénéité ?

Euzenat a choisi pour des raisons d'homogénéité d'implanter le TMS à l'aide d'objets SHIRKA. Cette solution conduit, comme cela a été montré dans le chapitre 5, à une multiplication importante du nombre des objets SHIRKA et des objets justifications en particulier.

Un enregistrement plus simple du graphe de dépendances peut être envisagé qui ne passe pas par l'utilisation de SHIRKA. Les gains concerneraient aussi bien les temps de réponse que l'occupation en mémoire.

Objets inutiles

A l'occasion d'une invalidation, un certain nombre d'objets deviennent inutiles lorsqu'ils ont été, par exemple, produits par une méthode devenue invalide. Dès lors ces objets occupent une place en mémoire qui peut être considérée comme perdue. Dans un contexte "grande base de connaissances" il

est intéressant de tenter de récupérer cette place. Ce problème sera évoqué dans la dernière section de ce chapitre.

7.5. UNE GRANDE BASE DE CONNAISSANCES

L'application ELSA peut contenir, comme le chapitre 5 l'a montré, un grand nombre d'objets. La limite de 20 000 a été dépassée, à l'aide, il est vrai, des objets du TMS. Toutefois, ce chiffre a été atteint sur des analyses de sites sans que des inférences spatiales aient lieu. Une prise en compte complète des caractéristiques du raisonnement spatial au sein d'ELSA doit conduire, même avec un TMS moins complet et implanté différemment, à une grande base de connaissances.

Les applications mentionnées au chapitre 6, et en particulier les bases de connaissances géographiques, sont également à ranger dans la catégorie des grandes bases de connaissances. De plus, contrairement aux applications orientées vers l'ingénierie, ces applications doivent pouvoir offrir un accès multiple à plusieurs utilisateurs et des capacités de sécurité et de persistance comparables à celles des systèmes de gestion de bases de données.

Différents aspects des grandes bases de connaissances sont maintenant évoqués. Il s'agit des problèmes liés aux limitations de taille de la base, à la sécurité et à la persistance de la base, à la rapidité du moteur et enfin au partage de la base. Pour chacun de ces problèmes, les situations propres au raisonnement spatial sont présentées, les capacités actuelles de SHIRKA exposées et des solutions évoquées.

7.5.1. Limites de la taille de la base

Il vient d'être expliqué que les applications des bases de connaissances spatiales évoquées précédemment comportent toutes de très grandes bases. En effet, la représentation d'ensembles dans l'espace passe peu ou prou par une numérisation de leurs contours qui implique la définition de nombreux points. Par ailleurs, les propriétés intéressantes de ces ensembles sont très nombreuses. Les relations de voisinage entretenues par une surface, par exemple, peuvent concerner de nombreuses autres surfaces.

Pour ce qui concerne la taille de l'application dans SHIRKA, elle est actuellement limitée, en première approximation, par la taille de la mémoire vive réelle allouée au processus (à partir du moment où la décision est prise de ne pas exploiter les capacités de la mémoire virtuelle). D'ores et déjà, des images LISP de 8 Mo sont couramment utilisées et les images de 16 Mo se répandent. Compte tenu, d'une part de l'augmentation ininterrompue des mémoires vives des ordinateurs, d'autre part de la baisse constante du prix de ces mémoires, il paraît raisonnable d'espérer dans les trois ans qui viennent des stations de travail à 100 Mo pour la somme de 100 kf 1990. Des processus LISP de cette taille doivent pouvoir être envisagés.

Un grand nombre d'applications bases de connaissances (celles consacrées à l'ingénierie par exemple) devraient se contenter de telles tailles.

Toutefois, les applications plus exigeantes existent. Il peut s'agir notamment des grandes bases de connaissances géographiques. Pour celles-ci, comme pour d'autres grandes bases de connaissances, il est possible que la taille de la base augmente plus vite que les capacités en mémoire vive. Celle-ci ne peut alors suffire.

Différentes solutions apparaissent. La première consiste à limiter la croissance de la base. Les autres reviennent à exploiter la mémoire secondaire.

Récupération de place

L'expérience d'ELSA a montré que la base de connaissances voyait sa taille augmenter assez rapidement au fur et à mesure des inférences spatiales. En particulier la modification de connaissances données revient à faire invalider par le TMS un grand nombre de méthodes. Tous les objets créés par ces méthodes deviennent alors inutiles et encombrants en mémoire.

Il est alors légitime de chercher à se débarrasser des objets qualifiés de façon générale d'inutiles.

Selon que l'objet à éliminer est un objet donné par l'utilisateur ou qu'il est construit par le moteur au sein d'une méthode, le travail est différent.

Dans le premier cas, il est très difficile de déterminer qu'un objet est inutile. En effet, même s'il n'est valeur d'aucun attribut d'aucun objet (sauf en tant qu'instance de sa classe), condition que tout un chacun serait tenté de choisir pour éliminer les objets, cet objet peut apparaître au cours d'une inférence, par exemple un filtrage.

Dans le deuxième cas, une condition est plus facile à trouver. Il suffit que la méthode qui a conduit à la création de l'objet soit invalidée pour que cet objet devienne réellement inutile. Il est alors possible de le détruire et de récupérer sa place. Cette fonction doit être contrôlée par le système de maintien de la vérité.

Il faut noter que cette récupération consiste, entre autres, à récupérer de la place dans la zone des symboles. Dans SHIRKA en effet, les objets créés par le système ont un nom qui est un symbole construit en accolant à la classe de l'objet un entier issu d'un compteur unique dans SHIRKA. Les entiers étant une ressource limitée, il est nécessaire pour de grandes bases de connaissances de gérer les noms des objets créés de façon plus économique en symboles et en entiers. Cette gestion des noms des objets créés doit être assurée par le module de récupération de place.

Un tel module serait particulièrement intéressant dans la gestion de bases de connaissances spatiales dynamiques. Les exemples du chapitre 4 montrent en effet le grand nombre d'instances de méthodes créées lors des inférences spatiales dans ELSA.

Recours à la mémoire secondaire

Il n'est pas toujours possible de limiter la croissance de la base. Il est alors nécessaire de recourir à un stockage en mémoire secondaire pour la base de connaissances. De nombreuses recherches sont consacrées à ce thème.

Il faut préciser toutefois que ce recours à la mémoire secondaire peut se faire, si seul se pose le problème de la taille, à l'aide des mécanismes de mémoire virtuelle, si ceux-ci sont utilisables pour le processus LISP.

L'utilisation de bases de données (relationnelles, étendues ou à objets) est particulièrement étudiée. Il s'agit le plus souvent d'utiliser ces bases de données pour enregistrer les objets ou les règles de la base de connaissances [LEBA90]. Les chercheurs parlent alors de "mapping" entre le modèle de représentation de connaissances et le modèle relationnel.

Une approche telle que celle retenue par les concepteurs de GEODE paraît tout aussi intéressante [PUCH&88]. GEODE est un gestionnaire d'objets

extensible orienté grande mémoire *vive*. Il permet cependant un stockage partiel en mémoire secondaire.

Enfin il est possible de réécrire directement un noyau de gestion des objets en mémoire secondaire. Cette solution consiste à refaire en grande partie le travail réalisé dans GEODE.

7.5.2. Persistance et sécurité

Le problème de la persistance n'est pas perçu de la même façon selon les applications. Là encore, dans des applications d'ingénierie, ce défaut n'est pas catastrophique. En revanche, il peut le devenir dans des applications plus transactionnelles comme les bases de connaissances géographiques.

SHIRKA offre la possibilité de sauvegarder sur fichier les instances d'une base à un instant donné. Toutefois, si une panne survient à l'improviste, toutes les modifications faites depuis la dernière sauvegarde sont perdues.

De ce point de vue, les solutions présentées ci-dessus peuvent là encore être appliquées : liaison avec un système de gestion de bases de données, utilisation d'un gestionnaire d'objets ou recours direct à la mémoire secondaire. En revanche, la solution de la mémoire virtuelle utilisée directement n'est pas satisfaisante sauf si un dispositif de "journalisation" est implanté.

Dans les trois premières solutions, la persistance et la sécurité de la base sont assurées respectivement par le système de gestion de bases de données, par le gestionnaire d'objet (GEODE possède par exemple un mécanisme de journalisation) ou par le noyau de gestion des objets qui devra en être capable.

7.5.3. Rapidité et accès

Le nombre des inférences nécessaires à une requête dans une base de connaissances spatiales peut être très grand. Les temps de réponse peuvent par conséquent devenir eux aussi importants. Le chapitre 4 donne des exemples de temps de réponse non négligeables sur des requêtes a priori peu complexes. Il a été montré que l'utilisation du TMS apportait des gains en temps de réponse également très importants.

Origine des temps de réponse dans SHIRKA

Les temps de réponse trouvent leur origine dans trois actions faciles à distinguer : l'accès à un objet et aux valeurs de ses attributs déjà connues, le lancement d'un attachement procédural et le filtrage. Ces dernières opérations sont particulièrement nombreuses au sein d'inférences spatiales.

Les problèmes d'accès aux objets ne constituent pas dans SHIRKA un problème important. En effet, la taille des bases actuelles permet jusqu'à présent leur stockage en seule mémoire vive et l'accès à un objet se fait par son nom à l'aide de la P-liste de Le-Lisp. L'objet SHIRKA est un vecteur rangé comme valeur de la propriété *schema* du symbole qui est le nom de l'objet.

Les temps de réponse dus à l'exécution des fonctions LISP associées aux attachements procéduraux ne peuvent être réduits que par la compilation de ces fonctions. Ces temps de réponse échappent par définition au gestionnaire de base de connaissances. Ils ne dépendent que de la qualité de la programmation ou de celle de l'interpréteur ou compilateur LISP.

En revanche, la taille des bases a une influence précise sur les temps de réponse en filtrage. Plus exactement, le nombre d'instances de la classe sur laquelle porte le filtrage est le paramètre le plus important. Il faut préciser

qu'un travail d'optimisation a été réalisé dans SHIRKA pour améliorer les performances de ce filtrage [BLOC86]. La solution consiste à compiler les filtres et à classer les instances.

Accès aux objets et filtrage

Avec l'apparition de grandes bases de connaissances les problèmes de temps de réponse vont devenir cruciaux sur deux des points qui viennent d'être mentionnés : l'accès à un objet donné et le filtrage.

Sur ce dernier point, il faut insister sur une caractéristique des bases de connaissances spatiales : le nombre d'instances est en général beaucoup plus grand que le nombre de classes. La nature même du filtrage le conduit donc à des performances dégradées.

Par ailleurs, comme la taille de la base de connaissances conduit à l'utilisation d'un gestionnaire d'objet ou à un stockage en mémoire secondaire, les solutions qu'il est possible d'envisager seront intimement liées au mode de gestion choisi.

Les chercheurs sur les systèmes de gestion de bases de données et les gestionnaires d'objets ont étudiés les problèmes d'accès sur des objets spatiaux. De nombreux index spatiaux ont été développés à cette fin (cf. chapitre 1 et [PUCH&88]). Le recours à un système de gestion de bases de données ou à un gestionnaire d'objets doit être fait en exploitant ces capacités pour les objets spatiaux.

De même, l'efficacité du filtrage dépendra en partie de son traitement par le système de gestion de bases de données ou le gestionnaire d'objets. En particulier, l'implantation des chemins d'accès dans ce système de gestion de bases de données ou dans ce gestionnaire d'objet pourrait se faire en exploitant les différences de pertinence entre connaissances spatiales telles qu'elles ont été évoquées au chapitre 2.

Exemple 7.23 : Il semble intéressant d'attacher, au fil de l'analyse dans ELSA, plus d'importance à l'attribut *a-cote* de l'objet *petit-panneau* qu'à son attribut *centre*. L'implantation des chemins d'accès devrait tenir compte de cet écart et donner plus de priorité au premier qu'au second.

7.5.4. Partage de la base

Dans SHIRKA un seul accès à la base est possible, celui du processus LISP. Dans les applications d'ingénierie, cette limitation n'est pas gênante. En revanche dans une application des bases de connaissances géographiques, un accès multiple peut se révéler particulièrement intéressant.

Un tel accès multiple peut être envisagé à l'aide, comme pour les bases de données, de plusieurs processus communicants. Il faut préciser que cette solution n'implique pas nécessairement un recours à la mémoire secondaire. Des problèmes de priorité et d'autorisation se posent alors. Ils n'ont jamais été abordés dans le cadre des bases de connaissances. Les solutions développées pour les bases de données ou pour les gestionnaires d'objet doivent sans doute pouvoir être exploitées.

CONCLUSION

Ce chapitre recense les problèmes rencontrés dans la construction, à l'aide de SHIRKA, d'une base de connaissances spatiales. Il suggère également un certain nombre de solutions. Les plus importantes sont celles relatives à la prise en compte de *relations* pour la représentation et les inférences (exploitation des régularités), le *filtrage constructif* et l'introduction de *langage de requête* plus ouvert. Cependant la construction d'ELSA ne s'est pas faite uniquement en manipulant des connaissances spatiales. D'autres problèmes ont été rencontrés ailleurs qui n'étaient pas apparus à propos de l'espace ; enfin, il est fort probable que des problèmes malgré tout typiquement spatiaux n'aient pas été rencontrés. Ce chapitre n'a par conséquent aucune prétention d'exhaustivité.

Toutefois, il est difficile de ne pas remarquer que les problèmes rencontrés pour l'espace sont également présents dans la manipulation d'autres types de connaissances ; les solutions proposées joueraient également un rôle intéressant dans un autre contexte que celui des connaissances spatiales.

En ce sens, le raisonnement spatial apparaît plus comme un cas particulier du raisonnement en général que comme un cas à part. Il sert à ce propos de bon révélateur pour les limites des systèmes de représentation de la connaissance.

CONCLUSION

A l'occasion de l'étude de différents systèmes informatiques déjà existants, il est apparu que *traiter de l'espace* n'était pas toujours, tant s'en faut, *raisonner sur l'espace*. Une définition du raisonnement spatial a donc été proposée qui repose, d'une part sur une *classification* des connaissances spatiales, d'autre part sur la prise en compte, par un véritable système de raisonnement, d'une décomposition ou *modèle* de l'espace.

Le rôle des connaissances *topologiques* dans de nombreuses formes de raisonnement a été particulièrement développé ainsi que trois de ces formes : *construction de topologies, inférences de propriétés, localisation d'objets dans une topologie*. Ces observations ont été faites dans un nombre important de situations rencontrées.

Les systèmes à base de connaissances sont apparus intéressants pour implanter ce raisonnement spatial, en particulier pour leur capacités à représenter des objets sous différents aspects et à inférer des connaissances nouvelles de différentes façons. L'aspect *dynamique* surtout a été développé en exploitant des caractéristiques "spatiales" du graphe de dépendances. En effet, le raisonnement *sur l'espace* présente des propriétés remarquables *dans l'espace* du raisonnement. Tout d'abord, ces propriétés exigent l'enregistrement des résultats d'inférence. D'autre part, elles permettent d'exploiter le concept de *localité* dans ce graphe pour le maintien de la vérité devenu nécessaire à cause de l'enregistrement.

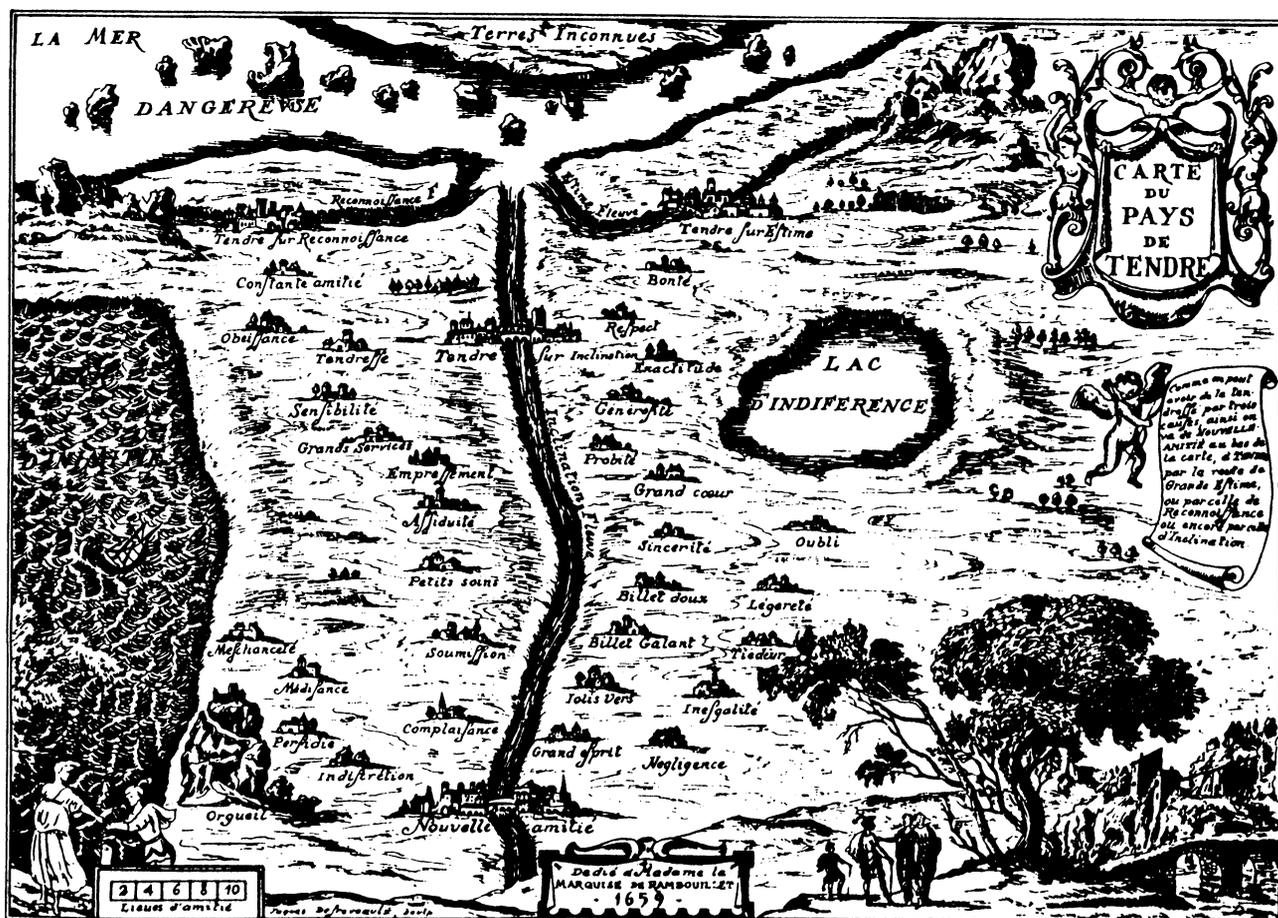
Ces considérations sur la représentation et l'inférence de connaissances spatiales ont été confrontées, de façon approfondie, à une application bien réelle, l'analyse de sites avalanches. Le système ELSA a été développé à cette occasion. Il serait, d'ailleurs, plus exact de dire que cette application est à l'origine de la plupart des réflexions conduites dans cette étude. De plus, outre cette application, d'autres domaines d'utilisation possibles des bases de connaissances spatiales ont été recensées. Deux d'entre eux sont déjà représentés dans le système ELSA : les environnements de résolution de problèmes et la modélisation symbolique de phénomènes ayant une réalité spatiale. Le troisième concerne les applications géographiques.

A partir de l'expérience du développement du système ELSA, et malgré quelques améliorations techniques à apporter, il est apparu que le modèle centré objet de représentation des connaissances, plus particulièrement celui de SHIRKA, était bien adapté aux bases de connaissances spatiales. Des extensions, plus importantes, peuvent néanmoins être suggérées ; elles concernent la prise en compte des relations entre objets pour la représentation et l'inférence, l'implantation d'un filtrage constructif et l'introduction d'un langage de requêtes plus ouvert. Ces extensions ne sont pas, d'ailleurs, destinées aux seules bases de connaissances spatiales. Des systèmes de représentation de connaissances génériques pourraient en être dotés, car des applications non spatiales en bénéficieraient.

Ceci repose la question d'une éventuelle spécificité du raisonnement spatial. En d'autres termes, faut-il développer des systèmes de représentation de connaissances particuliers destinés au raisonnement spatial ou est-il

possible de se contenter de systèmes génériques ? L'expérience du développement d'ELSA, qui n'est certes qu'une expérience, incite à choisir la deuxième proposition. En effet, la seule spécificité rencontrée est celle du nombre : nombre d'objets, nombre de leurs propriétés, nombre d'inférences plus encore que nombre de règles ou de méthodes d'inférence. Les systèmes à base de connaissances n'ont été, jusqu'à présent, que rarement confrontés à des bases de connaissances très volumineuses. Or le propre d'une base de connaissances spatiales est d'abord sa taille. Celle-ci met donc en évidence des faiblesses, jusqu'alors sans conséquence et face auxquelles des améliorations techniques, parmi celles évoquées ci-dessus, prennent tout leur sens. Là encore, d'autres bases de connaissances qui pourraient être qualifiées de très grandes bases de connaissances peuvent bénéficier de ces améliorations.

Si, effectivement, la spécificité du raisonnement spatial se révélait être toute relative, celui-ci apparaîtrait comme un bon illustrateur, voire même un révélateur, des besoins du raisonnement en général. Cette idée n'est pas choquante ; l'être humain n'exploite-t-il pas en permanence la métaphore spatiale pour représenter ses connaissances ou mener des raisonnements. De l'éventail politique, qui va de droite à gauche, aux espaces d'états de la robotique, qui n'ont rien à voir avec notre espace, les exemples sont infinis où, pour plus de facilité, l'être humain projette, dans l'espace, des concepts qui n'ont rien de spatiaux. La carte du Tendre reste sans doute l'exemple le plus achevé⁸ de ce comportement...



⁸ ... le plus étrange aussi

REFERENCES

- [ABEL&89] H. Abelson, M. Eisenberg, M. Halfant, J. Katzenelson, E. Sacks, G.J. Sussman, J. Wisdom, K. Yip
"Intelligence in Scientific Computing"
Communications of the ACM, Vol. 32, n° 5, mai 1989.
- [ALAM&90] R. Alami, R. Chatila, M. Ghallab, C. Laugier, J.P. Laumond
"Robotique et intelligence artificielle"
PRC-GDR Intelligence artificielle, Paris, mars 1990.
- [ANTO&90] J.L. Antoine, M. Pugelli
"ERASME : un système multi-experts d'entretien routier"
Convention IA 90, Paris, janvier 1990.
- [APT&87] K.R. Apt, J.M. Pugin
"Management of stratified databases"
TR-87-41, DCS, U.Texas, Austin, TX, 1987.
- [AUDF86] P. Audfray
"Rapport final sur l'approche système-expert dans l'analyse de sites
avalancheux"
ITMI, mars 1986.
- [BAKK&83] S. Bakkehøi, U. Domaas, K. Lied
"Calculation of snow avalanche run-out distance".
Annals of Glaciology, 4, 1983.
- [BEGH&83] P. Beghin, G. Brugnot
"Contribution of theoretical and experimental results to powder-snow avalanche
dynamics"
Cold Regions Science and Technology, Vol. 8, pp.66-73, 1983.
- [BERT86] B. Berthier
"Evaluation statistique des limites maximales atteintes par les avalanches à
partir des données topographiques"
Rapport CEMAGREF-RTM Haute-Savoie, Grenoble, septembre 1986.
- [BONN&82] A. Bonnet, J. Harry, J. G. Ganascia
"LITHO, un système expert inférant la géologie du sous-sol"
T.S.I.vol.1, n°5, 1982.
- [BRUG&85] G. Brugnot, J.P. Vila
"Investigation théorique et expérimentale des caractéristiques dynamiques des
avalanches de neige dense"
La Houille Blanche, n°2, 1985.

- [BUCH&84] B.G. Buchanan, E.H. Shortliffe
 "Rule Based Expert Systems"
 Addison-Wesley, 1984.
- [BUIS&88] L. Buisson, C. Charlier
 "Avalanche starting zone analysis with a knowledge-based system"
 International Glaciological Society, Symposium on snow and glacier research
 relating to human living conditions, Lom, Norvège, septembre 1988.
- [BUIS&89a] L. Buisson, C. Charlier
 "L'aide à l'expertise en avalanches : le système ELSA"
 Colloque de l'Association des Géologues du Sud-Est sur les risques naturels,
 Avignon, octobre 1989, à paraître dans *Géologie Alpine*.
- [BUIS&90a] L. Buisson, F. Rechenmann
 "ELSA : Un système à base de connaissances centré objet pour l'analyse de sites
 avalancheux"
 Convention IA90, Paris, janvier 1990.
- [BUIS&90b] L. Buisson, C. Charlier
 "Analyse de sites avalancheux : le système ELSA"
 Revue de l'Association Nationale d'Etude de la Neige et des Avalanches, mars
 1990.
- [BUIS87a] L. Buisson
 "Le raisonnement spatial dans les systèmes à base de connaissances. Une
 application: l'analyse de sites avalancheux"
 Rapport de DEA Informatique, INPG, juin 1987.
- [BUIS87b] L. Buisson
 "Spatial reasoning in knowledge-based systems. An application: avalanche
 path analysis"
 International Symposium on Expert Systems, IASTED, pp 157-161, Genève, juin
 1987.
- [BUIS87c] L. Buisson
 "Utilisation de modèles numériques de terrain"
 Rapport d'étudiant, échange ENGREF-NGI, Section avalanches, Oslo,
 septembre 1987.
- [BUIS89a] L. Buisson
 "ELSA : a problem solving environment for avalanche path analysis"
 SCS Symposium on "Artificial Intelligence in Numerical and Symbolic
 Simulation". Lyon, mars 1989.
- [BUIS89b] L. Buisson
 "Reasoning on space with knowledge object centered representation"
 Symposium on Design and Implementation of Very Large Spatial Data Bases.
 Santa-Barbara, 17-18 July 1989. Paru dans *Lecture Notes in Computer Science*,
 Springer Verlag, 1990.
- [BUIS89c] L. Buisson
 "Bases de connaissances et information géographique"
 Colloque sur les systèmes d'information géographiques SIGEO89, Rouen,
 octobre 1989.

- [CHAI&89] M. Chaillot, P. Girard
 "Projet EVE : Environnement de Résolution des Equations aux Dérivées Partielles"
 Mémoire d'ingénieurs, ENSIMAG, Grenoble, Juin 1989.
- [CHAR&85] E. Charniak, D. V. McDermott
 "Introduction to Artificial Intelligence"
 Addison-Wesley, 1985.
- [CHAT&87] J.N. Chatain, A. Dussauchoy
 "Systèmes experts : méthodes et outils"
 Eyrolles, Paris, 1987.
- [CHEN&79] T.T. Cheng, R. Perla
 "Numerical computation of avalanche motion"
 NHRI, paper n°5, Ottawa, 1979.
- [CLAN86] W.J. Clancey
 "Viewing knowledge bases as qualitative models"
 Technical Report, Stanford Knowledge Systems Laboratory, Palo Alto, 1986.
- [CLAP89] A. Clappier
 "Modélisation numérique du transport de neige par le vent"
 Rapport de DEA, IMG, 1989.
- [COHN87] A. G. Cohn
 "Qualitative Reasoning"
 in Advanced Topics in Artificial Intelligence, Springer Verlag, 1987
- [COLE&89] C. Coléou, C. Gendre
 "La prévision du risque d'avalanche"
 in Neige et avalanche, N.48, avril 1989.
- [CORD&85] M.O. Cordier, M.C. Rousset
 "Le contrôle dans les moteurs d'inférence"
 5^{es} Journées Internationales sur les Systèmes Experts et leurs Applications, mai 1985.
- [CORD87] M.O. Cordier
 "Unification contextuelle et raisonnement hypothétique : le moteur d'inférence SHERLOCK"
 6^e congrès "Reconnaissances des Formes et Intelligence Artificielle", pp 787-797, Antibes, novembre 1987.
- [CROW&86] J. L. Crowley, J. Coutaz
 "Navigation et Modélisation pour un Robot Mobile"
 T.S.I.vol.5, n°5, 1986.
- [DAMI89] C. Damier
 "OMEGA : un SGBD multimédia orienté objet pour les applications géographiques"
 Thèse de Doctorat, Université Joseph Fourier, Grenoble, juillet 1989.

- [DAVI88] B. David
 "Le modèle spatiael : une extension du modèle relationnel pour gérer les données spatiales surfaciques"
 4è journées bases de données avancées, Bénodet 1988.
- [DAVI84] R. Davis
 "Diagnostic Reasoning Based on Structure and Behavior"
 Artificial Intelligence, 24, pp.347-410, 1984.
- [DAVI88] E. Davis
 "Error correction in large-scale cognitive maps"
 SPIE Vol. 1003 Sensor Fusion Spatial Reasoning and Scene Interpretation. 1988.
- [DELO&82] C. Delobel, M. Adiba
 "Bases de données et systèmes relationnels"
 Dunod Informatique, 1982, Paris.
- [DENT&80] J.D. Dent, T.E. Lang
 "Modelling of snow flow"
 Journal of Glaciology, Vol.26, pp.131-140, 1980.
- [DOYL79] J. Doyle
 A truth maintenance system
 Artificial intelligence 12-III, pp. 231-272, 1979
- [DUDA&79] R. Duda, J. Gaschnig, P. Hart
 "Model design in the prospector consultant system for mineral exploration"
 in "Expert systems in the micro-electronic age", Edinburg University Press, 1979, pp.153-167.
- [DUFO88] H.M. Dufour
 "Quelques idées générales concernant l'établissement et l'amélioration des modèles numériques de terrain"
 in Bulletin d'information de l'IGN, N. 55, Saint-Mandé, février 1988, pp. 3-18.
- [EDOR87] EDORA
 "Activity report"
 INRIA, INRA, Laboratoire de Biométrie / Université Claude Bernard de Lyon, 1987.
- [EGEN&90] M.J. Egenhofer, A.U. Frank, J.P. Jackson
 "A topological data model for spatial databases"
 Symposium on Design and Implementation of Very Large Spatial Data Bases. Santa-Barbara, 17-18 july 1989. Paru dans Lecture Notes in Computer Science, Springer Verlag, 1990.
- [ESRI89] ESRI
 "ARC-INFO : Users Guide"
 Redlands, California, 1989.
- [EUZE87] J. Euzenat
 "Un système de maintenance de la vérité dans une base de connaissance centrée objet"
 Rapport de DEA, INPG, juin 1987.

- [EUZE90] J. Euzenat
 "Un système de maintenance de la vérité à propagation de contextes"
 Thèse de l'Université Joseph Fourier, Grenoble, février 1990.
- [FARR85] H. Farreny
 "Les systèmes experts ; principes et exemples"
 Cepadues éditions, 1985.
- [FINI88] T. Finin, R. Fritzson, D. Matuszek
 "Adding forward chaining and truth maintenance to Prolog"
 TR PRC-LBS-8802, Paoli, Pennsylvania, septembre 1988.
- [FORB83] K. D. Forbus
 "Qualitative Reasoning About Space and Motion"
 in D.Gentner, A.L.Stevens Eds, "Mental models", Lawrence Erlbaum, London,
 1983, pp53-73.
- [FROS86] R. Frost
 "Introduction to knowledge base systems"
 Collins, London, 1986.
- [GIAN&85] F. Giannesini, H. Kanoui, R. Pasero, M. Van Caneghem
 "Prolog"
 Interéditions, Paris 1985.
- [GIRA87] G. Giraud
 "Intelligence Artificielle et prévision du risque d'avalanche"
 in *Neige et avalanches*, N. 50, Grenoble, 1987.
- [GOAR88] C. Goardou
 "Réalisation d'un système à base de connaissances pour l'étude de zones
 soumises à des risques de congères".
 Rapport de stage Cemagref-Artemis, Grenoble, septembre 1988.
- [GOND83] M. Gondran
 "Introduction aux systèmes experts"
 Eyrolles, Paris, 1983.
- [GRAN&84] T. Granier, J. M. Lefèvre
 "Méthode empirique pour la prévision des risques d'avalanches".
 Rapport de DEA, INPG, septembre 1984.
- [GUIL&90] F. Guillaud, F. Souverain
 "Evaluation du modèle PERIDOT-Jeux Olympiques"
 Rapport de stage, ENM, mai 1990.
- [HARE&85] P. Haren, B. Neveu, O. Corby, M. Montalban
 "MEPAR : un moteur d'inférence pour la conception en ingénierie"
 5e congrès RFIA, Grenoble, novembre 1985.
- [HUET&84] G. Huet, D. Vincent
 "ROSACE, un outil de représentation de connaissances sous formes d'objets et
 d'actions; notice de présentation et d'utilisation"
 CNET, octobre 1984.

- [IGN90] Institut Géographique National
"La base de données topographiques"
Saint-Mandé, 1990.
- [ILOG1] Société ILOG
"SMECI ; manuels d'utilisation et de références"
Gentilly, 1988.
- [ILOG2] Société ILOG
"AIDA ; manuels d'utilisation et de référence".
- [INGR85] F. Ingrand
"Inférence de la description géométrique d'un objet à partir de ses spécifications fonctionnelles."
5^{es} Journées Internationales sur les Systèmes Experts et leurs Applications, mai 1985.
- [KE87] C. Ke
"Système expert de prévention de mouvements de terrain".
BRGM, octobre 1987.
- [KODR86] Y. Kodratoff
"Leçons d'apprentissage symbolique automatique"
CEPADUES Editions, 1986.
- [KOSS83] S. M. Kosslyn
"Descriptions and Depictions"
TR-1275, DCS, University of Maryland, mai 1983.
- [KOWA86] J.S. Kowalik
"Coupling Symbolic and Numerical Computing in Expert Systems"
Elsevier Science Publishers B.V. (North-Holland), 1986.
- [KOWA88] J.S. Kowalik, C.T. Kitzmiller
"Coupling Symbolic and Numerical Computing in Expert Systems II"
North-Holland, 1988.
- [KRIE90] H.P. Kriegel, M. Schiwietz, R. Schneider, B. Seeger
"Performance comparison of point and spatial access methods"
Symposium on Design and Implementation of Very Large Spatial Data Bases.
Santa-Barbara, 17-18 July 1989. Paru dans *Lecture Notes in Computer Science*,
Springer Verlag, 1990.
- [KUIP&88a] B. J. Kuipers, T. S. Levitt
"Navigation and Mapping in Large-Scale Space"
AI Magazine, Vol. 9, N° 2, Summer 1988, pp25-42
- [KUIP&88b] B. J. Kuipers, Y-T. Byun
"A robust, qualitative approach to a spatial learning mobile robot"
SPIE Vol. 1003 Sensor Fusion Spatial Reasoning and Scene Interpretation. 1988.
- [KUIP78] B. J. Kuipers
"Modeling spatial knowledge"
Cognitive Science, 2, 1978, pp129-153.

- [LAUG88] C. Laugier
 "Geometric reasoning in motion planning"
 Rapport de Recherche INRIA n°932, Rocquencourt, décembre 1988.
- [LAUM88] J.P. Laumond
 "Raisonnements géométriques"
 Actes des journées nationales PRC-GRECO-IA - Toulouse 14-15 mars 1988.
- [LAUR90] R. Laurini
 "Introduction au raisonnement spatial"
 4^e Université d'été sur les méthodes de raisonnement en intelligence artificielle, Lyon, juillet 1990.
- [LAWT&88] D.T. Lawton, T.S. Levitt, P. Gelband
 "Knowledge based vision for terrestrial robots"
 SPIE Vol 1003 Sensor Fusion : Spatial Reasoning and Scene Interprétation, 1988.
- [LEBA90] F. Lebastard
 "DRIVER : couche objet virtuelle persistante et raisonnement"
 10^{es} Journées Internationales sur les Systèmes Experts et leurs Applications, Avignon, mai-juin 1990.
- [LEFE&85] J.M. Lefèvre, T. Granier
 "Etude de faisabilité d'un système expert d'aide à la prévision des risques d'avalanches: MEPR".
 5^{es} Journées Internationales sur les Systèmes Experts et leurs Applications, Avignon, mai 1985.
- [LIED&80] K. Lied, S. Bakkehøi
 "Empirical calculations of snow avalanches runout distance based on topographic parameters".
 Journal of Glaciology, vol 26, n°94, 1980.
- [LYNC60] K. Lynch
 "The image of the city".
 Cambridge, Mass. MIT Press 1960.
- [MAIN&89] M. Mainguenaud, M.A. Portier
 "Un langage graphique pour l'interrogation de système d'information géographique"
 Rapport interne MASI, décembre 1989.
- [MANO&86] F.A. Manola, J. A. Orenstein
 "Toward a General Spatial Data Model for an Object-Oriented DBMS"
 Twelfth International Conference on Very Large Data Bases, Kyoto, August 1986.
- [MARI&90] O. Mariño, F. Rechenmann, P. Uvietta
 "Multiple Perspectives and Classification Mechanism in Object-Oriented Representation"
 ECCAI 90, Stockholm, 1990.
- [MARK&83] D. Marks, J. Dozier, J. Frew
 "Automated Basin Delineation from Digital Terrain Data"
 Geo-Processing, 1983.

- [MART86] M. Martinelli
 "A test of the avalanche runout equations developed by the Norwegian Geotechnical Institute"
 Cold Regions Science and Technology, 13, 1986, pp. 19-33.
- [MASI89] G. Masini, A. Napoli, D. Colnet, D. Léonard, K. Tombre
 "Les langages à objets"
 InterEditions, Paris, 1989.
- [MASS87] R. Massa-Rolandino
 "NEVAI: a knowledge-based system for avalanche fall risk forecasting"
 7^{es} Journées Internationales sur les systèmes Experts et leurs Applications, Avignon, 1987.
- [MCLO84] R. Martin-Clouaire
 "ELFIN: une approche système expert et théorie des possibilités appliquée en géologie pétrolière"
 Thèse de 3^{ème} cycle, Université Paul Sabatier de Toulouse, octobre 1984.
- [MCLO85] R. Martin-Clouaire
 "Un système expert capable de raisonnement spatial: l'approche ELFIN développée pour un problème de géologie pétrolière"
 RFIA, Grenoble, novembre 1985.
- [MCLU&84b] D. McClung, K. Lied
 "Statistical and geometrical definition of snow avalanche runout".
 Cold Regions Science and Technology, 13, 1987, pp.107-119.
- [MDER&84] D.V. McDermott, E. Davis
 "Planning Routes through Uncertain Territory"
 Artificial Intelligence, Vol 22, n°2, mars 1984, 107-156.
- [MDER80] D.V. McDermott
 "A theory of metric spatial inference"
 First AAAI, Stanford, CA, USA, 1980.
- [MENA90] S. Ménager
 "Neige et vent : viabilité hivernale au Puymorens"
 Mémoire de 3^e année, ENITEF, juillet 1990.
- [MOHA&88] L. Mohan, R.L. Kashyap
 "An Object-Oriented Knowledge Representation for Spatial Information"
 IEEE Transactions on software engineering, vol. 14, n° 5, may 1988.
- [MONT90] E. Montceyron
 "EXPORT : un poste de travail d'ingénierie portuaire"
 Rapport interne STCPMVN, Compiègne, avril 1990.
- [MOUT90] F. Moutte
 "Protection hivernale contre les congères : développement d'un système d'aide à l'expertise"
 Mémoire de 3^e année, ENITRS, Strasbourg, septembre 1990.

- [MORI&89] S. Morille, G. Borrel
 "Les cartes de localisation probable d'avalanches"
 in Neige et avalanches, Grenoble, n°49, Nov. 89.
- [NCGI89] National Center for Geographic Information and Analysis
 "The research plan of the National Center for Geographic Information and Analysis"
 International Journal of Geographical Information Systems, Vol. 3, N. 2, 117-136.
- [NICO&83] J.M. Nicolas, K. Yazdanian
 "An outline of BGEN : A Deductive DBMS"
 Information Processing 83, IFIP, Elsevier Publishers (North Holland).
- [NIEV90] J. Nievergelt
 "7±2 criteria for assessing and comparing spatial data structures"
 Symposium on Design and Implementation of Very Large Spatial Data Bases. Santa-Barbara, 17-18 july 1989. Paru dans Lecture Notes in Computer Science, Springer Verlag, 1990.
- [NORE&86^a] H. Norem, F. Irgens, E. Schieldrop
 "A continuum model for calculating snow avalanche velocities".
 International Symposium on Avalanche Formation, Movements and Effects, Davos, 1986.
- [NORE&86^b] H. Norem, K. Kristensen, K. Tronstad
 "The Ryggfonn project avalanche data from the 1984/85 winter"
 NGI, Report 58120-8, Oslo, juillet 1986.
- [PAVE&86] A. Pavé, F. Rechenmann
 "Computer-aided modelling in biology: an Artificial Intelligence approach"
 Artificial Intelligence and Simulation, SCS Series, 1986.
- [PIAG&47] J. Piaget, B. Inhelder
 "La représentation de l'espace chez l'enfant"
 Presses Universitaires de France, Paris 1947.
- [PONC90] T. Poncabaré
 "SCAI : un environnement de développement de systèmes à base de connaissances en calcul scientifique et technique"
 Mémoire d'ingénieur CNAM, Grenoble, à paraître.
- [PUVI&89] P. Puvilland, E. Marcoux, F. Rechenmann
 "SADIG : un système expert de classification précoce de gisements d'or"
 Actes de la Convention IA 88->89, Paris, janvier 1989.
- [PUCH&88] P. Pucheral, J.M. Thevenin, H. Steffen
 "GEODE : un gestionnaire d'objet extensible orienté grande mémoire"
 4^{es} journées bases de données avancées, Bénodet 1988.
- [RECH&89] F. Rechenmann, P. Fontanille, P. Uvietta
 "Shirka : manuel d'utilisation"
 Rapport interne lab. Artemis, 1989.

- [RECH86] F. Rechenmann
Représentation des connaissances dans les logiciels de calcul scientifique, in P. Chenin, C. DiCrescenzo, F. Robert (eds.), *Informatique et Calcul, Computers and Computing*, Masson et Wiley, 1986.
- [ROSE&85] C. Rosenthal-Sabroux, D. Schlienger, J. C. Sabroux
"Un nouveau champ d'application des systèmes experts: la volcanologie prédictive"
Cognitiva 85, Paris, juin 1985.
- [ROUS88] B. Rousseau
"Vers un environnement de résolution de problèmes en biométrie. Apports des systèmes à base de connaissances et de l'interaction graphique"
Thèse de Doctorat, Université Claude Bernard, Lyon, 1988.
- [SAME90] H. Samet
"Hierarchical Spatial Data Structure"
Symposium on Design and Implementation of Very Large Spatial Data Bases. Santa-Barbara, 17-18 july 1989. Paru dans *Lecture Notes in Computer Science*, Springer Verlag, 1990.
- [SCHO&90] M. Scholl, A. Voisard
"Thematic Map Modeling"
Symposium on Design and Implementation of Very Large Spatial Data Bases. Santa-Barbara, 17-18 july 1989. Paru dans *Lecture Notes in Computer Science*, Springer Verlag, 1990.
- [SMIT&87] T. Smith, D. Peuquet, S. Menon, P. Agarwal
"KBGIS-II : a knowledge-based geographical information system"
IJGIS, Vol. 1, N°2, pp149-172, 1987.
- [TABL75] R. Tabler
"Predicting profiles of snowdrifts in topographic catchment"
Western Snow Conference, vol n°43, pp 87-89, 1975.
- [TALO87] J.Y. Talon
"Algorithmes de génération et d'amélioration de maillages en 2 dimensions"
Rapport Technique 20, Laboratoire Artemis, Grenoble, mars 1987.
- [TESC&76] T. W. Tesche, M. A. Yocke
"Application of numerical wind models to snow avalanche forecasting"
Avalanche Workshop, Banff, Alberta, Canada, novembre 1976.
- [TOPP86] R. Toppe
"Terrain models. A tool for natural hazard mapping".
International Symposium on Avalanche Formation, Movements and Effects, Davos, 1986.
- [TROU89] B. Trousse
"Coopération entre systèmes à base de connaissances et outils de CAO : l'environnement multi-agent ANAXAGORE"
Thèse de Doctorat de l'Université de Nice - Sophia Antipolis, décembre 1989.

- [TRAU85] P. Trau
 "SYMATRAU : système expert de maillage tri-dimensionnel automatique"
 5e journées internationales sur les systèmes experts et leurs applications,
 Avignon, mai 1985.
- [VARE89] F. J. Varela
 "Connaitre les sciences cognitives ; tendances et perspectives"
 Editions du Seuil, Paris, 1989.
- [VILA84] J. P. Vila
 "Modélisation mathématique et simulation numérique d'écoulements à surface
 libre"
 La Houille Blanche, n°6/7, 1984.
- [VOEL55] Voellmy
 "Über die Zerstörungskraft von Lawinen"
 Schweizerische Bauzeitung, 73 Jahrgang, Hefte 12, 15, 17, 19, and 37, 1955.
- [WALK&88] E.L. Walker, M. Herman, T. Kanade
 "A framework for representing and reasoning about three-dimensional objects
 for vision"
 AI Magazine, Vol.9, N. 2, summer 1988.
- [WILL&86] G. J. Williams, P. M. Naninga, J. R. Davis
 "GEM. A micro-computer based expert system for geographic domains"
 6^{es} Journées Internationales sur les Systèmes Experts et leurs Applications,
 Avignon, avril 1986.
- [WRIG&83] J. M. Wright, M. S. Fox
 "SRL/1.5 User manual"
 CMU Robotics Institute, Pittsburgh, décembre 1983.
- [XIAN&85] Z. Xiang, S. N. Srihari
 "Spatial structure and function representation in diagnostic expert systems"
 5^{es} Journées Internationales sur les Systèmes Experts et leurs Applications,
 Avignon, mai 1985.

ANNEXE : LES ESPACES MATHÉMATIQUES

Le chapitre 2 propose une définition des connaissances spatiales basée sur des structures mathématiques d'espace. Quatre structures ont été retenues : les espaces topologiques, métriques, vectoriels et euclidiens. Cette annexe contient quelques éléments relatifs à ces structures. Elle ne prétend à aucune exhaustivité.

ESPACES TOPOLOGIQUES

Les espaces topologiques sont les structures mathématiques minimales qui permettent de formaliser les notions de frontières, d'intérieur, de connexité. Ils sont basés sur des objets particuliers appelés ouverts.

Topologie et ouverts

Soit E un ensemble quelconque

Soit \mathcal{T} une partie de $\mathcal{P}(E)$.

\mathcal{T} est une topologie de E si et seulement si :

i) $\emptyset \in \mathcal{T}, E \in \mathcal{T}$

ii) $\forall (A, B) \in \mathcal{T}, A \cap B \in \mathcal{T}$

iii) pour toute famille $(A_i)_{i \in I}$ d'éléments de \mathcal{T} , $\bigcup_{i \in I} A_i \in \mathcal{T}$

(E, \mathcal{T}) est un espace topologique.

Les éléments de \mathcal{T} sont les ouverts.

Fermés

Un fermé d'un espace topologique (E, \mathcal{T}) est le complémentaire d'un ouvert de \mathcal{T} dans E .

Une topologie est aussi bien caractérisée par l'ensemble de ses ouverts que par celui de ses fermés.

A partir de ces définitions, on définit les notions de voisinage, d'intérieur, d'adhérence, de frontière et de connexité pour un espace topologique (E, \mathcal{T}) . Elles correspondent aux notions habituellement manipulées par le bon sens.

Voisinage

Soit $a \in E$.

$V \subset E$ est un voisinage de a si et seulement si il existe un ouvert U de \mathcal{T} qui contienne a et qui soit inclus dans V .

L'ensemble des voisinages de a est noté $\mathcal{V}(a)$.

Intérieur

Soit $a \in E$ et $A \subset E$.

a est intérieur à A si et seulement si A est un voisinage de a .

L'intérieur de A est l'ensemble des points intérieurs à A . Il est noté $\overset{\circ}{A}$.

Adhérence

Soit $a \in E$ et $A \subset E$.

A est adhérent à $A \Leftrightarrow (\forall V \in \mathcal{U}(a), V \cap A \neq \emptyset)$

L'adhérence de A est l'ensemble des points intérieurs à A . Il est noté \bar{A} .

Frontière

La frontière de A est l'ensemble des points adhérents à A qui n'y sont pas intérieurs. Elle est notée $\delta(A)$.

$$\delta(A) = \bar{A} \setminus \overset{\circ}{A}.$$

Connexité

(E, \mathcal{T}) est un espace topologique.

E est connexe si et seulement s'il n'existe pas de partition de E en deux fermés non vides.

D'autres propriétés peuvent être définies sur les espaces topologiques : compacité, continuité de fonctions... Leur correspondance dans l'espace de tous les jours est moins évidente.

ESPACES METRIQUES

La notion d'espace métrique est basée sur celle de distance et de boules ouvertes.

Distance

E est un ensemble.

Soit une d application de $E \times E$ dans \mathbb{R}^+ vérifiant les propriétés suivantes :

- i) $\forall (x, y) \in E^2 : d(x, y) = 0 \Leftrightarrow x = y$
- ii) $\forall (x, y) \in E^2 : d(x, y) = d(y, x)$
- iii) $\forall (x, y, z) \in E^3 : d(x, y) + d(y, z) \geq d(x, z)$

d est une distance.

(E, d) est un espace métrique.

Boules ouvertes

(E, d) est un espace métrique.

$B_0(a, r)$ est la boule ouverte de centre a et de rayon $r > 0$.

$$B_0(a, r) = \{ x \in E \mid d(a, x) < r \}$$

Topologie sur un espace métrique

(E, d) est un espace métrique.

L'ensemble des réunions quelconques de boules ouvertes de (E, d) constitue une topologie de E .

Un espace métrique est donc un espace topologique.

Exemples de distance dans le plan

Distance usuelle

$$d_1(p_1, p_2) = [(x_1 - x_2)^2 + (y_1 - y_2)^2]^{1/2}$$

p_1, p_2 ont pour coordonnées (x_1, y_1) et (x_2, y_2) .

d_1 est la distance usuelle. Les boules sont effectivement circulaires.

Distance du max

$$d_2(p_1, p_2) = \max [|x_1 - x_2|/s_x, |y_1 - y_2|/s_y]$$

p_1, p_2 ont pour coordonnées (x_1, y_1) et (x_2, y_2) et s_x et s_y sont des réels strictement positifs qui caractérisent d_2 .

Avec d_2 les boules sont des rectangles centrés sur les centres des boules.

Distance SNCF

$d_{SNCF}(p_1, p_2) = d_1(p_1, p_2)$ si p_1, p_2 et Paris sont alignés

$d_{SNCF}(p_1, p_2) = d_1(p_1, \text{Paris}) + d_1(\text{Paris}, p_2)$ sinon

Cette "distance SNCF" caricature le réseau de chemins de fer français qui privilégie les trajets passant par Paris.

ESPACES VECTORIELS

Les espaces vectoriels sont la structure minimale nécessaire à la manipulation des notions de parallélisme, direction, proportionnalité dans l'espace, base...

Définition

\mathbb{R} ensemble des réels.

E est un ensemble.

E est un espace vectoriel s'il est muni d'une loi de composition interne, notée $+$, et d'une loi de composition externe à opérateur dans \mathbb{R} , notée \cdot , qui vérifient les propriétés suivantes.

i) $(E, +)$ est un groupe commutatif

ii) La loi \cdot est distributive par rapport à $+$ dans E et à l'addition dans \mathbb{R}

iii) $\forall (\lambda, \mu, x) \in \mathbb{R}^2 \times E : (\lambda \mu) \cdot x = \lambda \cdot (\mu \cdot x)$

iv) $\forall x \in E : 1 \cdot x = x$

Bases et dimension

$(e_i)_{i \in I}$ est une base de E si et seulement si :

$\forall x \in E : \exists ! (x_i)_{i \in I} \in \mathbb{R}^{(I)} : x = \sum_{i \in I} x_i \cdot e_i$.

Si E admet une base finie, alors le cardinal de cette base est la dimension de l'espace.

Tout espace vectoriel de dimension finie admet une base.

ESPACES EUCLIDIENS

Les espaces euclidiens possèdent toutes les propriétés des espaces précédents. Il est communément admis de considérer que l'espace de tous les jours est un espace euclidien de dimension 3.

Produit scalaire

E est un espace vectoriel.

Un produit scalaire sur E est une forme bilinéaire φ symétrique sur E dont la forme quadratique associée est définie positive.

Muni d'un tel produit scalaire, E est un espace euclidien.

Norme et distance

La fonction qui à x associe $[\varphi(x, x)]^{1/2}$ est une norme. Cette norme permet de définir une distance d dans E .

$$d(x, y) = [\varphi(x - y, x - y)]^{1/2}.$$

Muni de cette distance, l'espace euclidien peut être considéré comme un espace métrique et, par conséquent, comme un espace topologique.

Orthogonalité et angles

Deux vecteurs x et y sont orthogonaux si et seulement si $\varphi(x, y) = 0$

Les angles sont également définis à partir du produit scalaire.

Le raisonnement spatial dans les systèmes à base de connaissances

Résumé

Les connaissances spatiales sont définies à partir d'une classification qui met en évidence l'importance de la topologie. L'examen d'applications différentes montre que le raisonnement spatial revêt souvent trois formes particulières non exclusives les unes des autres : la construction de topologies, l'inférence de propriétés spatiales et la localisation d'objets.

Les données et les mécanismes d'inférence qui sont utilisés peuvent être introduits dans des systèmes à base de connaissances qui servent alors de support au raisonnement spatial. Son graphe de dépendances possède, par ailleurs, des particularités qu'il est intéressant et, parfois, nécessaire d'exploiter pour gérer la dynamique de ces bases de connaissances spatiales. Les résultats d'inférences doivent en particulier être enregistrés et le maintien de la vérité qu'exige cet enregistrement peut être facilité en exploitant une forme de localité au sein de ce graphe.

Ces considérations prennent appui sur l'expérience du développement du système ELSA, destiné à l'analyse de sites avalanches. D'autres utilisations des bases de connaissances spatiales ont également été étudiées : modélisation symbolique de phénomènes spatiaux, environnements de résolution de problèmes et bases de connaissances géographiques.

ELSA utilise une représentation de connaissances par objets, qui se révèle bien adaptée même si des extensions apparaissent utiles. Il s'agit notamment de la prise en compte, au sein du modèle de représentation, des relations entre objets, de l'insertion d'un filtrage constructif et d'une ouverture plus grande du langage de requête.

Le fait que ces extensions ne soient pas propres à l'espace et qu'elles puissent concerner d'autres types de connaissances suggère que le raisonnement sur l'espace, à défaut d'être spécifique, apparaît comme un révélateur des besoins du raisonnement en général.

Mots-clés : Raisonnement spatial, bases de connaissances, connaissances spatiales, environnements de résolution de problèmes, avalanches.

Abstract

Spatial knowledge can be defined through a classification which shows the large part played by topology. Three forms of spatial reasoning can be identified in several different application domains : topology building, inferences of spatial properties and localisation of objects.

Knowledge based systems can handle data and inference mechanisms involved in those three kinds of spatial reasoning and, as a result, support such a reasoning. Its dependancies graph has some features which imply a management of the dynamicity of spatial knowledge bases. The results of inferences must be recorded and the truth maintenance requested by this caching can take advantage of a locality concept defined in the graph.

These conclusions are based on the experience of the development of ELSA, a system dedicated to avalanche paths analysis. Some other domains can use spatial knowledge bases : symbolic modelling of spatial phenomena, problem solving environments and geographic knowledge bases.

ELSA uses an object-centered knowledge representation which is well adapted, even if some extensions appear useful, such as inter-objects relations, constructive pattern-matching and open query language.

Those extensions are interesting not only for spatial information handling but also for other kind of knowledge. This suggests that, far from being specific, spatial reasoning reveals some needs of general reasoning.

Key-words : spatial reasoning, knowledge bases, spatial knowledge, problem solving environments, avalanches.