

UNIVERSITE MOHAMED KHIDER DE BISKRA

**Faculté des sciences exactes et
sciences de la nature et de la vie**



Département d'informatique

N°d'ordre :

N°de Série :

Thèse

En vue de l'obtention du diplôme de

Doctorat en Sciences en Informatique

Présentée par:

LEJDEL Brahim

TITRE

Mise en œuvre d'un modèle de données à base d'agents pour optimiser
le processus de la généralisation automatique des données
géographiques

Soutenue devant le jury

Kholladi Mohamed Kheireddine	Professeur , Université d'El-Oued	Président
Kazar Okba	Professeur , Université de Biskra	Rapporteur
Laurini Robert	Professeur , INSA de Lyon (France)	Examineur
Chiki Salim	Professeur , Université de Constantine 2	Examineur
Terissa Labib Sadek	Maître de conférences A , Université de Biskra	Examineur
Bennoui Hammadi	Maître de conférences A , Université de Biskra	Examineur

Remerciements

*Je remercie, tout d'abord mon promoteur, Monsieur **Kazer Okba**, professeur à l'Université de Biskra, pour avoir accepté de diriger cette thèse, sans lui je n'aurais jamais eu l'idée de la lancer, pour sa patience et sa confiance, pour m'avoir fait partager sa passion pour la recherche, ses critiques constructives, ses explications, ses suggestions pertinentes, ses conseils précieux et ses compétences scientifiques m'ont conduit à l'aboutissement de ce travail.*

*Je remercie énormément Monsieur **Kholladi Mohamed Kheireddine**, Professeur à l'Université Hamma Lakhder d'EL-Oued, de me faire l'honneur et le plaisir de présider ce jury.*

*Mes remerciements vont particulièrement à Monsieur **Robert Laurini**, professeur à l'INSA de Lyon, pour avoir accepté de participer à ce jury et d'évaluer mon travail, aussi pour son aide et son soutien et pour ses commentaires pour les articles rédigés lors de cette thèse.*

*Je remercie également Monsieur **Chiki Salim**, Professeur à l'Université de Constantine 2, Monsieur **Terissa Labib Sadek**, Maître de conférences A à l'Université de Biskra, Monsieur **Bennoui Hammadi**, Maître de conférences A, à l'Université de Biskra, d'avoir accepté de participer à ce jury et d'évaluer mon travail, qu'ils trouvent ici l'expression de ma très haute considération.*

*Pour finir, merci à **toute personne** ayant participé de près ou de loin à l'aboutissement de ce travail.*

Dédicaces

Je dédie ce modeste travail

À mes parents ;

À ma femme et ma petite fille Anfal ;

À mon frère Ferhat et sa femme ;

À tous mes amis et mes collègues ;

À tous ceux qui m'ont aimé et me souhaitent le bonheur et la réussite.

Résumé

Le processus de la généralisation de données géographiques consiste à dériver à partir de données géographiques détaillées, des données moins détaillées adaptées aux besoins des utilisateurs (exp. changement d'échelle, contexte d'application). Plusieurs approches sont proposées pour automatiser ce processus, les systèmes multi-agents et les approches d'optimisation sont très répandues pour résoudre les problèmes de ce domaine mais ils avaient des inconvénients majeurs. Le but de cette thèse est d'hybrider ces deux approches pour combler leurs inconvénients et bénéficier de leurs avantages. Dans cette approche, les objets géographiques (routes, bâtiments, etc.) sont modélisés comme des agents, qui s'auto-généralisent en se basant sur le résultat de l'algorithme génétique qu'ils les exécutent. Les agents sont dotés de capacités de perception de leur environnement et de capacités de communication et se composent de trois modules principaux; un module de généralisation, un module d'optimisation et un module de mutation des relations topologiques entre les régions et le nouveau concept de rubans. Chaque agent exécute son algorithme génétique pour trouver les solutions qui mettent les contraintes cartographiques en situation de satisfaction. L'agent superviseur synchronise les transformations à appliquer par les agents pour éviter autant que possible les conflits entre les agents voisins et aussi il peut appliquer un algorithme génétique global dans le cas où il y a des conflits qui ne sont pas résolus au niveau local de l'agent.

Mots clés : SIG, contraintes cartographiques, agent génétique, action optimale, Rubans, relations topologiques, généralisation automatique.

Abstract

The generalization process of geographic data consists to derive from detailed geographic data, less detailed data adapted to the users needs (for example level of detail, application context). Several approaches are proposed to automate this process, the Multi-agent system and optimization approaches are usually used to solve the problems in this domain but they had some disadvantages. The purpose of this thesis is to hybrid these two approaches to address their inconvenients and benefits from their advantages. In this approach, the geographical objects (roads, buildings, etc...) are modeled as agents which self-generalize basing on the results of the genetic algorithm that they run. Agents are given powers of perception of their environment and communication capabilities and are composed of three main modules, module of generalization, optimization module and module of transformation of topological relationships between regions and the new concept of ribbons. Each agent performs its genetic algorithm to find solutions to meet the cartographic constraints in satisfaction situation. The supervisor agent synchronizes the transformations applied by the agents to avoid as much as possible conflicts between neighboring agents and also it can apply a global genetic algorithm in the case in which there are conflicts are not resolved at the local level of the agent.

Keywords: GIS, cartographic constraints, agent genetic, optimal action, ribbons, topological relationships, automatic generalization.

الملخص

عملية تعميم المعطيات هي عبارة عن اشتقاق معطيات أقل تفصيل من معطيات أكثر منها تفصيل و ذلك حسب احتياجات المستعمل (تغيير المقياس ، موضوع التطبيق،...). هناك عدة طرق مقترحة لتألية هذه العملية، النظام متعدد الوكلاء و طرق التحسين من بين الطرق الأكثر استعمالا في هذا المجال، لكن لهما عيوب كثيرة. الهدف من هذه المذكرة هو الجمع بين هتين الطريقتين للحصول على طريقة جديدة تسمح بتسريع هذه العملية وتحسين نتائجها. في هذه الطريقة نمثل الكائنات الجغرافية (الطرق، المباني..الخ) بوكلاء قادرين على القيام بعملية التعميم بشكل ذاتي. الوكلاء مجهزين بقدرات عالية في الملاحظة و الاتصال و يتكون كل وكيل من ثلاث أجزاء و هي جزء التعميم، جزء التحسين و جزء العلاقات الطبوغرافية بين المساحات و المصطلح الجديد و هو الأشرطة. في هذه العملية كل وكيل قادر على تنفيذ خوارزمية جينية تسمح له بإيجاد الحل المناسب الذي يجعل كل الشروط الجغرافية في حالة تشبع (اكتفاء). الوكيل المراقب قادر على المزامنة بين كل التغيرات التي تطرأ على الوكلاء لتجنب الوقوع في تصادم و كذلك هو قادر على تنفيذ خوارزمية جينية عامة تسمح له بإيجاد الحل المناسب في حالة عدم قدرة الوكيل على إيجادها محليا.

الكلمات المفتاحية : نظام المعلومات الجغرافي ، الشروط الجغرافية ، الوكيل الجيني ، العمل المحسن ، الأشرطة ، العلاقات الطبوغرافية ، التعميم الالي.

**Table
des
matières**

Table des matières

Introduction générale

1. Contexte	1
2. Objectif de la thèse	2
2.1. Motivation	2
2.2. Objectif	3
3. Méthodologie	4
4. Plan de thèse	4

Chapitre I : Revue des concepts

1. Introduction	6
2. L'information géographique	6
2.1. Définition	6
2.2. Représentation de l'information géographique	6
2.3. Bases de données géographiques	7
3. Système d'information géographique	8
3.1. Définitions	8
3.2. Fonctions d'un SIG	9
3.3. Organisation des données en couche	10
4. Généralisation des données géographiques	11
4.1. Echelle et niveau de détail	11
4.2. Définitions	12
4.2.1. Généralisation de données géographiques	12
4.2.2. Généralisation à but cartographique	13
4.3. Contraintes cartographiques	14
4.4. Relations topologiques	15
4.5. Opérateurs de généralisation	16
5. Conclusion	18

Chapitre II : Etat de l'art

1. Introduction	19
2. Automatisation de la généralisation	19
2.1. Besoin d'automatisation de la généralisation	20
2.2. Enjeu de l'automatisation de la généralisation	20
2.3. Problématique	21
2.3.1. Algorithmes de généralisation et leurs paramétrages	21
2.3.2. Mesures cartographiques de la qualité	22
2.3.3. De l'algorithme au processus	23
3. Modèles de généralisation	23
3.1. Modèles par séquence prédéfinie	23
3.1.1. Principe	23
3.1.2. Discussion	24
3.2. Modèles globaux	24
3.2.1. Modèles par minimisation	24
3.2.1.1. Principes	25

3.2.1.2. Discussion	26
3.2.2. Modèles par optimisation globale	26
3.2.2.1. Généralisation basée sur la méthode du recuit simulé	27
3.2.2.2. Généralisation basée sur les algorithmes génétiques	28
3.3. Modèles basés AGENTS.....	29
3.3.1. Principes	29
3.3.2. Discussion	31
4. Modèles de relations topologiques	32
4.1. Modèle d'ALLEN	32
4.2. Modèle des 9-intersections.....	32
4.3. Modèle de Lee & Hsu	33
5. Synthèse sur les approches	34
5.1. Modèle AGENT	34
5.2. Méthodes d'optimisation.....	35
5.3. Synthèse et objectifs.....	36
6. Conclusion.....	36

Chapitre III : Modèle hybride pour optimiser le processus de généralisation

1. Introduction	38
2. Généralisation et systèmes multi-agents	38
2.1. Définitions.....	38
2.1.1. Agent	38
2.1.2. Système multi-agent	39
2.1.3. Environnement	40
2.2. Pourquoi utiliser l'approche agent ?.....	40
3. Généralisation et méthodes d'optimisation	42
3.1. Définition.....	43
3.2. Représentation des solutions.....	43
3.3. Pourquoi utiliser une approche d'optimisation ?.....	44
4. Généralisation et algorithmes génétiques.....	45
4.1. Définitions.....	45
4.1.1. Algorithmes évolutionnistes.....	45
4.1.2. Algorithmes génétiques.....	45
4.2. Origines.....	46
4.3. Terminologie utilisée pour les algorithmes génétiques.....	46
4.4. Eléments de base de l'algorithme génétique.....	47
4.4.1. Individu	47
4.4.2. Population.....	47
4.4.3. Génération.....	48
4.4.4. Fonction d'évaluation (<i>fitness</i>).....	48
4.4.5. Sélection.....	48
4.4.5.1. Sélection par roulette (wheel)	48
4.4.5.2. Sélection par Rang	49
4.4.5.3. Sélection steady-state.....	49
4.4.6. Croissement et mutation.....	49
4.4.7. Validité et cohérence.....	50
4.4.8. Conditions d'arrêt.....	50
4.4.9. Codage des solutions	50
4.4.9.1. Codage binaire.....	51

4.4.9.2.	Codage string.....	51
4.4.9.3.	Codage réel.....	51
4.5.	Procédure d'évolution d'un algorithme génétique.....	52
5.	Contribution 1: Modèle hybride pour optimiser le processus de généralisation.....	53
5.1.	Motivation.....	53
5.2.	Principes.....	53
5.3.	Diagramme de classe UML.....	54
5.4.	Architecture de système.....	54
5.4.1.	Architecture de l'agent génétique	54
5.4.2.	Structure d'agent	56
5.4.2.1.	Gène Route.....	56
5.4.2.2.	Gène bâtiment	56
5.4.3.	Modules de l'agent génétique	57
5.4.3.1.	Module de généralisation (Optimiseur).....	57
5.4.3.2.	Module de contrôle d'optimisation	59
5.4.4.	Types des agents.....	60
5.4.4.1.	Agent interface	60
5.4.4.2.	Agent superviseur.....	60
5.4.5.	Relations spatiales entre les agents	61
5.4.6.	Motivation des agents.....	62
5.4.7.	Interaction et communication intra et inter-agents.....	62
5.4.8.	Opérations spécifiques	63
5.4.8.1.	Agrégation des objets	63
5.4.8.2.	Elimination	64
5.4.9.	Cycle de vie d'un agent génétique	64
6.	Contribution 2 : Modèle de généralisation des relations topologiques	65
6.1.	Motivation.....	65
6.2.	Rubans (Ribbon.....	66
6.3.	Processus de réduction d'échelle.....	66
6.4.	Acuité visuelle appliquée à des objets géographiques.....	67
6.5.	Relations Ruban – Ruban.....	68
6.5.1.	Relation « Disjoint »	68
6.5.2.	Relation « Meet »	69
6.5.3.	Relation « Merge »	69
6.5.4.	Relation « Cross »	70
6.6.	Mutation des relations Ruban-Ruban.....	71
6.6.1.	Mutation de « disjoint » en « merge ».....	71
6.6.2.	Mutation de relation « cross » en « merge »	71
6.7.	Relations Ruban-Région.....	72
6.8.	Mutation de relations Rubans-Régions.....	73
6.8.1.	Mutation de « disjoint » en « meet ».....	73
6.8.2.	Mutation de « contain » en « cover ».....	74
7.	Conclusion.....	75

Chapitre IV : Implémentation & test

1.	Introduction.....	76
2.	Technologie choisie.....	76
2.1.	Plateforme JADE.....	76
2.2.	Langage de programmation (Java).....	77

3.	Architecture du prototype développé	77
4.	Adaptation niveau de détail / échelle	79
5.	Expérimentation et résultats	80
5.1.	Déroulement du processus de généralisation	80
5.1.1.	Objets traités	81
5.1.2.	Conflits traités	81
5.1.3.	Algorithmes implémentés	81
5.1.4.	Paramètres	82
5.1.5.	Mesures	82
5.2.	Etapes de l'algorithme génétique	82
5.3.	Opérations de la généralisation	83
5.4.	Résultats de Test.....	84
5.4.1.	Cas d'étude 1 : Généralisation des données spatiales	84
5.4.2.	Cas d'étude 2 : Mutation des relations topologiques	85
6.	Comparaison avec d'autres approches.....	87
6.1.	Comparaison avec l'approche de (I.D.Wilson et al, 2003)	87
6.2.	Comparaison avec l'approche Agent (Duchêne et al. , 2012).	88
6.3.	Comparaison avec les autres modèles topologiques	89
7.	Conclusion.....	90

Conclusion Générale & Perspectives

1.	Conclusion générale	91
2.	Contributions	91
3.	Perspectives	92

Bibliographie..... 94

Annexe1 103

1.	Introduction.....	103
2.	Article : Genetic agent to optimize the automatic generalization process of spatial data.	103
2.1.	Objectif.....	103
2.2.	Contribution	103
3.	Article : Genetic agent approach for improving on the-fly web map generalization. ...	104
3.1.	Objectif.....	104
3.2.	Contribution	104
4.	Article : Optimizing the generalization process of spatial data using multi agent system based on improved genetic algorithm	105
4.1.	Objectif.....	105
4.2.	Contribution	105
5.	Article : Mathematical framework for topological relationships between ribbons and regions	105
5.1.	Objectif.....	105
5.2.	Contribution	106
6.	Conclusion.....	106

Annexe2..... 107

**Liste
des
Figures**

Liste des figures

Introduction générale

Figure 1. Méthodologie suivie.	4
Figure 2. Plan du mémoire.	5

Chapitre I : Revue des concepts

Figure I-1. Deux objets représentés en mode raster et en mode vecteur.....	7
Figure I-2. Un exemple de ruban.	7
Figure I-3. Organisation des données en couches.....	10
Figure I-4. Extraits de cartes à différentes échelles (P. Taillandier, 2008).	11
Figure I-5. Généralisation cartographique (Taillandier, 2008).	13
Figure I-6. Problèmes de lisibilité lorsqu'on affiche des données avec une échelle de symbolisation nettement inférieure à l'échelle de référence (Duchêne, 2004).....	14
Figure I-7. Exemples d'opérations de simplification (Mustière, 2001).	17
Figure I-8. Exemples d'opérations de caricature (Mustière, 2001).....	17
Figure I-9. Exemples d'opérations d'harmonisation (Mustière, 2001).....	17

Chapitre II : Etat de l'art

Figure II-1. Un processus de généralisation avec l'outil ModelBuilder (Lee, 2003).....	24
Figure II-2. Données généralisées grâce à une technique de minimisation par moindres carrés (Harrie & Sarjakoski 2002).	25
Figure II-3 Données généralisées par une méthode itérative basée sur la technique du recuit simulé (Ware et al., 2003).	27
Figure II-4. Exemple de généralisation par algorithme génétique (I.D. Wilson et al, 2003)...	29
Figure II-5. Exemples de résultats obtenus avec le modèle de généralisation AGENT.	30
Figure II-6. Exemple de résultats obtenus avec CartACom (Duchêne, 2004).....	31
Figure II-7. Le modèle linéaire d'Allen.	32
Figure II-8. Le modèle de 9-intersection.....	33
Figure II-9. Les 169 types de relations spatiales.	34

Chapitre III : Modèle hybride pour optimiser le processus de généralisation

Figure III-1. Le modèle AGENT appliqué à l'espace urbain.	41
Figure III-2. Résolution d'un problème par une technique d'optimisation (Pitiot, 2009).	43
Figure III-3. Exemple de sélection par roulette.....	48
Figure III-4. Codage réel.....	51
Figure III-5. Algorithme génétique générique.	52
Figure III-6. Diagramme de Classe UML.	54
Figure III-7. Architecture générale de l'agent génétique.	55
Figure III-8. Exemple de gène de l'agent route.....	56
Figure III-9. Exemple de gène de l'agent bâtiment.....	57
Figure III-10. Opérateur de croisement.....	58
Figure III-11. Opérateur de mutation.	59
Figure III-12. Opération de croisement globale.	61

Figure III-13. Types d'agrégation.....	64
Figure III-14. Cycle de vie d'un agent.....	65
Figure III-15. Le modèle ruban appliqué sur les autoroutes.....	66
Figure III-16. La généralisation selon l'échelle.....	67
Figure III-17. Relations « disjoint » entre deux rubans.....	69
Figure III-18. Relation « Meet ».....	69
Figure III-19. Relation « Merge ».....	70
Figure III-20. La relation « Cross ».....	70
Figure III-21. La Mutation de relation « disjoint » en « merge ».....	71
Figure III-22. La Mutation de relation « crossing » en « merging ».....	72
Figure III-23. Relations de base entre régions et rubans.....	72
Figure III-24. Selon l'échelle : la route touche (meet) ou non la mer.....	73
Figure III-25. Mutation de relation « disjoint » en « meet ».....	73
Figure III-26. Mutation de relation « contain » en « cover ».....	74

Chapitre IV : Implémentation & test

Figure IV-1. Architecture générale de notre prototype.....	78
Figure IV-2. L'architecture détaillée de notre prototype.....	79
Figure IV-3. Processus d'adaptation niveau de détail / échelle.....	79
Figure IV-4. Le déroulement de notre processus de généralisation.....	80
Figure IV-5. Résultat de généralisation.....	84
Figure IV-6. Exemple de généralisation.....	86
Figure IV-7. La généralisation des relations topologiques.....	86
Figure IV-8. Procédure d'évolution de l'algorithme génétique (Wilson et al. 2003).....	87
Figure IV-9. Espace urbain et espace rural (Duchène, 2003).....	89

Introduction

générale

Introduction Générale

1. Contexte

Depuis des centaines d'années, l'information géographique était connue comme une nécessité lors de prises de décisions stratégiques ainsi que pour diverses activités de l'être humain. En effet, il l'utilisait dans différentes applications de la vie quotidienne (marine, militaire, juridique, routière.....) pour localiser un objet (bâtis, route, pont, lieu touristique,...etc.) dans l'espace, évaluer des distances, mesurer des surfaces d'une commune, d'un pays...etc. Pour ce faire, il a développé des systèmes, des algorithmes et a défini des formes d'organisation de ces données géographiques qui sont habituellement des supports papier appelés "cartes".

Jusqu'aux années 80, la carte était le seul support de l'information géographique et le seul moyen de communiquer cette information (**Ruas, 2002**). Différentes cartes sont alors conçues, elles présentent l'espace à différentes échelles pour satisfaire les besoins des différents utilisateurs. L'établissement d'une carte à une échelle donnée est une tâche complexe qui peut prendre énormément de temps (de plusieurs mois à plusieurs années) selon la nature de la surface étudiée.

Or le développement technologique étonnant (électronique, informatique, optique...) qui a touché tous les domaines de la vie a fait que la terre est observée et photographiée de l'espace par des satellites de différents types. Ce support "la carte" est petit à petit remplacé par des images numériques stockées dans des bases de données baptisées, Bases de Données Géographiques (BDG). Ces dernières permettent de stocker l'information géographique et d'y appliquer des traitements numériques spécifiques (des calculs d'itinéraires, des simulations, des recherches de lieux qui répondent au mieux à un ensemble des critères (des terroirs adéquats à bâtir,...etc.)). Tout ceci est effectué à travers des systèmes capables de gérer ce volume considérable de données et délivrer par conséquent des résultats en un temps de réponse raisonnable, d'où l'apparition d'un nouveau type d'outils appelé le SIG (Système d'Information Géographique). Un SIG avec ses différentes fonctionnalités est un moyen d'aide à la décision, il soutient les utilisateurs (décideurs) dans leurs prises de décisions quant à la gestion du territoire, à la planification dans des domaines très variés (urbanisme, environnement, transport, ...). Il permet le stockage, la manipulation, l'interrogation, le partage, la diffusion, la restitution à l'écran et sur papier d'informations à composante géographique. Etant donné que les besoins en information géographique ont fortement évolué particulièrement dans la dernière décennie, qui permet de produire des bases de données de

niveaux de détail plus fins, mais ce type de base de données ne satisfait bien les besoins des utilisateurs. En conséquence, nous devons appliquer certains traitements pour que ces bases de données permettent s'adapter aux besoins particulier des utilisateurs

Parmi les traitements les plus importants effectués sur les bases de données nous pouvons citer ; la généralisation. La généralisation des données géographique est le processus qui consiste à dériver à partir des données géographiques détaillées, des données moins détaillées adaptées aux besoins. Beaucoup de travaux de recherches ont été réalisés pour automatiser ce processus. L'automatisation de ce processus consiste alors à déterminer quelles séries d'actions à appliquer sur les objets de base de données en fonctions des certains contraintes cartographiques. **Ruas (2004)** définit trois questions principales que l'ordinateur doit répondre tout seul pour automatiser le processus de la généralisation :

- Quels objets doivent être généralisés ?
- Quels sont les objets à traiter à un instant donné ?
- Quel algorithme doit-il être appliqué à instant donnée et avec quelle valeur paramétrique ?

Disposer d'un processus le plus automatique possible de généralisation peut permettre de grandement limiter le temps nécessaire à la généralisation et de pouvoir dériver à partir d'une unique base de données très détaillée (dite de référence), toutes les bases de données souhaitées (**Ruas, 2002**). Notre travail consiste à améliorer le résultat de ce processus en qualité de données et en temps d'exécution. Par conséquent, notre objectif principal consiste à optimiser le processus de la généralisation de données spatiales.

2. Objectif de la thèse

2.1. Motivation

Notre travail s'inspire des travaux qui favorisent d'utiliser l'approche AGENT pour automatiser le processus de généralisation des données géographiques, comme le modèle de (**Duchêne et al., 2012**) qui se base sur l'exploration d'états par essai/erreurs pour déterminer la meilleure séquence d'action à appliquer par l'agent pour se généraliser. Mais la méthode proposée ne permet pas à l'agent de choisir l'action optimale à effectuer.

Nous pensons que si nous intégrons l'agent à une méthode d'optimisation. Cela pourrait lui permettre de choisir l'action optimale à effectuer. Dans notre nouvelle approche, nous avons proposé d'utiliser un algorithme génétique qui va aider l'agent à déterminer quelle est la séquence d'actions optimales à exécuter pour améliorer son état actuel qui va en général optimiser le processus de généralisation global. Aussi, nous avons étudié les transformations

des relations topologiques durant le processus de la généralisation, selon un modèle topologique basé sur un ensemble d'assertions mathématiques.

Dans notre approche, les agents interagissent afin de résoudre les problèmes qui ne sont pas résolus individuellement. Chaque agent correspond à un objet dans la carte. En outre, en raison d'utiliser le processus de généralisation à la généralisation à la volée, nous avons besoin de réduire le temps de traitement nécessaires à ce processus. Ce temps doit être conforme au délai d'attente donné par (Newell, 1990). Par conséquent, une autre motivation consiste à trouver une solution qui permet de réduire le temps d'attente des usagers. Cette solution doit être indépendante des contraintes techniques telles que ; la bande passante de connexion et la vitesse des processeurs.

2.2. Objectif

L'objectif principal de cette thèse consiste à proposer une approche basée agent qui est améliorée par l'utilisation des algorithmes génétiques dont l'objectif est de permettre à l'agent de choisir les actions à exécuter à chaque moment et selon quelles conditions. Pour réaliser notre objectif, nous devons répondre aux questions suivantes :

- Comment peut-on résoudre tous les conflits cartographiques qui permettent, en effet d'améliorer la qualité des données généralisées ?
- Quelles sont les mesures qui permettent de caractériser les données à généraliser ?
- Comment peut-on satisfaire au maximum les contraintes cartographiques ?
- Comment peut-on offrir des résultats précis autant que possibles en un temps raisonnable ?
- Comment peut-on trouver un compromis entre la qualité des données généralisées et le temps de réponse de processus de généralisation ?
- Comment et sous quelles conditions, on peut transformer les relations topologiques en d'autres relations durant le processus de la généralisation ?

Donc, nous pouvons résumer notre objectif en :

Mise en œuvre d'un modèle de données à base d'agents qui permet d'optimiser le processus de la généralisation des données géographiques.

3. Méthodologie

La **Figure 1** décrit la méthode que nous avons suivie pour développer notre modèle.

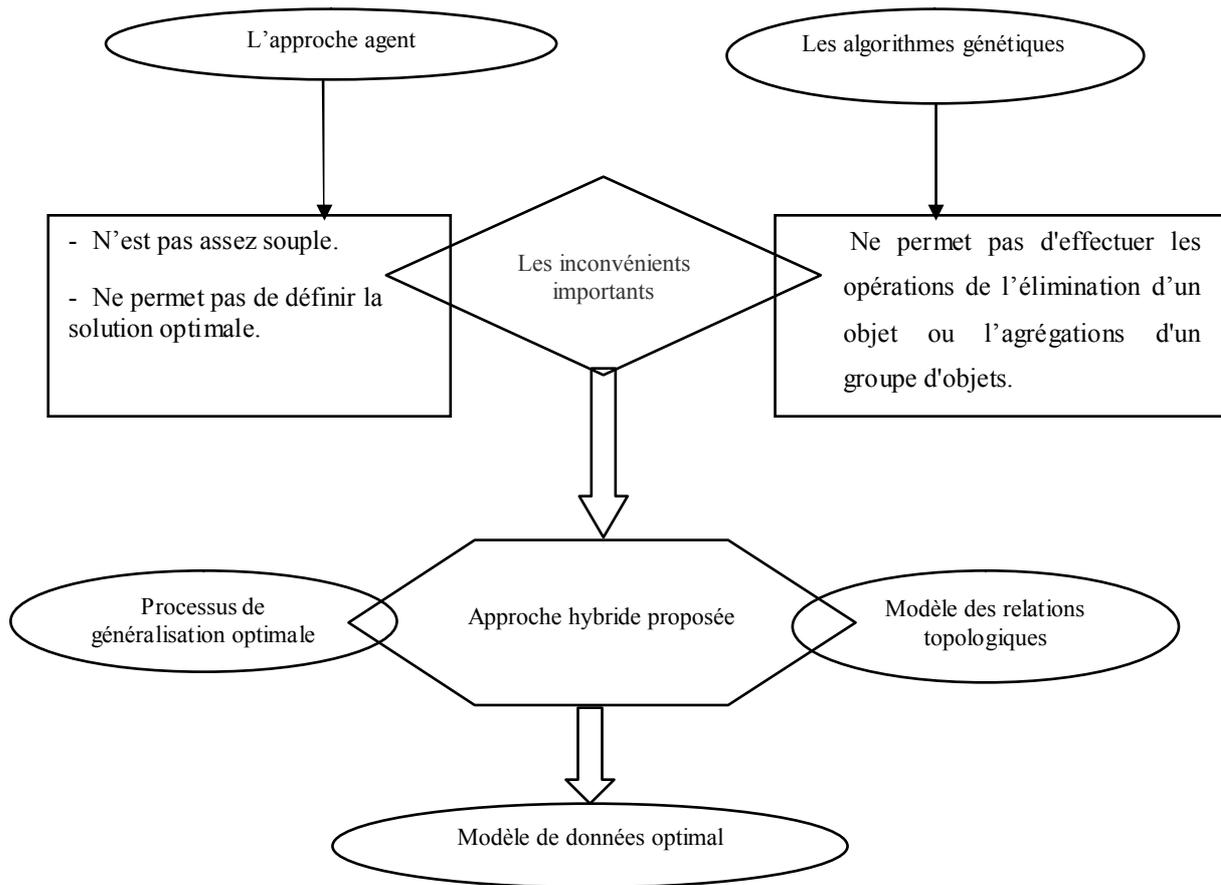


Figure 1. Méthodologie suivie.

4. Plan de thèse

Dans le premier chapitre, nous abordons les concepts de base liés au traitement de l'information géographique. Une étude théorique approfondie concernant les notions de généralisation automatique des données spatiales et de relations topologiques a été réalisée.

Le deuxième chapitre présente un état de l'art de la problématique étudiée et par rapport auquel nous situons notre apport.

Dans le troisième chapitre, nous définissons notre approche et nous décrivons les modèles conceptuels proposés en introduisant de nouveaux concepts.

Dans le quatrième chapitre, nous décrivons le prototype qui a été créé pour réaliser la vérification de la cohérence du modèle de données proposé.

Nous terminons par une conclusion et des perspectives possibles à nos travaux de recherche. Nous avons ajouté une partie d'annexes pour décrire les articles scientifiques que

nous avons fait durant les années de travail. Nous avons fait deux annexes, l'annexe-1 décrit un résumé des objectifs et des contributions de chaque article. L'annexe-2 contient les articles que nous avons publiés dans des revues internationales basés sur les modèles proposés. La **Figure 2** présente l'interaction entre les différents chapitres de ce mémoire.

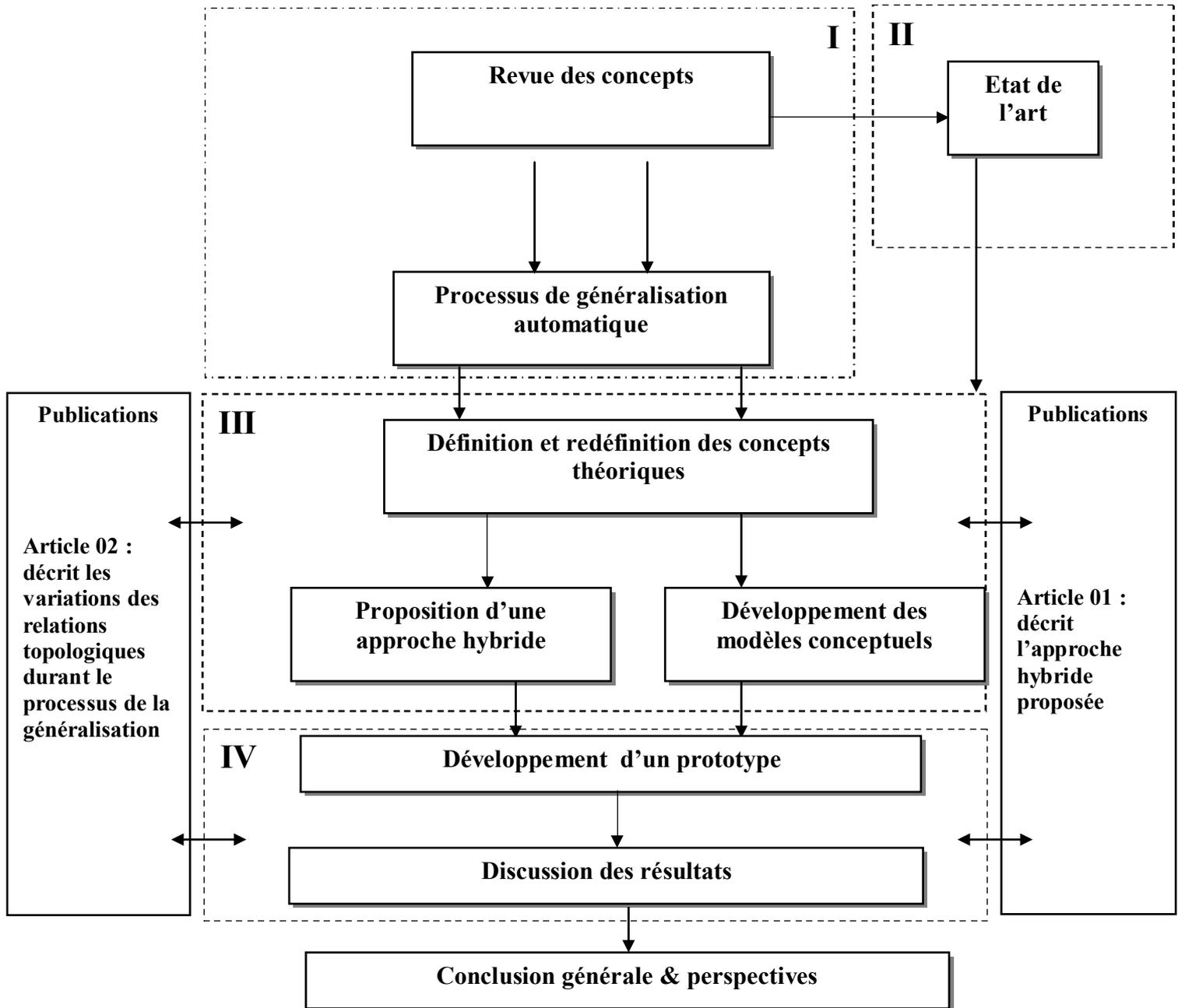


Figure 2. Plan du mémoire.

Chapitre I

Revue des concepts

1. Introduction

L'outil SIG reste l'environnement d'exploitation de l'information géographique organisée en Base de Données géographiques. Cette dernière ne répond plus aux besoins des différents utilisateurs de plus en plus exigeants, tels que ; le niveau de détail, préférences,...etc. La nouvelle aire de recherche consiste à satisfaire cette communauté qui utilise l'information géographique diversifiée en proposant des nouveaux outils. Il s'agit de modéliser le processus de la généralisation de données géographiques.

Dans ce chapitre, nous introduisons des concepts classiques liée à l'information géographique et des concepts spécifiques à la problématique traitée tels que, la notion du niveau de détail, la généralisation et relation topologique...etc.

2. L'information géographique

Nous allons présenter brièvement quelques notions essentielles relatives aux bases de données géographiques.

2.1. Définition

L'information géographique est l'information recueillie dans une base de données géographiques qui comprend une information relative à un phénomène du monde représenté par son aspect et ses caractéristiques, et sa localisation décrite dans un système de coordonnées de référence (**Denègre & Salgé, 1996**).

2.2. Représentation de l'information géographique

Il existe deux grands modes de représentation des données géographiques sous forme numérique : les représentations vecteur et raster (voir la **Figure I-1**).

— **Le mode raster** (ou maillé) se base sur le partitionnement de l'espace en un maillage constitué de cellules rectangulaires (et généralement carrées) appelées *pixels*. Chaque pixel porte une ou plusieurs informations descriptives caractérisant son contenu. La géométrie d'un objet géographique correspond à un ensemble de pixels. Ce mode de représentation est utilisé pour les photographies numériques.

— **Le mode vecteur** consiste à représenter la géométrie des objets géographiques sous la forme d'une primitive géométrique, qui peut être, en deux dimensions, un point (décrit par ses coordonnées dans un repère), une ligne (décrite par une suite de points), ou une surface (décrite par son contour qui est une ligne fermée). Les points constitutifs des lignes peuvent être reliés entre eux par des segments ou par des courbes plus complexes telles que des *splines* ou des courbes de *Bézièrs*. Ce mode de représentation est utilisé par la plupart des logiciels de CAO/DAO.

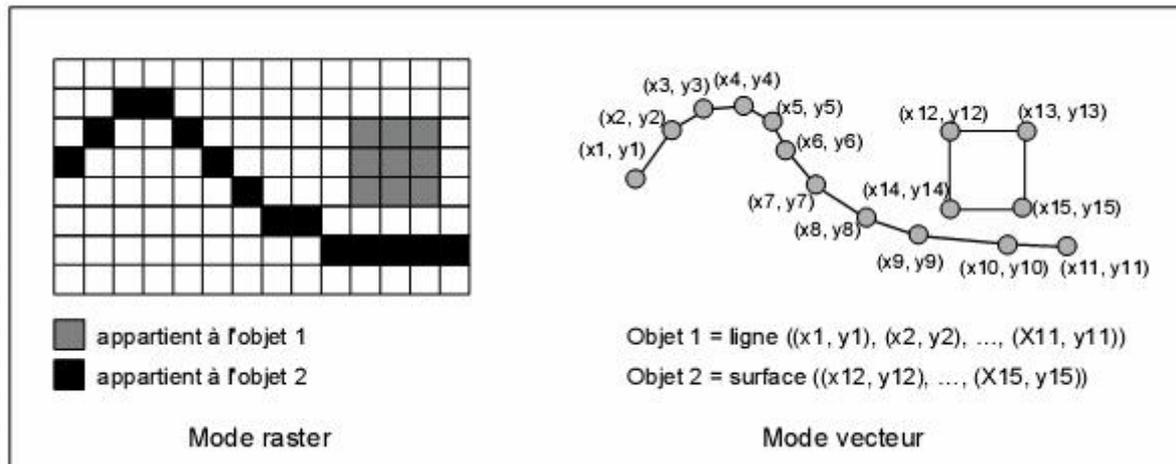


Figure I-1. Deux objets représentés en mode raster et en mode vecteur.

Dans cette thèse, on va aussi utiliser un autre objet baptisé « ruban » (ribbon en anglais) pour modéliser les rivières, les routes et les rues (que l'on appelle des objets linéaires): un ruban peut être défini comme une ligne ou une polyligne avec une certaine largeur. Mathématiquement, un ruban est défini comme un rectangle oblong (longish rectangle) (Laurini, 2014). Le ruban avait un squelette qui est peut être son axe. La Figure I-2 donne un exemple de ruban.

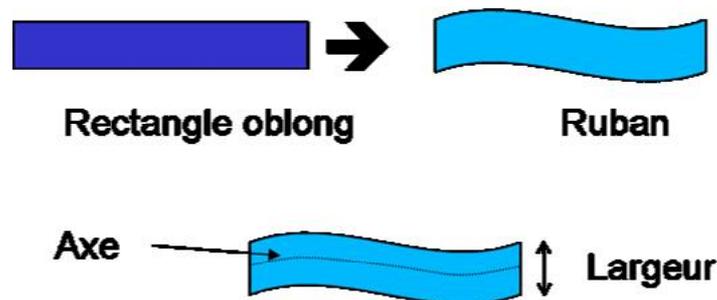


Figure I-2. Un exemple de ruban.

Dans le cadre de cette thèse, nous nous intéresserons uniquement aux données géographiques représentées sous forme vectorielle. Nous désignons donc par « base de données géographiques », une **base de données géographique vectorielle**.

2.3. Bases de donnée géographique

Une base de données géographiques est définie par (Ruas & Libourel, 2002) comme : "Une base de données géo-référencées c'est-à-dire de données qui possèdent des coordonnées permettant de les localiser, de connaître leur position sur un espace limité (la terre, un pays, une ville, un océan, mars, ... etc.)".

Une base de données géographiques est constituée d'un ensemble d'objets organisés en classes d'objets de même nature. Chaque objet de la base de données est décrit par :

- Une géométrie : elle est une primitive géométrique vecteur : point, ligne (ruban) ou surface, c'est-à-dire sous la forme d'une suite structurée de coordonnées de points. Par exemple, une surface peut être représentée par la suite des points de son contour.
- Des données attributaires (champs, ou données sémantiques) : ces données attributaires permettent de compléter la description de l'objet, similaires à ceux que l'on trouve classiquement dans toute base de données. Par exemple, un tronçon routier peut porter un attribut donnant des informations sur le type de la route qu'il représente (départementale, nationale, autoroute, ...etc.).

Il existe principalement deux types de Base de Données Géographiques (BDG) selon la destination de création de telle base de données. On parle de Base de Données Cartographique (BDC) si l'objectif est de cartographier cette Base de données, en anglais c'est "Digital Cartographic Model". Par opposition, le terme "Digital Landscape Model" désigne une base de données géographique non destinée à être cartographiée. Notons qu'il est pertinent de parler de "l'échelle" d'une base de données cartographique, puisqu'une telle base est destinée à être affichée à une échelle donnée (**Duchène, 2004**).

3. Système d'information géographique

L'homme a voulu, depuis longtemps, découvrir son environnement. Pour satisfaire ce besoin, il utilise des techniques afin de représenter et manipuler le flux d'informations géographiques. Différents supports d'informations sont utilisés : cartes topographique, images satellitaires, ...etc. Avec l'avènement des données numériques, l'information géographique a été intégrée dans des bases de données géographiques (BDG) et dans des outils spécifiques appelés systèmes d'information géographique (SIG). Ces deux outils se sont développés conjointement.

3.1. Définitions

Des nombreuses définitions ont été données au système d'information géographique (SIG) (**Pantazis & Donnay, 1996**). Plus précisément deux définitions, semblant très importantes à citer.

La première purement logicielle où un SIG est vu en tant qu'un outil informatique permettant de manipuler des données géographiques. Il dispose un système de gestion de base de données (SGBD), et un ensemble d'outils spécifiques dédiés à la composante spatiale des bases de données géographiques. Il permet de visualiser leur contenu en attribuant des

symboles cartographiques aux différents objets de la base. Il permet ainsi d'effectuer des calculs et des requêtes complexes s'appuyant sur les géométries des objets mais aussi d'appliquer des transformations sur ces dernières. La plupart des SIG proposent un environnement de programmation permettant de coder et intégrer de nouvelles fonctions, par exemple, le ARCGIS dispose le langage VBA, et MapInfo utilise le mpabasic (**EdiGéo, 1992**).

La seconde plus générale où le SIG est vu en tant que système d'information. Il est alors appelé SIRS (Système d'Information à Référence Spatiale). Il est défini comme un "ensemble organisé globalement comprenant des éléments (données, équipements, procédures, ressources humaines) qui se coordonnent, à partir d'une référence spatiale commune, pour concourir à un résultat" (**Bédard, 1982**) .

Un SIG permet d'analyser les données grâce à des outils d'analyse spatiale. Les calculs peuvent être simples (longueur d'une ligne, distance euclidienne entre deux objets), ou plus complexes (calcul d'intersections entre objets, recherche du plus court chemin dans un réseau, ...etc.). Il permet également de transformer les données en modifiant leur géométrie. Là aussi, les outils de transformation peuvent être simples (édition d'une ligne point à point), ou plus complexes (algorithme de simplification du contour,... etc.). La plupart des SIG disposent d'un langage de programmation qui permet de coder de nouvelles fonctions d'analyse ou de transformation.

Aussi, les SIG ouvrent plusieurs possibilités qui satisfont les besoins croissants des utilisateurs comme l'aide à la décision, à la gestion du territoire, à la planification dans divers domaines :

- Urbanisme : aménagement, gestion des plans d'occupation du sol (POS), gestion territoriale et municipale....
- Etudes économiques et socio-économiques : population, logement, géomarketing....
- Réseaux de communication et de transport : électricité, eau, routes, fleuves, réseaux souterrains, suivi et calcul d'itinéraires
- Applications militaires.
- Science de la terre : géologie, géotechnique....

3.2. Fonctions d'un SIG

Un SIG intègre généralement cinq fonctionnalités de base dites "5 A" (**Denègre & Salgé, 1996**).

- l'**Acquisition** pour la collecte des données grâce à des fonctions de saisie des données sous forme numérique
- l'**Archivage** grâce à un système de gestion de bases de données (SGBD)
- l'**Analyse** par des fonctions de manipulation, croisement et transformation des données spatiales au moyen de requêtes dans le SGBD
- l'**Affichage** pour la restitution des résultats par des fonctions de mise en forme et de visualisation
- l'**Abstraction** par des fonctions rendant compte de la modélisation de la réalité.

L'apport des SIG par rapport à la cartographie traditionnelle est d'une part le stockage d'une quantité importante de données et d'autre part une capacité de traitement de l'information accrue.

3.3. Organisation des données en couche

Il est fortement nécessaire d'organiser ou de structurer l'information géographique sous forme de couches. On définit une couche dans un SIG comme étant un ensemble d'un ou plusieurs thèmes associés au même territoire. On peut donc avoir des couches mono-thèmes ou multi-thèmes selon les applications des SIG considérés.

Par exemple, les thèmes : hydrographie, occupation du sol, orographie, agglomérations et géologie partagent le même territoire d'une commune (voir la **Figure I-3**). Bien qu'ils soient différents par leurs modes de saisie (méthode, précision, échelle...etc.) et par la fréquence de leur mise à jour (parfois de façon séparée), pour des raisons d'analyse spatiale, ces thèmes peuvent être associés, non seulement par voie de leur affichage à l'écran, mais également à l'intérieur de la base de données.

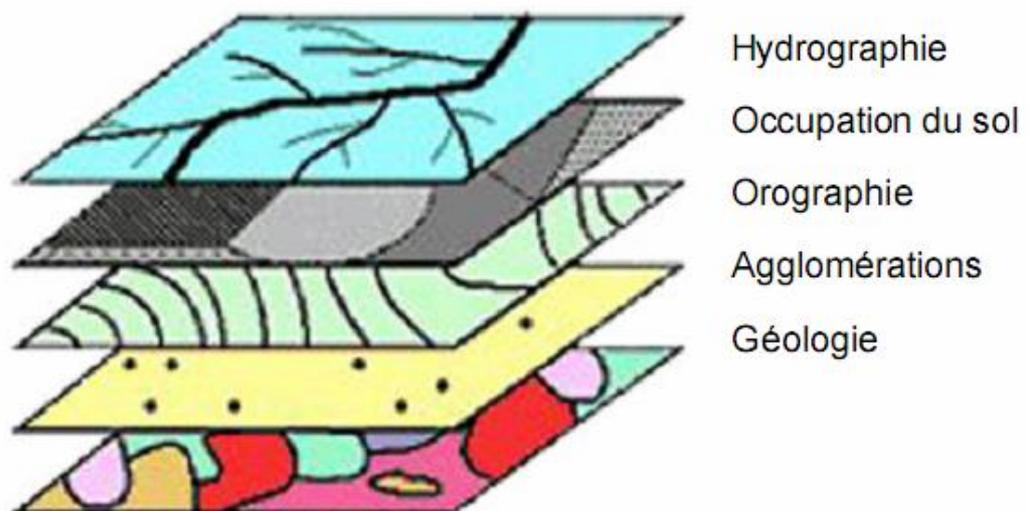


Figure I-3. Organisation des données en couches.

Le choix du nombre de couches ainsi que celui des thèmes à associer à une couche sera souvent un compromis entre efficacité lors de l'interrogation, difficulté de la construction et évolution de la couche. Il est clair qu'il ne faut pas mettre ensemble tous les thèmes dans une seule couche. Des thèmes consultés ensemble doivent constituer une couche. Mais une couche comportant trop de thèmes est illisible à l'écran et prend trop de temps à créer et à mettre à jour, voire à consulter.

4. Généralisation des données géographiques

Dans cette section, nous allons introduire les notions liées à la généralisation cartographique.

4.1. Echelle et niveau de détail

L'échelle de la carte représente le facteur d'homothétie existant entre la taille des entités sur le terrain et sur la carte. Plus l'échelle d'une carte est grande, plus la taille des objets est grande sur la carte. La **Figure I-4** présente des exemples de cartes de la même zone à différentes échelles.

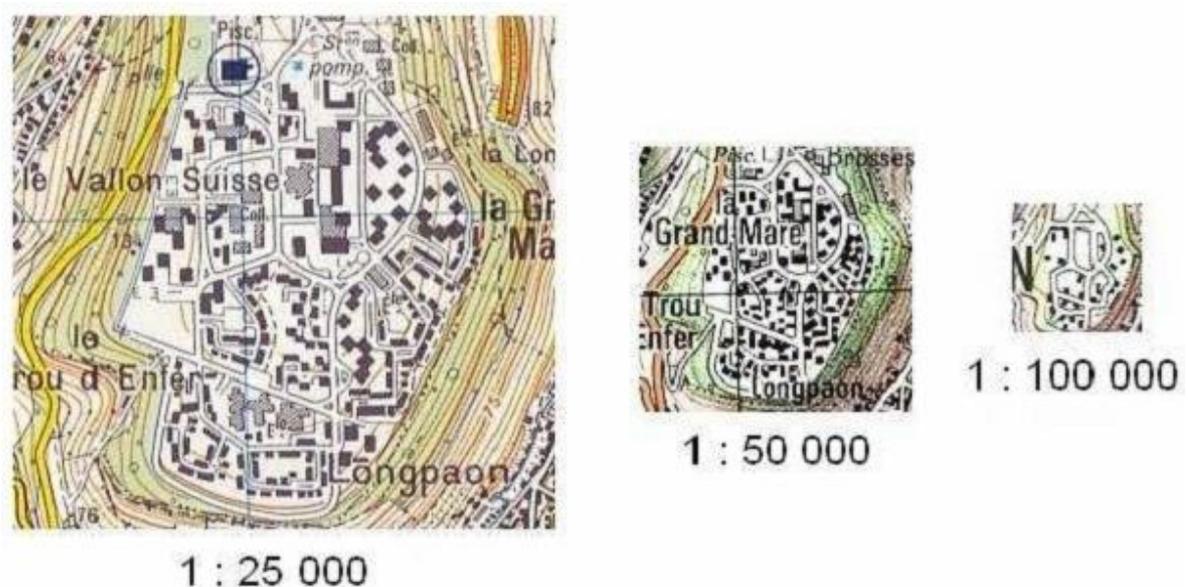


Figure I-4. Extraits de cartes à différentes échelles (P. Taillandier, 2008).

Contrairement à une carte, une base de données géographiques n'a pas d'échelle intrinsèque, puisqu'elle peut être affichée à différentes échelles simplement grâce aux outils de zoom des SIG. Dans certaines bases de données, on parle de l'échelle de référence. Elle désigne l'échelle la plus adaptée pour la visualisation des données (visualisation à l'écran avec un SIG ou sortie papier). On parle aussi du niveau de détail, il est lié à son échelle de référence. Il traduit la finesse des données contenues dans la base. **Ruas & Bianchin (2002)** distinguent deux types de niveaux de détail :

— Le niveau de détail géométrique : qui correspond à la finesse de description de la géométrie des objets. Il est lié au type d'implantation choisi pour représenter les géométries (ponctuelle, linéaire, surfacique,... etc.), à la précision des données géométriques (écart entre la position des objets et leur position réelle) ainsi qu'à leur granularité (taille du plus petit détail de forme) (**Gaffuri, 2008**).

— Le niveau de détail sémantique : qui correspond à la finesse de processus de l'abstraction. Il concerne le choix des objets représentés dans la base de données.

L'apparition des technologies avancées de l'acquisition des données géographique permettant d'acquérir des données en niveau de détail de plus en plus fin. Ces progrès permettent ainsi de décrire de mieux en mieux l'espace. Néanmoins, une base de données en un niveau de détail très fin n'est pas toujours convenable aux besoins des utilisateurs. Donc, un niveau de détail plus faible est souvent plus adapté aux besoins d'utilisateur, car il fait apparaître que les concepts nécessaires à ses besoins. Donc, parfois, il est très important de diminuer le niveau de détail de données, pour cela on utilise des méthodes, dites de généralisation, que nous allons présenter dans ce travail.

4.2. Définitions

Nous avons deux types de généralisation.

4.2.1. Généralisation de données géographiques

Plusieurs définitions ont été données à la généralisation.

1) La généralisation peut être comparée au processus de résumé de texte, cette définition a été donnée par (**Ruas, 2002**) :

« La généralisation est un processus de synthèse d'information. On peut comparer ce processus avec celui d'un résumé de texte dont les objectifs sont de réduire le nombre de mots, de retenir les idées principales, de ne pas faire de faux sens et si possible de conserver le style de l'auteur. Evidemment, le texte doit respecter les règles du langage tant au niveau de l'orthographe que de la grammaire. Pour bien généraliser l'information initiale, il faut également réduire la quantité d'information, mettre en valeur l'information la plus importante, rester fidèle à l'information initiale et, dans le cas d'une carte, respecter les règles de sémiologie qui permettent une bonne lecture de l'information ».

2) Pour une base de données, la généralisation permet de diminuer le niveau de détail :

« La généralisation de données géographiques est le processus qui consiste à simplifier l'information contenue dans une base de données géographique, pour répondre à un besoin donné (par exemple, l'analyse d'un phénomène géographique à un niveau d'abstraction

plus large que celui de la base de données initiale). On diminue ainsi le niveau de détail des données d'origine, tout en mettant en exergue les éléments importants au regard de ce besoin » (Duchène, 2004).

Pour le processus de généralisation de données géographiques, Duchène (2004) distingue deux types d'opérations :

- Les opérations qui agissent sur le schéma de données : suppression d'une classe, fusion de classes ou de valeurs d'attributs (par exemple les "céréales", "vignes" et "vergers" deviennent des "cultures"), changement de mode d'implantation spatiale d'une classe (par exemple la classe surfacique "rivière" devient linéaire).
- Les opérations qui agissent sur les objets eux-mêmes : élimination d'objets au sein d'une classe, agrégation d'objets, simplification de la géométrie des objets.

4.2.2. Généralisation à but cartographique

La généralisation cartographique aussi appelée généralisation à but cartographique, est un type particulier de généralisation de données géographiques qui vise à produire une carte (peu importe qu'elle doive être affichée à l'écran ou imprimée sur papier).

Avant l'apparition des données numériques, la généralisation cartographique était utilisée pour créer une carte à une échelle donnée à partir d'une autre carte à une échelle plus grande. De nos jours, la généralisation cartographique peut aussi être utilisée pour la création d'une carte à partir d'une base de données géographique trop détaillée par rapport à l'échelle souhaitée (voir la Figure I-5).

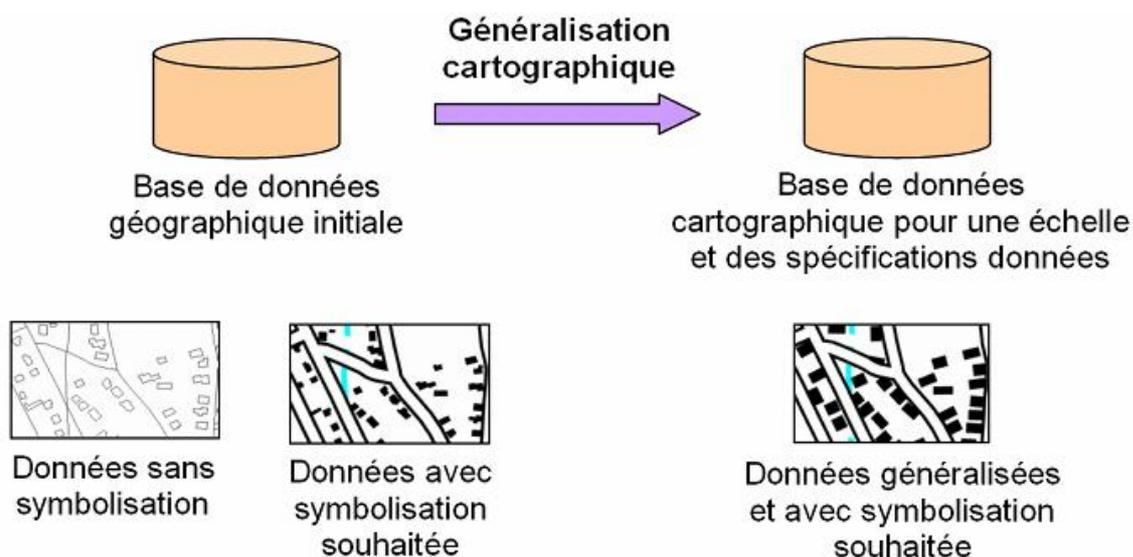


Figure I-5. Généralisation cartographique (Taillandier, 2008).

On applique la généralisation si on souhaite obtenir une carte ayant une échelle plus petite que l'échelle de référence de la base de données. Elle est motivée par les limitations physiologiques et cognitives de l'être humain (**Duchène, 2004**).

Les limites physiologiques concernent les limitations de l'œil humain. En effet, même sans tenir compte des limites de résolution des écrans et des imprimantes, l'œil humain ne peut percevoir des détails trop petits. Ainsi, certains objets trop petits peuvent ne plus être perçus une fois le passage à l'échelle souhaitée accompli. De même, des problèmes de séparation et de différenciation entre objets peuvent apparaître. Et les limites cognitives concernent la perception humaine qui concerne la difficulté de l'être humain à analyser simultanément de grandes quantités d'informations. Les capacités d'analyse de notre cerveau sont très inférieures à nos capacités visuelles : une carte efficace doit faire ressortir uniquement l'information importante. Une carte est considérée comme difficilement analysable lorsqu'elle contient trop d'éléments, ou lorsqu'elle n'a pas de structure globale claire (**Eastman, 1985**).

L'objectif de la généralisation cartographique est de rendre lisibles des données géographiques compte tenu des spécifications de la carte définies par l'utilisateur. Une généralisation particulière doit ainsi prendre en compte des éléments tels que l'échelle de la carte et la symbolisation souhaitées, les classes à représenter et les besoins de l'utilisateur (exemple: privilégier la conservation des positions des objets plutôt que leur forme), ...etc. (voir la **Figure I-6**).

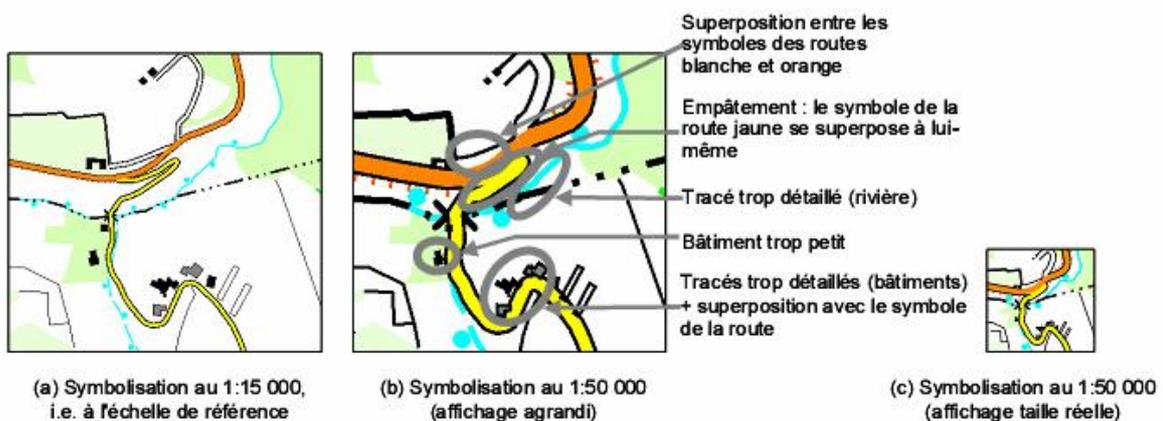


Figure I-6 . Problèmes de lisibilité lorsqu'on affiche des données avec une échelle de symbolisation nettement inférieure à l'échelle de référence (**Duchène, 2004**).

4.3. Contraintes cartographiques

Comme nous l'avons dit précédemment, la généralisation consiste à appliquer des transformations sur des données géographiques afin que celles-ci soient lisibles tout en

conservant les informations importantes des données initiales (**Taillandier, 2008**). Faire une meilleure généralisation revient à satisfaire un ensemble de contraintes cartographiques ((**Beard, 1991**) et (**Ruas, 1999**)) qui peuvent concerner aussi bien des problèmes de lisibilité que des problèmes de conservation de l'information importante. Donc, les objets de la base de données cartographiques doivent respecter un certain nombre de contraintes cartographiques. Une contrainte cartographique peut se diviser en deux catégories :

- Contraintes concernent un objet isolément (exp. pour qu'un bâtiment représenté par une surface soit lisible, sa taille doit être suffisamment grande et son tracé ne doit pas être trop détaillé).
- Contraintes concernent plusieurs objets (exemple. le symbole d'une route ne doit pas se superposer à celui d'une maison).

Si une contrainte cartographique n'est pas respectée, cette situation engendre un conflit cartographique. Ces conflits imposent une approche de solution pour être résolus. Les conflits cartographiques peuvent être résolus par une combinaison d'actions possibles telles que l'élimination, le déplacement, la fusion et la simplification, combinée avec des techniques appropriées pour évaluer la qualité du résultat. Toutefois, l'application d'un opérateur individuel peut avoir un effet sur des objets de la carte qui n'étaient pas précédemment en conflit, ce qui entraîne la propagation des conflits au sein de l'espace de la carte, c'est-à-dire l'effet de l'absence de dialogue entre les objets.

Les contraintes cartographiques guident le processus de généralisation : le cartographe s'attache à résoudre les conflits cartographiques présentés par les données d'origine, sans en faire apparaître de nouveaux. Il est en général impossible de satisfaire toutes les contraintes cartographiques considérées. En effet, certaines contraintes sont contradictoires. Par exemple, la contrainte de non-superposition entre symboles conduit à déplacer les objets pour les éloigner les uns des autres, alors que la contrainte de maintien de la position initiale suppose qu'ils soient déplacés le moins possible. Donc, généraliser consiste à trouver le meilleur compromis possible entre les contraintes, compte tenu des besoins concernant les données dérivées.

4.4. Relations topologiques

La topologie est définie comme l'étude mathématique des propriétés qui sont conservées pendant les déformations ou élargissement d'objets géométriques. La topologie est une branche des mathématiques, mais certains concepts de topologie ont une grande importance dans la généralisation cartographique, tels que les relations topologiques (**Harrie,**

2001). Les relations topologiques décrivent les relations entre tous les objets dans l'espace, les points, les lignes (ruban) et les zones pour tous les types possibles de déformation. Plusieurs chercheurs ont défini les relations topologiques dans le contexte de l'information géographique (**Egenhofer & Franzosa, 1991**), (**Clementini et al., 1993**) et (**Winter & Frank, 2000**).

Plusieurs méthodes de généralisation cartographiques ont été développées durant la décennie précédente, mais la plupart de celles-ci sont applicables aux seuls objets simples. Les relations entre les objets d'une carte, ou d'une base de données cartographique sont très importantes et doivent être pris en compte dans le processus de généralisation. Dans cette thèse, nous allons traiter les relations topologiques entre les objets (ruban et/ou régions). Nous allons aussi proposer un modèle topologique basé sur la règle condition-action (Si <condition A> faire <action B>).

4.5. Opérateurs de généralisation

La clé de la généralisation est l'application des opérateurs qui permettent de transformer les données géographiques en modifiant leurs géométries et/ou leurs descriptions. Plusieurs classifications ont été proposées dans ((**McMaster & Shea, 1992**), (**Ruas, 1999**), (**Bader, 2001**), (**Li, 2007**) et (**Foerster & Stoter, 2007**)) pour distinguer les différents types d'opérateurs de généralisation. On adopte dans cette thèse, celle proposée par (**Mustière, 2001**) qui divise les opérations de généralisation en trois grandes familles :

- Les opérations de simplification (**Figure I-7**) qui visent à réduire la quantité d'informations en éliminant celles peu importantes.
- Les opérations de caricature (**Figure I-8**) qui visent à accentuer certains caractères des objets au détriment d'autres.
- Les opérations d'harmonisation (**Figure I-9**) qui visent à éliminer les différences entre les objets.

Pour chaque opérateur, il existe une famille d'algorithmes. Chaque algorithme possède des valeurs paramétriques qui permettent d'appliquer un processus de généralisation optimale. Des mécanismes de choix des algorithmes et leurs valeurs paramétriques sont un objectif clé parmi les objectifs de cette thèse.

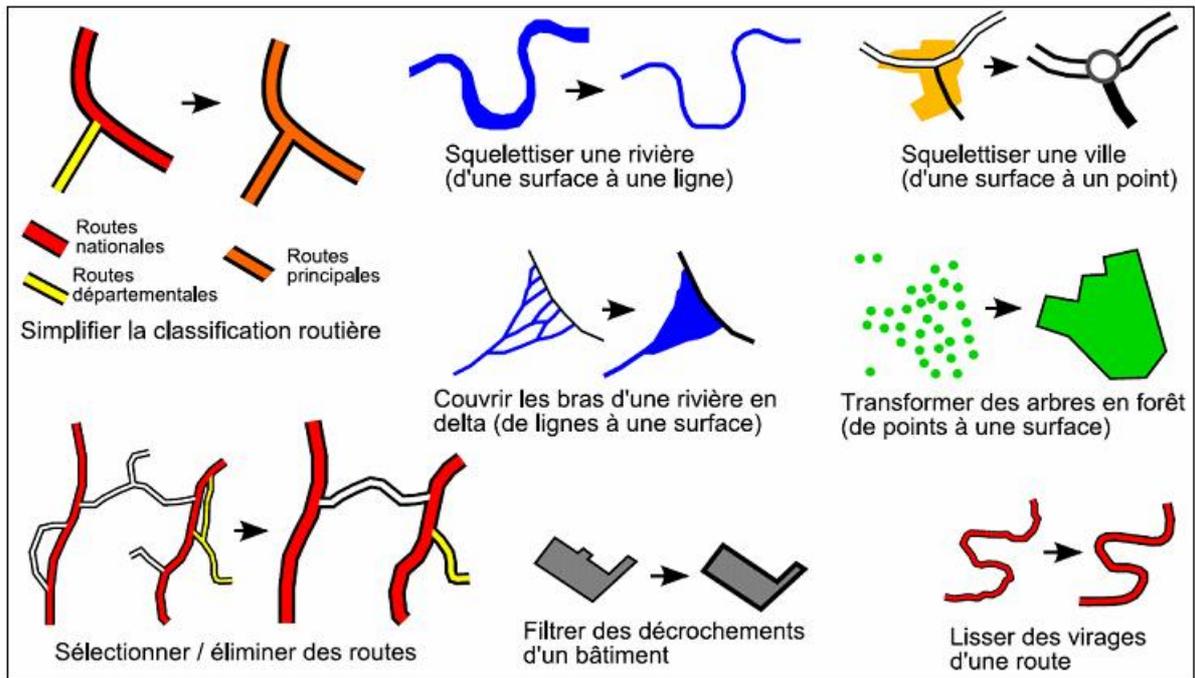


Figure I-7. Exemples d'opérations de simplification (Mustière, 2001).

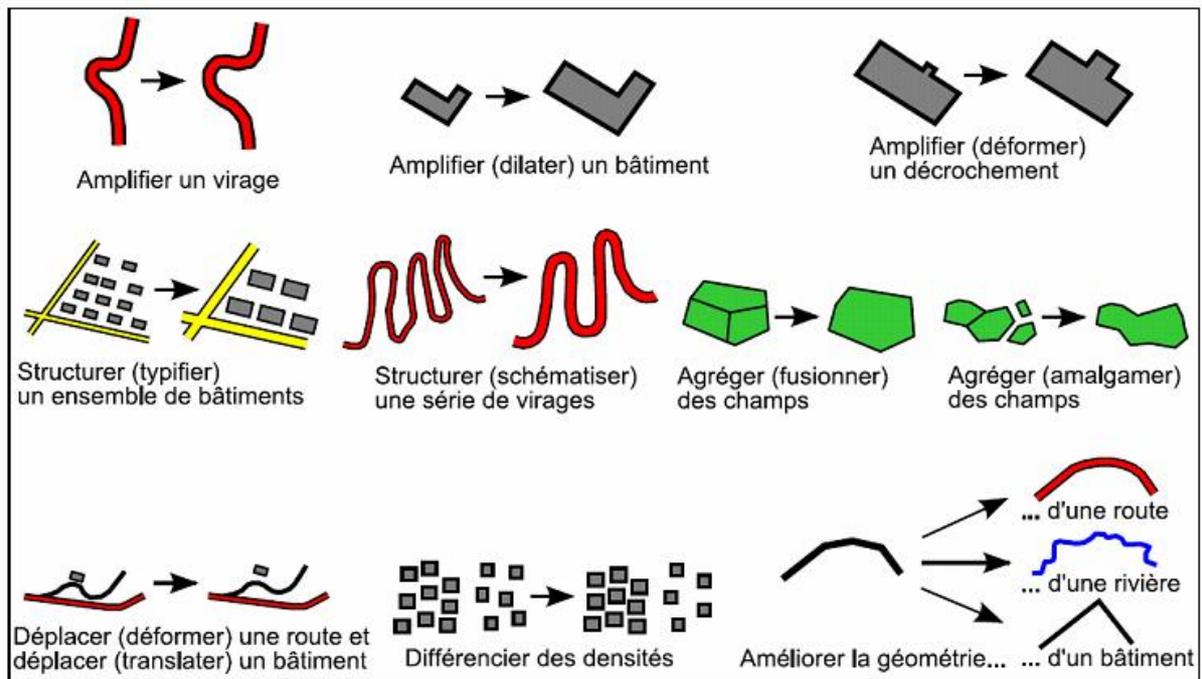


Figure I-8. Exemples d'opérations de caricature (Mustière, 2001).

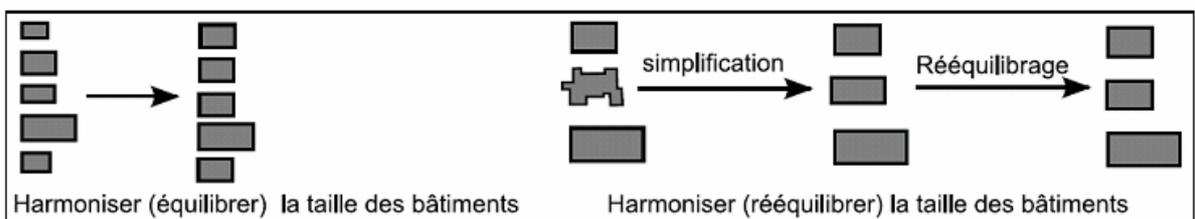


Figure I-9. Exemples d'opérations d'harmonisation (Mustière, 2001).

5. Conclusion

Nous avons présenté dans ce premier chapitre; les concepts liés à l'information géographique, sa description et sa manipulation au cœur des systèmes d'informations géographiques. Nous nous sommes focalisé sur les concepts liés à la problématique traitée; la généralisation de l'information géographique, cette dernière peut être dérivée en appliquant un processus de la généralisation.

Le prochain chapitre sera donc consacré à l'**état de l'art**, nous présentons une synthèse des travaux de recherches réalisés à la modélisation d'un processus de généralisation optimale.

Chapitre II

Etat de l'art

1. Introduction

Les objets géographiques représentés sur une carte sont une approximation de la réalité (ils subissent plusieurs changements dans leur géométrie (simplification de forme, dilatation, déplacement,...etc.)) car ces objets sont de taille inférieure à leur taille réelle d'un facteur qui est celui de l'échelle de la carte, rappelons que l'échelle est "le facteur d'homothétie entre la taille des entités sur le terrain et la taille de leurs représentations sur la carte". Lorsque l'échelle est petite, certaines informations sont éliminées pour que les objets représentés ne rentrent pas en collision les uns avec les autres, alors il y a une réduction de la quantité d'informations (remplacement des bâtiments par une zone urbaine, réduction du nombre de rue et de route, ...etc.).

Dans les bases de données, ce problème ne se pose pas parce qu'il n'y a aucune contrainte de place. En effet, le problème d'échelle se transforme plutôt en un problème de niveau de détail sémantique et géométrique que nous avons brièvement décrit dans le premier chapitre. L'adaptation d'une représentation dépend de l'objectif de l'utilisateur concerné et de son champ d'application, ceci juge que les besoins de l'utilisateur (niveau de détail, préférences et contexte d'utilisation) constituent des facteurs fondamentaux dans la représentation de l'information dans une base de données géographiques. La généralisation automatique représente la meilleure solution pour donner à l'utilisateur des données adaptables à ces besoins à partir d'une base de données géographiques.

Rappelons que la généralisation de données géographiques est le processus qui consiste à dériver à partir de données géographiques détaillées, des données moins détaillées, adaptées à un besoin particulier. Il s'agit d'un processus très long et complexe. Son automatisation est un enjeu majeur pour les chercheurs dans ce domaine. De nombreux travaux de recherche ont porté sur ce problème. Parmi ceux-ci, certains se basent sur les techniques d'optimisation et d'autres se basent sur l'approche agent.

Dans ce deuxième chapitre nous présentons un état de l'art de la problématique, par rapport auquel nous situons notre apport.

2. Automatisation de la généralisation

La généralisation est un processus complexe car les décisions de généralisation (appliquer quel algorithme? avec quels paramètres ou, à quel moment) dépendent du type des objets géographiques de la carte – qui sont très variés - et des besoins qui sont en général difficilement formalisables. On peut aussi dire plus simplement que lorsqu'on généralise, on passe d'une information exhaustive à une information représentative : on déforme localement

l'espace, on réduit la quantité d'information en tentant de conserver une représentation qui « fait penser à » la représentation initiale.

2.1. Besoin d'automatisation de la généralisation

Parfois, les utilisateurs veulent avoir des données moins détaillées car elles sont plus utiles que des données très détaillées pour satisfaire leurs besoins. Si l'utilisateur veut consulter de façon lisible sa carte, il a besoin de réduire la densité de données en utilisant un processus de la généralisation automatique. En fonction des besoins de l'utilisateur, des données avec un niveau de détail plus général, sont nécessaires, car elles font apparaître des concepts utiles qui ne sont pas explicites dans les données les plus fines (**Gaffuri, 2008**). Le choix du niveau de détail peut avoir un impact sur le résultat de la généralisation.

La généralisation s'inscrit comme l'un des processus indispensables, tant pour la réalisation de produits dérivés que pour l'intégration de données multi-sources. S'il existe de nombreux algorithmes, aucun système ne permet d'automatiser le processus de généralisation, essentiellement parce que la méthode de résolution ne se décide pas a priori mais est en fonction des caractéristiques de chaque conflit. De plus le choix d'un algorithme dépend des caractéristiques géométriques des objets à modifier. Nous pensons qu'il est donc indispensable d'analyser et de maintenir non seulement les propriétés géométriques mais aussi les relations spatiales (exp. connexité, proximité, inclusion,...etc.) pour pouvoir automatiser la généralisation.

Disposer d'un processus le plus automatique possible de généralisation, peut permettre de grandement limiter le temps nécessaire à la généralisation et de pouvoir dériver à partir d'une unique base de données très détaillée (dite de référence), toutes les bases de données souhaitées (**Ruas, 2002**).

2.2. Enjeu de l'automatisation de la généralisation

Traditionnellement, pour que le cartographe génère une nouvelle carte, il effectue la généralisation cartographique manuelle sur une carte existante. Ce travail manuel est très complexe et délicat.

Toutefois, au cours des trente dernières années, les progrès technologiques ont encouragé les recherches sur la généralisation cartographique automatique afin de faciliter et accélérer la dérivation d'une nouvelle carte, et donc diminuer le travail dur des cartographes. Ces recherches peuvent être associées à trois époques principales (**Kilpelainen, 2000**). Durant la période (1960-1975), des études ont porté sur le développement d'algorithmes afin de mettre en œuvre des différents opérateurs de généralisation cartographique (**Douglas &**

Peucker, 1972). Ensuite, dans les années 1980, les chercheurs ont examiné à la fois l'efficacité des algorithmes existants, ainsi que les aspects de conception (**McMaster & Shea, 1988**).

Enfin, depuis les années 90, la troisième époque a été marquée par l'utilisation des techniques d'intelligence artificielle pour la conception de systèmes à base de connaissances ((**Buttenfield & McMaster, 1991**), (**McMaster, 1991**), (**Weibel, 1991**)), ainsi que l'utilisation de réseaux de neurones (**Allouche & Moulin, 2001**), les algorithmes génétiques (**Wilson et al, 2003**), et enfin les systèmes multi-agents ((**Ruas, 1999**), (**Lamy et al., 1999**) (**Maozhen et al., 2002**),(**Ruas & Duchêne, 2007**) et (**Duchêne et al., 2012**)).

2.3. Problématique

Nous présentons tout d'abord les algorithmes et leurs paramétrages utilisés dans le processus de la généralisation. Nous montrons ensuite l'utilité d'utiliser les mesures cartographiques de la qualité de la généralisation. Enfin, nous présentons la distinction entre la notion d'algorithme et du processus de la généralisation.

2.3.1. Algorithmes de généralisation et leurs paramétrages

Les premières recherches visant à automatiser la généralisation cartographique se concentrent principalement sur l'implémentation d'algorithmes qui modélisent les opérations élémentaires de généralisation effectuées par le cartographe. Disposer d'algorithmes de généralisation n'est pas en soi suffisant pour assurer l'automatisation complète du processus de généralisation. Néanmoins, comme nous l'avons présenté au début, le problème est de savoir sur quel(s) objet(s), dans quelles conditions, appliquer quels algorithmes, et avec quelles valeurs paramétriques. Ce problème est abordé par (**Brassel & Weibel, 1988**), et par (**Shea & McMaster, 1989**) dont le titre pose explicitement la même question : "When and How To Generalize ?". De ces deux travaux de recherches, il ressort que la manière de généraliser des données dépend des spécifications du produit final (en particulier, de l'échelle et de l'utilisation qu'on veut en faire), mais aussi des données elles-mêmes : relations structurantes existant entre les objets, et les conflits cartographiques à résoudre. On ne peut spécifier un processus de généralisation (quels algorithmes appliquer à quel moment et sur quels objets) qu'en prenant en compte ces facteurs (**Duchêne, 2004**). Ceci a été confirmé par la suite, par des tests de généralisation interactive réalisés dans plusieurs organismes, dans le cadre d'un groupe de travail de l'Organisation Européenne d'Etudes Photogrammétriques Expérimentales (OEEPE) (**Ruas, 2001**).

Le paramétrage d'un algorithme joue un rôle très important dans le processus de généralisation. Le choix des algorithmes et de leurs paramètres influent de façon directe sur le processus de la généralisation de données. Le choix optimal de ces algorithmes et leurs paramètres qui permet de respecter une telle contrainte et comme objectif, résoudre un tel conflit guidant à finaliser un processus de généralisation optimal. Par conséquent, on s'est concentré dans ce travail sur la proposition d'une approche globale permettant principalement de trouver la séquence optimale d'algorithmes et de leurs paramètres qui guident un processus de généralisation optimale. Pour le faire, on a utilisé les algorithmes génétiques dont le rôle consiste à régler les valeurs paramétriques des algorithmes de généralisation utilisés, selon la configuration géographique souhaitable.

2.3.2. Mesures cartographiques de la qualité

Pour évaluer le respect d'une contrainte cartographique, on doit appliquer certaines mesures. Le choix de ces mesures est aussi très intrinsèque pour effectuer un processus de généralisation optimal car elle évalue la qualité des données généralisées. Elles se basent principalement sur les calculs de tailles d'objets, les distances entre les objets, la position ou l'orientation des objets à manipuler.

Voici quelques mesures utilisées pour évaluer la qualité de la généralisation :

- Sur une ligne : descripteurs de la forme d'une ligne ((**Buttenfield, 1984**) et (**Buttenfield, 1991**)), indicateurs de qualité de la simplification d'une ligne (**McMaster, 1991**), détection des points d'inflexion et des sommets (**Plazanet, 1996**), détection des portions empâtées sur une route (**Mustière, 1998**) et mesures de la forme d'un virage (**Barillot, 2002**) ;
- Sur deux lignes : distance euclidienne et distance de Hausdorff (**Abbas, 1993**) et distance de Fréchet (**Alt & Gadau, 1995**).
- Sur un bâtiment : indicateurs de forme d'un bâtiment ((**Clementini & Di Felice, 1997**), (**Regnauld, 1998**) et (**Rainsford & Mackaness, 2002**)), mesures de l'orientation d'un bâtiment ((**Hangouët, 1998**) et (**Regnauld, 1998**), (**Bader, 1999**) et (**Duchêne et al., 2003**)).
- Sur plusieurs bâtiments : mesures de régularité d'un alignement de bâtiment ((**Regnauld, 1998**) et (**Hangouët, 1998**)) et recherche de voisinages entre bâtiments ((**Hangouët, 1998**) et (**Ruas, 1998**)).

Par conséquent, guider un processus de généralisation optimale, consiste à utiliser des mesures permettant de bien quantifier les données géographiques.

2.3.3. De l'algorithme au processus

Les algorithmes de généralisation et les outils de mesures sont considérés comme de briques de base pour effectuer un processus complet de généralisation. La distinction entre algorithme de généralisation et processus de généralisation se retrouve assez souvent dans la communauté de recherche en généralisation (**Duchène, 2004**). Une bonne distinction entre eux, est donnée dans (**Duchène, 2004**) : un algorithme de généralisation est un algorithme qui réalise une opération élémentaire de généralisation sur un objet ou un groupe d'objets. Mais, un processus de généralisation est un ensemble d'algorithmes de généralisation qui tend à généraliser complètement un objet ou un groupe d'objets, en appliquant plusieurs opérations de généralisation, en fonction des valeurs prises par des diverses mesures utilisées pour caractériser l'espace géographique.

Les premiers processus de généralisation sont apparus au début des années 1990 (**Müller & Wang, 1992**) et concernaient seulement quelques thèmes géographiques. Puis, de nombreux processus de généralisation, plus génériques et pouvant s'appliquer à différents thèmes, ont été proposés.

Nous présentons, dans les sections suivantes, les différentes approches proposées dans la littérature pour l'automatisation et l'optimisation de la généralisation. La classification proposée est directement inspirée de (**Duchène 2004**), (**Gaffuri, 2008**) et de (**Touya, 2011**).

3. Modèles de généralisation

Nous appelons modèle de généralisation un modèle conceptuel (éventuellement implémenté) qui permet d'enchaîner et/ou combiner différentes opérations de généralisations pour des cas d'applications différents. La mise en œuvre de la généralisation pour un cas d'application particulier est considérée comme un processus de généralisation (**Touya, 2011**). Nous décrivons ici, les différents modèles utilisés pour automatiser et optimiser le processus de généralisation :

3.1. Modèles par séquence prédéfinie

L'approche par séquences prédéfinies consiste à définir des séquences d'opérations appliquées sur les objets géographiques. Par la suite, nous décrivons son principe. Puis, nous discutons les résultats obtenus en suivant cette approche.

3.1.1. Principe

Ce modèle se base sur l'approche par « séquence prédéfinie systématique ». Il consiste à prédéfinir des séquences d'algorithmes et l'appliquer sur les données de la base de données géographiques. Le choix d'un algorithme et de ses valeurs paramétriques est valable pour un

nombre important d'objets, souvent tous les objets d'une même classe ou d'un sous-ensemble sélectionné sur un critère simple (Duchène 2004).

Dans ce contexte, le logiciel ARCGIS intègre un outil très important appelé ModelBuilder (Lee, 2003). Il permet de définir graphiquement des séquences d'algorithmes fournis par le logiciel. La Figure II-1 montre un exemple de processus de généralisation qui peut être défini à l'aide de l'outil ModelBuilder du producteur de SIG d'ESRI : (1) sélectionner les courbes de niveau de la classe "Courbes maîtresses", (2) leur appliquer l'algorithme "Simplify Lines", (3) détruire celles qui ont été transformées en un point, (4) lisser les autres à l'aide de l'algorithme "Smooth Lines".

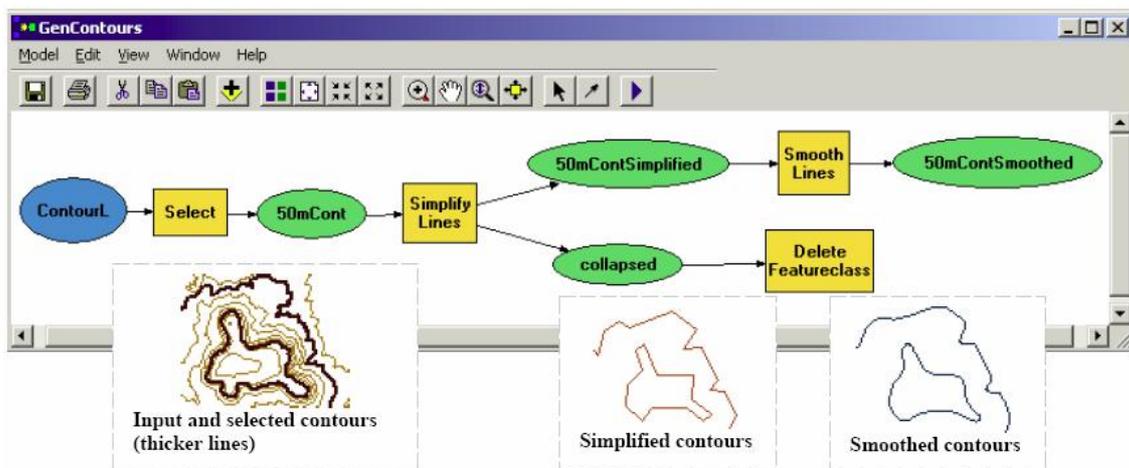


Figure II-1. Un processus de généralisation avec l'outil ModelBuilder (Lee, 2003).

3.1.2. Discussion

Le seul avantage de cette approche est sa simplicité de mise en œuvre, elle nécessite de disposer d'algorithmes de généralisation. Elle fonctionne parfaitement principalement dans le cas où il y a de faibles changements d'échelles, par exemple en zone urbaine (Duchène, 2004). En revanche, dans le cas général, il est très difficile de trouver un séquençement qui soit applicable à un jeu de données géographiques complet, comportant tous les types d'objets géographiques que l'on trouve sur une carte.

3.2. Modèles globaux

Il existe principalement deux modèles globaux.

3.2.1. Modèles par minimisation

Certaines opérations de la généralisation peuvent être vues comme des déformations de l'espace géographique donc des analogies avec des modèles physiques ont été très fertiles en modèles de généralisations. Dans cette section, nous discutons les principes et les résultats obtenus en suivant ces modèles.

3.2.1.1. Principes

Ce modèle se base principalement sur les modèles physiques. Il propose de résoudre le problème en utilisant les coordonnées des points des objets qui composent les objets au lieu des objets eux-mêmes. Les contraintes cartographiques sont traduites sous forme de forces appliquées sur les points, et on cherche à affecter des coordonnées aux points de manière à atteindre un équilibre des forces. Deux méthodes principales ont été proposées pour résoudre ce problème.

a) Généralisation basée sur la méthode par moindres carrés

Dans ce contexte, la méthode de moindres carrés qui est largement utilisée dans l'ingénierie arpentage¹ et photogrammétrie, a été adaptée par plusieurs chercheurs ((**Harrie, 2001**), (**Sarjakoski & Kilpelainen, 1999**) et (**Sester, 2000**)) pour résoudre les problèmes de généralisation.

Cette méthode consiste à modéliser chaque contrainte sous la forme d'une équation dont les inconnues sont les coordonnées finales des points composant les géométries des objets. Les équations dépendent des coordonnées initiales ainsi que des forces appliquées sur les différents points. L'ensemble de ces équations forme un système d'équations sur-contraint qui est résolu par une méthode de minimisation telle que la méthode des moindres carrés ((**Sester, 2000**), (**Harrie, 2001**), (**Harrie, & Sarjakoski, 2002**) et (**Sester, 2005**)) ou la méthode des éléments finis ((**Burghardt, & Meier, 1997**), (**Hojholt, 2000**) et (**Bader et al. 2005**)). La **Figure II-2** donne un exemple des résultats obtenus par la méthode des moindres carrés.

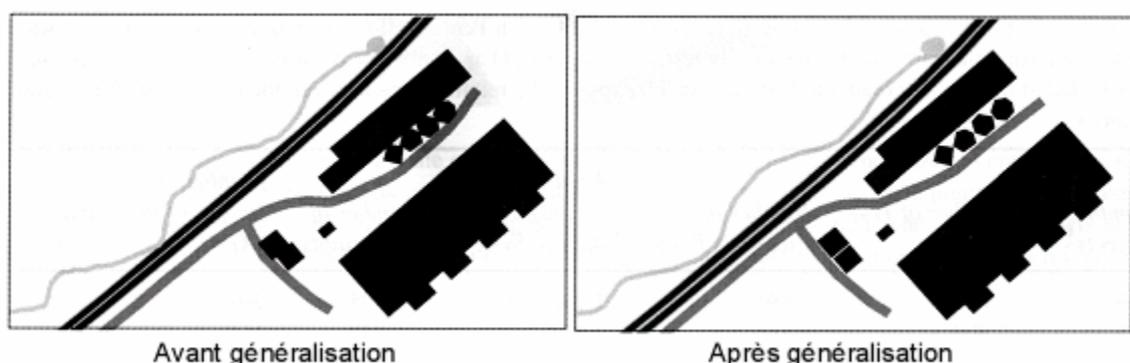


Figure II-2. Données généralisées grâce à une technique de minimisation par moindres carrés (Harrie & Sarjakoski 2002).

Cette méthode apparaît très intéressante pour résoudre les conflits spatiaux. Néanmoins, les approches proposées n'utilisent que l'opérateur de déplacement, elles

¹ L'arpentage est une méthode scientifique traditionnelle de mesure des caractéristiques naturelles ou artificielles de la surface terrestre

permettent la réduction des conflits spatiaux, sans les résoudre définitivement. **Sester (2000)** combine l'utilisation de la simplification et le déplacement des bâtiments, afin de résoudre les problèmes de généralisation. Elle déclare qu'il est nécessaire d'utiliser plusieurs opérateurs de généralisation cartographique en plus du déplacement et de la simplification, afin d'obtenir de meilleurs résultats (**Sester, 2000**).

b) Généralisation basée sur la méthode des éléments finis

La seconde méthode consiste à calculer, à chaque itération, l'ensemble des forces et à déduire pour chaque point, le micro-déplacement résultant de ces forces (**Fritsch, 1997**).

Le processus est réitéré jusqu'à ce qu'un point d'équilibre soit atteint. Les équations ne portent plus directement sur les coordonnées finales souhaitées, mais sur les évolutions élémentaires que subissent les différents points.

3.2.1.2. Discussion

Les approches par minimisation permettent d'obtenir de très bons résultats dans de nombreux cas. L'un de leurs principaux défauts est qu'elles ne permettent pas de recourir à des opérations brusques, telles que l'élimination d'objets, qui sont pourtant essentielles pour les grands changements d'échelle. Un autre problème vient de la prise en compte de l'ensemble des objets durant le processus de la généralisation, car ces approches nécessitent dans la plupart des cas des calculs lourds. Il est donc important de bien partitionner l'espace de façon à limiter la quantité d'objets considérés en même temps. Ce partitionnement peut avoir une grande influence sur la qualité des résultats (**Lemarié, 2003**).

3.2.2. Modèles par optimisation globale

Ce type de modèles se base principalement sur d'exploration locale utilisée en résolution de problèmes d'optimisation. Une solution globale est recherchée par exploration locale de l'espace de solutions à partir d'une solution initiale. Pour obtenir une solution optimale, on doit utiliser des techniques d'optimisation issues de l'intelligence artificielle. La solution retenue est la solution qui permette de rendre optimum la valeur de la fonction d'évaluation. La qualité générale de la solution est calculée de manière globale sous la forme d'une fonction de coût que l'on va chercher à minimiser ou à maximiser. Un inconvénient de ce processus est la possibilité de rester bloqué au niveau d'un minimum local (**Gaffuri, 2008**).

On peut citer principalement deux travaux sous cette classe. Ces travaux sont classés en fonction de la méthode de résolution qu'ils utilisent pour trouver une configuration maximisant la fonction d'évaluation.

3.2.2.1. Généralisation basée sur la méthode du recuit simulé

L'algorithme d'exploration par recuit simulé est un algorithme d'exploration locale utilisé en résolution de problèmes d'optimisation. Par la suite, nous discutons son principe et ses résultats.

a) Principe

Le travail de (Ware et al., 2003) vise à adapter cette méthode à la généralisation automatique. Il utilise la technique du recuit simulé qui est une méta-heuristique permettant de faire varier le taux accepté de détérioration de la fonction de coût (fitness) pour la validation des états. Ils ont proposé une approche basée sur l'utilisation multi-opérateurs de généralisation, tels que le déplacement, l'exagération, la suppression et réduction de la taille, actions qui permettent de modifier l'état de l'objet.

A chaque itération, sont tirés au hasard un objet, un algorithme de généralisation et des valeurs paramétriques. La transformation est appliquée en utilisant les opérateurs de généralisation tels que ; le déplacement, l'exagération, la suppression et réduction de la taille, puis la fonction de coût est recalculée. La transition est validée si l'évolution de la fonction de coût est jugée acceptable. Au début du processus, une dégradation assez importante de la fonction de coût est acceptée, puis au fur et à mesure du processus, la dégradation acceptée est de plus en plus faible. Cela permet d'assurer la convergence vers une bonne solution tout en évitant de tomber dans un minimum local (Duchène, 2004). En fin de processus, une descente de gradient est réalisée, c'est-à-dire que l'on n'accepte plus que des transitions qui améliorent strictement la valeur de la fonction de coût. La Figure II-3 donne un exemple de résultat obtenu par cette approche.

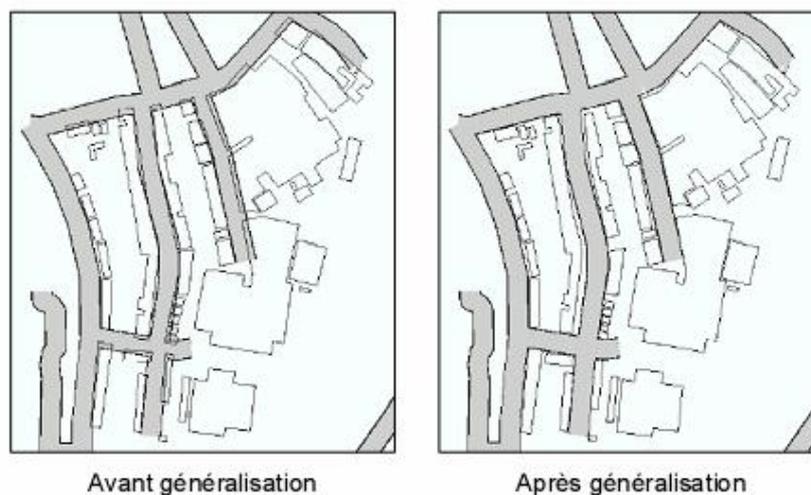


Figure II-3 Données généralisées par une méthode itérative basée sur la technique du recuit simulé (Ware et al., 2003).

b) Discussion

L'inconvénient principal de l'utilisation de cette méthode à la généralisation automatique de données spatiales est que le guidage de processus repose sur la définition de la fonction de coût. Or, on peut se demander s'il est réaliste de définir une telle fonction qui intègre toutes les contraintes de généralisation, tant il est difficile de résumer par un chiffre l'évolution de la satisfaction d'un grand nombre de contraintes (**Duchène, 2004**).

De plus, les opérations de généralisation sont appliquées sur les objets de la carte de façon individuelle mais la fonction d'évaluation est définie globalement, c'est-à-dire concerne tous les objets de la carte. Il peut paraître plus pertinent d'évaluer le résultat d'une opération uniquement sur les objets l'ayant subie (**Gaffuri, 2008**). Aussi, il semble que le critère d'arrêt des algorithmes soit difficile à définir, et que de plus la fonction de coût mérite d'être affinée.

3.2.2.2. Généralisation basée sur les algorithmes génétiques**a) Principe**

Comme son nom s'indique, cette approche se base sur l'utilisation des algorithmes génétiques. C'est une approche bien répandue, proposée par (**Holland, 1975**). Son principe est de faire subir à une population d'individus (qui est généralement au départ tirée en hasard) différentes opérations d'évolution (sélection, croisement entre individus et mutations). La sélection favorise les individus les plus prometteurs vis-à-vis de la fonction de coût (fitness).

Le travail le plus intéressant qui peut être cité ici, est le travail de (**Wilson et al., 2003**). Ce travail se base sur les principes des algorithmes génétiques. Les individus représentent ici l'ensemble des coordonnées, traduites en mode binaire pour coder les gènes. Il utilise l'opération de déplacement pour réduire le nombre des conflits cartographiques dû à la généralisation des bâtiments. Des vecteurs de déplacement de tous les points des objets sont définis. Le travail présenté permet de calculer le déplacement (dx, dy) par rapport l'abscisse X et l'ordonnée Y respectivement, dans un rayon de diamètre D autour du bâtiment. L'algorithme génétique s'exécute en ses étapes classiques ; sélection, croisement et mutation.

La Figure II-4 donne un exemple de résultat obtenu par cette approche.

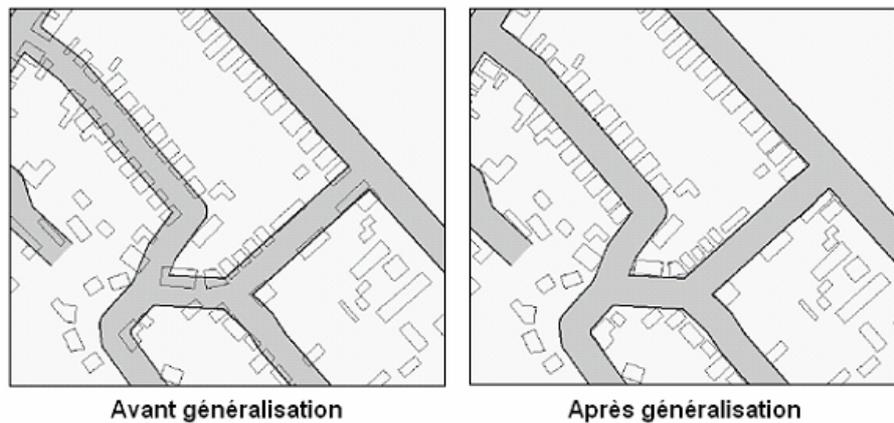


Figure II-4. Exemple de généralisation par algorithme génétique (I.D. Wilson et al, 2003).

b) Discussion

La méthode de déplacement utilisée par (Wilson et al., 2003) fonctionne bien quand il y a beaucoup d'espace libre dans la carte dans laquelle les objets peuvent être déplacés. Cette condition n'est pas toujours garantie, en particulier parce que les cartes peuvent être compactes et seront probablement affichées sur les écrans dont la taille est réduite.

Aussi, l'utilisation d'un seul opérateur, celui de déplacement ne peut pas conduire à la solution optimale en un temps de calcul acceptable. Ce défaut nous pousse à utiliser d'autres opérateurs de généralisation, comme l'élimination, la fusion, et la réduction de taille. En outre, le déplacement peut conduire aux conflits cartographiques, quand il n'y a pas assez d'espace libre dans lequel les objets peuvent être déplacés. Le déplacement de certains objets peut entraîner une propagation des conflits à d'autres objets qui ne sont pas en conflits qui font certains objets déplacés, à être déplacé à nouveau. Enfin, cette méthode ne permet pas d'éliminer des objets, puisque tous les points sont conservés (Duchêne, 2004).

3.3. Modèles basés AGENTS

Le concept d'agent est issu du domaine de l'intelligence artificielle. Dans cette section, nous détaillons l'utilisation de ce concept pour modéliser les objets géographiques dans le processus de la généralisation et nous discutons les résultats obtenus en suivant ce modèle.

3.3.1. Principes

Plusieurs travaux principaux s'inscrivent dans ce modèle ; (Ruas, 1999), (Duchêne, 2004), (Ruas & Duchêne, 2007), (Duchêne et al., 2012). Le modèle présenté dans (Duchêne et al., 2012) se base à son tour sur le projet AGENT ((Ruas, 1999), (Lamy et al., 1999), (Duchêne & Ruas 2001) et (Ruas & Duchêne, 2007)). Dans le modèle AGENT, les objets géographiques sont modélisés par les entités décisionnelles du système de la généralisation. Ces entités sont des agents logiciels dont leur objectif est satisfaire que

possible leurs contraintes cartographiques. Ces contraintes sont classées en deux catégories: les contraintes internes ou externes (avec d'autres objets).

Les contraintes sont transformées en objet. Afin de résoudre un conflit donné, L'objet contrainte propose à l'agent des plans (algorithmes de généralisation) pour résoudre ce conflit. L'agent s'auto-généralise en réalisant un cycle qui consiste à essayer plusieurs opérations de la généralisation, préconisées par les contraintes internes de l'agent, jusqu'à ce que l'agent atteigne une solution (contraintes cartographiques respectées). Il utilise un mécanisme d'arbre pour le stockage des états intermédiaires, décrit dans (Regnauld, 2004).

Chaque contrainte indique à son agent les opérations possibles qui permettent de la satisfaire. L'agent synthétise les propositions obtenues à partir de toutes ses contraintes et choisit les meilleures actions à mener pour résoudre un tel conflit cartographique. Lorsqu'un ou plusieurs contraintes impliquent plusieurs agents *micro*, ces agents se retournent vers leur agents *méso*, qui coordonne la résolution du conflit. Dans ce cas, l'agent *méso* demande à ses agents composants d'effectuer les transformations nécessaires. Dans ce modèle, aucune interaction n'est possible entre les agents *micro*.

La structure hiérarchique du prototype AGENT peut être appliquée aux cartes des zones urbaines, dont les routes et les bâtiments sont présentés de façon bien organisée. Toutefois, dans les zones rurales, la structure hiérarchique de l'espace n'est pas facilement représentée par ce modèle, le modèle est beaucoup plus difficile à mettre en œuvre. En effet, la densité des objets est trop basse et par conséquent, il est difficile à distinguer les groupes disjoints d'objets. La **Figure II-5** donne des exemples de résultats obtenus avec le modèle de généralisation AGENT.

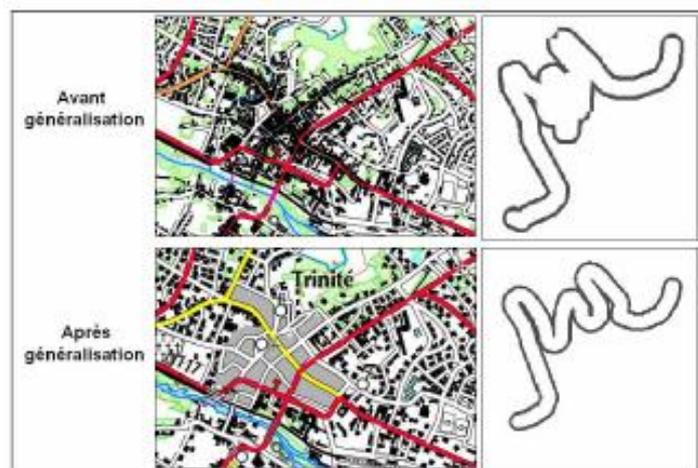


Figure II-5. Exemples de résultats obtenus avec le modèle de généralisation AGENT.

Afin de surmonter les limitations du prototype AGENT pour la généralisation cartographique des zones rurales, **Duchêne (2003)** a proposé un autre modèle appelé

CartaCom à base d'agents qui traitent des contraintes relationnelles. Une contrainte relationnelle indique une relation entre deux objets géographiques, telle que la contrainte qui interdit la superposition de symboles. **Duchêne (2003)** a proposé une approche qui utilise un seul niveau d'agents. Contrairement au prototype AGENT, les agents voisins de ce niveau interagissent bilatéralement afin de résoudre leurs conflits cartographiques. Grâce à sa «perception» partielle de son environnement, un agent peut identifier les agents voisins. La **Figure II-6** donne un exemple de résultat obtenu avec *CartACom*.

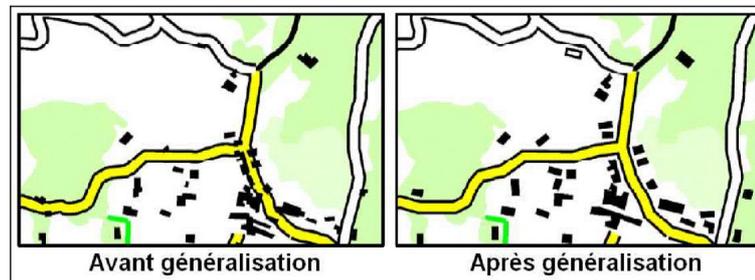


Figure II-6. Exemple de résultats obtenus avec *CartACom* (**Duchêne, 2004**).

Une autre plateforme *CartAGen* est proposée pour regrouper trois modèles de généralisation à base d'agents, développés au cours des dix dernières années à l'IGN: AGENT (**Ruas, 1999**), *CartACom* (**Duchêne, 2004**) et GAEL (**Gaffuri, 2008**), chacun de ces modèles est dédié pour résoudre différents types de conflits. L'architecture de cette plateforme est discutée et implémentée dans (**Renard et al., 2010**) et (**Renard et al., 2011**). Dans le même contexte, **Renard & Duchêne (2014)** ont proposé un modèle basé sur le prototype *CartAGen* pour traiter le problème de préservation de l'alignement des bâtiments durant le processus de la généralisation, en introduisant le concept d'agents réactionnels. Ce nouveau modèle améliore la qualité des données générées par le modèle AGENT et donne des meilleurs résultats spécialement dans les zones urbaines d'une grande densité.

3.3.2. Discussion

Les conflits dans le modèle de (**Duchêne, 2004**) sont résolus par la communication entre les agents. Ce modèle et ses postérieurs ne décrit pas un système intra-agent qui lui permet de choisir les actions (les algorithmes de généralisation) satisfaisant au maximum les contraintes cartographiques. Ce modèle aussi ne garantit pas le paramétrage optimal des algorithmes de généralisation. Bref, la stratégie suivie dans ce modèle n'est pas assez flexible, car elle ne permet pas à l'agent de choisir la meilleure action à exécuter dans une situation donnée et ne permet pas aussi de traiter les relations topologiques entre les objets spatiaux. Ainsi, les scénarios prédéfinis exprimant les connaissances du cartographe sont difficiles à

automatiser ((Rigaux, 1994) et (Plazanet, 1996)) et à modéliser en totalité, en utilisant ce modèle.

4. Modèles de relations topologiques

Il existe plusieurs façons de déterminer les relations topologiques entre deux objets. Au cours des dernières années, plusieurs propositions ont été faites : le modèle d'Allen (Allen, 1991), le modèle des 9-intersections (Egenhofer & Herring, 1990) et le modèle de Lee & Hsu (1990, 1992). Par la suite, nous allons présenter ces trois modèles :

4.1. Modèle d'ALLEN

Allen (1991) propose un formalisme basé sur une logique temporelle utilisée pour raisonner sur les événements, des actions, des convictions, des intentions, des causalités et également utilisée pour servir de cadre à des résolution des problème. Nous pouvons utiliser le modèle d'Allen pour décrire les relations topologiques entre les objets spatiaux telle que les relations topologiques entre deux segments. Dans ce modèle linéaire, Allen a défini 13 relations binaires. La Figure II-7 présente les relations qui sont définies par le modèle d'ALLEN.

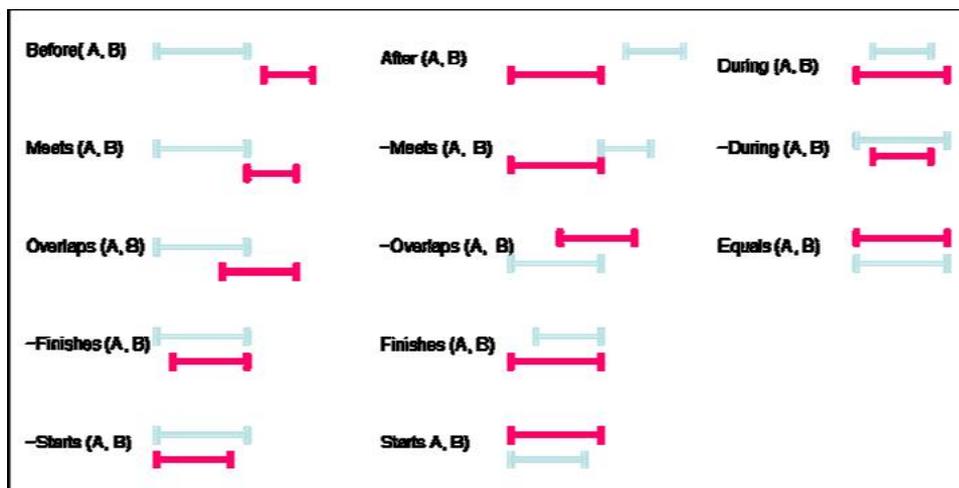


Figure II-7. Le modèle linéaire d'Allen.

4.2. Modèle des 9-intersections

Egenhofer & Herring (1990) proposent un modèle de données spatiales basé sur l'analyse topologique à 2D (2 dimensions). Ce modèle topologique est basé à son tour sur les primitives géométriques appelées (Cell) qui sont définies pour différentes dimensions spatiales.

- Un 0-complexe cellulaire est un nœud (qui correspond au plus petit objet 0-dimensionnel),
- Un 1-complexe cellulaire est le lien entre deux 0-complexes cellulaires distincts,

— Un 2-complexe cellulaire est la surface décrites par la séquence de trois 1-complexes cellulaires qui ne s'intersectent pas.

Le modèle des 9-intersection est un modèle binaires entre deux cellules A et B . il est basé sur la combinaison des six primitives topologiques qui sont les intérieurs (interior), frontières (boundary) et extérieurs (exterior) de A ($A^\circ, \partial A, A^-$) et de B ($B^\circ, \partial B, B^-$). Ces six primitives topologiques peuvent donc être combinées de telle sorte à former neuf combinaisons possibles, représentant la relation topologique entre ces deux cellules. Ces 9 intersections sont représentées sous la forme d'une matrice 3*3 :

$$R(A,B) = \begin{pmatrix} A^\circ \cap B^\circ & A^\circ \cap \partial B & A^\circ \cap B^- \\ \partial A \cap B^\circ & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^\circ & A^- \cap \partial B & A^- \cap B^- \end{pmatrix}$$

La valeur représentée dans la matrice sera uniquement un symbole précisant si l'intersection est nulle (\emptyset) ou pas ($\neg\emptyset$). Lorsque la valeur de l'intersection n'a pas d'importance, elle est représenté comme (-). En basant sur ces 9 intersections possibles, on peut avoir 512 relations théoriques. Cependant, elles ne sont pas toutes possibles. La détection des relations possible se fait à l'aide des conditions négatives qui empêchent l'association d'intersections entre les couples de primitives. Enfin, on obtient alors 8 relations topologiques possibles pour deux régions dans \mathfrak{R}^2 . La **Figure II-8** représente ces relations.

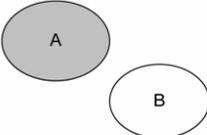
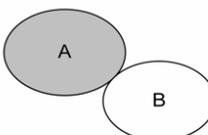
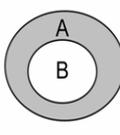
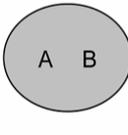
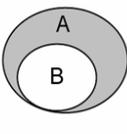
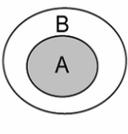
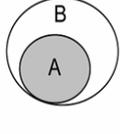
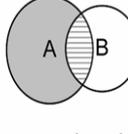
 $R_{disjoint}(A,B) = \begin{pmatrix} \emptyset & \emptyset & \neg\emptyset \\ \emptyset & \emptyset & \neg\emptyset \\ \neg\emptyset & \neg\emptyset & \neg\emptyset \end{pmatrix}$	 $R_{meet}(A,B) = \begin{pmatrix} \emptyset & \emptyset & \neg\emptyset \\ \emptyset & \emptyset & \neg\emptyset \\ \neg\emptyset & \neg\emptyset & \neg\emptyset \end{pmatrix}$	 $R_{contains}(A,B) = \begin{pmatrix} \neg\emptyset & \neg\emptyset & \neg\emptyset \\ \emptyset & \emptyset & \neg\emptyset \\ \emptyset & \emptyset & \neg\emptyset \end{pmatrix}$	 $R_{equal}(A,B) = \begin{pmatrix} \neg\emptyset & \emptyset & \emptyset \\ \emptyset & \neg\emptyset & \emptyset \\ \emptyset & \emptyset & \neg\emptyset \end{pmatrix}$
 $R_{covers}(A,B) = \begin{pmatrix} \neg\emptyset & \neg\emptyset & \neg\emptyset \\ \emptyset & \neg\emptyset & \neg\emptyset \\ \emptyset & \emptyset & \neg\emptyset \end{pmatrix}$	 $R_{inside}(A,B) = \begin{pmatrix} \neg\emptyset & \emptyset & \emptyset \\ \neg\emptyset & \emptyset & \emptyset \\ \neg\emptyset & \neg\emptyset & \neg\emptyset \end{pmatrix}$	 $R_{coveredBy}(A,B) = \begin{pmatrix} \neg\emptyset & \emptyset & \emptyset \\ \neg\emptyset & \neg\emptyset & \emptyset \\ \neg\emptyset & \neg\emptyset & \neg\emptyset \end{pmatrix}$	 $R_{overlap}(A,B) = \begin{pmatrix} \neg\emptyset & \neg\emptyset & \neg\emptyset \\ \neg\emptyset & \neg\emptyset & \neg\emptyset \\ \neg\emptyset & \neg\emptyset & \neg\emptyset \end{pmatrix}$

Figure II-8. Le modèle de 9-intersection.

4.3. Modèle de Lee & Hsu

Lee & Hsu (1990,1992) étudient les relations qui peuvent exister entre deux rectangles. Ils ont proposé un tableau représentant toutes les relations spatiales entre ces deux rectangles. Ils ont trouvé un total de 169 types. Parmi lesquelles : 48 disjointes, 40 meet, 50 chevauchements partiels (overlap), 16 contient et 16 appartient (= l'intérieur). En raison de la

sémantique des rubans (qui est défini comme un rectangle oblong), beaucoup d'entre eux peuvent être éliminés. Nous ne examinerons pas tous ces relations, mais les relations de rubans les plus intéressants, à savoir ; la disjonction (disjoint), l'adjacence (meet), le chevauchement (overlap) et le croisement (cross). La **Figure II-9** présente les relations qui sont définies par le modèle de Lee et Hsu.

DISJOINT (48)			JOIN (40)			PART_OVLP (50)			CON TAIN (16)	BEL* ONG (16)

Figure II-9. Les 169 types de relations spatiales.

5. Synthèse sur les approches

Nous avons présenté dans la section précédente des travaux visant à automatiser et à optimiser le processus de la généralisation des données spatiales. En général, il y a une distinction entre deux principaux modèles ; les modèles par optimisation et les modèles AGENT.

5.1. Modèle AGENT

Dans le modèle AGENT, chaque agent communique avec les agents voisins pour résoudre les conflits. En cours de cette communication, des scénarios prédéfinis de conversation sont utilisés. Ces scénarios sont stockés dans le système. Ils indiquent la séquence des performatifs échangés entre les agents. Quand un agent reçoit un message, il se

réfère à ces scénarios pour connaître les réponses à donner au message. Ainsi, afin d'aider un agent à choisir la meilleure réponse à une situation donnée, le système proposé consiste à sauvegarder pour chaque message, le nom de l'action à effectuer, ainsi que l'état qui indique comment choisir le prochain message à envoyer en fonction du résultat de cette action (**Duchêne, 2003**).

Duchêne (2004) propose une approche de résolution des conflits spatiaux basée sur des scénarios prédéfinis de communication. Cette stratégie n'est pas assez souple, car elle ne permet pas à l'agent de choisir l'action optimale à effectuer en fonction de la situation donnée (**Jabeur, 2006**). Aussi, la majorité des méthodes et les approches qui ont été présentées pour améliorer les résultats obtenus (en terme de carte) du modèle AGENT et ses variantes, ne traitent pas la généralisation des relations topologiques entre les objets spatiaux.

5.2. Méthodes d'optimisation

Comme nous l'avons constaté, il existe un grand nombre de méta-heuristiques d'optimisation utilisé pour résoudre les problèmes de la généralisation. Dans ce contexte, on peut distinguer deux groupes : les méthodes de recherche globales et locales. Les méthodes globales (recuit simulé, algorithmes évolutionnaires, ...etc) sont plus adaptées à l'exploration globale de l'espace de recherche. Ceux-ci sont des meilleurs mécanismes mais ils peuvent être bloqués dans des minima locaux.

A partir d'un point de départ quelconque, les méthodes locales permettent d'obtenir la solution optimale le plus proche du point de départ. Elles essaient toujours d'améliorer la solution courante. Il permet de surmonter le minimum local puisque cela impliquerait d'accepter de dégrader la solution courante (**Paris et al., 2001**).

Nous allons résumer les principales méthodes d'optimisation globale et particulièrement, les algorithmes génétiques, utilisée pour la modélisation du processus de la généralisation, que nous qualifions de stochastiques. Ils utilisent tous un processus itératif de recherche avec trois éléments : la génération de candidats (solutions), le critère d'acceptation d'un candidat et un critère d'arrêt du processus itératif :

- La génération de candidats peut être réalisée à partir d'une ou plusieurs solutions de base. L'algorithme introduit une perturbation aléatoire des coordonnées de la ou des solutions de base suivant une loi probabiliste. Les candidats sont construits en utilisant la valeur ou les coordonnées des solutions de base pour les méthodes par voisinage,
- le critère d'acceptation des candidats pour le pas suivant dépend soit d'une loi de probabilité choisie, soit de la performance des candidats,

— le critère d'arrêt est exprimé en fonction d'un coût de calcul (temps, nombre d'itérations,...etc.) ou en fonction de la qualité des solutions obtenues.

Pour cette raison, on a proposé une nouvelle approche, c'est-à-dire fournir à un agent une méthode d'optimisation comme l'algorithme génétique qui lui permet de choisir l'action optimale à effectuer en fonction des paramètres prédéfinis. Nous allons nous concentrer, dans ce travail, sur l'étude de l'approche AGENT et les algorithmes génétiques étant donné leurs performances avérées sur le problème de la sélection de scénarios. Aussi, on va proposer un modèle topologique pour gérer les transformations des relations topologiques durant le processus de la généralisation des données spatiales. On va détailler tout cela au prochain chapitre.

5.3. Synthèse et objectifs

Nous avons présenté dans ce chapitre les différents travaux effectués pour automatiser le processus de la généralisation. Ces modèles se basent sur des approches différentes. La plupart des modèles proposés pour automatiser le processus de la généralisation, donc pour résoudre les conflits cartographiques entre les objets spatiaux après la généralisation. Ces modèles ne garantissent pas le paramétrage optimal des algorithmes de généralisation et aussi ils ne traitent pas les variations des relations topologiques entre les objets spatiaux durant le processus de la généralisation. Nous nous concentrons principalement sur les travaux basés sur les modèles AGENT et les méthodes d'optimisation. Donc, notre objectif consiste à proposer un modèle hybride de la généralisation. Ce modèle se base sur le modèle AGENT amélioré par un algorithme génétique permettant à l'agent de choisir l'action optimale à exécuter. Dans notre modèle hybride, nous avons utilisé le concept de ruban pour modéliser les objets linéaires comme les routes, les rues et les rivières puisque ces objets en réalité ont une certaine largeur. Aussi, nous avons étudié les transformations des relations topologiques durant le processus de la généralisation.

6. Conclusion

Cette exploration des différents travaux menés par différents groupes de chercheurs dans des laboratoires de recherche à travers le monde, nous a permis d'identifier les différentes approches adoptées dans la modélisation des données spatiales dans un processus de la généralisation et aussi la modélisation des relations topologiques entre les objets spatiaux.

A l'heure actuelle, il n'existe pas une approche qui puisse résoudre en lui-même, l'ensemble des problèmes liés à la généralisation des données spatiales et leurs relations

topologiques. L'automatisation complète reste difficile à atteindre, en utilisant une seule approche. Par conséquent, il est difficile à avoir la généralisation des données spatiales en temps-réel.

L'idée qui nous a envahi en étudiant de près ces travaux, est de trouver une approche hybride qui exploite la puissance des approches décrites dans ce chapitre et que nous allons détailler dans la suite de cette thèse.

Selon notre réflexion, nous étions guidé par un objectif principal : servir efficacement les besoins des utilisateurs par des cartes de bonne qualité et de bonne cohérence topologique. Donc, satisfaire les exigences des nouvelles applications comme par exemple, la gestion de l'aménagement urbain (bâtis, réseau routière, réseau d'assainissement, réseau hydraulique,...etc.) en différentes résolutions et selon différents préférences des utilisateurs.

Dans le prochain chapitre, nous abordons tous les aspects et les concepts conduisant à la modélisation du processus optimal de la généralisation.

Chapitre III

Modèle hybride pour optimiser le processus de généralisation

1. Introduction

Notre approche est une approche hybride des systèmes multi-agents, qui profite au fond des avantages des algorithmes génétiques pour permettre principalement aux agents de déterminer quelle séquence d'actions optimale à appliquer (les algorithmes de généralisation). D'où, notre approche consiste à utiliser un système multi-agent guidé par une méthode d'optimisation performante, à savoir les algorithmes génétiques. La population d'un algorithme génétique est un ensemble de solutions possibles au problème posé, dans notre cas, une solution est une suite d'algorithmes de généralisation avec des valeurs paramétriques permettant de résoudre les conflits cartographiques apparents sur une carte géographique à une échelle donnée. Aussi, pour augmenter la qualité de la carte et maintenir sa cohérence topologique, nous avons proposé d'utiliser un autre type d'objet appelé ruban et nous avons étudié les éventuelles mutations des relations topologiques en d'autres relations en cours du processus de la généralisation, suivant un modèle topologique basé sur certaines assertions mathématiques.

2. Généralisation et systèmes multi-agents

Dans cette section, nous présentons plus en détail l'utilisation de systèmes multi-agents pour automatiser le processus de généralisation.

2.1. Définitions

Nous présentons quelques notions essentielles relatives aux systèmes multi-agents, à savoir l'agent et l'environnement.

2.1.1. Agent

Une définition est donnée par **(Wooldridge & Jennings, 1995)** : « Ils définissent l'agent comme une entité logicielle, situé dans un environnement donné, et qui est capable d'agir de façon autonome dans son environnement afin d'atteindre les objectifs pour lesquels il a été conçu ».

“The agent as a software entity, situated in a given environment, and which is able to act autonomously in its environment in order to reach the goals for which it was conceived”.

Nous pouvons aussi citer, la définition de **(Ferber, 1997)**.

« On appelle agent, une entité physique ou virtuelle

- Qui est capable d'agir dans un environnement,
- Qui peut communiquer directement avec les autres agents,
- Qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser),

- Qui possède des ressources propres,
- Qui est capable de percevoir (mais de manière limitée) son environnement,
- Qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune),
- Qui possède des compétences et offre des services,
- Qui veut éventuellement se reproduire,
- Dont le comportement tend à satisfaire les objectifs, en tenant compte des ressources et des compétences dont il dispose, et en fonction de sa perception, de ses représentations et des communications qu'il reçoit ».

Une autre définition donnée par **(Chaib-Draa et al., 2001)** est très importante à citer : « Un agent est une entité autonome, réelle ou abstraite, qui est capable d'agir sur elle-même et sur son environnement, qui, dans un univers multi-agents, peut communiquer avec d'autres agents, et dont le comportement est la conséquence de ses observations, de ses connaissances et des interactions avec les autres agents ».

2.1.2. Système multi-agent

Un système multi-agents est un système distribué qui contient plusieurs agents qui interagissent pour atteindre un but privé, et éventuellement commun. Un agent peut être vu comme un objet amélioré qui possède des capacités de communication, de perception et surtout d'autonomie et peut également être vu comme un objet vivant qui décide lui-même de s'appliquer des méthodes particulières en fonction de son but et de son environnement **(Duchène , 2004)**. **Ferber (1995)** définit le système multi-agents comme suit.

"On appelle un système multi-agent (ou SMA), un système composé des éléments suivants.

- Un environnement E , c'est-à-dire un espace disposant généralement d'une métrique.
- Un ensemble d'objets O . Ces objets sont situés, c'est-à-dire que, pour tout objet, il est possible, à un moment donné, d'associer une position dans E . Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.
- Un ensemble A d'agents, qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système.
- Un ensemble de relations R qui unissent des objets (et donc des agents) entre eux.
- Un ensemble d'opérations Op permettant aux agents de A de percevoir, produire, consommer, muter et manipuler des objets de O .
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers."

2.1.3. Environnement

Ferber (1995) définit l'environnement comme suit.

"L'environnement d'un agent est caractérisé par tout ce qui n'est pas lui. Par ailleurs, les agents n'ont qu'une représentation partielle de leur environnement, c'est-à-dire qu'ils n'ont pas de vision globale de tout ce qui se passe". Dans ce contexte l'environnement est constitué d'agents géographiques, il définit l'espace qu'un agent perçoit (**Duchêne, 2004**). La délimitation de l'environnement de l'agent se base principalement sur la distance qui sépare les objets géographiques.

Pour décrire l'environnement de l'agent, on s'intéresse particulièrement à leurs relations topologiques d'adjacence avec les autres agents de même type, d'adjacence et d'inclusion avec les objets d'autres types. En termes de généralisation, du fait qu'ils doivent former une partition de l'espace, toute modification ou élimination d'un objet entraîne nécessairement des modifications sur les agents adjacents.

2.2. Pourquoi utilise-t-on l'approche agent ?

Chaib-draa et al. (2001) ont indiqué que les systèmes multi-agents conviennent pour représenter des problèmes qui ont plusieurs solutions, comme c'est le cas de la généralisation automatique. **Galanda (2003)** a aussi désigné que SMA peuvent offrir la méthodologie nécessaire pour orchestrer les algorithmes, les contraintes et les stratégies...etc., pour réaliser un processus de généralisation automatique, dynamique et flexible.

Plusieurs travaux dédiés à l'utilisation de la technologie multi-agents dans le processus de généralisation ont vu le jour comme (**Baeijs et al., 1996**), (**Lamy et al., 1999**), (**Lee, 2003**), (**Duchêne, 2004**), (**Jabeur et al., 2006**) et (**Duchêne et al., 2012**). Dans une telle approche, une autonomie est donnée aux objets cartographiques afin de leur permettre de choisir les outils nécessaires pour leur généralisation.

Dans le contexte du projet AGENT, chaque objet géographique (bâtiment, route et arbre... etc.) devient un agent géographique. Il peut s'agir d'objets initialement présents dans la base de données à généraliser ou d'objets créés à partir de ceux-ci par regroupement (exp. un groupe de bâtiments) ou par division (exp. une portion de route) (**Duchêne & Regnauld, 2002**). Dans l'approche AGENT, les agents géographiques sont divisés en trois types principaux selon le niveau de la hiérarchie :

Les **agents micros**, ceux sont les objets géographiques (une route, un bâtiment ou une partie d'objet) et les **agents mesos** se composent d'agents micros et d'agent méso. Les agents macros qui représentent un niveau supérieur de la hiérarchie, ils correspondent à des

populations d'objets (exp. les îlots). Par exemple, pour l'espace urbain, les trois niveaux précédents ont été identifiés comme pertinents : la ville, composée d'îlots urbains (bâtiments délimités par un cycle minimal de rues), qui sont eux-mêmes composés de bâtiments. Les bâtiments sont des agents micros, les îlots urbains sont des agents *mesos* et la ville est un agent *macro* (voir la **figure III-1**). Cela aboutit à une structure multi-niveaux hiérarchique (pyramidale) (**Duchêne, 2004**).

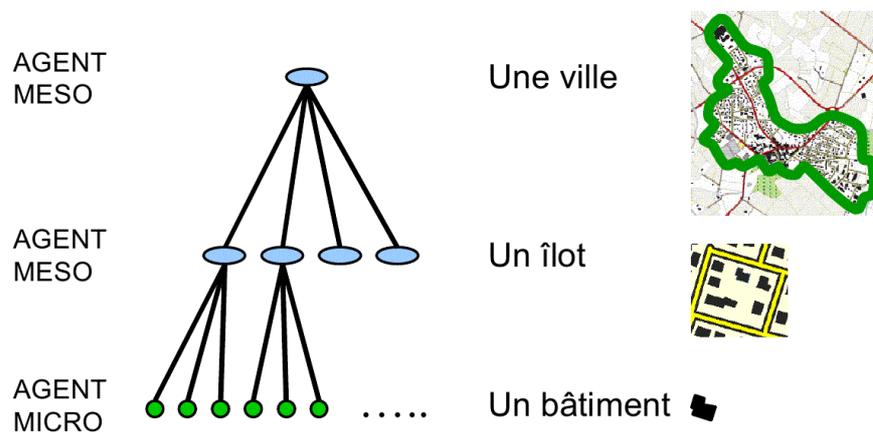


Figure III-1. Le modèle AGENT appliqué à l'espace urbain.

L'agent *meso* est d'une part, le responsable des opérations de généralisation qui ont lieu au niveau du groupe (exp. l'élimination d'objets au sein du groupe), elles sont guidées par les contraintes et d'autre part, il supervise la généralisation individuelle de ses composants (**Duchêne, 2004**). Ce modèle est très avantageux mais si le nombre d'objets dans un îlot est relativement élevé, il est difficile que le système atteigne des meilleures performances. D'autre part, une telle structure est inefficace en dehors des villes, c'est le cas dans les espaces hétérogènes en types d'objets géographiques, c'est-à-dire dans lesquels il n'existe pas vraiment de thème dominant comme en ville ; le réseau routier et les bâtiments. Pour adopter l'approche au contexte des zones rurales, **Duchêne (2004)** propose des interactions transversales aux agents *micros*.

On a dit dans le chapitre précédent que, ce modèle ne décrit pas un système intra-agent qui lui permet de choisir les actions optimales (les algorithmes de généralisation) satisfaisant au maximum les contraintes cartographiques. Ce modèle aussi ne garantit pas le paramétrage optimal des algorithmes de généralisation. Donc, la stratégie suivie dans ce modèle n'est pas assez flexible, car il ne permet pas à l'agent de choisir l'action optimale à exécuter à un moment et dans une situation donnée. Ainsi, les scénarios prédéfinis expriment les connaissances du cartographe, difficiles à automatiser ((**Rigaux, 1994**) et (**Plazanet, 1995**)) et à modéliser en totalité, en utilisant l'approche agent seulement. Donc, pour remédier

à ces problèmes, nous allons proposer dans cette thèse, de doter l'agent géographique, de patrimoines génétiques qui lui permettent de trouver la solution optimale, d'où la notion **agent génétique**.

Ainsi, notre approche est hybride des systèmes multi-agents qui profite au fond des avantages des algorithmes génétiques pour permettre principalement aux agents de trouver l'action optimale. L'objectif de cette approche consiste à guider un processus de généralisation optimale. L'agent exécute un algorithme génétique qui suit les étapes classiques ; sélection, croisement et mutation. La population d'un algorithme génétique est un ensemble de solutions possibles pour résoudre un conflit. La solution est la séquence optimale des algorithmes de généralisation avec des bonnes valeurs paramétriques permettant de résoudre les conflits apparents sur une carte à une échelle donnée.

Nous allons étudier dans les sections suivantes, en détail, les algorithmes génétiques et leur utilisation pour optimiser le processus de la généralisation des données spatiales.

3. Généralisation et méthodes d'optimisation

Les techniques d'optimisation sont bien connues pour la résolution des problèmes sur-contraints en physique et en ingénierie. Elles permettent de trouver la solution pour les problèmes de recherche parmi un ensemble de solutions potentielles ou d'un ensemble de solutions optimales par rapport à une fonction d'évaluation ou fonction « coût ». Si la fonction est multi-valuée, on parle alors de méthode multi-objectif. Donc, on parle ici de problèmes qui n'ont pas une solution unique. Ainsi, la solution est considérée comme étant celle qui répond au mieux au compromis parmi toutes les contraintes.

La résolution des problèmes par une méthode d'optimisation est illustrée par la **Figure III-2**. Elle passe principalement par deux étapes successives (**Pitiot , 2009**) :

- L'analyse du problème, regroupant la caractérisation du problème puis sa modélisation, qui aboutit en général à sa représentation informatique ou codage (codage des solutions potentielles et codage d'opérateurs de recherche adaptés),
- Le choix d'une méthode de résolution ou méthode de recherche, qui permet à partir de la représentation du problème dans un espace de recherche d'obtenir une solution ou un ensemble de solutions optimales par rapport à la fonction d'évaluation.

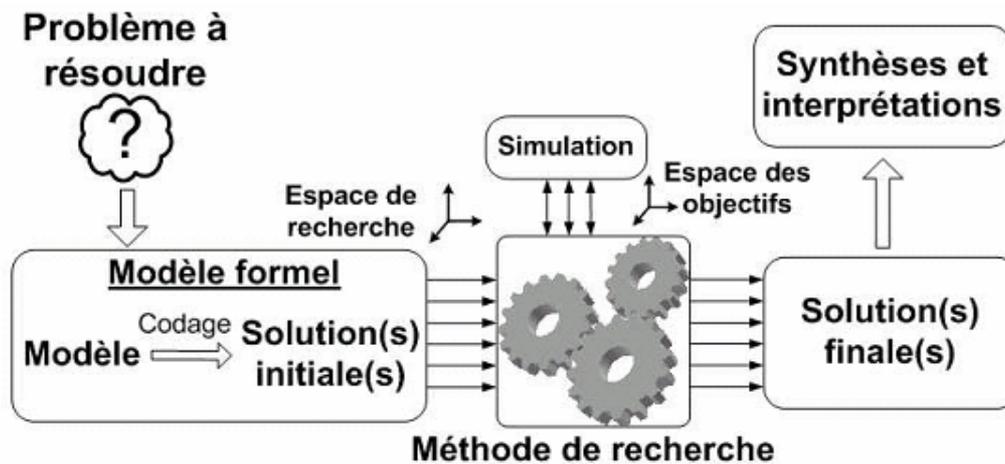


Figure III-2. Résolution d'un problème par une technique d'optimisation (Pitiot, 2009).

3.1. Définition

Selon la définition de (Larousse, 2014) :

- « **Optimiser** » : consiste à donner à quelque chose, à une machine, à une entreprise...etc., le rendement optimal en créant les conditions les plus favorables ou en tirant le meilleur parti possible.
- L'**Optimisation** est :
 - Action d'optimiser ou d'optimiser ; fait d'être optimisé ou optimisé,
 - Démarche consistant à rendre optimal le fonctionnement d'un système.

3.2. Représentation des solutions

Un bon codage assure un compromis entre la qualité de l'information intégrée dans le codage et la complexité des opérateurs. Il faut également trouver un compromis entre la complexité des opérateurs et la présence de solutions invalides ou redondantes. Dans le cas de solutions redondantes, la taille de l'espace de recherche peut s'avérer très importante puisqu'il contient toutes les configurations ainsi que les redondances. La plupart des méthodes d'optimisation utilisent une représentation sous forme de vecteurs des paramètres à régler. De nombreux problèmes réels se présentent néanmoins sous la forme d'arbres ou de graphes de paramètres (Paris et al., 2001). Le codage de tels problèmes sous forme de vecteurs est alors fortement "épistatique" (ou déceptif) puisque certains paramètres peuvent être totalement inhibés par la sélection d'autres paramètres situés en amont dans le graphe ou l'arbre (Pitiot, 2009). Cette représentation correspond à une perte d'informations dans le sens où l'algorithme n'utilise pas cette connaissance (Pitiot, 2009). Cet aspect sera étudié plus en détail dans la section 4.4.9 pour le problème de sélection de scénarios (volume et impact des solutions redondantes, modification des opérateurs, ...etc.).

3.3. Pourquoi utiliser une approche d'optimisation ?

Le processus de généralisation des données spatiales peut également être modélisé comme un problème d'optimisation, où les différentes contraintes doivent être satisfaites simultanément le plus possible (**Sester, 2000**). Pour ce faire, les problèmes et les contraintes de généralisation doivent être formalisées analytiquement comme des dépendances fonctionnelles (**Sarjakoski & Kilpelainen, 1999**). Ainsi, pour satisfaire les contraintes de généralisation, on peut appliquer simultanément tous les déplacements et les déformations nécessaires sur l'ensemble de données. Pour ce faire, les objets de la carte doivent être modélisés comme des agents et interagir pour trouver une solution optimale (le meilleur compromis qui satisfait les contraintes concurrentes). En d'autres termes, les techniques d'optimisation supportent une approche holistique de la généralisation. Lors d'un processus de généralisation manuelle, le cartographe a une vision globale de la carte, il ne regarde pas les objets de la carte d'une manière isolée, mais dans leur contexte tout en considérant à la fois leurs caractéristiques internes (exp. la grandeur, la sinuosité) et leurs relations avec les autres objets de la carte (**Sabo, 2007**). De nombreuses méthodes d'optimisation ont été introduites à la généralisation des cartes, elles sont en particulier associées à des objets linéaires. Parmi lesquelles, on peut citer :

- La méthode des moindres carrés,
- la méthode des éléments finis,
- Les algorithmes génétiques,
-etc.

Le potentiel des techniques d'optimisation consiste à combiner différentes opérations de généralisation. **Harrie & Sarjakoski (2002)** proposent un concept appelé ; généralisation graphique simultanée basé sur la technique des moindres carrés. Cette approche combine les opérations d'application de généralisation (exp. simplification, lissage, élargissement et déplacement) et la maintenance des contraintes structurels (courbure, croisement,...etc.) pour réaliser un processus de généralisation compréhensive. Les méthodes d'optimisation réalisent un compromis entre les opérations de généralisation concurrentes et les contraintes. **Burghardt (2002)** présente un algorithme qui établit un lissage et un déplacement simultané des objets linéaires basés sur la technique des éléments finis (Snakes). Dans ces techniques d'optimisation, la généralisation n'effectue que les opérations de déplacement et l'élargissement, mais la généralisation se compose d'autres opérations comme ; élimination, agrégation ou la typification...etc., qui ne peuvent être implémentées par une telle approche.

Donc, nous proposons de combiner une méthode d'optimisation avec l'approche AGENT pour combler ces manques.

Dans la prochaine section, nous allons étudier en détail les algorithmes génétiques, étant donné leurs performances avérées dans le problème de sélection de scénarios et l'ensemble des recherches effectuées par rapport à leur capacité à manipuler des connaissances du domaine étudié.

4. Généralisation et algorithmes génétiques

Les algorithmes génétiques reposent sur plusieurs concepts, à savoir la sélection, le croisement, la mutation et une fonction de coût. Dans cette section, nous allons détailler ces différents concepts.

4.1. Définitions

Pour définir les algorithmes génétiques, nous devons tout d'abord définir les algorithmes évolutionnistes.

4.1.1. Algorithmes évolutionnistes

Les algorithmes évolutionnistes ou algorithmes évolutionnaires sont une méthode d'optimisation, s'inspirant de la théorie de l'évolution pour résoudre des problèmes divers (Eiben & Smith, 2003). Ils font ainsi évoluer un ensemble de solutions à un problème donné, dans l'optique de trouver les meilleurs résultats. Ce sont des algorithmes stochastiques car ils utilisent itérativement des processus aléatoires. La grande majorité de ces méthodes sont utilisées pour résoudre des problèmes d'optimisation.

4.1.2. Algorithmes génétiques

Les algorithmes génétiques initiés à partir des années 1970 par John Holland. ils appartiennent à la famille des algorithmes évolutionnistes. Ils sont inspirés du concept de sélection naturelle élaboré par Charles Darwin (Goldberg, 1989). Celle-ci énonçait que les individus les plus aptes à survivre (les meilleurs) se reproduiront plus souvent et auront plus de descendants. Ainsi, la qualité du pool génétique de la population sera augmentée, les gènes les plus efficaces deviendront plus fréquents ; la population s'améliore.

Le but principal des algorithmes génétiques est d'obtenir une solution approchée à un problème d'optimisation, en un temps raisonnable, lorsqu'il n'existe pas de méthode exacte (ou que la solution est inconnue). Les algorithmes génétiques utilisent la notion de sélection naturelle et l'appliquent à une population de solutions potentielles d'un problème donné. Ils partent d'une population de solutions initiales. Une solution représente un individu de cette population. Ces individus se reproduisent (les meilleurs individus ont plus de chances de se

reproduire), pour créer une nouvelle génération d'individus. Ils s'appuient sur des techniques dérivées de la génétique et de l'évolution naturelle : sélections, croisements, mutations. En répétant ce dernier cycle plusieurs fois, pour obtenir une population composée des meilleures solutions (des meilleurs individus) après un certain nombre de générations.

4.2. Origines

Dès 1960, John Holland et son équipe travaillent sur l'utilisation d'un algorithme génétique à la résolution des problèmes. Selon ce groupe de chercheurs l'utilisation de l'opérateur de croisement, en complément de mutations permet le plus souvent de se rapprocher à l'optimum d'une fonction en combinant les gènes contenus dans les différents individus de la population (**Holland, 1975**).

Les fruits de ces travaux de recherches ont été publiés dans (**Holland, 1975**) en 1975, dans lesquels est introduit le premier modèle formel des algorithmes génétiques. Il explique comment ajouter de l'intelligence dans un programme informatique avec les croisements (échangeant le matériel génétique) et la mutation (source de la diversité génétique). En 1989, David Goldberg publie un ouvrage de vulgarisation des algorithmes génétiques (**Goldberg, 1989**). Ce livre est encore édité aujourd'hui et reste la première référence des algorithmes génétiques. Depuis 1990, Les chercheurs ont développé une panoplie d'algorithmes génétiques implémentés en C⁺⁺, appelée GALib. Cette librairie contient des outils pour des problèmes d'optimisation en utilisant les algorithmes génétiques (AG). Elle est conçue pour servir de support de programmation.

4.3. Terminologie utilisée pour les algorithmes génétiques

Les algorithmes génétiques étant basés sur des phénomènes biologiques, il est très important de rappeler préalablement quelques termes de génétique :

En biologie, les organismes vivants sont constitués de cellules, dont les noyaux comportent des **chromosomes**. L'élément de base de chromosome est le **gène**. Chaque gène a une fonction, (couleur des yeux, type de cheveux,...etc.). La position d'un gène sur le chromosome est son **locus**. L'ensemble des gènes d'un individu est son **génotype** et l'ensemble du patrimoine génétique d'une espèce est le **génom**e. Les gènes ayant une même fonction sont appelés **allèles** (**Pitiot, 2009**).

On utilise aussi, dans les algorithmes génétiques, une analogie avec la théorie de l'évolution qui propose qu'au fil du temps, les gènes conservés au sein d'une population donnée sont ceux qui sont le plus adaptés aux besoins de l'espèce vis-à-vis de son environnement.

4.4. Eléments de base de l'algorithme génétique

Pour modéliser un problème d'optimisation en utilisant les algorithmes génétiques, nous devons définir les principaux éléments suivants.

4.4.1. Individu

Les individus correspondent aux « solutions » du problème à optimiser. Ces solutions doivent être « codées » pour que le traitement puisse être effectué par l'algorithme génétique. Cette représentation codée d'une solution est appelée chromosome et est composée de gènes. Chaque gène peut représenter une variable, un élément de la solution, ou encore une partie plus abstraite. La manière la plus utilisée de codage par l'algorithme génétique est le codage en vecteurs. Chaque solution est représentée par un vecteur. Ce vecteur peut être binaire ou encore de n'importe quel type discret dénombrable (entier, caractères,... etc.) (Pitiot, 2009). On pourrait également utiliser un type continu (ex : nombres réels), mais dans ce cas, il faut également revoir les opérations qui modifient le contenu des chromosomes (la fonction qui crée aléatoirement les chromosomes et les opérateurs génétiques).

La simplicité veut que les chromosomes soient uniformes, c'est-à-dire que tous les gènes soient du même type. Cependant, si on tient compte encore une fois des opérations qui modifient le contenu des chromosomes, on peut assez aisément construire des vecteurs d'éléments de type différent. On demande habituellement que les chromosomes soient tous de même longueur, basés sur la même architecture, les gènes homologues étant au même endroit sur leur chromosome respectif. De fait, le codage par vecteur est si utilisé (très grande simplicité comparée aux autres méthodes de codage de chromosome) que les algorithmes sont souvent identifiés comme étant des méthodes de traitement vectoriel. Cette affirmation n'est pas tout à fait vraie car d'autres types de codage existent, bien que n'étant pas très fréquents. Par exemple, l'utilisation des algorithmes génétiques pour faire de la programmation génétique utilise un codage en arbre (ce qui permet entre autres d'avoir des chromosomes de longueurs différentes). Dans ce travail, nous avons une collection de codage ; vecteur, réel, string et booléen.

4.4.2. Population

C'est l'ensemble des individus, ou encore l'ensemble des chromosomes d'une même génération. Habituellement, la taille de la population reste constante tout au long de l'algorithme génétique.

4.4.3. Génération

Habituellement, au départ d'un algorithme génétique, il faut créer une population d'individus initiale. Ces individus sont générés par une fonction simple. Cette fonction affecte à chaque individu qu'elle crée une valeur aléatoire pour chacun de ses gènes. L'algorithme génétique peut également utiliser comme population de départ, une population déjà créée a priori.

4.4.4. Fonction d'évaluation (*fitness*)

C'est la fonction à optimiser. Elle permet de mesurer la qualité d'un individu. Cette fonction retourne une valeur numérique (habituellement réelle). C'est selon cette valeur numérique que sont calculées les chances de sélection de cet individu. Comme nous avons dit précédemment (voir la **section 4.1.2**), Les algorithmes génétiques étant une technique d'optimisation, ils cherchent la qualité maximale, donc l'optimisation de la fonction d'évaluation. Si on cherche plutôt à minimiser une fonction, il faudra la modifier de sorte que la fonction de qualité se maximise.

4.4.5. Sélection

Selon la catégorie des individus, chacun se voit attribuer un pourcentage de chances d'être choisi pour la reproduction qui correspond à l'importance relative à la qualité de l'individu par rapport à la qualité totale de la population. Il existe plusieurs méthodes de sélection, nous citons ici les plus importantes.

4.4.5.1. Sélection par roulette (*wheel*)

Dans ce type de sélection, Il faut imaginer une sorte de roulette de casino sur laquelle chaque individu (chromosome) est représenté par une valeur correspond à son adaptation au problème (**Goldberg, 1989**). La **Figure III-3** donne un exemple de sélection par roulette.

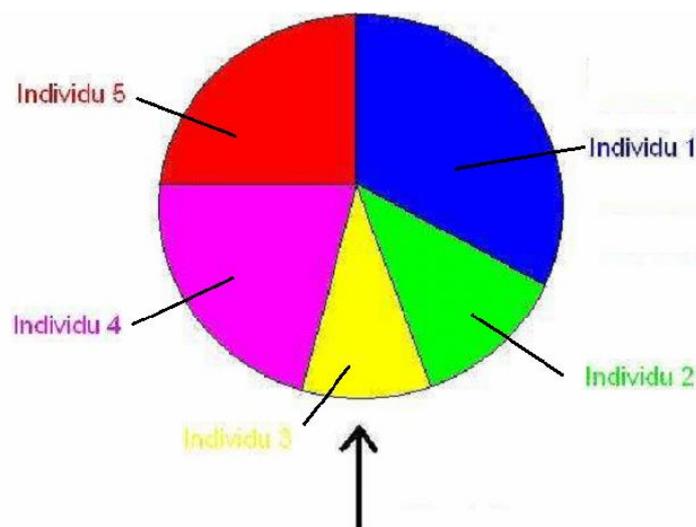


Figure III-3. Exemple de sélection par roulette (Goldberg, 1989).

Afin de sélectionner un individu, la bille est lancée et s'arrête sur un chromosome. Les meilleurs chromosomes peuvent ainsi être tirés plusieurs fois et les plus mauvais ne jamais être sélectionnés.

4.4.5.2. Sélection par Rang

La sélection précédente rencontre des problèmes lorsque la valeur d'adaptation des chromosomes varie énormément. Si la meilleure fonction d'évaluation d'un chromosome représente 90% de la roulette alors les autres chromosomes auront très peu de chance d'être sélectionnés et on arriverait à une stagnation de l'évolution. Cette technique de sélection choisit toujours les individus possédant les meilleurs scores d'adaptation, le hasard n'entre donc pas dans ce mode de sélection. En fait, si n individus constituent la population, la sélection appliquée consiste à conserver les k meilleurs individus (au sens de la fitness) suivant une probabilité qui dépend du rang (et pas de la fitness) (Whitley, 1994).

4.4.5.3. Sélection steady-state

L'idée principale de cette stratégie est qu'une grande partie de la population puisse survivre à la prochaine génération. L'algorithme génétique s'exécute alors de la manière suivante.

A chaque génération, on sélectionne quelques chromosomes (parmi ceux qui ont le meilleur fitness) pour créer des chromosomes fils. Ensuite les chromosomes les plus mauvais sont retirés et remplacés par les nouveaux. Le reste de la population survit à la nouvelle génération (Syswerda, 1989).

Dans cette thèse, nous allons utiliser ce type de sélection car elle apparaît la méthode la plus adéquate pour sélectionner nos solutions. Elle permet de préserver la meilleure solution rencontrée lors du processus évolutif de l'algorithme génétique car à chaque itération, le meilleur individu est sauvegardé pour être réinjecté dans la population de l'itération suivante. De cette manière nous garantissons la survie du meilleur chromosome pour les générations suivantes.

4.4.6. Croisement et mutation

Le croisement (*crossover*) s'effectue généralement en croisant deux individus, ce qui produit deux nouveaux individus à placer dans la nouvelle population. De la manière classique, le croisement consiste à appliquer les opérateurs génétiques sur les deux chromosomes sélectionnés et mettre les deux chromosomes résultant dans la nouvelle population. Chaque chromosome fils reçoit environ la moitié des gènes de chacun de ses parents. La probabilité de croisement est presque toujours de 50% (effectue de cette manière un mélange plus efficace). Selon cette probabilité, on échange les gènes homologues des deux

chromosomes. De cette façon, après le croisement, on obtient deux fils qui sont complémentaires par rapport à leurs parents (si un fils reçoit un gène d'un tel parent, l'autre fils tiendra ce même gène de l'autre parent). Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global.

L'opérateur de mutation, bien qu'ayant une probabilité bien moindre (habituellement entre 0,5% et 5%) joue un rôle très important. Ainsi, les gènes des fils sont limités par les gènes des parents et si un gène n'est pas présent dans la population initiale (ou s'il disparaît à cause des reproductions), il ne pourra jamais se développer chez les descendants. L'opérateur de mutation est là pour combler ce problème. La mutation permet de conserver ce qu'on appelle la diversité génétique. Habituellement, la mutation crée des individus faibles peu aptes à survivre. Cependant, une certaine mutation pourrait se révéler « géniale » et permet d'augmenter grandement l'évolution de la population.

4.4.7. Validité et cohérence

Selon la méthode de codage et sa signification, on doit être toujours certain que les individus de notre population soient valides. En effet, que ferait-on d'une solution donnant une bonne note de qualité, mais n'ayant aucun sens pratique une fois interprétée. Il faut donc s'assurer que la fonction de création des individus crée toujours des individus valides et que les opérateurs génétiques conservent la validité des individus traités. Ceci vise à conserver la cohérence générale de l'algorithme génétique.

4.4.8. Conditions d'arrêt

Généralement, un algorithme génétique se termine après un certain nombre de générations, mais on peut également terminer l'exécution de l'algorithme lorsqu'une certaine condition soit atteinte, par exemple lorsque le temps d'exécution de l'algorithme ou la qualité d'un individu dépasse un certain seuil.

4.4.9. Codage des solutions

L'étape de codage des solutions est une étape très importante dans la modélisation des problèmes utilisant les AG, c'est la façon de codage des solutions (ce que l'on a appelé ici les chromosomes), c'est-à-dire les structures de données qui coderont les gènes. Elle se place généralement après une phase de modélisation mathématique du problème traité. La qualité du codage des données conditionne le succès des algorithmes génétiques. Les codages binaires ont été très utilisés à l'origine. Les codages réels sont désormais largement utilisés, notamment dans les domaines applicatifs pour l'optimisation de problèmes à variables réelles. Il existe trois principales façons de codage de solution.

4.4.9.1. Codage binaire

Ce codage a été le premier à être utilisé dans le domaine des AG. Ce type de codage est certainement le plus utilisé car il présente plusieurs avantages. Son principe est de coder la solution selon une chaîne de bits (qui peuvent prendre les valeurs 0 ou 1). **Holland (1975)** utilise ce type de codage dans ses premiers travaux et prouve la supériorité de ce type de codage par rapport au codage de chaînes de caractères. Les avantages présentés par ce type sont :

- Il utilise un alphabet minimum $\{0,1\}$,
- Facilité de mise en point d'opérateurs génétiques,
- Existence de fondements théoriques (théorie sur les schémas).

A coté des avantages, ce type de codage présente quelques inconvénients:

- Ce codage est souvent peu naturel par rapport à un problème donné (l'évolution des poids d'arcs dans un graphe par exemple est difficile à coder sous la forme d'une chaîne de bits).
- Les performances de l'algorithme sont dégradées devant les problèmes d'optimisation de grande dimension à haute précision numérique. Pour de tels problèmes, les AG basés sur les chaînes binaires ont de faibles performances comme il est montré dans (**Michalewicz, 1992**).
- La distance de Hamming entre deux nombres voisins (nombre de bits différents) peut être assez grande dans le codage binaire: l'entier 7 correspond à la chaîne 0111 et la chaîne 1000 correspond à l'entier 8. Or, la distance de hamming entre ces deux chaînes est de 4, ce qui crée bien souvent une convergence, et non pas l'obtention de la valeur optimale (**Guenounou, 2009**).

4.4.9.2. Codage string

Dans ce type, on utilise les caractères pour le codage, par opposition de codage de bits. Souvent, ce type de codage est plus naturel que celui énoncé auparavant. Il est utilisé dans de nombreux cas de résolution des problèmes qui utilise les algorithmes génétiques.

4.4.9.3. Codage réel

Chaque chromosome est représenté par une série de valeurs quelconque, du contexte de problème. La **Figure III-4** donne un exemple de codage d'une solution en codage réel.

556	59.12	456	18.33	661
-----	-------	-----	-------	-----

Figure III-4. Codage réel.

4.5. Procédure d'évolution d'un algorithme génétique

De manière globale, l'algorithme génétique suit les quatre étapes suivantes.

- Sélectionner les individus pour le croisement,
- Effectuer le croisement et la mutation,
- Évaluer la qualité (*fitness*) des individus et leurs chances de survie,
- Remplacer l'ancienne population par la nouvelle.

Cette itération est répétée autant de fois que demandée. L'évaluation de la qualité (*fitness*) d'un individu permet d'illustrer avec une valeur numérique (et donc plus facile à manipuler) la qualité des gènes qui forment l'individu. Plus la qualité d'un individu est grande, plus ses chances d'être sélectionné pour le croisement sont grandes. En général, on calcule les chances de croisement d'un individu en regard de sa qualité en relation avec la qualité totale des individus de la population. Le croisement se fait en croisant deux individus. On applique sur les deux individus choisis les opérateurs génériques, le croisement et la mutation. Le croisement (reproduction) donne deux fils (*offspring*), sont placés dans la nouvelle population. On répète le croisement jusqu'à ce qu'on ait rempli la nouvelle population (la taille de la population devrait rester constante). On remplace alors l'ancienne population par la nouvelle et on recommence le processus jusqu'à la satisfaction des conditions d'arrêts.

La **Figure III-5** présente le corps général d'un algorithme génétique.

- 1- Générer une population de individus de taille N : $x_1, x_2, x_3, \dots, x_N$.
- 2- Calculer les chances de survie (qualité ou encore *fitness*) de chaque individu. $f(x_1), f(x_2), f(x_3), \dots, f(x_N)$.
- 3- Vérifier si le critère d'arrêt est atteint. Si oui, terminer.
- 4- Choisir une paire d'individus pour le croisement (selon les chances de survie de chaque individu).
- 5- Selon les probabilités associées à chaque opérateur génétique (croisement et mutation), appliquer ces opérateurs.
- 6- Placer les individus produits dans la nouvelle population.
- 7- Vérifier si la taille de la nouvelle population est correcte. Si non, retourner à l'étape 4.
- 8- Remplacer l'ancienne population d'individus par la nouvelle.
- 9- Retourner à l'étape 2.

Figure III-5. Algorithme génétique générique.

5. Contribution 1 : Modèle hybride pour optimiser le processus de généralisation

Dans cette section, nous allons discuter notre première contribution.

5.1. Motivation

Nous avons discuté précédemment que la généralisation est un processus subjectif et holistique, durant lequel, le cartographe peut appliquer des règles très complexes. Donc, il est très important de modéliser ce processus en utilisant une approche d'optimisation car il permet de résoudre les conflits de la généralisation. Aussi, l'approche AGENT montre beaucoup d'avantages quand elle est adaptée à la généralisation mais le problème est que l'agent ne peut pas détecter la solution optimale ou la solution la plus proche parmi les scénarios qui sont proposés pour résoudre les conflits.

5.2. Principes

Dans cette thèse, nous avons proposé une autre solution qui permet d'atteindre toujours la meilleure solution qui représente la solution optimale ou celle qui est proche de l'optimale. Cette approche se compose de deux approches l'une de l'optimisation ; l'algorithme génétique et l'autre de l'approche AGENT. L'objectif général de notre proposition consiste à trouver un compromis entre la qualité de données généralisées et le temps d'exécution du processus de la généralisation.

Notre proposition se base sur les éléments suivants.

- Agents génétique : pour gérer le processus de généralisation locale et d'échanger des informations pertinentes avec les agents voisins.
- Flux d'informations : les agents voisins s'échangent des messages qui contiennent des informations sur l'état interne de chaque agent. Des informations sur l'environnement sont collectées potentiellement à partir des capteurs. Les informations récoltées sont utilisées comme entrée (input) à l'algorithme génétique.
- Algorithmes génétiques: l'agent utilise AG pour trouver dynamiquement la solution optimale qui est appliquée par le processus de généralisation.

Chaque agent exécute une séquence d'algorithmes avec des paramètres. Ces paramètres prennent leurs valeurs dans un domaine continu, défini par le concepteur. Les agents cherchent à respecter les différents types des contraintes ; internes et relationnelles. Pour chaque agent, Le déroulement de l'algorithme génétique permet d'affecter des valeurs aux paramètres des algorithmes de généralisation de manière à respecter les contraintes qu'il

subis. Le problème est résolu quand une valeur paramétrique a pu être affectée à chaque algorithme de généralisation de manière à ce que toutes les contraintes soient satisfaites.

5.3. Diagramme de classe UML

Dans les sections précédentes, nous avons présenté les fondements sur lesquels se base notre approche. Dans cette section, nous présentons le diagramme de classe que nous avons conçu pour modéliser notre système.

Selon notre approche, les objets géographiques sont modélisés sous forme d'agents instance. Il existe principalement trois agents ; agent route, agent bâtiment et agent point. Il existe aussi d'autres agents ; un agent groupe qui remplace un ensemble d'agents si on applique l'opération d'agrégation, agent interface et agent superviseur. L'agent exécute un ou plusieurs itérations de l'algorithme génétique pour définir la solution optimale qui se compose d'une séquence des algorithmes de généralisation avec des paramètres convenable pour satisfaire progressivement ses contraintes et effectuer ainsi sa propre généralisation. La **Figure III-6** présente le diagramme de classe du modèle proposé.

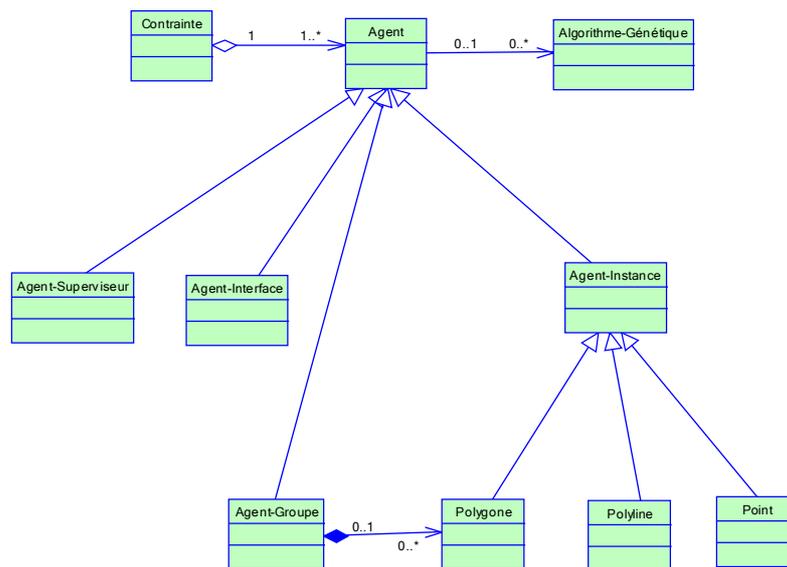


Figure III-6. Diagramme de Classe UML.

5.4. Architecture de système

Le modèle que nous avons fait se base principalement sur le modèle AGENT (**Ruas & Duchêne, 2012**) qui profite au fond des avantages des algorithmes génétiques pour trouver la solution optimale en un moment donnée et dans n'importe quelle situation donnée dont le but est d'optimiser le processus de la généralisation automatique de données spatiales.

5.4.1. Architecture de l'agent génétique

Les bases de données spatiales contiennent les objets géographiques de trois types différents; polygones, rubans (polylines) et les points. On considère dans ce travail, les deux

premiers types. Ces objets sont modélisés par les agents que l'on appellera dans ce contexte **agent génétique**. Chaque agent se généralise selon son état et son but pour satisfaire au maximum leur contraintes. Pour ce faire, chaque agent génétique doit être doté de trois composants principaux :

- Ressources génétiques sous forme de gène.
- *Un optimiseur* qui utilise l'algorithme génétique pour trouver la solution optimale en fonction des contraintes interne ou relationnelles.
- Capacité de communication avec les agents voisins.

L'architecture de notre agent se compose principalement de deux modules; l'optimiseur qui représente le module de généralisation et un autre, c'est le module de contrôle d'optimisation. Le rôle de l'optimiseur consiste à exécuter l'algorithme génétique correspond au type d'agent qui l'associe. L'objectif principal d'ajouter le module de contrôle d'optimisation est de compter le temps d'exécution du processus de la généralisation et le nombre d'agents en conflits. Ce dernier module communique avec les autres modules des agents pour acquérir les informations nécessaires au contrôle. **La Figure III-7** présente l'architecture de l'agent génétique qui est semblable à tous les types d'agent.

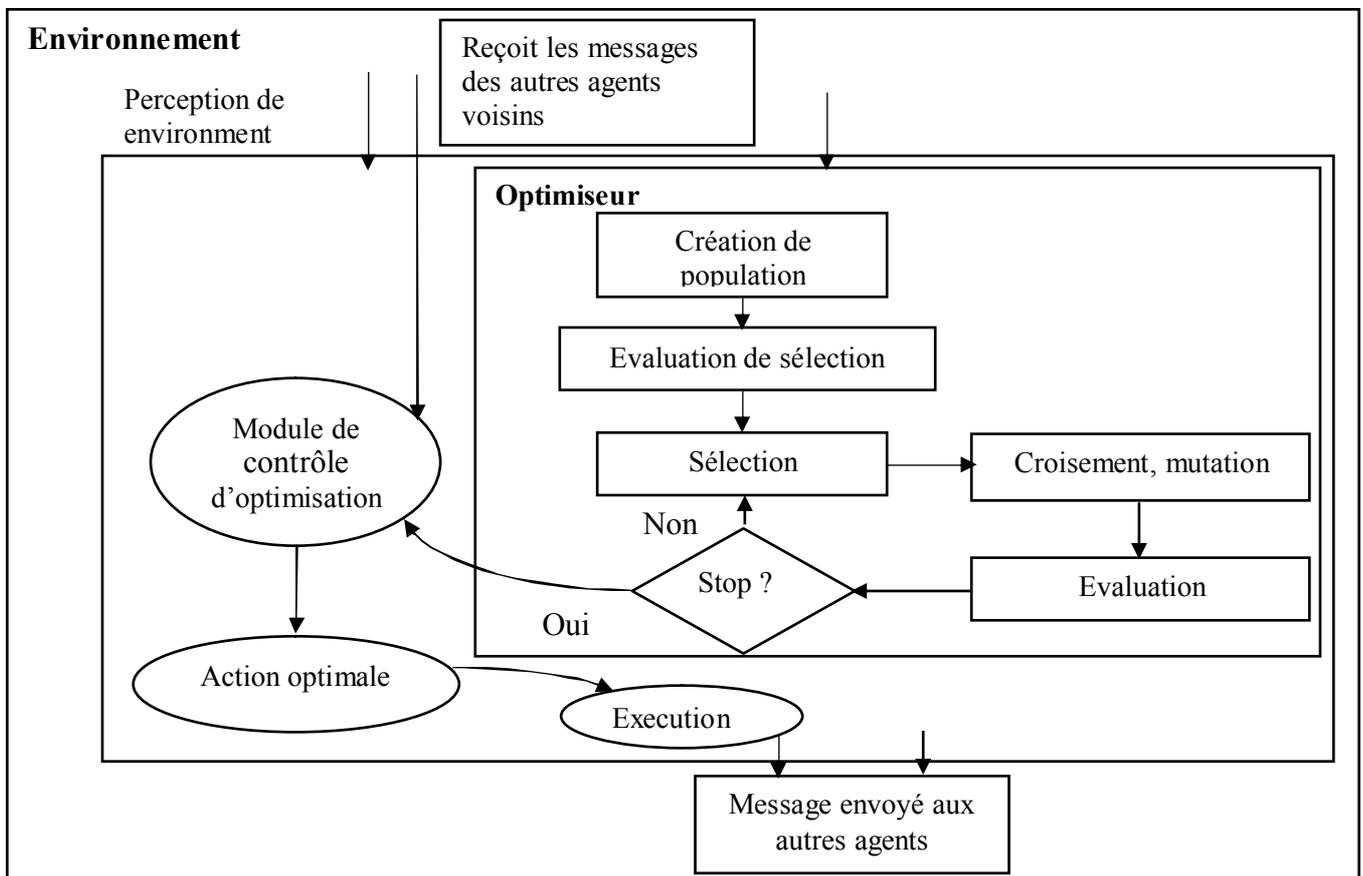


Figure III-7. Architecture générale de l'agent génétique.

L'agent perçoit son environnement à partir d'un capteur pour construire une représentation de cet environnement qui lui permet d'identifier les contraintes relationnelles qu'il partage avec les agents voisins. L'agent dispose d'un optimiseur qui exécute un algorithme génétique permettant de définir des actions possibles à exécuter pour augmenter la satisfaction de ces contraintes en fonction de leur nature et de leur état. Une action possible est une séquence des algorithmes avec leurs paramètres convenables qui vont optimiser le processus de la généralisation. L'agent, en tant qu'agent communiquant, reçoit des messages pour améliorer le processus de la généralisation. Un agent doit être capable de percevoir tous les agents voisins proche de lui.

5.4.2. Structure d'agent

On a considéré deux types principaux d'agent ; agent route et agent bâtiment. Dans ce qui suit, nous allons représenter les ressources génétiques de ces deux types (**Lejdel & Kazar, 2012**).

5.4.2.1. Gène Route

Un agent route est caractérisé par son identificateur et par les algorithmes qui peuvent être appliqués sur lui pour effectuer sa généralisation. Ces algorithmes peuvent prendre des paramètres. Les codages des gènes se fait sous forme multiple, on utilise les formes caractères pour coder les identificateurs, les formes binaire pour coder l'application ou non de tels algorithmes (0 pour dire que l'algorithme n'est pas appliqué, 1 pour dire que l'algorithme est appliqué) et les formes réelles pour coder les paramètres des algorithmes de généralisation, et tous ces éléments sont regroupés dans des vecteurs. La **Figure III-8** donne un exemple de gène de l'agent route.

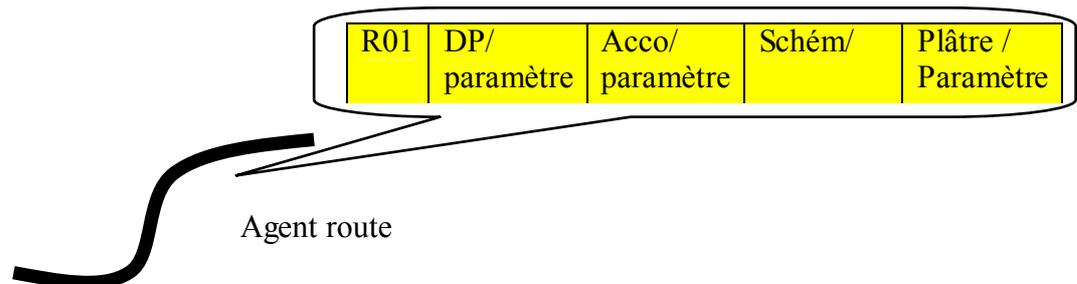


Figure III-8. Exemple de gène de l'agent route.

5.4.2.2. Gène bâtiment

Un agent bâtiment est caractérisé par son identificateur et par les algorithmes de généralisation : algorithme de simplification et algorithmes de déplacement. Les mêmes types de codage précédemment cités, sont utilisés pour coder ce type de gène. La **Figure III-9** donne un exemple de gène de l'agent Bâtiment.

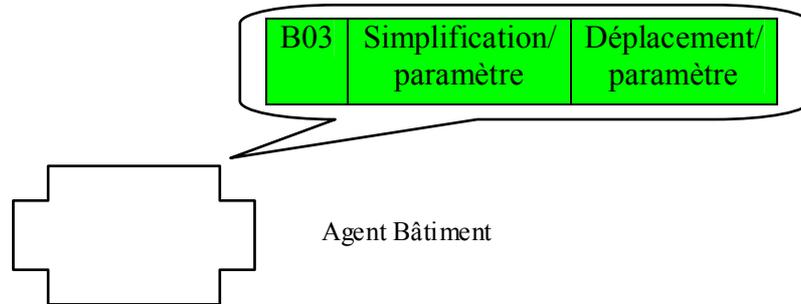


Figure III-9. Exemple de gène de l'agent bâtiment.

5.4.3. Modules de l'agent génétique

Le rôle principal d'un agent génétique est de se généraliser, afin de l'adapter au niveau de détail demandé par l'utilisateur. Ainsi, l'agent génétique est responsable de la satisfaction de ses contraintes. Il doit collaborer avec les agents voisins pour éviter une violation des contraintes. Elle s'applique sous forme d'une séquence d'algorithmes de généralisation avec leurs paramètres convenables, ce qui est généré par son optimiseur. L'architecture de l'agent génétique est composée de deux principaux modules.

5.4.3.1. Module de généralisation (Optimiseur)

Ce module réalise le processus de généralisation de données spatiales, il applique la solution trouvée par l'optimiseur. L'optimiseur exécute un algorithme génétique pour trouver le chromosome optimal. Il suit les étapes classiques d'un algorithme génétique qui sont la sélection, le croisement et la mutation. La solution est représentée par une séquence d'algorithmes et leurs meilleurs paramètres.

Chaque agent dispose d'un optimiseur et d'un simulateur qui sont utilisés pour découvrir la séquence d'algorithmes et leurs paramètres qui permettront d'optimiser le processus de la généralisation de données géographique selon les conditions de l'état actuel (l'état interne de l'agent et de l'environnement). L'utilisation d'un algorithme génétique pour le processus de la généralisation avait un avantage majeur par rapport aux systèmes qui s'appuient sur des séquences prédéfinies, car l'algorithme génétique permet à chaque agent de découvrir les séquences qui peuvent se ressembler pas à n'importe quelle séquence prédéfinie mais peut être optimal selon les conditions courantes de l'environnement.

Afin d'assurer un processus d'optimisation plus performant, le simulateur doit atteindre un équilibre entre le temps de la découverte de la séquence et la qualité de résultat obtenu. L'*optimiseur* exécute un algorithme génétique. La solution est affinée petit à petit au fil des itérations jusqu'à une convergence vers une solution qui s'approche de la solution optimale, une certaine marge d'imperfection est donc acceptée. L'AG suit les étapes

classiques d'un algorithme génétique qui sont la sélection, le croisement et la mutation. A chaque itération, l'AG choisit le chromosome qui avait une fitness minimale ç-a-d qui réduit principalement le nombre des conflits et améliore sa qualité. Les étapes à suivre sont :

1) Sélection

Dans cette thèse, nous avons utilisé la méthode sélection steady-state (voir la **section 4.4.5.3**). L'idée principale de cette méthode, est qu'une grande partie de la population puisse survivre à la prochaine génération. L'algorithme génétique marche alors de la manière suivante : à chaque génération, on sélectionne quelques chromosomes (parmi ceux qui ont le meilleur coût) pour créer des chromosomes fils. Ensuite les chromosomes les plus mauvais sont retirés et remplacés par les nouveaux. Le reste de la population survie à la nouvelle génération.

2) Croisement

C'est un opérateur qui est utilisé par les algorithmes génétiques. Son rôle fondamental est de permettre la recombinaison des informations présentes dans le patrimoine génétique de la population. La **Figure III-10** donne un exemple de croisement.

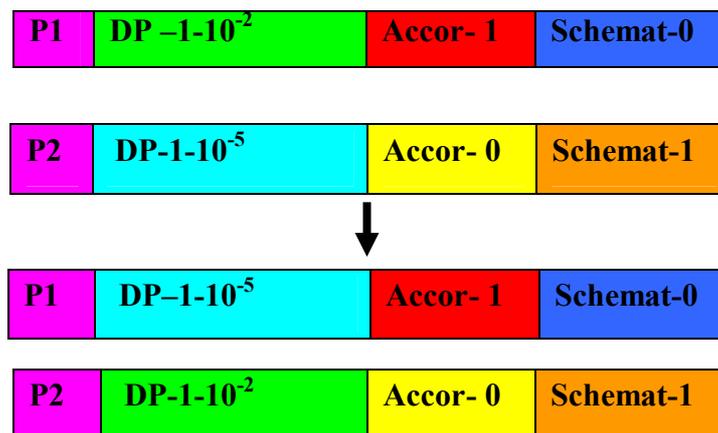


Figure III-10. Opérateur de croisement.

3) Mutation

C'est une modification qui intervient de manière aléatoire sur le génome de chromosome, elle a pour but de maintenir une certaine diversité dans la population. Cette mutation n'intervient que sur une partie suffisamment petite pour ne pas détruire les caractéristiques qui ont été sélectionnées mais suffisamment grande pour apporter des éléments nouveaux à un individu. Le pourcentage de la mutation doit être très faible. Il est appliqué aux tous paramètres qui représentent un chromosome, si le point de mutation est une valeur booléenne oui, il est remplacé par non et vice versa. Si ce point est une valeur réelle alors la

nouvelle valeur est générée aléatoirement à partir d'un intervalle donné. La **Figure III-11** donne un exemple de mutation.

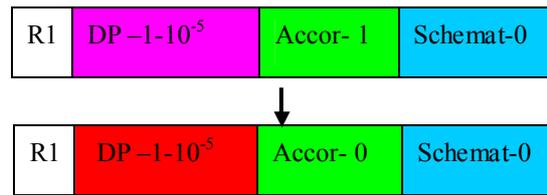


Figure III-11. Opérateur de mutation.

4) Evaluation

La fonction d'évaluation se base sur certaines mesures qui permettent d'évaluer la qualité des données géographiques, on restreint sur un ensemble des mesures :

FO : qui calcule la perte de la forme de l'objet lors de la transformation. Pour les bâtiments, on calcule la différenciation dans les surfaces *S* et pour les routes, on utilise les mesures de McMaster.

$$FO = \sum \Delta S + \Delta (\text{McMaster})$$

Un nombre minimal est une bonne solution.

NC : c'est le nombre d'objets en conflit, un nombre minimal est une bonne solution.

DP: c'est la somme des distances (*dx* et *dy*) normalisées et absolues de déplacements effectués sur les bâtiments par rapport aux positions initiales.

$$DB = \sum_1^n \sqrt{(dx_i^2 + dy_i^2)}$$

Donc la fonction totale serait :

$$F = NC + DP + FO$$

La solution optimale est donc celle qui possède la plus petite valeur de fonction totale.

5.4.3.2. Module de contrôle d'optimisation

Le module de contrôle d'optimisation doit atteindre un équilibre satisfaisant entre le temps de la découverte de la solution optimale et la qualité des résultats. Ainsi, ce module contrôle le temps de généralisation pour ne pas dépasser les limites maximales et recevoir les messages d'agents voisins qui contiennent des informations pertinentes, telles que le nombre d'agents en conflits, la distance entre les objets voisins ... etc., pour accomplir le calcul de la fitness générale de solution. Ce module arrête l'exécution de l'algorithme génétique pour les trois raisons suivantes : lorsqu'on a atteint un certain fitness, lorsqu'on a atteint un nombre fixe d'itérations ou lorsqu'on a dépassé un certain temps d'exécution. Dans les deux derniers cas, on choisit la solution qui possède le meilleur fitness pour la restituer.

5.4.4. Types des agents

Plus des agents mentionnés ci-après, il existe aussi dans notre système deux agents supplémentaires, ils sont créés pour guider en mieux le processus de la généralisation :

5.4.4.1. Agent interface

On peut ajouter également un agent *interface* entre l'utilisateur et le module de la généralisation des cartes dont l'objectif est de gérer la communication entre eux. Il permet à l'utilisateur de transmettre ses requêtes. La requête transmise comporte des informations importantes pour la recherche des données, telles que l'identification de la zone, le type de carte et son niveau de détail (Lejdel et al., 2009). Le niveau de détail est arbitraire et non prédéfini. Ce module permet également de tracer les données finales généralisées.

Dans cette approche, les agents génétiques négocient avec leurs agents voisins, via des messages dont les informations sont utilisées comme des entrées de l'algorithme génétique, ces données permettent à l'agent de résoudre les différents conflits à la fois et éviter des nouveaux conflits à apparaître dont le but est de réaliser un processus de généralisation optimale.

5.4.4.2. Agent superviseur

Pour guider un processus de généralisation optimal, les agents doivent interagir pour résoudre les conflits spatiaux qui apparaissent lors de la génération des données souhaitées. Chaque agent peut planifier localement une séquence d'algorithmes de généralisation avec leurs paramètres convenables. Le rôle de l'agent superviseur se limite à la synchronisation de l'exécution des plans des différents agents de la carte. L'agent superviseur synchronise l'activation des agents génétiques, soit de manière séquentielle ou de manière parallèle selon l'état actuelle de l'environnement et selon les résultats des algorithmes génétique de différents agents. Il essaye toujours d'activer les agents de manière parallèle pour diminuer le temps d'exécution de processus de généralisation.

Durant le processus de la généralisation de données, des conflits peuvent apparaître et ils ne peuvent pas être résolu par les agents, par exemple, s'il ne trouve pas une solution en raison de manque d'espace. Dans cette situation, les agents en conflits envoient une demande à l'agent superviseur afin de résoudre ces conflits. L'agent superviseur s'intervient pour proposer une solution. Il peut exécuter un algorithme génétique générale pour résoudre ces conflits.

L'agent superviseur ordonne les gènes des routes et de bâtiments en conflit pour former le chromosome global de l'AG, la longueur de chromosome est égale au nombre de gènes qu'il contient. La **Figure III-12** présente l'opération de croisement globale.

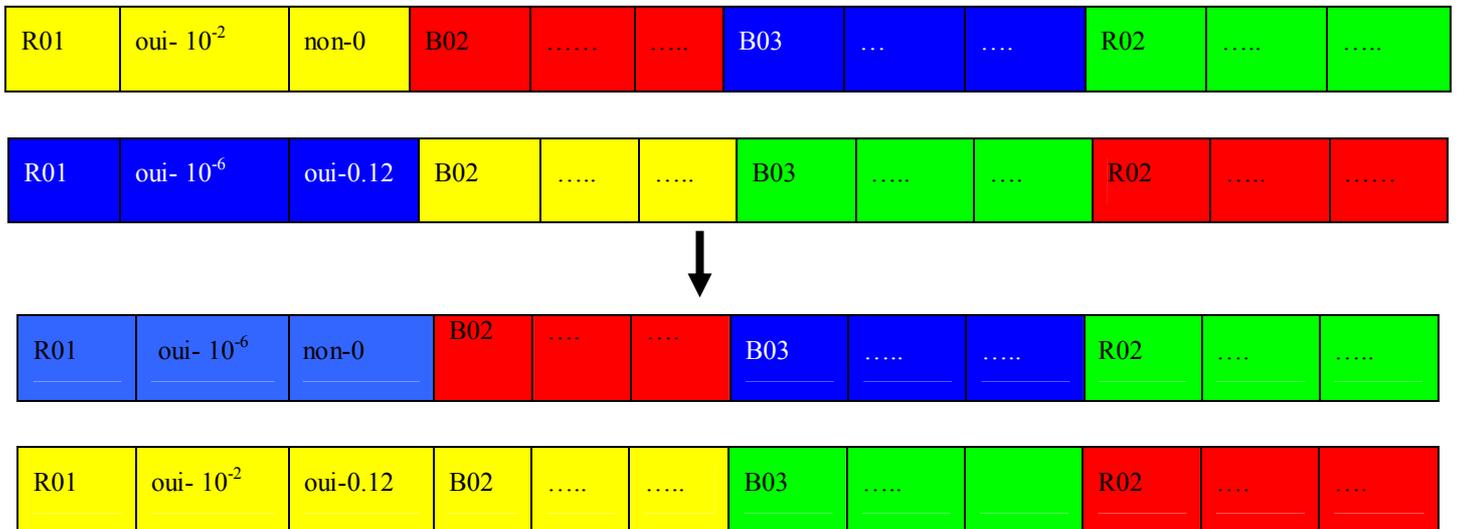


Figure III-12. Opération de croisement globale.

Contrôler le processus de généralisation de carte est une tâche très complexe qui nécessite un temps de traitement supplémentaire, cela peut ralentir le processus de généralisation entière de la carte. Il n'est pas possible d'attribuer cette tâche à un seul agent, nous avons besoin de partitionner l'espace de la carte en sous-espaces (Jabeur, 2006). Le contrôle de chaque sous-espace est alors attribué à un agent spécifique (agent superviseur). Dans la littérature, la méthode la plus populaire du partitionnement de l'espace de la carte est basé sur l'utilisation du réseau routier (Lamy et al., 1999). Cette méthode de partitionnement a été utilisée dans plusieurs travaux comme le projet AGENT.

L'agent concerné de contrôle d'une partition donnée doit superviser toutes les modifications au sein de cette partition et de déterminer, si nécessaire, l'action optimale appropriée pour résoudre les conflits en utilisant l'algorithme génétique. Aussi, à l'intérieur de chaque agent, il existe un module de contrôle d'optimisation qui permet de contrôler le processus de généralisation locale (interne).

5.4.5. Relations spatiales entre les agents

Les agents qui se trouvent dans une même zone partagent des relations spatiales entre eux, ces relations doivent être identifiées par chaque agent de cette zone, pour détecter facilement les conflits spatiaux qui peuvent survenir pendant la généralisation de la carte. Donc, selon la situation, il est nécessaire de déterminer, quand les objets sont considérés éloignés, quand les objets sont considérés comme des voisins et quand les objets sont considérés plus proche (proximité), ce dernier est un forme de conflit. En général, il est un peu difficile de définir formellement la notion de voisinage (Jabeur, 2006). Voisinage et proximité se sont des formes de voisinage. Lors de la généralisation cartographique, le conflit

de proximité est l'un des plus importants et difficiles à corriger. Sa correction nécessite le plus souvent le déplacement ou la fusion des objets, voire une élimination de l'objet.

Les relations spatiales sont attachées à la forme géométrique de l'objet et les distances qu'ils se séparent. Dans cette thèse, on utilise la triangulation de Delaunay (**Tsai, 1993**) afin d'identifier les relations de voisinage entre des objets spatiaux. Généralement, pour définir les relations de voisinage, on utilise la distance entre les objets. A ce point, nous allons proposer un modèle topologique basé sur des équations mathématiques permettant de définir les relations topologiques entre les objets spatiaux et d'effectuer les transformations nécessaires durant le processus de la généralisation automatique des données spatiales. Pour plus de détail, voir la **section 6**.

5.4.6. Motivation des agents

Les motivations des agents, c'est qui pousse les agents à soumettre des actions (**Ferber, 1995**) et (**Duchène, 2004**) classent les motivations en quatre catégories.

- les motivations personnelles, liées à sa satisfaction personnelle et à l'atteinte des buts qu'il s'est fixé,
- les motivations environnementales, liées à ce qu'il perçoit du monde extérieur,
- les motivations relationnelles, liées aux demandes et influences qui lui sont adressées par les autres agents,
- les motivations sociales, liées "aux demandes exprimées par l'organisation de niveau supérieur et par le concepteur" (**Ferber, 1995**).

Les agents s'auto-généralisent et pour ce faire, ils doivent satisfaire au mieux l'ensemble de ses contraintes cartographiques internes et relationnelles, ou éventuellement de s'éliminer de manière à ce que l'ensemble des contraintes de la base de données cartographique soient satisfaites au mieux (**Duchène, 2004**). Donc, il doit répéter l'exécution de son algorithme génétique (plusieurs itérations) jusqu'à atteindre une situation d'équivalence qui met la valeur de fitness le plus minime. Ainsi, l'agent doit communiquer avec les agents voisins. S'ils aboutissent à une situation de conflit qui ne peut être résolue par la généralisation individuelle, il sollicite l'agent superviseur d'exécuter l'algorithme génétique global (**voir la section 5.4.4.2**).

5.4.7. Interaction et communication intra et inter-agents

La généralisation cartographique s'attache à satisfaire au mieux l'ensemble des contraintes cartographiques auxquelles sont soumis les objets de la base de données cartographique (**Duchène 2004**). En effet, les agents constituant la carte cherchent à satisfaire

au mieux l'ensemble des contraintes cartographiques intra et inter-agents. Dans notre système, chaque agent génétique cherche à satisfaire au mieux ses contraintes cartographiques internes (taille, empâtement, ...etc.) et les contraintes relationnelles qu'il partage avec ses voisins (proximité, maintien d'alignement, ...etc.). L'objectif de notre thèse consiste à atteindre un état d'équilibre qui satisfasse au mieux les deux types de contraintes en même temps. La coordination d'actions est donc nécessaire pour "permettre à des agents ayant des objectifs distincts mais dépendants les uns des autres de satisfaire ces objectifs et d'accomplir leur travail en tirant éventuellement parti de cette dépendance"(Ferber, 1995). Nous adaptons ici la définition d'interaction donnée en (Weiss, 1999) :

"Interaction: Generally, everything that occurs between agents (agent-agent interaction) and between agents and their environment (agent-environment interaction). Agents can interact directly via –verbal– communication (by exchanging information) and indirectly via their environment (by passively observing one another or by actively carrying out actions that modify the environmental state). Interaction may result in changes in the internal state and the future course of activity of an agent".

De nombreux travaux ont été menés sur la définition et la standardisation de langages de communication entre agents au sein d'un système multi-agents (Duchène, 2004). Le fruit de ces travaux est un langage de communication appelé communément *ACL* (Agent Communication Language). Il est utilisé pour échanger de l'information entre les agents. Dans cette thèse, nous allons utiliser le langage *ACL* qui est mis au point par la FIPA 2000.

5.4.8. Opérations spécifiques

Nous allons utiliser deux opérations, l'agrégation et l'élimination.

5.4.8.1. Agrégation des objets

L'agrégation consiste à remplacer un ensemble d'objets par un seul objet sous certaines conditions. La géométrie de l'objet créé dérive de la géométrie des objets agrégés. Les agrégations spatiales sont toujours associées à des agrégations sémantiques. Les agrégations sont plus complexes pour trois raisons :

- Il faut d'abord détecter les objets à agréger, ce qui impose l'existence d'outils d'analyse des proximités et une prise de décision basée sur le résultat de l'analyse.
- Elles utilisent l'espace entre les objets à agréger, ce qui requiert une analyse de cet espace de séparation (son contenu sémantique, sa taille et sa forme),
- L'objet agrégé doit souvent posséder une forme compatible avec les formes des objets à agréger. Il existe donc des méthodes d'agrégation propres aux types d'objets à agréger.

Les agrégations sont dans la plupart des cas utilisées pour remplacer deux objets surfaciques par un seul, soit parce qu'un des deux objets est trop petit, soit parce qu'on considère que l'espace de séparation entre les objets est une information inutile. Un cas particulier est l'agrégation de bâtiments où la contrainte de forme résultante impose l'utilisation de méthodes spécifiques. On distingue deux types principaux d'agrégation (Ruas, 1999) :

- les agrégations par remblai qui remplissent l'espace entre les deux bâtiments,
- les agrégations par déplacement qui déplacent les bâtiments l'un vers l'autre (en pondérant éventuellement les déplacements en fonction de la taille des bâtiments).

La Figure III-13 décrit les différents types de l'agrégation.

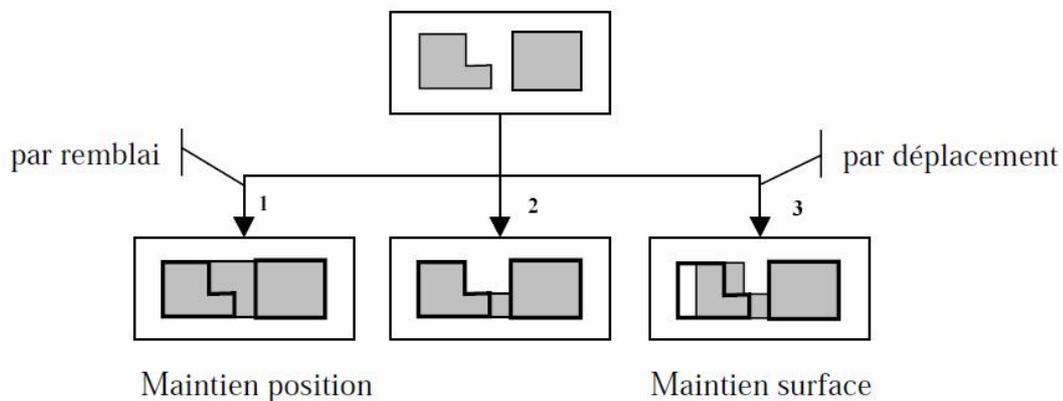


Figure III-13. Types d'agrégation.

5.4.8.2. Elimination

L'élimination est une opération spécifique qui est exécutée par l'agent lui-même, alors si la taille de l'agent est sous le seuil de lisibilité ou s'il ne trouve pas une solution pour une situation de conflits, dans ce dernier cas, c'est l'agent superviseur qui ordonne l'objet concerné à s'éliminer. Il se supprime et notifie l'agent superviseur ainsi que ses voisins possibles autour de son élimination pour permettre à relaxer leurs contraintes. Mais, il est noté ici, que nous devons vérifier l'importance de l'objet dans la carte, s'il est un repère dans la carte comme mairie ou hôpital...etc., on ne devra pas l'éliminer. Donc, l'opération d'élimination ne représente pas une perte d'information car les objets affectés par cette transformation n'ont pas une grande importance. Ces importances définies selon les objectifs de la création de la carte et son contexte d'utilisation.

5.4.9. Cycle de vie d'un agent génétique

Le cycle de vie d'un agent se commence depuis sa création Tous les agents commencent par l'identification de ses agents proches et de leurs contraintes internes et relationnelles. L'agent génétique crée son chromosome et identifié les algorithmes de

généralisation et leur valeurs paramétriques préalables, c'est-à-dire il propose les premiers plans possibles qui sont considérés comme une population initiales de l'algorithme génétique. L'*optimiseur* de l'agent exécute un algorithme génétique qui suit les étapes classiques : sélection, croisement, mutation et évaluation. Lorsqu'il trouve la solution optimale, il l'exécute. L'agent vérifie sa lisibilité ; s'il est lisible et il n'a pas des conflits avec les agents voisins, il passe à la fin de processus, c'est l'état pendant lequel il génère ses données finales et l'envoi à l'agent interface et met à jour la base de données. Sinon, il répète le cycle depuis le début. La **Figure III-14** présente le cycle de vie d'un agent génétique.

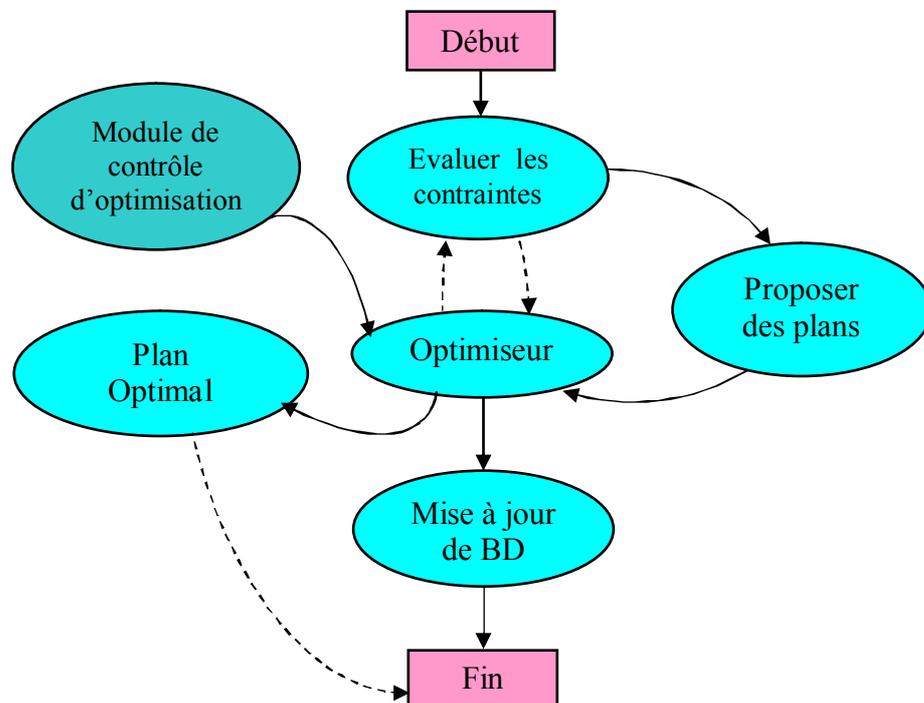


Figure III-14. Cycle de vie d'un agent.

6. Contribution 2 : Modèle de généralisation des relations topologiques

Dans cette section, nous allons détailler notre deuxième contribution.

6.1. Motivation

Pour modéliser les objets géographiques, nous avons principalement quatre objets géométriques : 0D (points), 1D (lignes), 2D (zones) et 3D (solides). Mais la réalité est beaucoup plus complexe que cela. Il est également fréquent de constater que les rues, les rivières et les routes peuvent être modélisés comme des lignes ou des polygones, mais en réalité (ground), ce sont des zones (surfaces) avec des propriétés spécifiques de sorte qu'ils peuvent être réduits à des lignes en cas de besoin. Afin de prendre en compte ces caractéristiques, le concept de *ruban* sera détaillé. L'application des opérateurs de généralisation sur les données géographiques peut entraîner des conflits topologiques. Pour éviter ces conflits, des conditions topologiques sont utilisées pour générer les relations topologiques comme : disjoint, meet, contain, overlap en d'autres relations (Lejdel et al., 2015). Avant de définir le modèle mathématique des relations topologiques pour les rubans et les régions, nous présentons quelques définitions mathématiques.

6.2. Rubans (Ribbons)

Un ruban a une largeur W , une longueur L et un rapport (ratio) R_l , tel que : $R_l = L / W$. Le ratio est plus grand qu'une valeur positive R_L , tel que $R_l > R_L$. La valeur minimale possible pour le seuil R_L est 10. Notons que $Skelton(R)$ est l'axe d'un ruban. Le ruban peut contenir des trous qui peuvent être utiles pour la modélisation des îles dans les rivières (Lejdel et al., 2015).

Pour simplifier la présentation, un ruban sera représenté par un rectangle oblong. Par exemple une autoroute peut être décrite par plusieurs rubans qui sont correspondant à plusieurs voies de circulation, les voies d'urgence et une médiane (voir la **Figure III-15**).

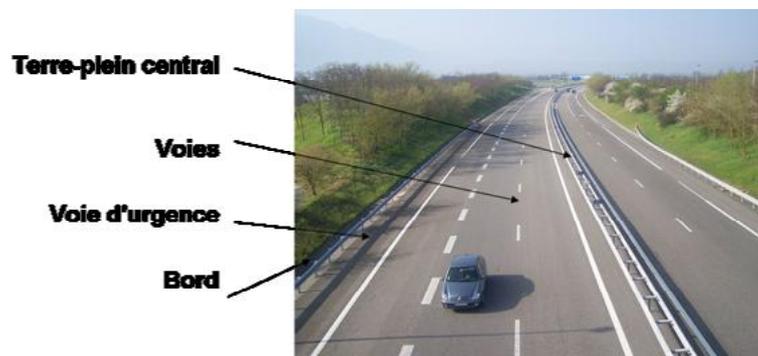


Figure III-15. Le modèle ruban appliqué sur les autoroutes.

6.3. Processus de réduction d'échelle

Dans la plupart des cas, la représentation de l'échelle requise, n'existait pas dans la base de données géographique (BDG), donc une dérivation de la représentation requise de l'une des représentations existantes est nécessaire. Ce processus d'adaptation et de réduction de la représentation existante à l'échelle demandée est appelé le processus de généralisation. Lors de la réduction de l'échelle d'une carte (représentation cartographique des objets géographique), les relations topologiques entre les objets peuvent varier de même que les modifications de leurs géométries. Dans ce contexte, nous allons traiter principalement deux types d'objets : les rubans (linges) et les régions (voir la **Figure III-16**). On peut utiliser le processus tel qu'il est décrit dans (**Laurini, 2014**) :

Étape 0: les objets géographiques originaux sont modélisés comme des régions et / ou des rubans,

Étape 1: quand l'échelle diminue, les petites surfaces (régions) et les rubans seront généralisés et éventuellement peuvent être fusionnés,

Étape 2: quand l'échelle continue à diminuer, les régions se transforment en points et les rubans se transforment en des lignes (le squelette du ruban),

Étape 3: quand l'échelle continue de diminuer, les points et les lignes peuvent disparaître.

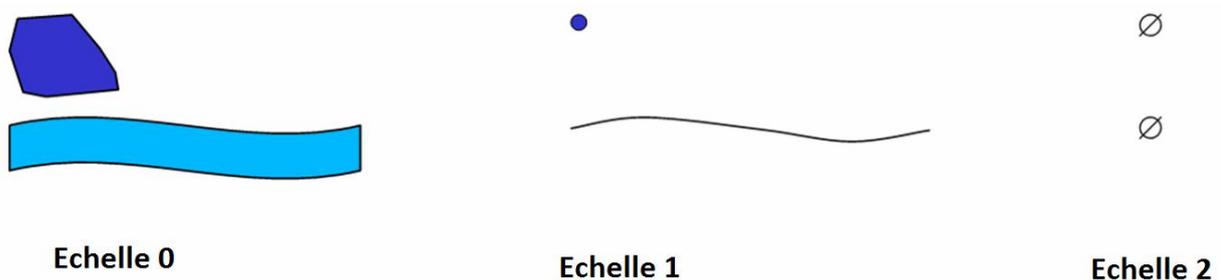


Figure III-16. La généralisation selon l'échelle.

6.4. Acuité visuelle appliquée à des objets géographiques

Dans un SIG, la représentation cartographique est liée à « l'acuité visuelle » (**Laurini, 2014**). Dans ce contexte, des seuils doivent être définis. En cartographie classique, la limite est comprise entre 1 mm et 0,1 mm. Si on prend une route dans certaine échelle et si sa transformation donne une largeur de ruban plus que 1 mm, cette route est une zone, entre 1 mm et 0,1 mm, c'est une ligne et si elle est moins de 0,1 mm, la route disparaît. Le même raisonnement est valable pour les villes ou les petits pays comme Andorre,

Liechtenstein, Monaco...etc. Dans ces cas, les «trous» en Italie ou en France disparaissent cartographiquement. Dans la suite de cette thèse, ces abréviations seront utilisées.

$Inters(R^1, R^2)$: qui représente l'intersection entre R^1 et R^2 .

$Dist(R^1, R^2)$: c'est la distance entre R^1 et R^2 .

$Area(R^1 \cap R^2)$: qui représente la surface de la zone de l'intersection entre R^1 et R^2 .

$2Dmap(R^1, \sigma)$: c'est une fonction de transformation d'un objet géographique en une certaine échelle avec la généralisation, en deux dimensions.

Avec des seuils définis comme : $\varepsilon_i = 0,1$ mm, $\varepsilon_{lp} = 1$ mm, On peut obtenir :

a. la disparition d'un objet géographique (O) à l'échelle σ :

$$\forall O \in \text{GeObject}, \forall \sigma \in \text{Scale} : O_\sigma = 2Dmap(O, \sigma) \wedge Area(O_\sigma) < (\varepsilon_i)^2 \Rightarrow O_\sigma = \phi.$$

b. La transformation d'une région en un point (par exemple, le centre de gravité de l'objet concerné, par exemple considéré comme le centre du Rectangle Englobant le Minimum (REM)):

$$\forall O \in \text{GeObject}, \forall \sigma \in \text{Scale} : O_\sigma = 2Dmap(O, \sigma) \wedge (\varepsilon_i)^2 < Area(O_\sigma) < (\varepsilon_{lp})^2 \Rightarrow O_\sigma = \text{Centroid}(O).$$

c. La transformation d'un ruban R en une ligne (par exemple, son squelette):

$$\forall R \in \text{Ribbon}, \forall \sigma \in \text{Scale} : R_\sigma = 2Dmap(R, \sigma) \wedge \varepsilon_i < Width(R_\sigma) < \varepsilon_{lp} \Rightarrow R = \text{Skel}(R).$$

6.5. Relations Ruban – Ruban

Dans cette section, nous allons classer les relations topologiques entre les rubans selon certaines caractéristiques, puis une description mathématique sera donnée pour chaque type de relation. Ainsi, deux rubans peuvent être «disjoint» ou «intersect». La disjonction est définie par une distance qui sépare les deux rubans mais l'intersection entre deux rubans peut être un Point (0D), un ligne (1D) ou une zone (2D). Dans ce qui suit, nous pouvons formellement définir la description mathématique pour chaque type de relation topologique lorsqu'on utilise des seuils et des mesures métriques, telle que la surface d'un objet, la distance entre deux objets ou plus, ...etc.

6.5.1. Relation « Disjoint »

Deux rubans peuvent être « disjoints », s'il existe une distance entre eux. La **Figure III-17** présente cinq cas possibles de ce type de relation topologique.

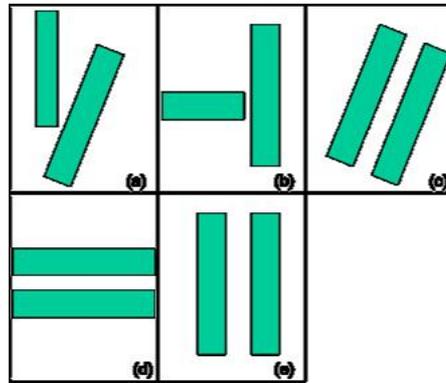


Figure III-17. Relations « disjoint » entre deux rubans.

Nous pouvons définir formellement ce type de relation topologique par l’assertion suivante.

$$\forall R^1, R^2 \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R^1_\sigma = 2Dmap(R^1, \sigma)) \wedge (R^2_\sigma = 2Dmap(R^2, \sigma)) \wedge Inters(R^1, R^2) = \emptyset \wedge (Dist(R^1, R^2) > \epsilon_{Ds}) \Rightarrow Disj(R^1_\sigma, R^2_\sigma).$$

6.5.2. Relation « Meet »

Deux rubans R^1 et R^2 sont liés par une relation Meet, notée par $Meet(R^1, R^2)$, lorsque l’intersection des ces deux rubans est un point P (0D) ou une ligne L(1D). La **Figure III-18** présente ce type de relation :

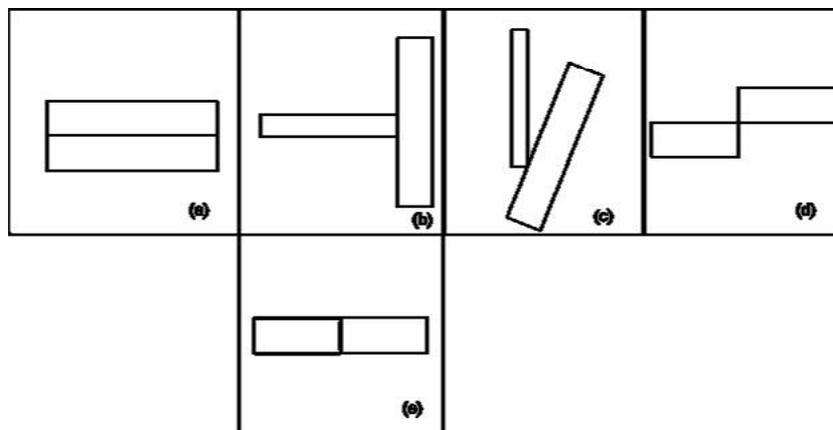


Figure III-18. Relation « Meet ».

Formellement, nous pouvons définir cette relation par l’assertion mathématique suivante.

$$\forall R^1, R^2 \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R^1_\sigma = 2Dmap(R^1, \sigma)) \wedge (R^2_\sigma = 2Dmap(R^2, \sigma)) \wedge (Inters(R^1, R^2) = \{P \vee L\}) \wedge (Dist(R^1, R^2) = 0) \Rightarrow Meet(R^1_\sigma, R^2_\sigma).$$

6.5.3. Relation « Merge »

Deux rubans R^1 et R^2 sont liés par une relation Merge, notée par $Merge(R^1, R^2)$, si l’intersection entre ces deux rubans est une zone. Pour ce type de relation, nous pouvons obtenir sept cas possibles (voir la **Figure III-19**).

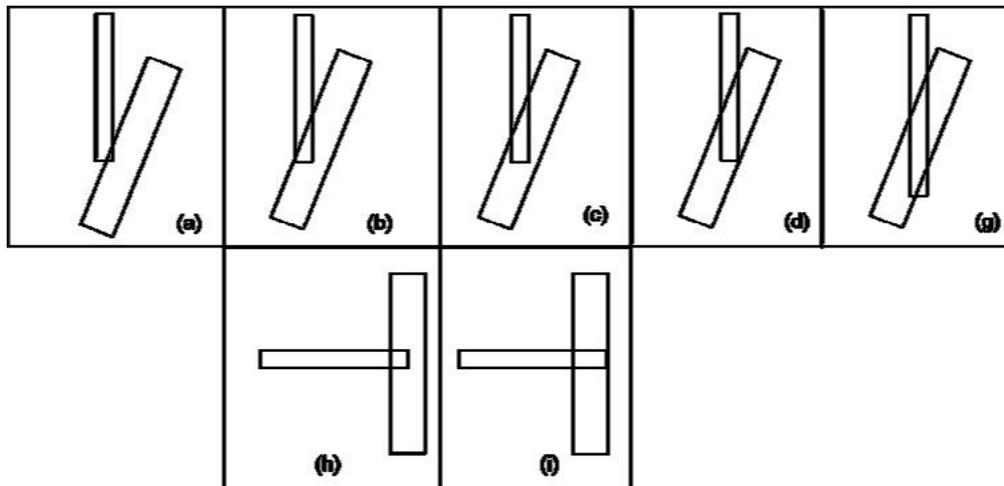


Figure III-19. Relation « Merge ».

Cette relation topologique peut être définie par l'assertion mathématique suivante.

$$\forall R^1, R^2 \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R_\sigma^1 = 2Dmap(R^1, \sigma)) \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge Inters(R^1, R^2) \neq \emptyset \wedge (Area(R^1 \cap R^2) > \varepsilon_{mr}^2) \wedge (Area(CMP(R^1 \cap R^2)) = 0) \Rightarrow Merge(R_\sigma^1, R_\sigma^2).$$

6.5.4. Relation « Cross »

Cette relation topologique est très importante. L'exemple commun qui représente le croisement est le croisement de route-route ou route-rivière. La **Figure III-20** présente deux cas possibles.

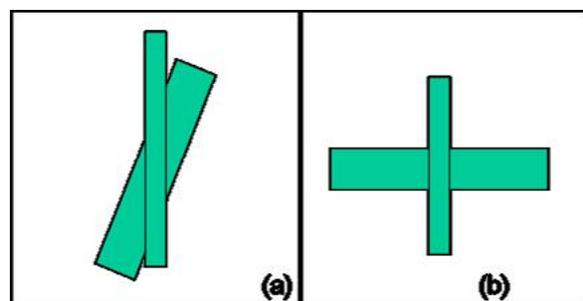


Figure III-20. La relation « Cross ».

La définition de cette relation est basée sur la surface de la zone d'intersection entre les deux rubans R1 et R2. Pour décrire mathématiquement cette relation, un seuil ε_{cr} doit être utilisé. Donc, nous avons:

$$\forall R^1, R^2 \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R_\sigma^1 = 2Dmap(R^1, \sigma)) \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge Inters(R^1, R^2) \neq \emptyset \wedge (Area(R^1 \cap R^2) > \varepsilon_{cr}^2) \wedge (Area(CMP(R^1 \cap R^2)) > 0) \Rightarrow Cross(R_\sigma^1, R_\sigma^2).$$

6.6. Mutation des relations Ruban-Ruban

Lorsque l'échelle se réduit, des mutations des relations topologiques peuvent être appliquées. Les relations topologiques entre les objets varient selon certains critères, nous présentons ces mutations ci-dessous :

6.6.1. Mutation de « disjoint » en « merge »

Lors de la réduction de l'échelle, la relation « disjoint » sera transformée en « merge ». La **Figure III-21** donne un exemple de cette mutation.

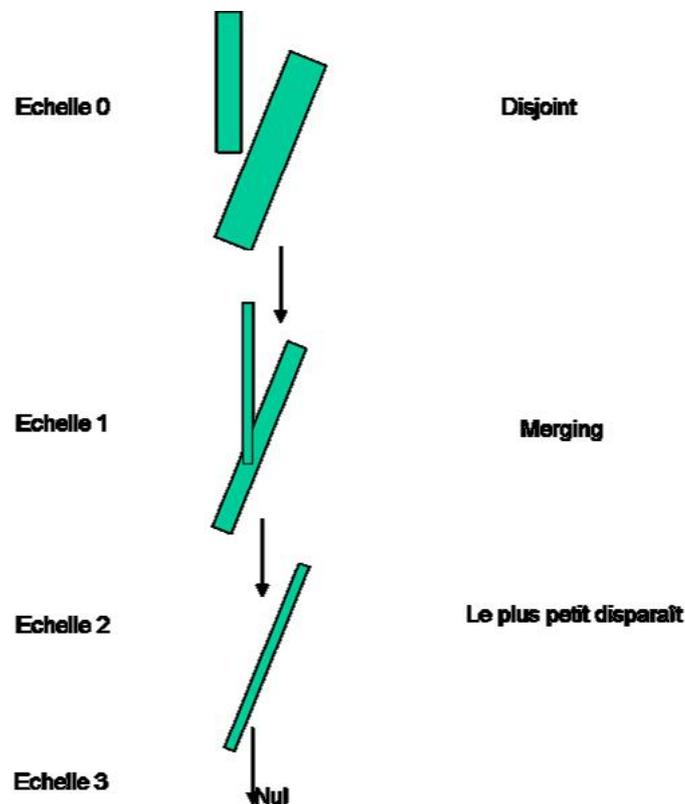


Figure III-21. La Mutation de relation « disjoint » en « merge ».

Ce processus de mutation peut être modélisée comme suit.

$$\forall R^1, R^2 \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R_\sigma^1 = 2Dmap(R^1, \sigma)) \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge Disj(R^1, R^2) \wedge (Dist(R^1, R^2) < \varepsilon_{Dj}) \Rightarrow Merg(R_\sigma^1, R_\sigma^2).$$

Lorsque le ruban deviendra très petit, nous appliquons cette assertion:

$$\forall R \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R_\sigma = 2Dmap(R, \sigma)) \wedge (Width(R_\sigma) < \varepsilon_i) \Rightarrow R_\sigma = \phi.$$

6.6.2. Mutation de relation « cross » en « merge »

La relation « crossing » peut se muter en relation « merge » en fonction de la surface du complément de l'intersection entre les deux rubans : $Area(CMP(R^1 \cap R^2))$. La **Figure III-22** présente la transformation de relation « cross » en « merge ».

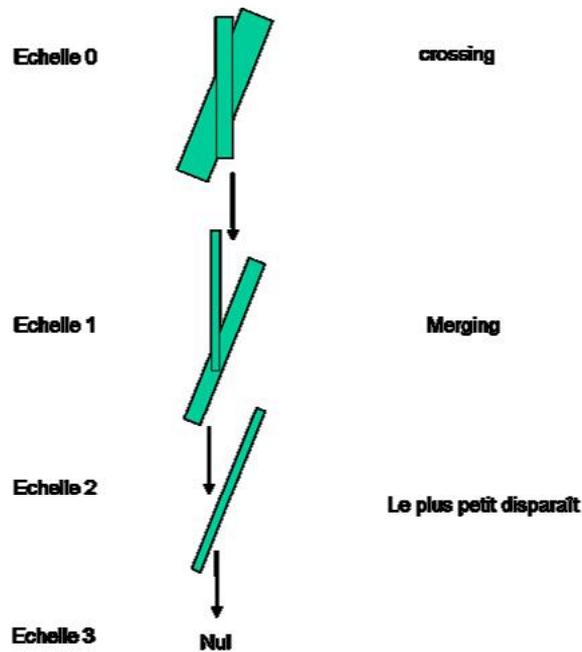


Figure III-22. La Mutation de relation « crossing » en « merging ».

La définition formelle de ce processus est:

$$\forall R^1, R^2 \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R_\sigma^1 = 2Dmap(R^1, \sigma)) \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge (Cross(R^1, R^2)) \wedge (Area(CMP(R^1 \cap R^2)) < Area(R^1 \cap R^2)) \Rightarrow Merge(R_\sigma^1, R_\sigma^2).$$

6.7. Relations Ruban-Région

Dans cette section, nous allons étudier les relations qui peuvent exister entre des rubans et des régions. Pour décrire ces types de relations, nous devons les classer en neuf (9) types, à savoir disjoints, meet, cross, covers (cover-by), overlap, contain (contained-by) et on-boundary. La Figure III-23 présente les différents types de relations Rubans-Région.

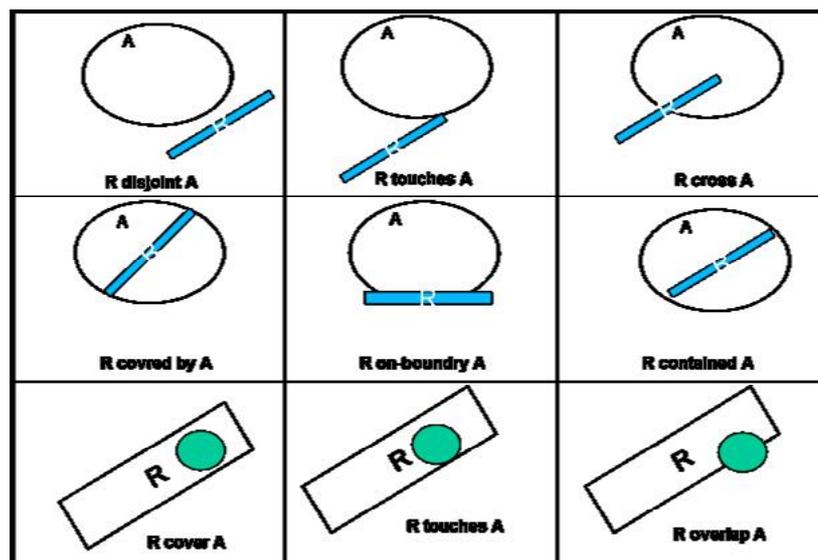
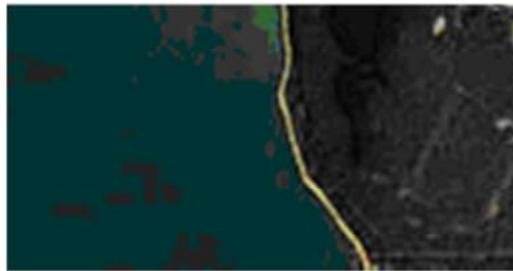


Figure III-23. Relations de base entre régions et rubans.

L'exemple qu'illustre ce cas est: lorsqu'une route passe à la coté de la mer (voir la **Figure III-24**), quels sont exactement les relations spatiales ou géographiques qui sont concernés? Parfois, soit la route touche (*meet*) la mer ou une petite plage est située entre la route et la mer, ...etc. D'un point de vue mathématique, généralement, il y a une relation « disjoint » entre la route et la mer alors que pour les utilisateurs, la relation topologique est différente. Donc, lorsque l'on est en train de lire une carte, selon l'échelle, la relation topologique peut être différent : *disjoints* ou *meet*.



a) La route touche la mer.



b) la route et la mer sont disjointes.

Figure III-24. Selon l'échelle : la route touche (*meet*) ou non la mer.

Par conséquent, les relations topologiques entre les régions et les rubans peuvent aussi varier selon l'échelle, par exemple, la relation « disjoint » peut muter en « *meet* » ou « *on-boundry* » lors de la réduction d'échelle.

6.8. Mutation de relations Rubans-Régions

6.8.1. Mutation de « disjoint » en « meet »

La relation « disjoint » peut se transformer en « *meet* » selon les conditions suivantes (voir la **Figure III-25**).

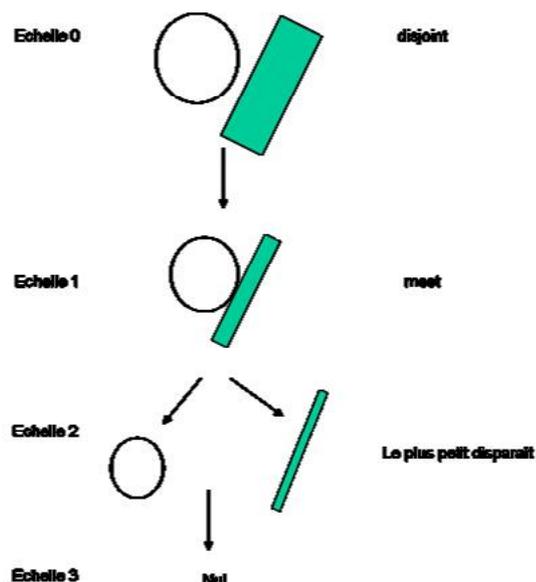


Figure III-25. Mutation de relation « disjoint » en « *meet* ».

La condition exacte est que la distance entre (région et /ou rubans) doit être supérieure à un seuil ε_{Ds} . Cette transformation peut être appliquée selon l’assertion suivante.

$$\forall R \in \text{Ribbon}, \forall G \in \text{Region}, \forall \sigma \in \text{Scale} : \wedge (R_\sigma = 2Dmap(R, \sigma)) \wedge (G_\sigma = 2Dmap(G, \sigma)) \wedge Disj(R, G) \wedge (Dist(R, G) > \varepsilon_{Ds}) \Rightarrow Meet(R_\sigma^1, G_\sigma^2).$$

Lorsqu’un ruban deviendra très petit, nous appliquons cette assertion:

$$\forall R \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (R_\sigma = 2Dmap(R, \sigma)) \wedge (Width(R_\sigma) < \varepsilon_1) \Rightarrow R_\sigma = \phi.$$

La région peut être éliminée si sa superficie est trop petite pour être bien visible. Donc, dans ce cas, la relation initiale ne se tient plus.

$$\forall O \in \text{GeObject}, \forall \sigma \in \text{Scale} : (O_\sigma = 2Dmap(O, \sigma)) \wedge (Area(O_\sigma) < (\varepsilon_{lp})^2) \Rightarrow O_\sigma = \phi.$$

6.8.2. Mutation de « contain » en « cover »

La mutation de relation « contain » en relation « cover » est exprimée par l’assertion suivante (voir la **Figure III-26**).

$$\forall G \in \text{Region}, \forall R \in \text{Ribbon}, \forall \sigma \in \text{Scale} : (G_\sigma = 2Dmap(O, \sigma)) \wedge (R = 2Dmap(R, \sigma)) \wedge (Contains(R, G)) \wedge (Dist(R, G) < \varepsilon_1) \Rightarrow Cover(R_\sigma^1, G_\sigma^2).$$

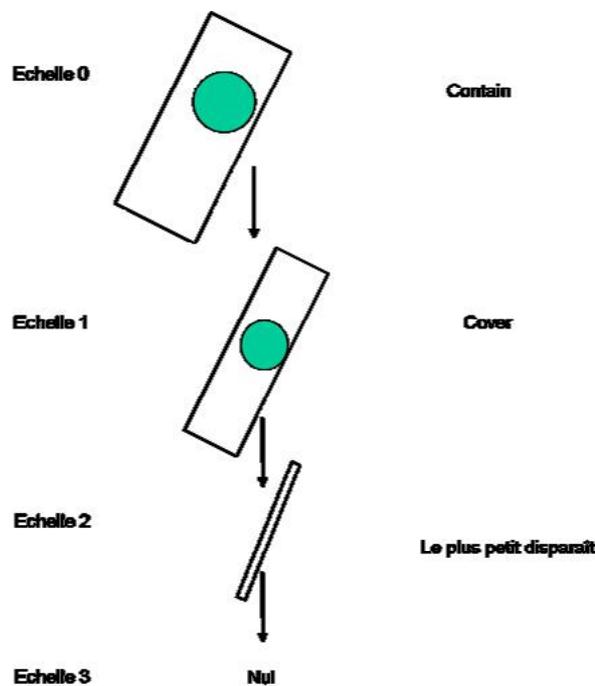


Figure III-26. Mutation de relation « contain » en « cover ».

Mais un objet géographique qui deviendra plus petit peut disparaître ou être éliminé si sa superficie est trop petite pour être bien visible. Donc, dans ce cas, la relation initiale ne se tient plus.

7. Conclusion

Dans notre approche, l'agent génétique est responsable du processus de généralisation et de son optimisation. Cet agent utilise ses connaissances internes et des capteurs en temps réel comme des inputs de l'algorithme génétique pour identifier la séquence d'algorithmes de généralisation et leurs paramètres permettant d'optimiser le processus de la généralisation selon son état et l'état de l'environnement. L'agent échange des messages avec les agents voisins afin d'assurer que ces agents ont également optimisé au maximum leur processus de généralisation.

L'utilisation d'algorithmes génétiques signifie que les agents ne sont pas associés à des séquences des algorithmes prédéfinis, mais ils sont capables de découvrir des nouvelles séquences optimales pour guider un processus de généralisation optimal. Aussi, pour assurer une bonne qualité de la carte et maintenir sa cohérence topologique, nous avons défini un modèle topologique basé sur des équations mathématiques dont l'objectif est guider les mutations de relations topologiques en d'autres relations, durant le processus de généralisation des données géographique.

Dans le prochain chapitre, nous allons présenter une validation des modèles proposés selon l'approche décrite au dessus.

Chapitre IV

Implémentation & test

1. Introduction

Nous avons choisi de valider notre modèle à travers le développement d'une application prototype. L'idée générale consiste à implémenter les modèles développés sur une plateforme Jade, en utilisant le langage de programmation Java avec une bibliothèque GeoTools qui permet de manipuler les objets géographiques. Notre objectif est donc d'implémenter un processus de généralisation facile à manipuler et permettant de gérer les mutations des relations topologiques entre les objets spatiaux.

La démarche suivie dans le développement est décrite dans les paragraphes suivants.

2. Technologie choisie

Nous allons utiliser la plateforme JADE et le langage de programmation Java pour implémenter notre prototype.

2.1. Plateforme JADE

Le meilleur moyen pour construire un système multi-agent (SMA) est d'utiliser une plateforme multi-agent. Une plateforme multi-agent est un ensemble d'outils nécessaire à la construction et à la mise en service d'agents au sein d'un environnement spécifique. Il existe plusieurs technologies pour implémenter les systèmes multi-agent, à savoir MadKit, AgentBuilder, Jack, Beegent et Jade. Dans cette thèse, nous avons choisi d'utiliser JADE (Java Agent Development Framework) qui a été développé par Bellifemine, Poggio, Rimassa et Turci dans la société *Telecom Italia Lab* «Tilab, anciennement CSELT» en 1999 (**Ramdani, 2005**), et plus précisément, nous avons utilisé la version 4.3 de JADE. Elle avait plusieurs avantages. Il permet de :

- simplifier la construction des systèmes multi-agents «SMA interopérables».
- exécuter des SMA.
- réaliser d'applications conformes avec le standard FIPA¹ 1997 pour faciliter la communication des agents JADE avec des agents non JADE.
- essayer d'optimiser les performances d'un système d'agent distribué.

Jade est une plateforme open source (**JADE, 2014**). Elle est utilisée pour le développement des applications d'agents selon les spécifications de FIPA 2000. Elle supporte les applications d'agent qui nécessitent d'exploiter les caractéristiques offertes par la

¹ FIPA (*Foundation for Intelligent Physical Agents*) est une organisation à but non lucratif fondée en 1996 dont l'objectif est de produire des standards pour l'interopération d'agents logiciels hétérogènes.

spécification standard de FIPA (telles que l'échange des messages, le cycle de vie d'agent, ...etc.). L'implémentation des agents dans la plateforme Jade est basée sur le langage Java. Jade fournit aux développeurs plusieurs comportements prédéfinis aux agents tels que des états de machine : simple, cyclique, séquentiel, parallèle et état de fin de la machine (**Jabeur, 2006**). Ces comportements permettent aux développeurs de définir les actions qui seront exécutées par les agents pour aboutir à leurs objectifs.

2.2. Langage de programmation (Java)

Dans tout système informatique, le choix d'un langage de programmation est très important. Bien que les systèmes multi-agent ne soient pas limités au langage Java, ce dernier a été récemment fortement utilisé. En ce qui nous concerne, la portabilité et l'universalité étant des facteurs prédominants dans la conception de notre système multi-agent. Comme on peut citer d'autres raisons :

- Java est un langage du type compilé une fois et exécuté sur toutes machines.
- Les agents développés sous la plate-forme *JADE*, sont entièrement écrits en Java. Ce langage s'est donc imposé comme étant une conséquence de notre précédent choix en termes de plateforme de développement de SMA.
- *Java* possède une bibliothèque immense d'objets prêts à l'emploi, ce qui facilite pleinement la procédure d'implémentation et particulièrement la bibliothèque des outils spatiales, *GeoTools*. Dans notre implémentation, nous avons utilisé la version 12.1 (**GeoTools, 2014**).

Pour ces raisons, on a utilisé le langage Java pour coder notre prototype. Il a été développé dans l'environnement de programmation *Netbeans*, avec la version 7 de *Java JDK* (Java Development Kit). *Netbeans* est un environnement très puissant et très évolué, il facilite la conception des applications et rend la programmation non fastidieuse avec la notion graphique.

3. Architecture du prototype développé

Le prototype que nous avons développé considère trois phases essentielles, chacune d'elle étant réalisée à travers un ou plusieurs modules comme décrits dans la **Figure IV-1**.

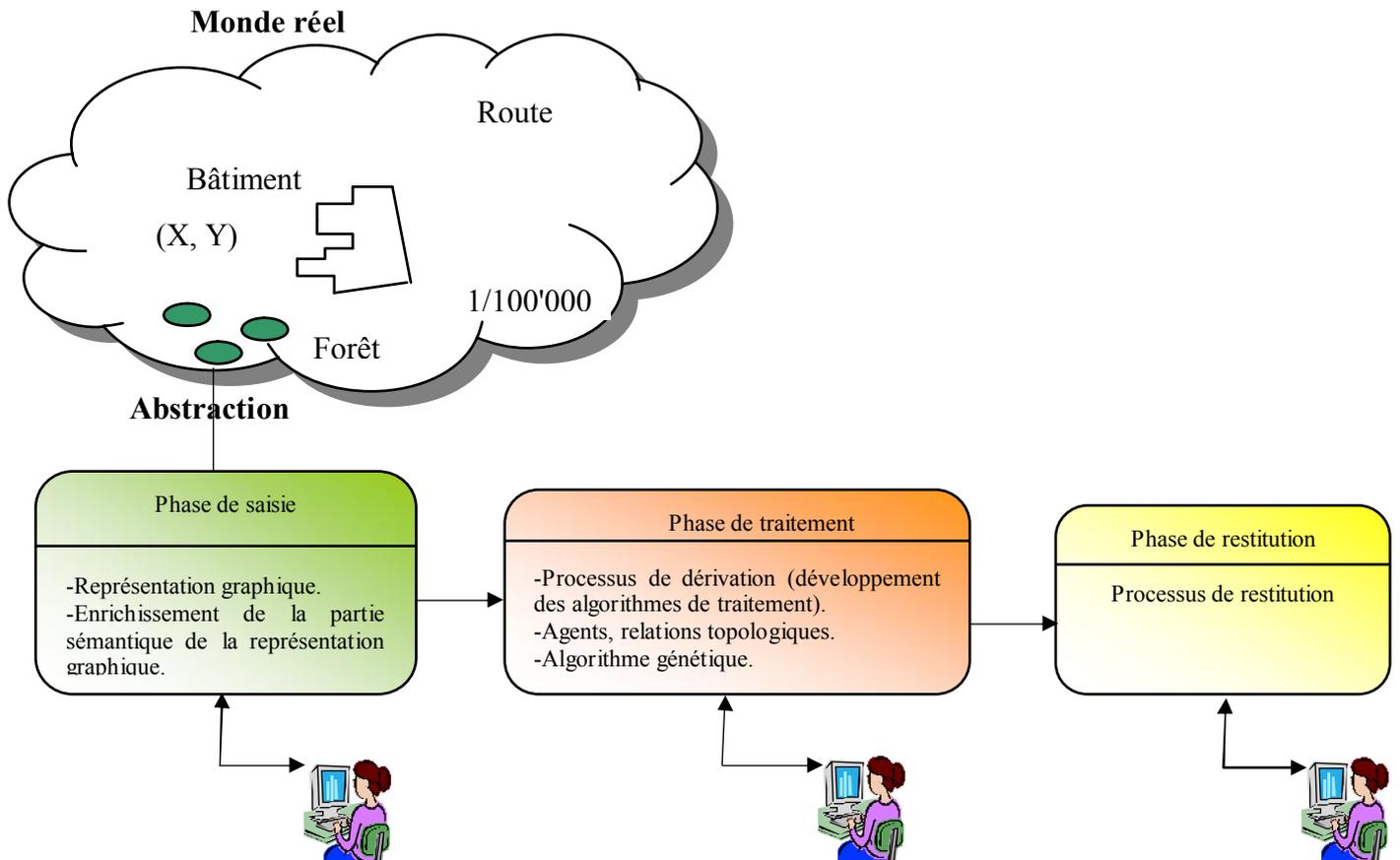


Figure IV-1. Architecture générale de notre prototype.

Rappelons que l'objectif final de notre prototype est de servir efficacement les utilisateurs (niveau de détail souhaité, préférence, contexte d'utilisation). La base de données géographiques est une représentation graphique enrichie par une représentation sémantique afin d'améliorer la personnalisation de la carte et son utilisabilité. Nous avons également développé un processus de généralisation optimale afin de répondre aux insuffisances des prototypes actuels. L'agent génétique se compose de trois modules principaux : module de généralisation et module de contrôle d'optimisation et le module de mutation des relations topologiques. La **Figure IV-2** présente l'architecture de notre prototype.

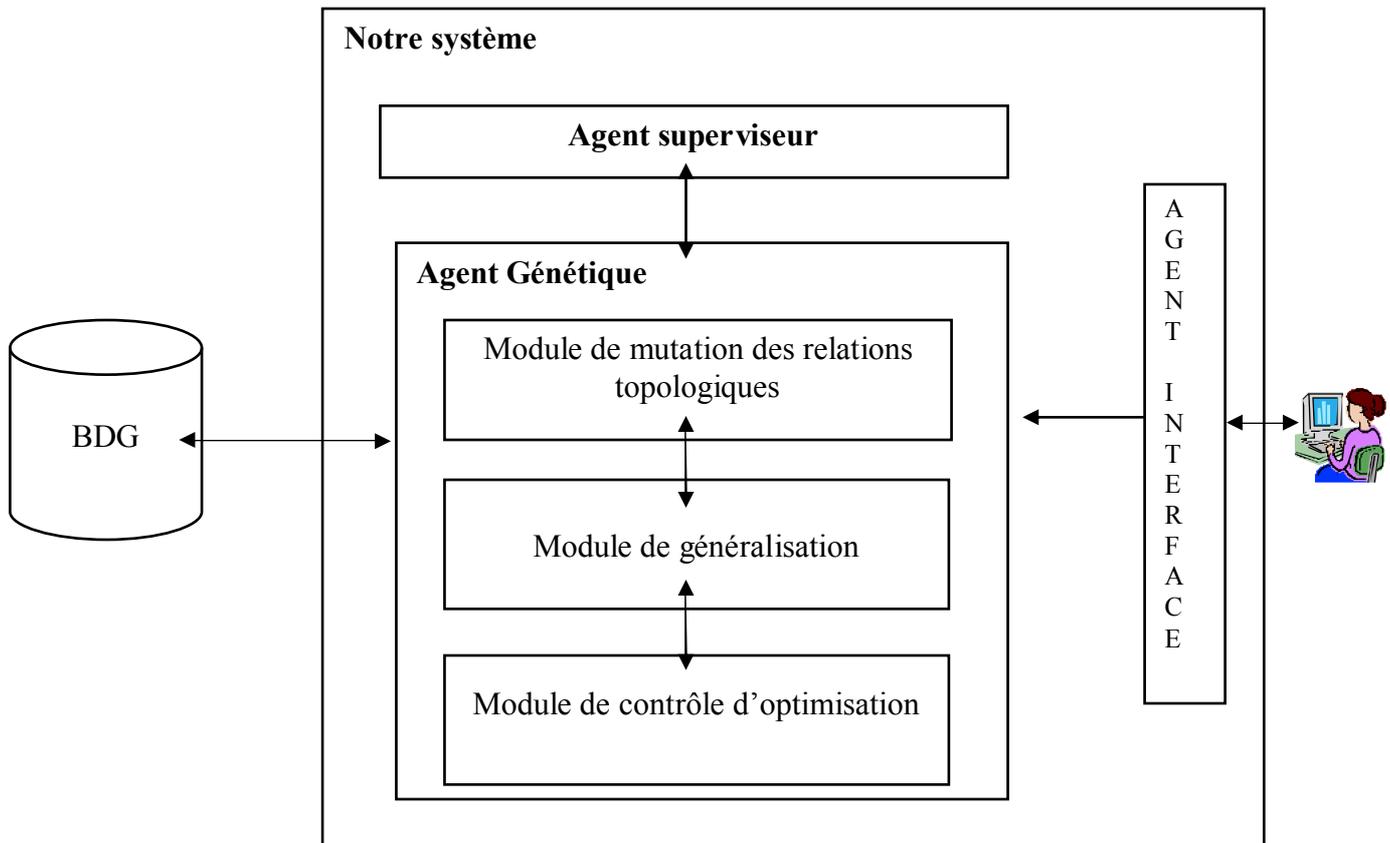


Figure IV-2. L'architecture détaillée de notre prototype.

4. Adaptation niveau de détail / échelle

La notion de niveau de détail dans une BDG correspond à la notion d'échelle couramment utilisée en cartographie. Cette dernière permet de visualiser un détail qui peut être présent dans une base de données en tant que format géométrique comme il peut être directement (sans sauvegarde préalable) affiché sur un support graphique en appliquant des processus de dérivation automatique. Pour trouver un équilibre entre les deux notions, nous devons définir un processus d'affichage pour que l'information soit correctement perçue par l'utilisateur. Le niveau de détail doit être adapté à l'échelle de visualisation (voir la **Figure IV-3**).

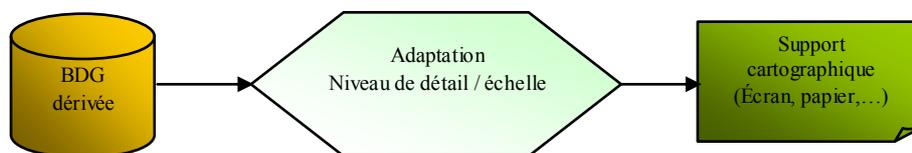


Figure IV-3. Processus d'adaptation niveau de détail / échelle.

5. Expérimentation et résultats

Dans cette section, nous allons présenter une mise en œuvre et une validation de notre travail : les solutions proposées pour optimiser le processus de la généralisation des données géographiques.

5.1. Déroulement du processus de généralisation

Le déroulement de la généralisation est représenté dans la **Figure IV-4**.

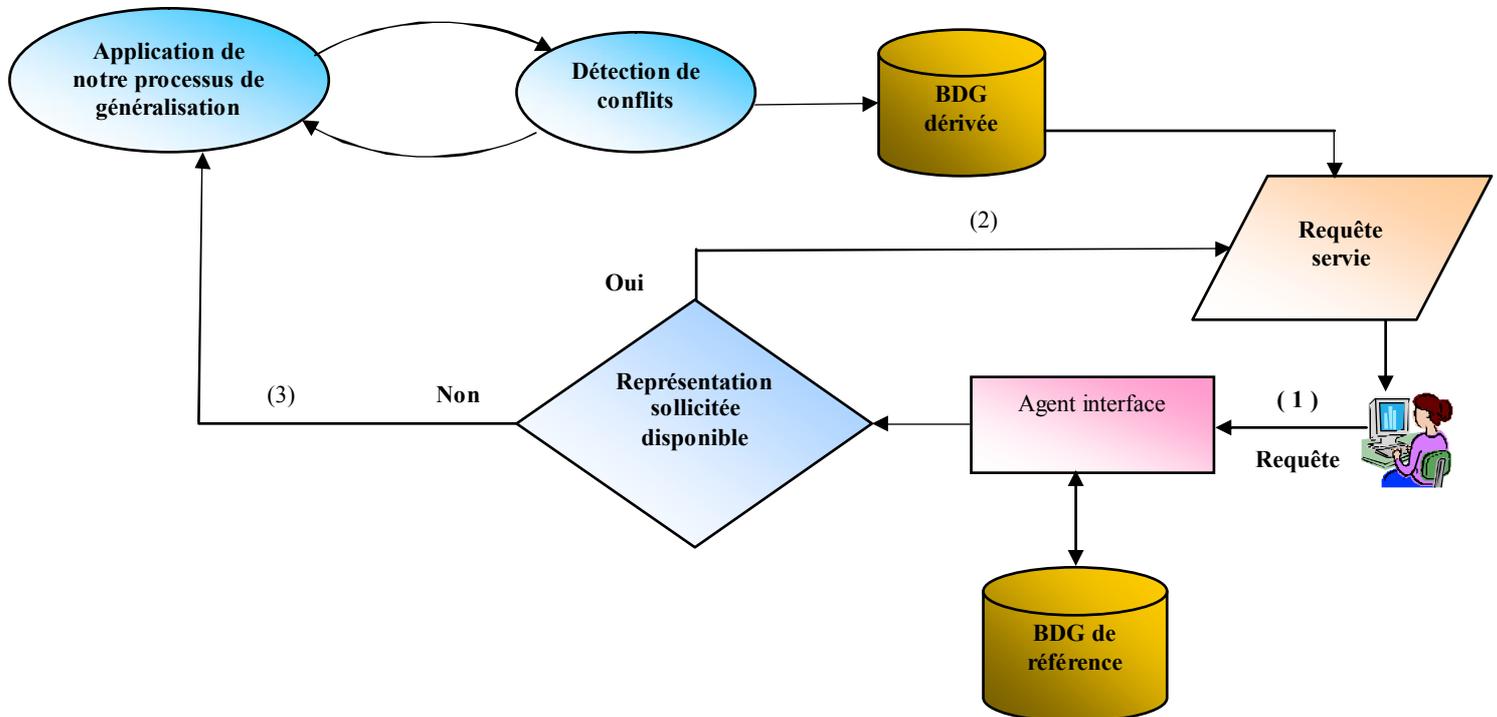


Figure IV-4. Le déroulement de notre processus de généralisation.

La requête transmise comporte des informations importantes pour l'agent interface telles que l'identification de la zone, le niveau de détail et le point de vue à travers le domaine d'application et les préférences de l'utilisateur... etc. (1). Le niveau de détail, les préférences de l'utilisateur et contexte d'application peuvent être arbitraires et non prédéfinis (Lejdel et al., 2009). L'agent interface interprète la requête et envoie une ou plusieurs demandes internes à la BDG de référence. Si l'information demandée est disponible, la requête est servie directement (2), si ce n'est pas le cas, un processus de généralisation a été activé (3). Les données dérivées stockées dans une base de données appelées base de données dérivée.

Pour bien comprendre le déroulement de notre processus de généralisation, on doit définir quels sont les algorithmes que doivent être pris en compte, quels paramètres qui résolvent tel ou tel contrainte. Le déroulement du processus automatique de généralisation s'effectue en plusieurs étapes qui sont ordonnées comme suit.

1. La définition de la zone de l'environnement qui sera généralisée.

2. Affectation des agents à chaque objet de la zone identifiée en 1,
3. Pour chaque agent faire :
4. L'exécution de l'algorithme génétique,
 - a. Le choix des conflits cartographiques à traiter,
 - b. le choix des algorithmes de généralisation à appliquer,
 - c. La mesure (évaluation de la solution), si la solution est optimale **allez à 5**, sinon **allez à 4**,
5. l'application de la solution.

Donc, pour réaliser un processus optimal, on doit définir dès le départ les objets à traiter. Ensuite, on doit définir le codage de la solution (chromosomes) de l'algorithme génétique. Puis, les algorithmes de généralisation appliqués sur un ou plusieurs objets en un instant donné et avec quels paramètres, seront identifiés en appliquant l'algorithme génétique.

5.1.1. Objets traités

Dans notre implémentation, on a utilisé des données géographiques appartenant à une zone urbaine, nous avons un ensemble de bâtiments représentés par des objets surfaciques (polygone) et un ensemble de rubans (ou segments) représente les routes.

5.1.2. Conflits traités

Pour réaliser notre application, nous avons choisi de traiter certains conflits car il est difficile de recenser tous les conflits qui peuvent apparaître entre les objets géographiques à un moment donné. A titre d'exemple : la superposition, empâtement, intersection qui n'existe pas dans la carte d'origine, la mutation des relations d'adjacence, disjonction, inclusion et de croisement.

5.1.3. Algorithmes implémentés

Nous avons utilisé les algorithmes qui permettent de généraliser les deux types d'objets de la carte : les polygones (bâtiment) et les rubans (routes). Les algorithmes implémentés sont :

- La simplification des contours pour simplifier les routes (rubans),
- L'algorithme de Douglas-Peucker pour généraliser les routes,
- La simplification des contours pour simplifier les bâtiments,
- Le déplacement de bâtiments,
- Le lissage de ligne.

Aussi, on a implémenté deux algorithmes génétiques différents, car chaque agent exécute son AG spécifique. Donc, on programme un agent route et un agent bâtiment.

5.1.4. Paramètres

Ce facteur est défini par l'utilisateur. Pour bien guider l'utilisateur, nous avons défini les intervalles auxquels la valeur introduite par l'utilisateur doit appartenir.

5.1.5. Mesures

Pour détecter la violation des contraintes, on a implémenté certaines mesures que nous avons considérées principales pour guider un processus de généralisation optimale. La violation d'une contrainte engendre des conflits qui nécessitent une méthode de détection spécifique, nous avons choisi d'utiliser des méthodes simples à implémenter.

- **la superposition** : la méthode utilisée pour détecter la superposition est le calcul de l'intersection. On a principalement deux types de superpositions, la superposition d'un bâtiment et une route et la superposition d'un bâtiment et un autre bâtiment. Pour coder le premier type, on a utilisé l'algorithme de calcul d'intersection entre un polyline et un polygone (Lejdel, 2005). Dans le deuxième cas, il suffit de vérifier si l'un des segments d'un bâtiment croise avec un des segments de l'autre.
- **La relation d'inclusion** : on dit que des points sont inclus dans un objet si tous ces points de l'objet inclus se trouvent dans l'objet englobant. Dans cette thèse, nous avons utilisé le code de détection d'inclusion introduit dans (Lejdel, 2005).

5.2. Etapes de l'algorithme génétique

On a implémenté les différentes étapes de l'algorithme génétique en forme de procédures, ce qui permet à l'optimiseur de l'agent de l'appeler autant de fois que nécessaire. Nous avons donc sept procédures principales : la procédure Initialisation, la procédure *Population_Initiale* chargée de générer une population initiale quelconque, la procédure *Sélection*, la procédure *Croisement*, la procédure *Mutation*, la procédure *Fitness* qui permet d'évaluer les solutions.. Les générateurs de nombres aléatoires que nous avons choisis d'utiliser sont des générateurs congruentiels mixtes car ils sont très utilisés.

L'utilisateur peut paramétrer l'algorithme génétique comme par exemple : le nombre d'individus de la population, le nombre de points de croisement et le taux de mutation. Aussi, il peut déterminer dès le début : la distance maximale de déplacement d'un bâtiment, la plage de paramétrage de l'algorithme de Douglas-Peucker (Douglas & Peucker, 1973). Il peut aussi décider quel type de conditions d'arrêt qui devra être appliquée par le module d'optimisation : les valeurs du fitness, du nombre d'itération et du temps d'exécution.

5.3. Opérations de la généralisation

Elle correspond aux processus permettant de dériver des données moins détaillées à partir des données sources plus détaillées. Nous avons implémenté les algorithmes suivants : l'algorithme de Douglas-Peucker, l'algorithme de déplacement, l'algorithme de réduction de taille de bâtiment et de lissage. Nous avons implémenté aussi un module pour gérer les mutations des relations topologiques entre les objets spatiaux.

Ces processus ont pour finalité l'adaptation de la résolution spatiale et sémantique des informations représentées à l'échelle demandée afin de produire une « bonne » carte répondant aux impératifs de précision, de contenu d'informations et de lisibilité pour l'utilisateur. Nous nous intéressons ici peu aux opérations qui esthétisent l'affichage final à l'écran (élargissement, affinement et l'esthétique) et nous ciblons les opérations de dérivation qui peuvent être appliquées sur les bases de données.

Chaque opération de dérivation correspond à un ensemble d'algorithmes ayant chacun des caractéristiques différentes :

- **Sélection** : qui correspond au tri des entités géographiques retenues en fonction de leurs natures et selon leur importance par rapport aux préférences et au contexte d'application demandés et le niveau de détail souhaité.
- **Simplification de rubans** (polyligne): on applique principalement deux algorithmes, l'un concerne la simplification des rubans et l'autre simplifie les polygones qui consiste à n'en retenir que les « points critiques » définissant sa forme essentielle : *L'algorithme de Douglas-Peucker*,
- **Déplacement** : L'opération de déplacement permet de résoudre des conflits cartographiques (proximité trop grande ou chevauchement) entre deux ou plusieurs objets. elle peut être appliquée à plusieurs objets distincts (point, ruban (polyligne) ou polygone) ou à des constituants (points ou segments) d'objets de type ruban (polyligne) ou polygone. L'objet sur lequel opère cette procédure est plutôt une paire d'entités spatiales qu'une entité indépendante.
- **Agrégation** : Les agrégations consistent à remplacer un ensemble d'objets proches par un seul objet (relation 1:n). La géométrie de l'objet créé dérive de la géométrie des objets agrégés. Les agrégations spatiales sont toujours associées à des agrégations sémantiques.
- **Classification** : Cette opération permet de regrouper des objets partageant des attributs géographiques similaires dans une nouvelle classe de plus haut niveau en utilisant le processus de généralisation/spécialisation de sémantique.

- **Symbolisation** : Cette opération permet de regrouper les graphiques des objets partageant des graphiques similaires dans une nouvelle classe de plus haut niveau en utilisant le processus de généralisation/spécialisation de graphique.

5.4. Résultats de Test

Nous allons présenter deux cas d'étude de notre modèle.

5.4.1. Cas d'étude 1 : Généralisation des données spatiales

Pendant le processus de la généralisation des données géographiques, l'optimiseur de chaque agent exécute son algorithme génétique pour résoudre les conflits cartographiques possibles, maintenir ou muter les relations topologiques entre les objets qu'ils manipulent et préserver leur lisibilité. A chaque cycle de traitement spatial, chaque agent surveille les changements dans son environnement immédiat et évalue son état actuel. Selon les modifications de l'environnement, elle planifie et exécute les actions pertinentes trouvées par son *optimiseur*. Les agents échangent des messages avec ses agents voisins pour le calcul de la fitness générale. Afin de réduire le temps de traitement, les *optimiseurs* de différents agents exécutent leurs algorithmes génétiques en parallèle autant que possible. Dans cette thèse, nous avons limité aux interactions des agents qui consistent à coopérer pour calculer la fitness générale. De plus, notre approche permet de réduire les délais d'attente des utilisateurs car elle génère et transfère les cartes demandées en même temps. Un exemple des premiers résultats est illustré par la **Figure IV-5**.

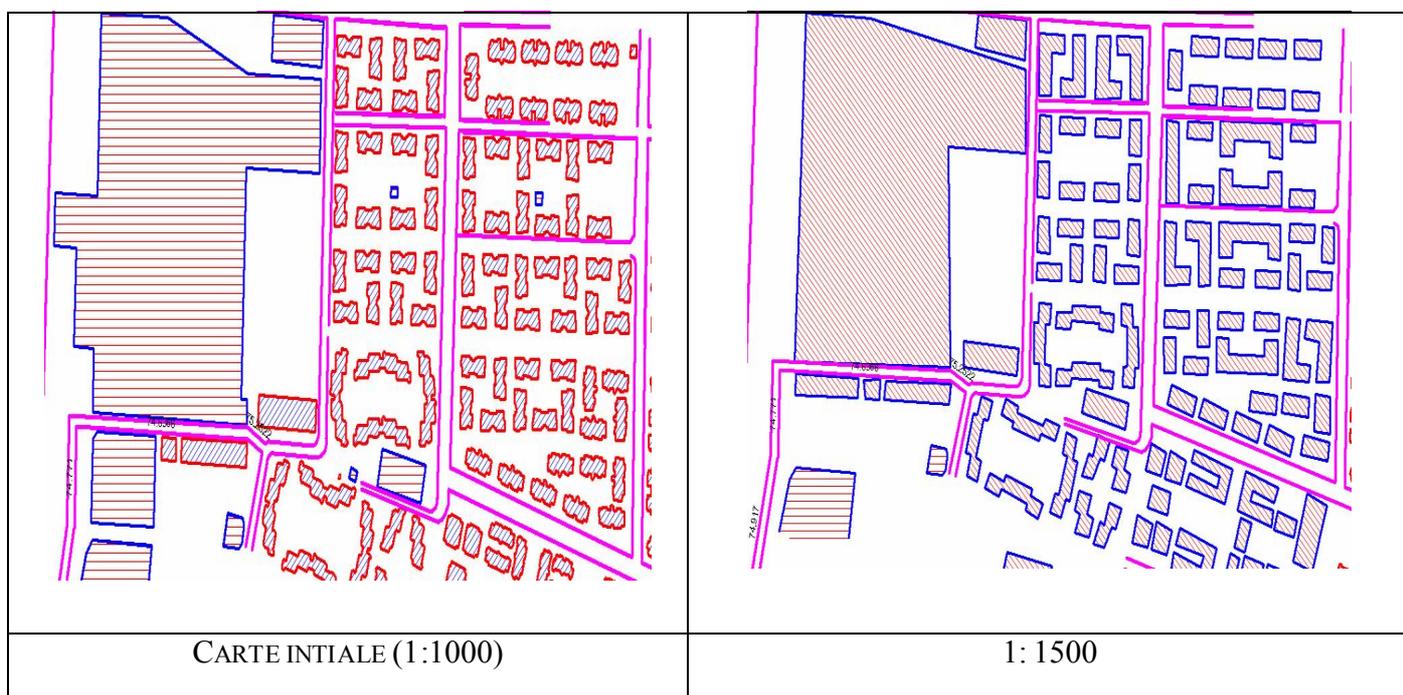


Figure IV-5. Résultat de généralisation.

5.4.2. Cas d'étude 2 : Mutation des relations topologiques

La réduction de l'échelle d'une carte implique que les relations topologiques entre les objets spatiaux (rubans et régions) devraient être mutées en d'autres relations (**Lejdel et al., 2015**). Pour appliquer ces mutation sur la carte, les conditions suivantes doivent être remplies: (1) les relations topologiques entre les objets spatiaux devraient être définis, (2) un modèle mathématique est nécessaire pour effectuer les mutations de ces relations topologiques en d'autre relations et (3) un ensemble des mesures métriques et des seuils doivent être définis pour guider ces mutations.

Pour analyser la variation des relations topologiques en utilisant notre modèle mathématique, un prototype est développé. Nous implémentons principalement trois fonctions principales :

- *Dist* (O^1, O^2): Calculer la distance entre deux objets spatiaux.
- *RelTOPO* (O^1, O^2): Définir les relations qui peuvent être existe entre deux objets spatiaux.
- *TransRel* (*Rel, Dist, seuil*): Appliquer les mutations possibles en utilisant les assertions mathématiques.

En utilisant des seuils prédéfinis, le prototype fonctionne comme les étapes suivantes.

- Calculer et stocker les relations topologiques entre les rubans et / ou régions, en utilisant les assertions mathématiques développées dans le chapitre III.
- Appliquer l'opérateur de la simplification du processus de généralisation.
- Utilisez les assertions développées dans le chapitre III pour muter les relations topologiques possibles entre les objets simplifiées (régions et rubans).

Le prototype peut détecter automatiquement les relations topologiques entre les objets et les muter en d'autres relations selon les assertions mathématiques décrites au chapitre III.

La **Figure IV-6** présente un exemple réel : la rivière R^1 est croisée avec une autre rivière R^2 et les bâtiments ($G1, G2$ et $G3$) sont disjoints. Les relations topologiques sont les suivantes.

- Disjoint*(G^1, G^2).
- Cross*(R^1, R^2).
- Disjoint*(G^1, G^3).
- Disjoint*(R^2, G^4).

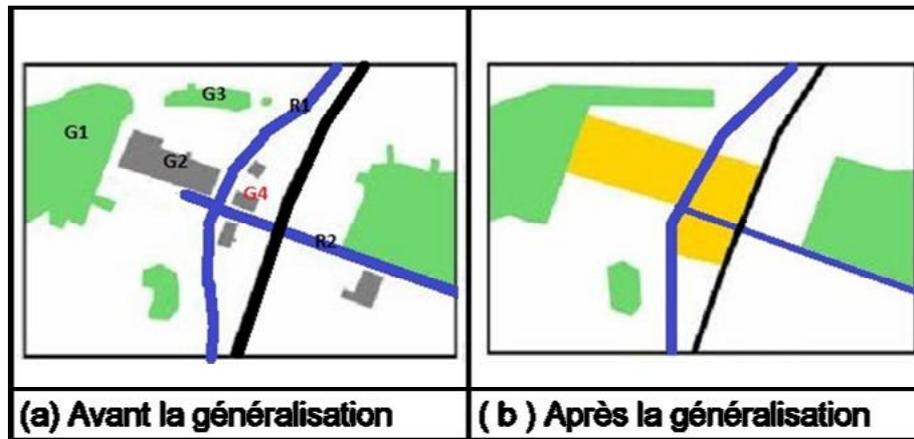


Figure IV-6. Exemple de généralisation.

Nous avons utilisé deux seuils, un pour l'invisibilité des objets et l'autre pour la réduction des objets (régions ou rubans) à point ou une ligne. Lorsqu'on réduit l'échelle, les rivières et les bâtiments sont généralisés et les relations topologiques sont mutées en d'autres relations. Le tableau suivant illustre ces mutations de relations topologiques:

Avant la généralisation	Après la généralisation
$Disjoint(G1, G2)$	$Meet(G1, G2)$
$Cross(R1, R2)$	$Merge(R1, R2)$
$Disjoint(G1, G3)$	$Meet(G1, G3)$
$Disjoint(R2, G4)$	$Meet(R2, G4)$

La **Figure IV-7** présente un autre exemple réel composé de plusieurs objets : rivières, côte, ville. Il y a aussi trois villes; Nice, Montpellier et Marseille. Le Rhône est liée à la mer par la relation topologique suivante : $Meet(Rivière, Côte)$.

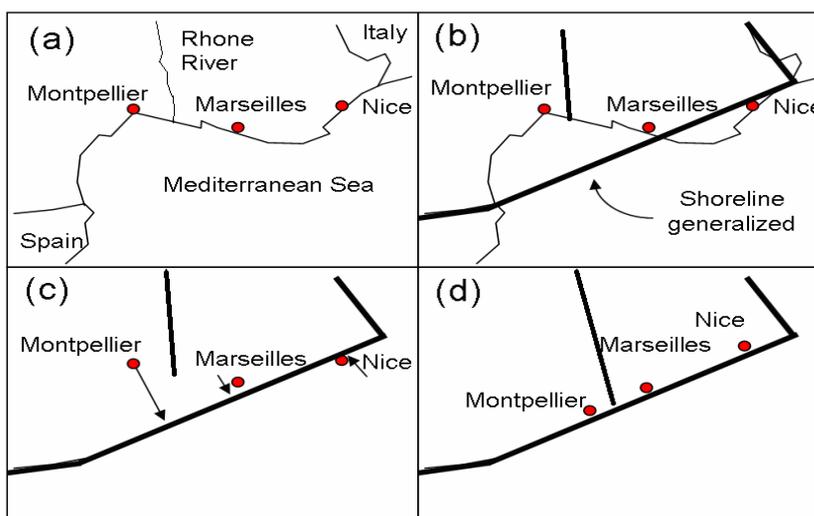


Figure IV-7. La généralisation des relations topologiques.

Lorsqu'on réduit l'échelle, la côte méditerranéenne est généralisée et la relation topologique spécifiée ci-après, est mutée en : *Merge(rivière, Côte)*.

6. Comparaison avec d'autres approches

Dans cette section, nous allons faire une comparaison entre notre approche hybride de la généralisation et les deux approches qui la composent. C'est l'approche de **(Wilson et al, 2003)** décrite en (section, II.3.2.2.2), et l'approche AGENT décrite en (section, II.3.3).

Ces deux approches sont utilisées pour automatiser le processus de généralisation et résoudre les conflits cartographiques entre les objets durant le processus de la généralisation mais elles ne permettent pas de fournir la solution optimale. Néanmoins, elle donne des solutions proches de la solution optimale. La qualité de la solution donnée par ces deux approches est affectée, soit par le temps limite de génération d'une solution ou la faisabilité de la solution finale. Enfin, nous avons comparé notre modèle topologique avec les modèles traditionnels qui sont présentés pour maintenir les consistances de relations topologiques entre les objets spatiaux.

6.1. Comparaison avec l'approche de (I.D.Wilson et al, 2003)

Les algorithmes génétiques sont des méthodes de recherche adaptative qui se base sur le processus génétique évolutif au sein des organismes biologiques. Ils commencent avec une population initiale et la modifient jusqu'à ce que l'une des solutions acceptables soient créées.

Wilson et al. (2003) propose une approche utilisant les algorithmes génétiques pour résoudre les conflits cartographiques. L'approche proposée (voir la **Figure IV-8**) se base sur le déplacement, elle utilise un index spatial afin d'identifier rapidement les objets spatiaux en conflits. Cette approche considère qu'un conflit se produit lorsque la distance minimale qui sépare deux objets est inférieure à un seuil prédéfini. Cette approche introduit le concept d'un vecteur de déplacement qui est utilisé pour résoudre un conflit. Ce vecteur détermine le taux de déplacement de l'objet.

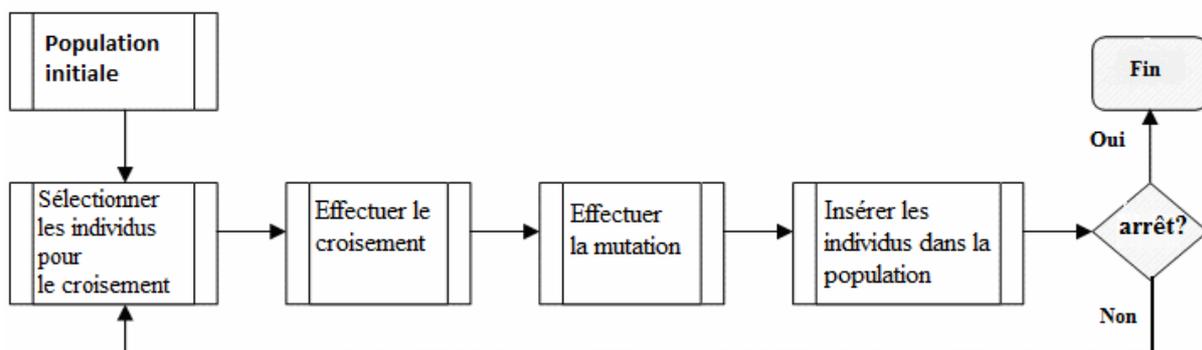


Figure IV-8. Procédure d'évolution de l'algorithme génétique (Wilson et al. 2003).

Par rapport à cette approche (**Wilson et al, 2003**), l'approche **Agent génétique** offre l'avantage d'effectuer des opérations très importantes, telles que l'élimination d'un objet ou l'agrégations d'un groupe d'objets. En revanche, la méthode de déplacement utilisée par (**Wilson et al, 2003**) fonctionne bien lorsqu' il y a un espace libre suffisant dans la carte pour les objets qui peuvent être déplacés. Mais, cette condition n'est pas toujours garantie, surtout si les cartes sont compactes et seront probablement affichées sur des écrans dont les tailles sont réduites.

6.2. Comparaison avec l'approche Agent (**Duchêne et al., 2012**)

Comme nous avons mentionné dans la section **II.3.3**, dans le projet AGENT ((**Ruas 1999**), (**Lamy et al. ,2001**), (**Duchêne & Ruas, 2001**), (**Duchêne & Ruas, 2007**) et (**Duchêne et al., 2012**)), les objets géographiques sont modélisés par des entités décisionnelles du processus de la généralisation. Ces entités sont des agents logiciels dont les objectifs sont de satisfaire autant que possible les contraintes cartographiques individuelles. Afin de résoudre un conflit spatial donné, un agent essaye de satisfaire ses propres contraintes liées à différents facteurs, tels que de la lisibilité et de la position. Ces contraintes sont classées en deux catégories: les contraintes qui déclenchent le processus de généralisation et de celles qui tentent d'empêcher la dégradation de données initiales.

Dans le cadre du projet AGENT, les contraintes sont transformées en objets qui agissent comme adversaire de leurs agents correspondants. Chaque contrainte indique à son agent le plan possible qui permet de satisfaire cette contrainte. L'agent synthétise les plans proposés obtenus à partir de toutes ses contraintes et choisit les meilleures actions à exécuter. Quand une ou plusieurs contraintes impliquent plusieurs agents, ces agents se tournent vers leur agent *meso*, qui coordonne la résolution du conflit. Dans ce cas, l'agent *meso* demande à ses agents composants d'effectuer les transformations nécessaires. Dans ce modèle, aucune interaction n'est possible entre les agents micro. La structure hiérarchique du prototype AGENT peut être appliquée à des cartes des zones urbaines où les routes structurent l'espace et où les bâtiments doivent être réorganisé en raison de leur densité.

Toutefois, dans les zones rurales, la configuration de l'espace ne peut pas être facilement représentée par ce modèle hiérarchique. En effet, la densité des objets est faible et, par conséquent, il est difficile à distinguer les groupes disjoints d'objets (**Duchêne, 2003**) (voir la **Figure IV-9**).



Figure IV-9. Espace urbain et espace rural (Duchêne, 2003).

Duchêne (2004) a proposé un modèle très intéressant qui permet de proposer une solution dans tous les cas mais elle ne garantit pas que cette solution soit optimale. Pour toutes ces raisons, nous avons combiné l'approche AGENT et la technique des algorithmes génétiques pour combler les inconvénients des deux approches et profiter de leurs avantages afin de résoudre les problèmes du processus de la généralisation.

Notre approche est très souple puisque les agents peuvent trouver l'action optimale dans tous les cas. La solution optimale est obtenue en un temps limité, ce qui fait l'approche proposée utile pour générer des cartes à un temps réduit parce que les optimiseurs des agents exécutent les algorithmes génétiques en parallèle, autant que possible, ce qui nous encourage à utiliser notre approche dans le contexte de web pour la généralisation à la volée. Les agents collaborent entre eux pour trouver la solution optimale. La forme de collaboration étudiée dans cette thèse est l'échange de message (conversation entre les agents). Les résultats sont très encourageants qui montrent la pertinence de l'approche proposée et nous encourage à poursuivre dans cette voie.

6.3. Comparaison avec les autres modèles topologiques

Dans notre cas d'étude 2, nous avons utilisé les rubans pour représenter les objets linéaires afin de vérifier l'exactitude des concepts du modèle mathématique des relations topologiques proposé en section III.6. Dans cette étude, nous avons présenté quelques exemples qui décrivent les mutations de relations topologiques lors de la réduction de l'échelle.

La cohérence topologique de la carte doit être maintenue lorsqu'on réduit l'échelle de la carte. Cependant, les méthodes traditionnelles qui sont présentées pour maintenir les cohérences de relations topologiques sont inefficaces car elles n'associent pas la simplification de forme avec la mutation des relations topologiques (Lejdel et al., 2015).

Donc, elles ne peuvent pas analyser les mutations des relations topologiques, ce qui les rend inefficaces et faibles pour préserver la cohérence topologique de la carte.

Le modèle topologique qui est présenté dans cette thèse consiste à muter les relations topologiques en d'autres relations durant le processus de la généralisation des données spatiales afin de maintenir la cohérence topologique de la carte générée. Par conséquent, il convient de maintenir la haute qualité de la carte générée lors de la réduction d'échelle. La collection des exemples que nous avons testés au cours du **cas d'étude 2** correspondant à différentes relations topologiques entre les objets géographiques (rubans et/ou régions) qui peuvent être disjoint, meet, cross, merge, contain ou cover. Ils ont été testés avec succès, ce qui indique la cohérence de nos concepts et la validité de nos assertions mathématiques pour muter les relations topologiques de n'importe quelle carte donnée.

Dans cette étude, nous nous sommes concentrés uniquement sur la mutation des relations topologiques lors de la réduction de l'échelle. Les aspects mathématiques de ce modèle topologique peuvent être intégrés sur n'importe quel algorithme de simplification implémenté par le SIG comme l'algorithme de Douglas-Peucker.

7. Conclusion

Dans ce chapitre, nous avons présenté le prototype dédié à l'optimisation du processus généralisation automatique des données spatiales. Ce prototype permet d'effectuer un processus de généralisation optimale et traiter les mutations des relations topologiques entre les données géographiques durant ce processus, ce qui garantit une bonne visualisation des données spatiales. Nous avons testé notre application par l'utilisation des données spatiales réelles pour concrétiser l'utilité d'utiliser les modèles proposés pour optimiser le processus de la généralisation des données géographiques.

Conclusion
générale
& Perspectives

1. Conclusion générale

Rappelons que l'objectif de notre thèse est de proposer une nouvelle méthode de généralisation des données géographiques en se basant sur le paradigme multi-agents.

Cette approche permet aux agents de coopérer afin de trouver la meilleure solution possible parmi les scénarios proposés. Donc, chaque agent génétique est équipé d'un optimiseur et d'un patrimoine génétique. L'optimiseur exécute un algorithme génétique pour déterminer l'action optimale à exécuter en fonction de son état actuel et selon la situation courante de l'environnement, à condition de satisfaire autant que possible les contraintes cartographiques. Nous avons proposé aussi d'utiliser un agent superviseur permettant de synchroniser l'exécution des algorithmes génétiques des différents agents et il peut aussi exécuter un algorithme génétique global dans le cas où les conflits ne sont pas résolus au niveau interne de l'agent.

Les expérimentations initiales nous montrent les avantages potentiels de l'utilisation d'une communauté d'agents, dans laquelle chaque agent est équipé d'un patrimoine génétique et un optimiseur pour optimiser le processus de la généralisation. Donc, chaque agent peut:

1. Définir les actions optimales de généralisation et il peut s'auto-généraliser.
2. Adapter sa généralisation selon l'état des autres agents géographiques du voisinage.
3. Collaborer avec les autres agents pour améliorer le résultat du processus de la généralisation cartographique et de résoudre les conflits cartographiques en un temps raisonnable. Pour améliorer la qualité de la carte et maintenir sa cohérence topologique, les agents doivent collaborer entre eux pour effectuer des mutations des relations topologiques en d'autres relations selon le modèle topologique proposé.

2. Contributions

Nos contributions sont rédigées en forme d'articles qui sont présentés à l'annexe 2, nous citons ici, un résumé de ses travaux :

1. En premier lieu, nous avons proposé une approche appelée **agent génétique** qui représente le fruit de notre premier travail (**Lejdel & Kazar, 2012**).
2. Dans un autre travail et en se basant principalement sur la notion de l'agent génétique présenté dans l'article 01 qui représente une nouvelle méthode hybride de l'approche SMA et les algorithmes génétiques, nous avons proposé une autre contribution qui consiste à adapter cette méthode à la généralisation à la volée, c'est-à-dire la généralisation des donnée sur Internet (**Lejdel & Kazar, 2012a**).

3. Dans une troisième contribution, nous avons proposé une approche très importante permettant à l'agent de définir la solution la plus optimale possible parmi les scénarios trouvés par l'AG car ce dernier permet seulement de définir la zone des solutions optimales, et non pas la solution exacte. Donc, pour améliorer les résultats de notre algorithme génétique, nous devons utiliser une méthode de recherche locale permettant de définir la valeur exacte de la solution optimale. En conséquence, pour affiner le résultat de l'AG, nous utilisons une méthode de recherche locale très répandue, c'est l'algorithme de *tabu*¹. Ainsi, chaque objet géographique sera modélisé par un agent équipé d'optimiseur. Il exécute un algorithme hybride de GA (Genetic Algorithm) et TS (Tabu search) pour trouver le plan optimal à exécuter selon l'état actuel de l'agent, à condition de respecter les contraintes internes et relationnelles. Les agents génétiques s'échangent des messages. Ces messages sont reçus par l'agent, ils peuvent l'aider à trouver le plan optimal en un temps d'exécution plus réduit. (Lejdel & Kazar, 2012b).

4. Dans une récente contribution, nous avons proposé un modèle topologique composant des assertions mathématiques pour accomplir le modèle hybride proposé pour optimiser le processus de la généralisation automatique. Ce modèle permet de muter les relations topologiques afin d'améliorer la qualité de la carte et maintenir sa cohérence topologique durant le processus de la généralisation. Nous avons utilisé la notion de *ruban* au lieu de *polyline* pour représenter les objets linéaires car dans la réalité, ces objets avaient une certaine surface. (Lejdel et al., 2015).

3. Perspectives

Les perspectives offertes par ce travail de recherche sont diverses et multiples. Elles concernent non seulement l'amélioration du fonctionnement du modèle que nous avons développé mais aussi l'intégration de ce processus dans les SIG est fortement nécessaire.

— Le prototype développé dans cette thèse permet d'appliquer la simplification, le déplacement, l'agrégation et le lissage. Au futur, nous allons essayer d'intégrer les autres algorithmes de généralisation.

— En général, le processus de généralisation ne concerne pas seulement la partie graphique de données, mais aussi la partie sémantique est très intéressante à être analysée. Ainsi, il faut ajouter d'autres méthodes qui peuvent réaliser cet aspect.

— La personnalisation de la carte en fonction des besoins de l'utilisateur. Dans un autre travail, nous allons modéliser les contraintes spatiales sous la forme d'une ontologie pour les différents types de carte. Par exemple, nous pouvons avoir l'ontologie des contraintes

¹ C'est une métaheuristique d'optimisation présentée dans (Glover & Laguna, 1997).

sémantiques des cartes touristiques. Ainsi, lors de l'utilisation du système, l'ontologie la plus appropriée peut être choisie en fonction du type de carte qui sera généré.

- Améliorer la performance de notre approche avec l'utilisation des autres méthodes d'optimisation qui permettent à l'agent de définir la solution optimale, comme l'hybridation entre les algorithmes génétiques et les méthodes des moindres carrés.
- Les assertions du modèle mathématique proposé dans ce travail utilisent les géométries des objets représentés dans le domaine de dimension 2D. Dans un travail futur, nous pouvons étendre ce modèle pour gérer les géométries de dimensions supérieures, comme le 3D.
- La carte ne contient pas seulement les objets simples tels que les régions et de rubans mais il y a aussi des ronds-points et des autoroutes qui sont des objets complexes. Un modèle mathématique complémentaire pour prendre en compte ce type d'objets sera un bon travail futur.

Bibliographie

Bibliographie

- Abbas I., 1993:** “Contrôle du tracé planimétrique d'une carte – Contrôle ponctuel et contrôle linéaire”, Bulletin d'information de l'IGN, n° 61: Bilan de la Recherche, pp.30-36.
- Allen J., 1991:** “Time and Time Again: the Many Ways to Represent Time”, International Journal of Intelligent Systems, 6 (4), 341–355.
- Allouche M. & Moulin B., 2001:** “Reconnaissance de patterns par réseaux de neurones: application à la généralisation cartographique”, Revue Internationale de Géomatique, 11(2), pp. 251-279.
- Alt H. & Gadau M., 1995:** “Computing the Fréchet Distance Between two Polygonal Curves”, International Journal of Computational Geometry and Applications, 5(1-2), pp.75-91.
- Bader M., Barrault M. et Weibel R., 2005:** “Building Displacement Over a Ductile Truss”, International Journal of Geographical Information Science, 19(8-9), pp. 915-936.
- Baeijs C. , Demazeau Y. et Alvares L., 1996:** “Sigma: Application of Multi-Agent Systems to Cartographic Generalization”, In W. Van de Vel de and J. W. Perram (Eds.), Agents Break Away, Proceedings of the European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Volume 1038 of LNAI, Springer, pp.163-176.
- Barillot X., 2002:** “Characterization of Complex Bends”, Proceedings of 10th Annual Conference on GIS Research in United Kingdom (GISRUK), Sheffield, pp.179-181.
- Beard K., 1991:** “Constraints on Rule Formation”, In B. Buttenfield and R. McMaster (Eds.), Map Generalization, Longman Pages, pp.121-135.
- Bédard Y., 1982 :** “Recommandations relatives à l'implantation d'un système d'information sur le territoire pour des fins de gestion municipale ”, Thèse de maîtrise, Université Laval, Québec.
- Brassel K. & Weibel R., 1988:** “A Review and Conceptual Framework of Automated Map Generalization”, IJGIS, 2(3), pp.229-244.
- Burghardt D. & Meier S., 1997:** “Cartographic Displacement Using the Snakes Concept”, in W. Foerstner & L. Pluemer, ed., Semantic Modelling for the Acquisition of Topographic Information from Images and Maps, Birkhaeuser Verlag, Basel, pp. 59-71.
- Burghardt D., 2002 :** “Smoothing of Line Objects with Optimisation Techniques”, Proceedings of Joint ISPRS /ICA work shop on Multi-Scale Representations of Spatial Data, Ottawa, Canada, URL: <http://www.ikg.uni-hannover.de/isprs/wg43-workshop.html>.

- Buttenfield B. & McMaster R., 1991:** “Map Generalization: Making Rules for Knowledge Representation”, Longman, pp.245.
- Buttenfield B., 1984 :** “Line Structure in Graphic and Geographic Space”, thèse de doctorat, département de géographie, université de Washington.
- Buttenfield B., 1991:** “A Rule for Describing Line Feature Geometry”, in Map Generalization: Making Rules for Knowledge Representation, Buttenfield and McMaster, eds, Longman Scientific and Technical, Harlow, Essex, pp.150-171.
- Cecconi C., 2003:** “Integration of Cartographic Generalization and Multi-scale Databases for Enhanced Web Mapping”, thèse de doctorat, Université de Zurich.
- Chaib-Draa B., Jarras I. et Moulin B., 2001:** “Systèmes multiagents: Principes généraux et applications”, In Briot, J. P. and Demazeau, Y., editeurs, Principes et architecture des systèmes multi agents. Hermès, Paris, France.
- Clementini E. & Di Felice P., 1997:** “A Global Framework for Qualitative Shape Description”, *GeoInformatica*, 1(1), pp.11-27.
- Clementini E., Di Felice, P. et Van Oosterom P., 1993:** “A Small Set of Formal Topological Relationships Suitable for End-User Interaction”, In Abeland, B. C. (ed), *Advances in Spatial Databases, Lecture Notes in Computer Science*. Springer, 692, pp.277-295.
- Denègre J. & Salgé F., 1996 :** “Les systèmes d'information géographique”, Que sais-je? , Presses universitaires de France.
- Douglas D. & Peucker T., 1973:** “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature”, *The Canadian Cartographer*, 10(2), pp.112–122.
- Duchêne C. & Regnauld N., 2002 :** “Le modèle AGENT. Généralisation et représentation multiple ”, dir. Anne Ruas, Hermès, chap.21, pp.369-385.
- Duchêne C. & Ruas A., 2001 :** “Généralisation de données géographiques: Présentation des résultats du projet AGENT ”, *Bulletin d'information de l'IGN*, n°72 : Bilan de la Recherche.
- Duchêne C., 2003:** “Automated Map Generalisation Using Communicating Agents”, *Proceedings of the 21st International Conference of Cartography (ACI/ICA)*, Durban, Afrique du Sud, pp.160-169.
- Duchêne C., 2004:** “Généralisation cartographique par agents communicants : le modèle CartACom”, thèse de doctorat, Université Pierre et Marie Curie Paris VI, laboratoire COGIT, URL : <ftp://ftp.ign.fr/ign/cogit/theses/>.

- Duchêne C., 2012:** “The CARTACOM Model: Transforming Cartographic Features into Communicating Agents for Cartographic Generalization”, *International Journal of Geographical Information Science*, 26(9), pp.1533-1562.
- Duchêne C., Bard S., Barillot X., Ruas A., Trevisan J. et Holzapfel F., 2003:** “Quantitative and qualitative description of building orientation”, *Proceeding of 5th ICA Workshop on progress in automated map generalisation*, Paris, URL: <http://www.geo.unizh.ch/ICA/docs/paris2003/papers03.html>.
- Eastman J., 1985:** “Cognitive Models and Cartographic Design Research”, *Cartographic Journal*, 22(2), pp. 95-101.
- EdiGéo, 1992 :** “Traitement de l'information géographique numérique”, AFNOR, Norme Z, pp.13-150.
- Egenhofer M. & Herring J., 1990:** “A Mathematical Framework for the Definition of Topological Relationships”, *Proceedings of the 4th International Symposium on Spatial Data Handling*, Zurich, Switzerland, pp.803-813.
- Egenhofer M., & Franzosa R., 1991:** “Point-Set Topological Spatial Relations”, *International Journal of Geographical Information Systems*, 5(2), pp.161-174.
- Eiben E., Smith E., 2003:** “Introduction to evolutionary computing”, Springer Verlag, Natural Computing Series.
- Ferber J., 1995 :** “Les systèmes multi-agents ”, Inter Editions, Paris, France.
- Ferber J., 1997 :** “Les systèmes multi-agents : Vers une intelligence collective”, Inter Editions, Paris, France.
- Foerster T. & Stoter J., 2007:** “Towards A Formal Classification of Generalization Operators”, *International Cartographic Conference, Moscow, Russia, CD-ROM, Theme 10: Cartographic generalization and multiple representation, Session Oral 8: Semantics and terminology in generalization.*
- Fritsch E., 1997:** “Représentations de la géométrie et des contraintes cartographiques pour la généralisation du linéaire routier”, *Thèse de doctorat, Université de Marne-la-Vallée, France.*
- Gaffuri J., 2008 :** “ Généralisation automatique pour la prise en compte de thèmes champ: le modèle GAEL”, *thèse de doctorat, Université de Paris-Est, France.*
- Galanda M., 2003:** “Automated Polygon Generalization in a Multi Agent System”, *Dissertation zur Erlangung der Doktorwürde, Mathematisch-naturwissenschaftliche Fakultät, Université de Zurich.*

- GeoTools, 2014:** “The open source java GIS toolkit”, URL: <http://docs.geotools.org/>, [Consulté le 10 décembre 2014].
- Glover F. & Laguna M., 1997:** “Tabu Search”, Kluwer Academic Publishers, Norwell, MA, USA.
- Goldberg D., 1989:** “Genetic Algorithms in Search, Optimisation and Machine Learning”, Addison-Wesley Publishing Company, MA, USA.
- Guenounou O., 2009:** “Méthodologie de conception de contrôleurs intelligents par l'approche génétique- application à un bioprocédé”, thèse de doctorat, université de Toulouse, France.
- Hangouët J., 1998:** “Approches et Méthodes pour l'Automatisation de la Généralisation Cartographique ; Application en bord de ville ”, Thèse de doctorat, Université de Marne-la Vallée, France.
- Harrie L. & Sarjakoski T., 2002:** “Simultaneous Graphic Generalisation of Vector Data Sets”, *GeoInformatica*, 6(3), pp. 233-261.
- Harrie L., 2001:** “An Optimisation Approach to Cartographic Generalisation”, Thèse de doctorat, département de technologie et société, Université de Lund, Suède.
- Hojholt P., 2000:** “Solving Space Conflicts in Map Generalisation: Using a Finite Element Method”, *Cartography and Geographic Information Sciences*, 27(1), pp. 65-74.
- Holland J., 1975:** “Adaptation in Natural and Artificial Systems”, Université de Michigan Press, Ann Harbor, Mich, USA.
- Jabeur N., 2006:** “A Multi-Agent System for On-the-fly Web Map Generation and Spatial Conflict Resolution”, thèse de doctorat, Université Laval, Canada, URL: <http://www.theses.ulaval.ca/2006/23356/23356.html>.
- JADE, 2014:** “Java Agent Development Framework”, URL: <http://jade.tilab.com/> [Consulté le 10 décembre 2014].
- Kilpelainen T., 2000:** “Maintenance of Multiple Representation Databases for Topographic Data”, *The Cartographic Journal*, 37(2), pp.101-107.
- Lamy S., Ruas A., Demazeau Y., Jackson M., Mackaness W. et R.Weibel, 1999:** “The Application of Agents in Automated Map Generalisation”, *Proceedings of 19th ICA meeting*, Ottawa, pp.160-169.
- Larousse, 2014 :** “Des dictionnaires et une encyclopédie gratuite et ouverte aux contributions des internautes”, Français-Français, URL: <http://www.larousse.fr/dictionnaires/francais>, [consulté le 10 décembre 2014].
- Laurini R., 2014:** “A Conceptual Framework for Geographic Knowledge Engineering”, *Journal of Visual Languages and Computing*, 25(1), pp.2-19.

- Lee D., 2003:** “Recent Generalisation Development and Road a Head”, Proceedings of 5th Workshop on Progress in Automated Map Generalisation, Paris, France.
- Lee Y. & Hsu J., 1990:** “2D C-string: A new Spatial Knowledge Representation for Image Database Systems”, Pattern Recognition, 23(10), pp.1077-1087.
- Lee Y. & Hsu J., 1992:** “Spatial Reasoning and Similarity Retrieval of Images Using 2D C-String Knowledge Representation”, Pattern Recognition, 25(3), pp.305-318.
- Lejdel B. & kazar O., 2012:** “Genetic agent for optimizing generalization process of spatial data”, International Journal of Digital Information and Wireless Communications (IJDIWC), 1(3), pp-729-737.
- Lejdel B., 2005 :** “Les Algorithmes de Manipulations des Données Intégrées dans un SIG”, Mémoire d’ingénieur, Université de Ouargla, Algérie.
- Lejdel B., Derbal K. et Alimazighi Z., 2009:** “Modelling and Managing Multiple Representations of Spatial Data by an Hybrid Approach (Application Them: Road Network)”, Proceeding of International Conference on Information and Communication Systems (ICICS), Amman, Jordan, pp.219-226.
- Lejdel B., Kazar O. et Laurini R., 2015:** “Mathematical Framework for Topological Relationships Between Ribbons and Regions”, Journal of Visual Languages and Computing, 26, pp.66-81.
- Lemarié C., 2003:** “Generalisation Process for Top100: Research in Generalisation Brought to Fruition”, Proceedings of 5th ICA Workshop on Progress in Automated Map Generalisation, Paris, URL: <http://www.geo.unizh.ch/ICA/docs/paris2003/papers03.html>.
- Li Z., 2007:** “Essential Operations and Algorithms for Geometric Transformations in Digital Map Generalization”, Proceedings of International Cartographic Conference, Moscow, Russia, CD-ROM, Theme 10: Cartographic Generalization and Multiple Representations, Session Oral 8: Semantics and Terminology in Generalization.
- Maozhen L. Sheng Z. et Jones B., 2002:** “Multi-agent Systems for Web-Based Map Information Retrieval”, In: M. J. Egenhofer and D.M. Mark (Eds.): GIScience, Springer-Verlag Berlin, pp.161-180.
- McMaster R. & Shea K., 1988:** “Cartographic Generalization in Digital Environment: Framework for Implementation in a Geographic Information System”, Proceedings of GIS/LIS’88 San Antonio, Texas, Vol. 1, pp.240-249.
- McMaster R. & Shea K., 1992:** “Generalization in Digital Cartography”, Washington, DC: Association of American Geographers Press.

- McMaster R., 1991:** “Conceptual Frameworks for Geographic Knowledge, Map Generalization”, Ed. B. Buttenfield & R. McMaster: Longman, pp.21-39.
- Michalewicz Z., 1992:** “Genetic Algorithms + Data Structures = Evolution Programs”, Springer-Verlag, International Journal of Geographical Information Science, 19(8-9), pp. 871-897.
- Müller J. & Wang Z., 1992:** “Area-Patch Generalisation: a Competitive Approach”, The Cartographic Journal, 29(2), pp.137-144.
- Newell A., 1990:** “Unified Theories of Cognition”, Press de l’Université de Harvard, MA Cambridge, pp. 549.
- Pantazis D. & Donnay J., 1996:** “La Conception de SIG. Méthode et formalisme”, Paris, Hermès, pp. 343.
- Paris J., Tautou L. et Pierreval H., 2001:** “Dealing with Design Options in the Optimization of Manufacturing Systems: an Evolutionary Approach”, Inter. Journal of Production Research, 39(6), pp. 1081-1094.
- Pitiot P., 2009 :** “ Amélioration des techniques d'optimisation combinatoire par retour d'expérience dans le cadre de la sélection de scénarios de Produit/Projet ”, Thèse de doctorat, Université de Toulouse, France.
- Plazanet C., 1996 :** “Enrichissement des bases de données géographiques : Analyse de la géométrie des objets linéaires pour la généralisation cartographique. Application aux routes”, Thèse de doctorat, Université de Marne-la-Vallée, France.
- Rainsford D. & Mackaness W., 2002:** “Template Matching in Support of Generalisation of Rural Buildings”, Proceedings of Advances in Spatial Data Handling, Ottawa, Canada, pp.137-151.
- Ramdani L., 2005 :** “Systèmes Multi Agents”, Université de Paris Sud XI, Paris, France.
- Ramer U., 1972:** “An Interactive Procedure for the Polygonal Approximation of Plane Curve”, Computer Graphics and Image, Processing 1, pp. 244-256.
- Regnauld N., 1998 :** “Généralisation du bâti: Structure spatiale de type graphe et représentation cartographique”, Thèse de doctorat, Université de Provence -Aix-Marseille 1.
- Regnauld N., 2004:** “Enhanced Hill Climbing Technique to Solve Cartographic Constraint Satisfaction Problems in the Context of the AGENT Project”, Computers Environment and Urban Systems, Numéro spécial sur le projet AGENT.
- Renard J., Gaffuri J., Duchêne C., et Touya G., 2011 :** “Automated Generalisation Results using the Agent-based Platform CartAGen”, Proceedings of the Twenty-fifth International Cartographic Conference (ICC’11), Paris, France.

- Renard J., Duchêne C., 2014:** “Urban Structure Generalization in Multi-Agent Process by Use of Reactional Agents”, *Transactions in GIS*, 18(2), pp. 201-218.
- Renard J., Gaffuri J. et Duchêne C., 2010:** “Capitalisation Problem in Research: Example of a New Platform for Generalisation, CartAen”, Proceedings of the Thirteenth ICA Workshop on Generalisation and Multiple Representations, Zurich, Switzerland.
- Rigaux P., 1994 :**“La représentation multiple dans les systèmes d’information géographique”, *Revue Internationale de Géomatique*, 4(2), pp.137-164.
- Ruas A. & Duchêne C., 2007:** “A Prototype Generalisation System Based on the Multi-Agent System Paradigm”. In W.A.Mackaness, A.Ruas et L.T. Sarjakoski, éditeurs: *Generalisation of Geographic Information: Cartographic Modelling and Applications*, Elsevier Ltd, Chapitre14, pp. 269-284.
- Ruas A. & Libourel T., 2002 :** “Introduction. Généralisation et représentation multiple”, dir. Anne Ruas, Hermès, pp.17-22.
- Ruas A., & Bianchin A., 2002 :**“Echelle et niveau de détail, Généralisation et représentation multiple”, dir. Anne Ruas, Hermès Lavoisier, chapitre 1, pp.25-44.
- Ruas A., 1998:** “A Method for Building Displacement in Automated Map Generalization”, *International Journal Geographic Information Systems (IJGIS)*, 12(8), pp.789-803.
- Ruas A., 1999 :** “Modèle de généralisation de données géographiques à base de contraintes et d’autonomie”, Thèse de doctorat, Université de Marne-la-Vallée, France.
- Ruas A., 2001:** “Automatic Generalisation Project: Learning Process from Interactive Generalisation”, OEEPE Official Publication n°39, pp. 98.
- Ruas A., 2002 :**“Les problématiques de l’automatisation de la généralisation”,Généralisation et représentation multiple, dir. Anne Ruas, Hermès, chapitre 4, pp.75-90.
- Ruas A., 2004 :**“le changement de niveau de détail dans la représentation de l’information géographique”, Mémoire d’habilitation à diriger des recherches, Université de Marne-la-vallée, France.
- Sabo M., 2007 :** “Généralisation et des patrons géométriques pour la création des objets auto-généralisants (SGO) afin d’améliorer la généralisation cartographique à la volée ”, Thèse de doctorat, Université laval, Quebec, Canada.
- Sarjakoski T. & Kilpelainen T., 1999:** “Holistic Cartographic Generalization by Least Squares Adjustment for Large Data Sets”, Proceedings 19th International Cartographic Conference, Ottawa, Canada, pp.1091–1098.
- Sester M., 2000:** “Generalization Based on Least Squares Adjustment”, *International Archives of Photogrammetry and Remote Sensing*, 33(B4), ISPRS, Amsterdam, Netherlands.

- Sester M., 2005:** “Optimization Approaches for Generalisation and Data Abstraction”, *International Journal of Geographical Information Science*, 19(8-9), pp.871-897.
- Shea K. & McMaster R., 1989:** “Cartographic generalization in a digital environment, When and how to generalize”, in *AutoCarto 9*, pp. 56-67.
- Syswerda G., 1989:** “Uniform Crossover in Genetic Algorithms”, J.D Shaeffer, éditeur, *Proceeding of 3th International Conference on Genetic Algorithms and Their Applications*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp.2-9.
- Taillandier P., 2008:** “Révision automatique des connaissances guidant l’exploration informée d’arbres d’états, Application au contexte de la généralisation de données géographiques”, Thèse de doctorat, Université de Paris-est, France.
- Touya G., 2011:** “Le Modèle CollaGen Collaboration de processus automatiques pour la généralisation cartographique de paysages hétérogènes”, Thèse de doctorat, Université de Paris-Est, Ecole doctorale MSTIC, COGICT, URL : <ftp://ftp.ign.fr/ign/coGIT/theses/>.
- Tsai V., 1993:** “Delaunay Triangulations in TIN Creation: an Overview and a Linear-Time Algorithm”, *International Journal of Geographic Information Systems*, 7(6), pp.501-524, URL : http://aci.ign.fr/BDpubli/paris2003/papers/lee_v0.pdf.
- Ware M., Thomas N. et Jones C., 2003:** “Resolving Graphic Conflicts in Scale Reduced Maps: Refining, the Simulated Annealing Technique”, *Proceeding of 5th ICA Workshop on Progress in Automated Map Generalisation*, Paris, France, URL : <http://www.geo.unizh.ch/ICA/docs/paris2003/papers03.html>.
- Weibel R., 1991:** “Specification for a Platform to Support Research in Map Generalization”, *Proceeding of the 15th Conference of the International Cartographic Association*, Bournemouth, UK, pp.12-20.
- Weiss G., 1999:** “Multi-Agent Systems, A Modern Approach to Distributed Artificial Intelligence”, the Mit Press, Cambridge, MA.
- Whitley F., 1994:** “A Genetic Algorithm Tutorial”, *Statistics and Computing*, 4, pp.65-85.
- Wilson I., Ware M. et Ware A., 2003:** “Reducing Graphic Conflict in Scale Reduced Maps Using a Genetic Algorithm”, *Proceeding of 5th ICA Workshop on Progress in Automated Map Generalisation*, Paris, France.
- Winter S. & Frank A., 2000:** “Topology in Raster and Vector Representation”, *GeoInformatica*, 4(1), pp.35-65.
- Wooldridge M. & Jennings N., 1995:** “Intelligent Agents: Theory and Practice”, the *Knowledge Engineering Review*, 10(2), pp.115-152.

Mes publications¹

Lejdel B., Derbal K. et Alimazighi Z., 2009: “Modelling and Managing Multiple Representations of Spatial Data by an Hybrid Approach (Application Them: Road Network)”, proceeding of International Conference on Information & Communication Systems (ICICS), Amman, Jordan, pp.219-226.

Lejdel B. & Kazar O., 2012: “Genetic Agent for Optimizing Generalization Process of Spatial Data”, **International Journal of Digital Information and Wireless Communications (IJDIWC), 1(3), pp-729-737.**

Lejdel B. & Kazar O., 2012a: “Genetic Agent Approach for Improving On-the-fly Web Map Generalization”, International Journal of Information Technology Convergence and Services (IJITCS), 2(3), pp.1-10.

Lejdel B. & Kazar O., 2012b: “Optimizing the Generalization Process of Spatial Data Using Multi Agent System Based on Improved Genetic Algorithm”, International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), 1(2), pp.155-160.

Lejdel B., Kazar O., et Laurini R., 2015: “Mathematical Framework for Topological Relationships Between Ribbons and Regions”, **Journal of Visual Languages and Computing, 26, pp.66-81.**

¹ Les publications en gras ont été choisies pour figurer en annexe 2.

Annexe 1

1. Introduction

Le processus de la généralisation est l'opération de simplification d'une représentation cartographique (carte géographique) suivant le niveau de détail demandé par l'utilisateur et le contexte d'utilisation de la carte. Nous avons commencé à partir de l'idée de proposer une approche permettant d'optimiser ce processus car jusqu'à l'heure actuelle, il n'existe pas d'une approche complète permettant de réaliser ce processus de façon flexible et efficace. Les travaux de cette thèse sont décrits par un ensemble de contributions qui est considéré comme une continuation de travaux de l'automatisation du processus de la généralisation de données géographique. Deux principaux facteurs sont pris en compte dans cette étude, la résolution des conflits cartographique et la réduction de temps de traitement.

Durant les années de préparation de notre thèse, nous avons finalisé quatre articles, publiés dans des revues internationales. Le premier article décrit l'approche **agent génétique** qui est utilisé pour optimiser le processus de la généralisation de données géographique. Le deuxième article présente l'utilisation de l'approche *agent génétique* pour améliorer les résultats de la généralisation à la volée. Le troisième article a été rédigé pour améliorer l'approche d'agent génétique par l'utilisation d'un algorithme de recherche local, c'est l'algorithme de *tabu*. Dans le quatrième article, nous avons proposé un modèle topologique basé sur un ensemble d'assertions mathématiques, pour guider les mutations des relations topologiques durant le processus de la généralisation des données géographiques.

2. Article : Genetic agent to optimize the automatic generalization process of spatial data

2.1. Objectif

Le processus de généralisation de données géographique consiste à dériver de données moins détaillées à partir de données géographiques détaillées, adaptées aux besoins des utilisateurs (par exemple, niveau de détail, préférence, contexte d'application). Ce processus est guidé par un ensemble de contraintes qui doivent être satisfaits. L'objectif de cet article consiste à optimiser ce processus, c'est-à-dire trouver dans un temps raisonnable, une solution qui met les contraintes cartographiques en un état de satisfaction maximale. L'utilisation d'un seul opérateur ne peut pas résoudre tous les conflits existants. Donc, dans tous les articles, nous allons utilisé plusieurs opérateurs de généralisation (déplacement, élimination, simplification, agrégation...etc.) afin de générer des données d'une bonne qualité.

2.2. Contribution

L'approche proposée est une approche hybride de système multi-agents qui profite au fond les avantages des algorithmes génériques qui aident principalement les agents à trouver

l'action optimale pour effectuer un processus de généralisation optimale. La population de l'algorithme génétique est un ensemble de solutions possibles à ce problème, dans notre cas, une solution est une suite d'algorithmes de généralisation avec des bonnes valeurs paramétriques pour résoudre les conflits apparents sur une carte à une échelle donnée.

Cette approche permet aux agents de coopérer afin de trouver l'action optimale parmi les scénarios proposés. Donc, chaque agent géographique a équipé d'optimiseur. Les optimiseurs de chaque agent exécutent leurs algorithmes génétiques pour trouver l'action optimale à exécuter en fonction de son état actuel, à condition de satisfaire le plus maximum possible les contraintes cartographiques.

3. Article : Genetic agent approach for improving on-the-fly web map generalization

3.1. Objectif

Avec l'arrivée du Web, une grande interactivité a été offerte entre les utilisateurs et la base de données géographiques. L'utilisateur a maintenant l'opportunité de naviguer parmi les données spatiales en modifiant par exemple les thèmes ou le niveau de détail visualisés. Cette nature dynamique et interactive requiert toutefois des temps de réponse quasi-instantanés. Par conséquent, pour pouvoir changer d'échelle des données spatiales sur le web, on doit effectuer en temps réel, un processus dit de généralisation à la volée (on-the-fly generalization). Cette généralisation doit permettre à l'utilisateur de générer des produits cartographiques à des échelles arbitraires (définies par l'utilisateur) et cela sans aucun délai d'attente notable. Donc, l'objectif de cet article consiste à utiliser l'approche agent génétique pour optimiser le processus de la généralisation sur Internet afin de délivrer des cartes à la volée c'est-à-dire, la généralisation à la volée.

3.2. Contribution

Ce travail est basé sur l'approche proposée dans l'article 01. Nous combinons les agents génétiques, le processus de généralisation cartographique et l'approche représentations multiples pour optimiser la dérivation des données spatiales moins détaillées à partir des données trop détaillées afin de résoudre des conflits cartographiques, ce qui améliore la qualité des données obtenues. Cette approche vise exclusivement à délivrer dans un temps raisonnable des données moins détaillées pour satisfaire les besoins de l'utilisateur. Les objets spatiaux sont modélisés en tant qu'agent. Un agent est un concept d'intelligence artificielle. Chaque agent est équipé d'un patrimoine génétique et d'un optimiseur qui permet d'exécuter un algorithme génétique. Ainsi, chaque agent génétique a une certaine connaissance de son état interne et des informations sensorielles concernant son contexte environnemental qui

permettent de lui aider à choisir l'action (ou la séquence d'action) optimale à exécuter afin d'atteindre ses objectifs .

4. Article : Optimizing the generalization process of spatial data using multi agent system based on improved genetic algorithm

4.1. Objectif

La grande force des algorithmes génétiques est leur capacité à trouver la zone de l'espace des solutions contenant l'optimum. En revanche, ils sont inefficaces lorsqu'il s'agit de trouver la valeur exacte de l'optimum dans cette zone. Or, c'est précisément ce que les algorithmes locaux d'optimisation réalisent le mieux.

Il est donc naturel de penser à améliorer l'algorithme génétique par un algorithme de recherche locale comme l'algorithme de *tabu* de façon à trouver la valeur exacte de l'optimum. On peut aisément le faire en appliquant à la fin de l'algorithme génétique un algorithme de recherche locale sur les meilleurs éléments trouvés. Donc, l'objectif de cet article est d'affiner la solution trouvée par l'algorithme génétique.

4.2. Contribution

Notre contribution consiste à doter l'agent, un optimiseur qui exécute un algorithme génétique. A chaque itération, l'optimiseur choisit le chromosome qui permet d'effectuer la meilleure généralisation selon le *fitness*. À la fin de l'exécution de l'AG, l'algorithme de Tabu va lancer pour affiner les résultats trouvés par l'AG. Dans la nouvelle configuration, des conflits peuvent être résolus. Cependant, d'autres peuvent apparaître ou même réapparaître. Pour éviter de retourner à une configuration antérieure, nous proposons de stocker les chromosomes qui sont classés, sans intérêts ou tabu pour éviter de les croiser dans les prochaines générations.

5. Article : Mathematical framework for topological relationships between ribbons and regions

5.1. Objectif

Pour améliorer la qualité de la carte et maintenir sa cohérence topologique, nous avons construit un modèle topologique en forme d'assertions mathématiques, basés sur des seuils et des mesures métriques. Dans ce modèle, nous avons utilisé un nouvel objet appelé *ruban* pour modéliser les objets linéaires car ces objets dans la réalité avaient une certaine largeur. L'objectif de ce modèle consiste à guider les mutations topologiques durant le processus de la généralisation des données géographiques.

5.2. Contribution

Notre contribution consiste à étudier les éventuelles mutations des relations topologiques durant le processus de la généralisation des données géographiques. Donc, si l'échelle de la carte est changée et si certains objets géographiques sont généralisées, non seulement les formes de ces objets vont changer (par exemple, une petite zone devient un point et puis disparaît lors de la réduction de l'échelle) mais aussi leurs relations topologiques peuvent varier selon à l'échelle. Jusqu'à l'heure actuelle, il n'existe pas de modèle mathématique qui modélise la mutation de ces relations topologiques. Un nouveau modèle topologique doit être décrit. Il est basé sur la notion des *rubans* qui sont définis comme des rectangles oblongs. Si on change l'échelle de la carte, les petits rubans se transforment à des lignes, puis ils disparaîtront. Supposons qu'une route passe à la côté d'une mer, à certaine échelle, elles semblent tous les deux disjoints tandis que dans des autres échelles plus petites, elles se touchent (meet). Donc, les relations topologiques mutent selon l'échelle. Dans cet article, les différentes composantes de ce modèle mathématique sont discutées. Pour chaque situation, des assertions mathématiques sont définies pour formuler les mutations des relations topologiques en d'autres relations durant le processus de la généralisation des données géographiques.

6. Conclusion

Ce chapitre a été consacré pour présenter les objectifs des quatre articles que nous avons publiés. Ces articles représentent le fruit d'un travail de plusieurs années de recherche. En premier lieu, nous avons défini une nouvelle approche pour modéliser le processus de la généralisation optimale des données géographiques. Cette approche se base sur une approche hybride de l'approche AGENT et l'approche de l'algorithme génétique, celle qui est utilisé pour combler les problèmes de choix de l'action optimale exécutée par l'agent. Ensuite, nous avons utilisé cette approche pour générer des cartes à la volée, en temps réel. Puis, nous avons utilisé une autre méthode de recherche locale pour affiner les résultats de l'algorithme génétique. En fin, nous avons proposé un modèle topologique basé sur des équations mathématiques, pour guider les mutations des relations topologiques lors de la réduction de l'échelle. L'objectif de ce modèle consiste à améliorer la qualité de la carte et maintenir sa consistance topologique en cours de l'exécution du processus de la généralisation des données géographiques.

Annexe 2

Article 01

**GENETIC AGENT TO OPTIMIZE THE AUTOMATIC GENERALIZATION
PROCESS OF SPATIAL DATA**

**International Journal of Digital Information and Wireless Communications (IJDIWC)
1(3): 729-737.**

**The Society of Digital Information and Wireless Communications, 2012
ISSN 2225-658X.**

Genetic agent to optimize the automatic generalization process of spatial data

Brahim lejdel¹ and Okba kazar²

¹ Institute of Science and Technology, University Center of el-Oued, Algeria.
lejdel82@yahoo.fr

² Computer science department, University of Biskra, Algeria.
kazarokba@yahoo.fr

ABSTRACT

The automatic generalization processes consist to derive less detailed spatial data from data too detailed. The objective of this process is to provide the user a spatial data adapted to those needs. Several approaches are proposed to automate this process. These approaches include the agent-based approach. But the central problem of this approach is the selection of the optimal action performed by the agent in a given moment. In this paper, we proposed an approach that can optimize this process by satisfying the cartographic constraints. Our approach consist to provide agents geographic, genetic patrimony, to enable them choosing the optimal action, from where the concept of genetic agent.

KEYWORDS

Generalization process, multi agent system, genetic algorithm, cartographic constraints, optimizer.

1 INTRODUCTION

Several works recommend to model the spatial objects by agents such as the works of [1], [2], [3]. These agents carried a set of actions to reach a goal. To do this, they must have some capacity of environmental perception, and communication with other agents. Each agent can be generalized itself

by applying the suitable algorithm with good parameters which satisfy certain constraints [4]. These constraints can divide on two types; internal and external. However, the main problem of this approach does not guarantee an optimal generalization process.

According to our idea, achieve optimal cartographic generalization process implied that the agent performs an optimal action which satisfy the different constraints. In this paper, we propose to provide the agent a set of geographic capacity; perception, communication and genetic patrimony that allow it to select the optimal action.

The paper is structured as follows. Part 2 briefly reminds the generalization process. Part 3 presents the proposed approach for optimising the generalization process. In part 4, we present some results. Finally, part 5 concludes and cites some perspectives of this work.

2 GENERALIZATION PROCESS

The generalization process is defined as a set of algorithms used to generalize an object or a set of objects [5]. This process is guided by a set of constraints that must be

satisfied [3]. The main problems of this process are:

- Response time is very long.
- The quality of generalized data is too degraded.

To optimize this process, we must find a solution that puts all the constraints in a state of maximum satisfaction. Therefore, we must find within a reasonable time, a compromise that satisfied these constraints in better for minimize the cartographic conflicts.

1.1 Generalization algorithms

To generalize a spatial object there are several algorithms, these algorithms can be classified by the objects type:

- Buildings: simplification algorithm of contour, displacement [6], etc.
- Roads: plaster algorithm [7], Douglas-Peucker algorithm, accordion algorithm [8], and *faill*_max,*faill*_min[9], schematization algorithm, etc.

Each algorithm is parameterized by a set of values. The choice of these values is very intrinsic to guide a good process of generalization.

1.2 Cartographic Constraints

The objects in the map database must respect a number of cartographic constraints. They can be divided into two categories:

- Constraints relate an isolated object (e.g for a building represented by a surface to be readable, its size must be large enough and its route should not be too detailed).
- Constraints involve multiple objects (e.g the symbol of a road should not overlap with that of a house).

If a cartographic constraint is not respected, this creates a cartographic conflict. These conflicts require a solution for be remedied. The cartographic conflict can be addressed by a combination of possible actions such as elimination, displacement, amalgamation and boundary simplification, combined with appropriate techniques for evaluating the quality of the result. However, the application of an individual operator may have an effect on map objects that was not previously in conflict, resulting in propagation of conflict within the map space. So, to avoid these problems, we propose in this paper to model spatial objects by agents with capabilities of perceptions and communication and also it has an optimizer which running a genetic algorithm.

3 PROPOSED APPROACH

The Architecture of action selection is a set of mechanisms that allows an artificial agent to choose at any time, the action to perform based on its perceptions, its memory, its goals and skills.

The proposed approach is a hybrid multi-agent system, which benefits in high the advantages of generic algorithms to permit principally the agents to select optimal action which optimize the generalization process in entirety. The population of a genetic algorithm is a set of possible solutions to the problem, in our case a solution is a suite of generalization algorithms with good parametric values to resolve all apparent conflicts on a scale map data.

3.1 Multi Agent Model

Agent is a concept of artificial intelligence [10]. An agent can be defined as an object which has improved communication ability, perception, and especially autonomy. An agent can be viewed as a living object decides itself to

apply special methods depending on its purpose and its environment. This distribution of decision-making allows us to give all staff a particular behaviour. This set, called Multi Agent System. But selecting the optimal action to execute is an intrinsic problem in these systems.

In this context, Multi Agents are groups of collaborative agents, which cooperate to achieve a common goal. Each “Agent”, as described above, is a member of a group whose purpose is to optimise its generalization process.

3.2 Genetic Algorithms

Genetic algorithms are developed by John Holland in 1975 [11] to imitate the phenomena adaptation of living beings.

They are optimisation techniques based on the concepts of natural selection and genetics. It searches an optimal solution among a large number of candidate solutions within a reasonable time (the process of evolution takes place in parallel). Each of these solutions contains a set of parameters that completely describe the solution. This set of parameters can then be considered the “genome” of the individual, with each parameter comprising of one or more “chromosomes”. [12]

They allow a population of solutions converging step by step toward optimal solutions [13]. To do this, they will use a selection mechanism of the population of individuals (potential solutions). The selected individuals will be crossed with each other (crossover), and some will be mutating by avoiding, whenever possible, local optima.

They are used primarily to treat both problems [13]:

- The search space is large or the problem has a lot of parameters to be optimized simultaneously.

- The problem can not be easily described by a precise mathematical model.

We are combining multi-agent systems with Genetic Algorithms, for optimizing the generalization process. Therefore, our proposal is based on the following three points:

- Collaborative Agent: to manage the local generalization optimisation process and to exchange relevant information with neighbouring Agents (agents which have a conflict with it).
- Genetic patrimony which transformed between agents, are used as inputs to the genetic algorithm.
- Genetic Algorithms: to research optimal solution for the generalization process.

4 System architecture

4.1 Geographic Agent

Each geographic agent has three main components:

- Patrimony genetic.
- An optimizer: using the genetic algorithm to find the optimal solution according to the internal constraints or relationships ones.
- Ability to communicate with neighbouring agents.

4.2 Structure of agent

In this context we consider two main types of agent; road agent and building agent. In the following, we represent the patrimony of these two types:

4.2.1 Road gene

An agent road can be characterized by its identifier, and a set of algorithms that can be applied on it to perform its

generalization. These algorithms can take parameters. The coding of gene is made by multiple forms; we use character code for coding the identifiers, the binary form to encode the application or not of such algorithms (0 to say that the algorithm is not applied, 1 for say that the algorithm is applied) and the real forms to encode the parameters of the algorithms. So, we can represent agent roads by the following schema:

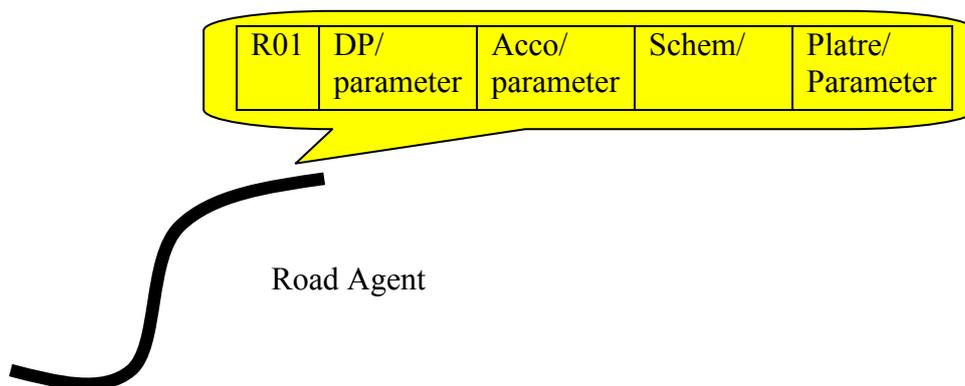


Figure 1. Genetic road Agent

4.2.2 Building gene

An agent Building can be characterized by its identifier and by their generalization algorithms: simplification algorithm and

algorithms of displacement. The same type of coding the road gene is used to encode this type of gene.

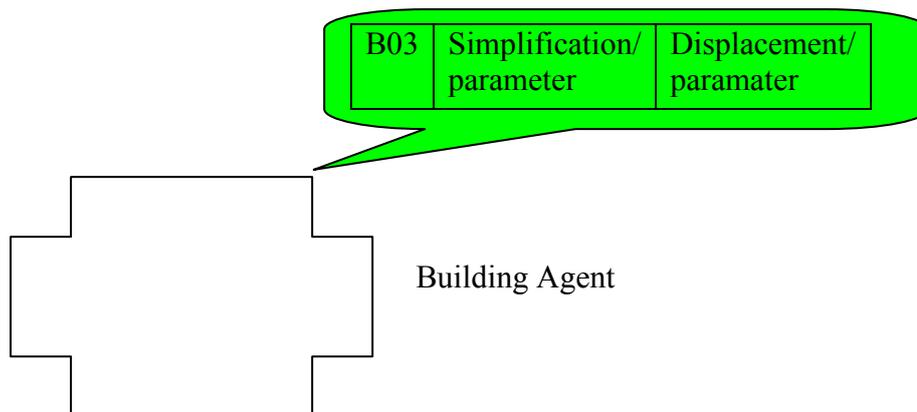


Figure 2. Genetic building agent

4.3 Architecture of genetic agent

Figure 3 presents the architecture of a genetic agent:

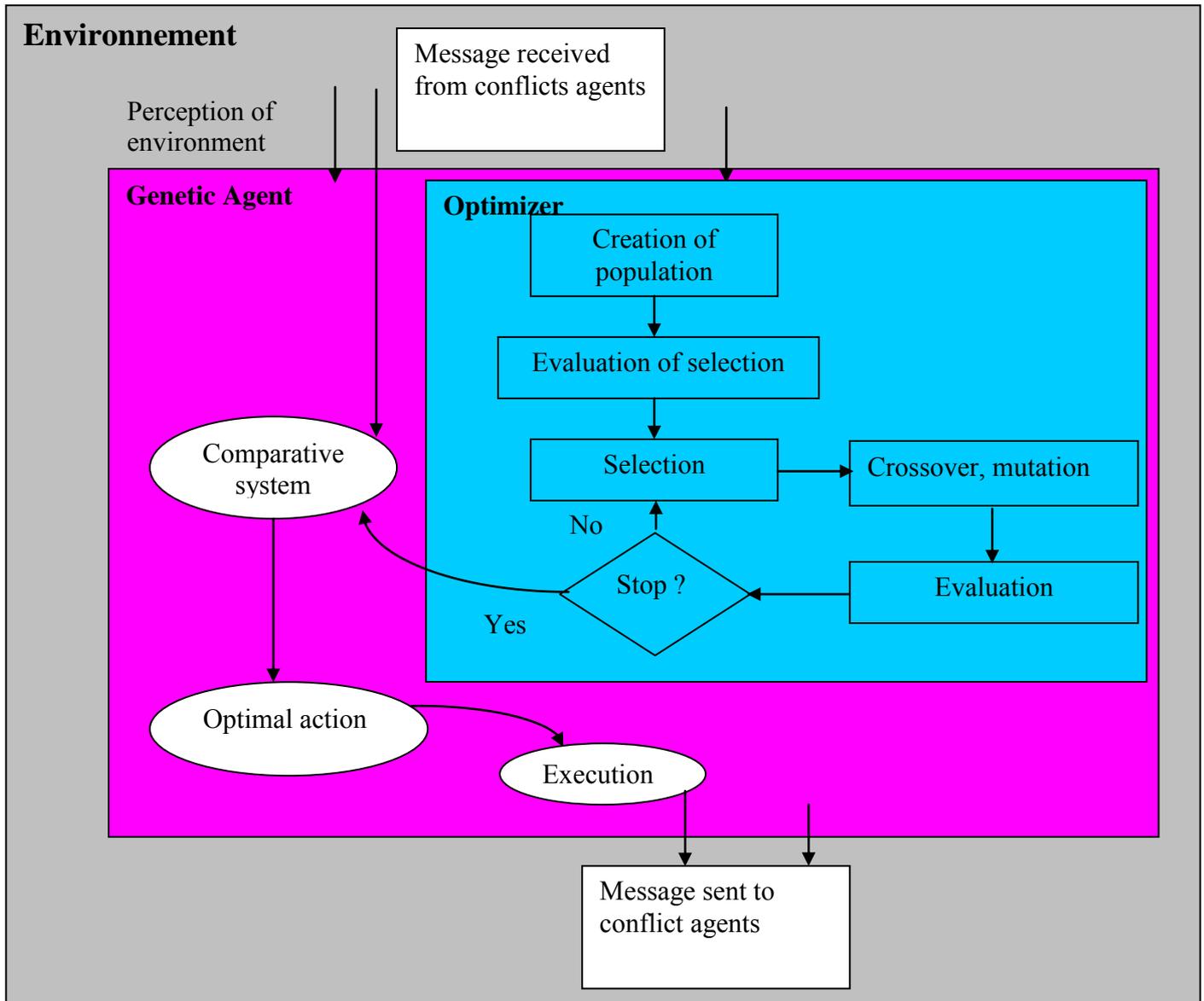


Figure 3. Architecture of Genetic agent

4.4 Optimizer

The optimizer executes the genetic algorithm. It follows the classical steps of a genetic algorithm are selection, crossover and mutation. The solution is refined gradually over the iterations until convergence to a solution that approaches the optimal solution, a certain degree of imperfection is acceptable. The solution here is a set of algorithms with good parameters which minimize conflicts:

a) Selection

In this work, we used steady-state selection [13]. The main idea is that much of the population can survive in the next generation. The genetic algorithm then works as follows: in every generation a few chromosomes are selected (among those with the best cost) to create children chromosome. Then, the worst chromosomes are removed and replaced

with new ones. The rest of the population survives in the next generation.

b) Crossover

It is an operator that is used by genetic algorithms. It is the transposition of the computer mechanism which permits, in

nature, producing chromosomes which partially inherit the characteristics of parents. Its fundamental role is to allow the recombination of information contained in the genetic population.

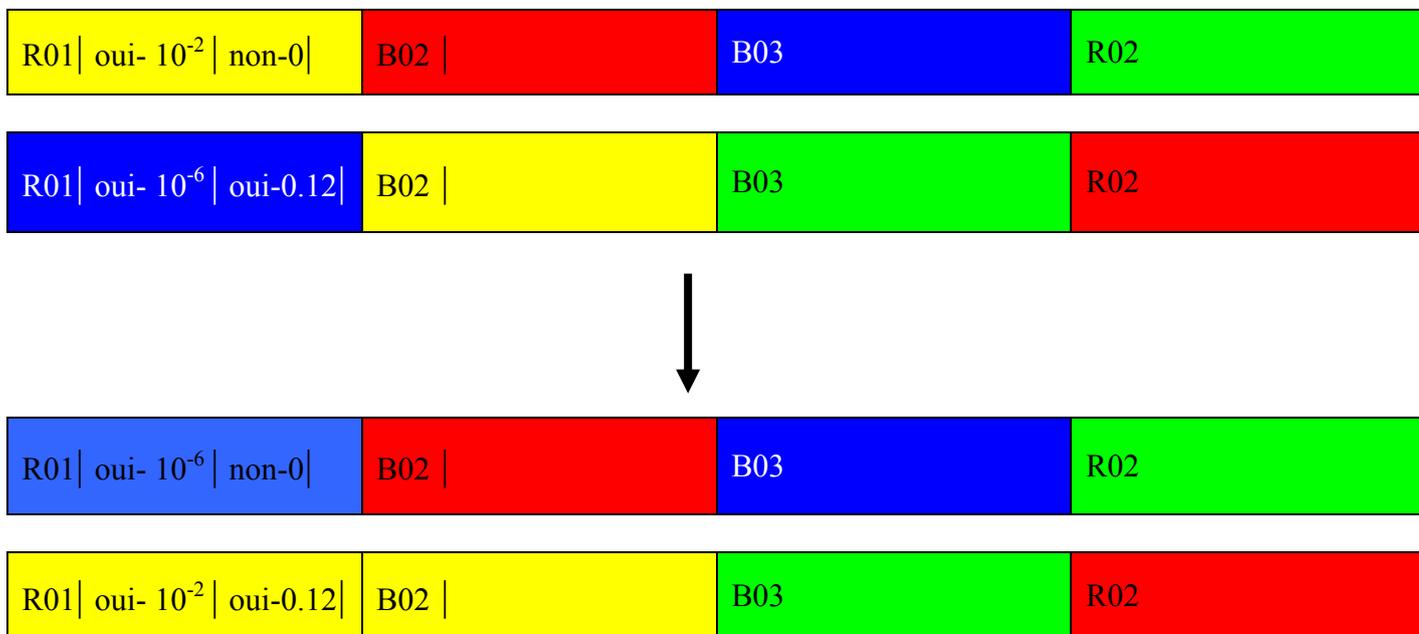


Figure 4. Crossover operator

c) Mutation

A change that occurs randomly on the chromosome [13], it aims to maintain some diversity in the population. This mutation occurs only on a sufficiently small population to avoid destroying the features that have been selected but large enough to bring new elements to an individual. The percentage of the mutation must be very low. It is applied to all parameters that represent a chromosome, if the mutation point has a Boolean values **yes**; it is replaced by **no** and vice versa. If it has a real value, the new value is randomly generated from a given interval.

d) Evaluation

The evaluation function is based on certain measures that assess the quality of spatial data, is restricted to a set of measures:

OS (object shape): calculates the loss of the shape of the object during processing. For buildings we compute the differentiation in the surface S and for roads, using measures of McMaster [14]:

$$OS = \sum \Delta S + \Delta (\text{McMaster}) \quad (1)$$

A minimum number is a good solution.

NC: number of objects in conflict, a minimum number is a good solution.

DP: sums the normalised, absolute, distance of each object has been displaced from its starting position.

$$DB = \sum_1^n \sqrt{(dx_i^2 + dy_i^2)} \quad (2)$$

Also, a minimum number is a good solution.

So, the general function is:

$$f = NC + DP + OS \quad (3)$$

The optimal solution is one that has the smallest value of general function. After each iteration (selection, crossover, mutation), agents in conflicts exchange the messages for calculate the value of fitness which permit to select the best chromosome, i.e. chromosomes with minimum fitness. After this step, it retains only the most relevant solutions.

e) Stop condition

The algorithm can stopped for three reasons, when it achieve certain fitness, when it achieve a set number of iterations or when it have passed a certain running time. In the two latter cases, we choose the solution that has the best fitness. The agent

receive the fitness values of other agents, the comparative system has an important role to choose the optimal action.

4 THE INITIAL RESULTS

This experimentation is intended to both clarify and illustrate the ideas and concepts of our approach. So, all spatial objects are represented by genetic agents. We have implemented the different algorithms of generalization such as the algorithms of reducing the size and displacement one. We implement also the various steps of the genetic algorithm; selection, crossover, mutation, and the fitness function for evaluate the solution which composed of a sequence of generalization algorithms with good parametric values to guide an optimal generalization process.

The algorithms and the agents have been implemented in Java using the platform JADE. The optimizer of each genetic agent executes a genetic algorithm which follows the classical steps: selection, crossover and mutation (see section 4.4). We use in this experimentation sets of data (buildings and roads) extracted from cadastral data. An example of initial results is illustrated in Figure 5.

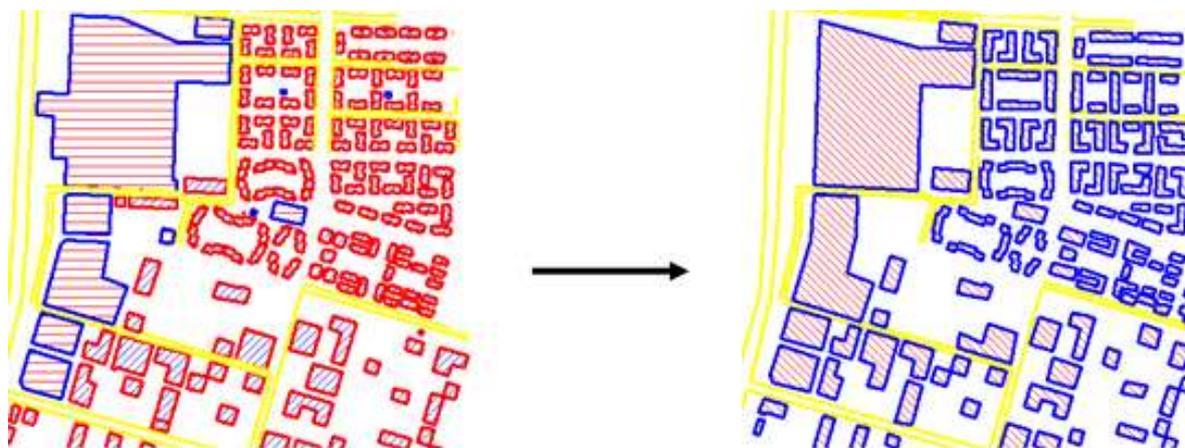


Figure 5 Generalization results

These results are very interesting and the method is very promising, it is highly recommended to follow this way to make the results better.

5 CONCLUSION

This approach will allow agents to cooperate so as to find the best action possible between proposed scenarios. So, each geographic agent has equipped of patrimony genetic, and then each ones executes a genetic algorithm to determine the optimal action executed according to its current state, in condition to satisfy the most possible cartographic constraints. Then, the geographic agents in conflict exchanged the messages. These messages are received by a comparison system that allows determining the optimal action which will execute by the geographic agent.

The initial experiments have shown us the potential advantages of using collaborative agent communities which each agent has equipped by patrimony genetic to optimise the generalization process. These advantages are:

- Each geographic agent can define the optimal actions of generalization and it can generate its self.
- Each agent can adapt its generalization with the other geographic agent.

Own work open very direction of research:

- In generally, the map not only contains polyline and polygon features. Thus, it will must expanded this method for generate another features as point features.
- Adapted this method in on-demand web mapping will be our future work.

6 REFERENCES

1. Cécile Duchêne, “The CartACom model: a generalisation model for taking relational constraints into account”, 7th ICA Workshop on Generalisation and Multiple representations, Leicester, UK (2004).
2. Ruas A., “Modèle de généralisation de données géographiques à base de contraintes et d’autonomie”, PhD thesis, université of “Marne-la-Vallée”, France (1999).
3. Ruas A., Dyevre A., Duchêne C., Taillandier P., “Methods for improving and updating the knowledge of a generalization system”, 16th AutoCarto conference, Vancouver, Canada (2006).
4. Regnauld N., “Constraints based Mechanism to achieve automatic generalization using agent modelling”, Proceedings of GIS Research UK 9th Annual Conference, pp.329–332, university of Glamorgan, UK (2001).
5. Bader M., “Energy Minimizing Methods for Feature Displacement in Map Generalization”, PhD thesis, Department of Geography, University of Zurich (2001).
6. Ruas A., “A Method for building displacement in automated map generalisation”, International Journal Geographic Information Systems (IJGIS) Vol.12 n°8, pp 789-803(1998).
7. Fritsch E., “représentation de la géométrie et des contraintes cartographiques pour la généralisation du linéaire routier”, PhD thesis, university of “marne-la-valée”, France (1997).
8. Corinne P., “Enrichissement des bases de données géographiques : analyse de la géométrie des objets linéaires pour la généralisation cartographique (application aux routes) ”, PhD thesis, University of “Marne-la-Vallée”, France (1996).
9. Mistière S., “GALBE: Adaptive generalization, the need for an adaptative

- process for automated generalisation, an example of road”, proceedings of 1st GIS’PlaNet conference, lisbonne, Potugal (1998).
10. Weiss G., “Multi-agent Systems, A modern Approach to Distributed Artificial intelligence”, The Mit Press, Cambridge, MA (1999).
 11. Holland J., “Adaptation in Natural and Artificial Systems”, University of Michigan Press, Ann Harbor(1975).
 12. Wilson I.D., Ware J.M. and Ware J.A., “Reducing Graphic Conflict in Scale Reduced Maps Using a Genetic Algorithm”, 5th ICA Workshop on Progress in Automated Map Generalisation, Paris, France (2003).
 13. DeJong K., Sarma J., “Generation Gaps Revisited”, “Foundations of Genetic Algorithms 2”, D. Whitley, Morgan-Kaufmann Publishers, San Mateo (1993).
 14. McMaster, R.B., “Mathematical measures for the evaluation of simplified lines on maps”, PhD thesis, University of Kansas., USA (1983).

Article 02

**MATHEMATICAL FRAMEWORK FOR TOPOLOGICAL RELATIONSHIPS BETWEEN
RIBBONS AND REGIONS**

Journal of Visual Languages and Computing (JVLC)
Volume 26, 66-81, Février 2015

ISSN: 1045-926X



Mathematical framework for topological relationships between ribbons and regions [☆]



Lejdel Brahim ^{a,b,*}, Kazar Okba ^a, Laurini Robert ^c

^a Linfi, University Mohamed Khider of Biskra, Biskra 7000, Algeria

^b Department of Computer Science, University Hamma Lakhder of EL-Oued, El-Oued 39000, Algeria

^c INSA de Lyon, Université de Lyon, Villeurbanne 69621, France

ARTICLE INFO

Article history:

Received 7 July 2014

Received in revised form

17 November 2014

Accepted 18 November 2014

Available online 28 November 2014

Keywords:

Topological relationships

Ribbon topology

Visual acuity

Ribbon–region relation

Downscaling

ABSTRACT

In a map, there are different relationships between spatial objects, such as topological, projective, distance, etc. Regarding topological relations, if the scale of the map is changed and if some spatial objects are generalized, not only the shapes of those objects will change (for instance, a small area becomes a point and then disappears as the scale diminishes), but also their topological relations can vary according to scale. In addition, a mathematical framework which models the variety of this category of relationships does not exist. In the first part of this paper, a new topological model is presented based on ribbons which are defined through a transformation of a longish rectangle; so, a narrow ribbon will mutate to a line and then will disappear. Suppose a road is running along a lake, at some scales, they both appear disjointed whereas at some smaller scales, they meet. So, the topological relations mutate according to scale. In this paper, the different components of this mathematical framework are discussed. For each situation, some assertions are defined which formulate the mutation of the topological relationships into other ones when downscaling.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

When somebody is saying “this road runs along the sea,” what are exactly the spatial or geographical relations which are concerned? Sometimes, either the road touches the sea or a small beach is located between the road and the sea, etc. From a mathematical point of view, mostly there is a disjointed relation between the road and the sea whereas for people the relation is different. In addition, when one is reading a map, according to scale, the topological relation can be different, disjoint or meet. So, topological relations can vary according to scale. Suppose a decision-maker wants to create a new motorway running

along a lake with the help of a computer. Taking this consideration into account, any reasoning system will generate difficulties because the spatial relations hold differently: any conceptual framework dealing with spatial relationships must be robust against scales.

Another problem comes from mathematical modeling of streets and rivers. Often, they are considered as linear objects even if they have some widths or areas. By considering a road as a line or as an area, topological relationships can be different. In order to solve this problem, the concept of ribbon will be developed. Depending on the scale, or more exactly on visual acuity and granularity of interest, a ribbon will be a longish rectangle (area), a line or it will disappear. In other words, ribbons can be seen as an extension of poly-lines. Moreover, in order not to be stuck to cartography, the concept of granularity of interest will be introduced.

This paper will be organized as follows. Firstly, definitions and a state of the art review for the generalization

[☆] This paper has been recommended for acceptance by Shi Kho Chang.

* Corresponding author at: El-gharbia, Hassani abdel Kerim, BP 39079, El-oued, Algeria. Tel.: +213 663831425.

E-mail address: lejdel82@yahoo.fr (L. Brahim).

process will be given (Section 2). Also, definitions and a state of the art review for topological relationships will be given (Section 3). Then, the framework of topological relationships for ribbons will be defined (Section 4). Finally, we present a conclusion and future work (Section 5).

2. Geographic object generalization

2.1. Definition

Many definitions have been given for the generalization process. The International Cartographic Association [12] has defined it as “the selection and simplified representation of detail appropriate to scale and/or the purpose of a map.” The geographic object generalization is a very complex process. In order to reduce its complexity, the overall process is often decomposed into individual sub-processes, called operator [13], such as simplification, displacement etc. Each operator defines a transformation that can be applied to a single spatial object, or to a group of spatial objects.

2.2. State of the art for generalization process

Historically, cartographic production was a matter of cartographers who generate maps for different users, generally for a specific domain (e.g., geological maps). This cartographic production always includes steps of possible control to ensure the quality of the generalized map. This crucial phase is usually performed by experienced cartographers. But today, with computer technologies that allow users not experts in the field of cartographic production to generate maps without the intervention of experts.

The first generalization process appeared in early 1990s [14]. It involved only a few geographical areas. The first algorithm for generalizing polylines was published in Ref. [15]. Then, several variants were published essentially to improve the results of the initial algorithm. However, this algorithm does not take into account many aspects, such as the topological relationships between objects.

Now, several methods and concepts have been proposed to model and implement the generalization process but a framework for their combination into a comprehensive generalization process is still missing [16].

Ruas and Plazanet [17] proposed a framework controlled by a set of constraints. The dynamic generalization model is based on avoiding constraint violations and on the local qualification of a set of objects represented by means of an object situation. A situation is described by the geographical objects involved, their relationships and the constraint violations. They concentrated only on constraints related to objects and not on the constraints between objects such as the topological constraints.

Many other works use the least squares adjustment theory to solve the generalization problems such as [18,19,20]; these works aim to globally reduce all spatial conflicts. The idea is to solve spatial conflicts by modeling different constraints using mathematical expressions. Moreover, Harrie [21] proposed to formulate the geometrical and topological constraints as linear

functions of the object coordinates. The least squares adjustment seems to be an interesting technique but these constraints are difficult to express by a linear equation.

In the same context and for reducing the spatial conflicts in the map, many interesting methods were proposed in [21,22]. In those approaches, a cost function (fitness) must be defined for validating the statements. However, it is questionable whether it is realistic to define such a function that integrates all the constraints of generalization such as the topological constraints.

Then several works model the spatial objects by agents such as the works of [23–25]. In the agent-based model, the spatial objects are modeled by the decisional entities in the generalization system. These entities are software agents the goal of which is to satisfy their cartographic constraints as much as possible. In Ruas [23], the constraints are subdivided into four types: metric, topological, structural and procedural constraints. The topological constraints ensure that any topological relationship between objects is maintained or modified consistently, for example, self-intersections of an object or any intersection between two objects must be avoided.

Also to improve the map generalization process, another approach was proposed in [26], which is based on a new concept called SGO (self-generalizing object). An SGO is able to generalize a cartographic object automatically using one or more geometrical patterns, simple generalization algorithms and spatial integrity constraints, but this approach does not define a pattern for topological constraints.

In the EuroSDR project, cartographic experts of four NMAs (National Mapping Agencies) were called to evaluate the results of the automation generalization process according to certain constraints [29]. The objective of this project is to illustrate the state-of-the-art of automated generalization in practice, exchange of knowledge between research community, NMAs and software vendor and to contribute to the development of constraint specification. Four test cases were selected and provided by the participating NMAs. The NMAs defined their map specifications for automated generalization in template which were developed by the EuroSDR team [29]. These map specifications were formalized as a set of cartographic constraints to be followed. They distinguished between two main categories of constraints: legibility constraints and preservation constraints. After the analysis of constraints composition, the EuroSDR project team derived a list of generic and specific cartographic constraints which must be respected in the generalization process.

Lejdel and Kazar [27] proposed an approach for optimizing the automatic generalization process by satisfying cartographic constraints. This approach consists of providing agents with geographical genetic properties to enable them to choose the optimal actions, thus giving the concept of genetic agent. Each geographical agent is equipped with an optimizer, and each one executes a genetic algorithm to determine the optimal action to be executed according to its current state in order to satisfy cartographic constraints as much as possible. The genetic algorithm follows the classical steps as selection, crossover and mutation. The solution is refined gradually over the

iterations until reaching convergence to a solution that approaches the optimal solution and a certain degree of imperfection is acceptable. The solution here is a set of algorithms with adapted parameters which minimize conflicts. The model of the topological constraints of this approach is not addressed in this paper.

3. Topological relations

3.1. Definition

Topology is defined as the mathematical study of the properties that are preserved through deformations of objects. Many works can be cited here such as the work of Thom and Zeeman [30]; they study the evolution of forms in nature. Thus, this theory can be applied in mapping and more exactly in the transformation of topology.

Topology is foremost a branch of mathematics, but some concepts are of importance in the GIS domain, such as topological relationships [1]. Topological relationships describe relationships between all objects in space, the points, lines and areas for all possible kinds of deformation. Several researchers have defined topological relationships in the context of geographical information [2–4].

3.2. State of the art for topological relations

From a historical point of view, different topological models were proposed. Firstly, Allen [5] proposed a model organizing pieces of a linear model which can also be used for temporal reasoning, Max Egenhofer [6] with his colleagues proposed the first topological model for two-dimensional objects, and then Lee and Hsu [7,8] defined the relations between rectangles. Let us examine them rapidly.

3.2.1. Allen model

The objective of the Allen model is to represent the relations between two segments [5], as illustrated in Fig. 1.

3.2.2. Egenhofer region topological relationship

To define a model of topological relationships, Egenhofer and Herring [6] proposed a spatial data model based on topological algebra. The algebra topological model is based on geometrical primitives called cells that are defined for different spatial dimensions 0-D, 1-D, and 2-D. A variety of topological properties between two cells can be expressed in terms of the 9-intersection model [10]. The 9-intersection model between two cells A and B is based on the combination of six topological primitives that are interiors, boundaries and exteriors of A (A^+ , ∂A , A^-) and B (B^+ , ∂B , B^-).

These six topological primitives can be combined to form nine possible combinations representing the topological relationships between these two cells. These

9-intersections are represented as one 3×3 matrix [28]

$$R(A, B) = \begin{pmatrix} A^+ \cap B^+ & A^+ \cap \partial B & A^+ \cap B^- \\ \partial A \cap B^+ & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^+ & A^- \cap \partial B & A^- \cap B^- \end{pmatrix}$$

The value represented in the matrix will be only a symbol indicating whether the intersection is null (ϕ) or not null ($-\phi$). When the value of the intersection is not important, it is represented by (-). Based on these nine possible intersections, one can construct 512 theoretical relationships. However, they are not all available. The detection of possible relations is made using negative conditions which prevent the association between pairs of primitives (non-existing topological relations). Therefore, the result implies eight possible topological relations between two regions in \mathfrak{R}^2 . These eight relations are explicitly represented in Fig. 2 (note that sometimes the MEET relation is called TOUCHES in some papers).

3.2.3. Egenhofer line topological relationship

Egenhofer and Herring define 33 relations can be realized between two simple lines [6]. Fig. 3 shows the different types of intersections and their mathematical interpretations.

3.2.4. Lee and Hsu model

In this model, Lee and Hsu [7,8] study the rectangle relations; they proposed a table representing all spatial relations between two rectangles. They found a total of 169 types (see Fig. 4) in which they number: 48 disjoint, 40 joints, 50 partial overlaps, 16 contains and 16 belongs (=inside). Due to the semantics of ribbons, a lot of them can be discarded. We shall not examine all of them, but the more interesting ribbon relations, namely, disjointing, meeting, merging and crossing.

All the models presented above define topological relationships between objects but they do not treat the transformation of topological relationships between the spatial objects when downscaling. In this paper, we will discuss the transformation of these categories of relationships during the generalization process.

4. Mathematical framework for topological relations

As previously told, it is common to state that there are 0D (points), 1D (lines), 2D (areas) and 3D (solids) geometrical objects for modeling geographical objects. But the reality is much more complex. It is also common to state that streets and rivers can be modeled as lines or polylines, but in reality (ground) they are areas with specific properties so that they can be reduced to lines when needed. In order to take these characteristics into account, the concept of ribbon will be detailed. But before defining a mathematical framework of topological relationships for ribbons and regions, let us present some mathematical background.

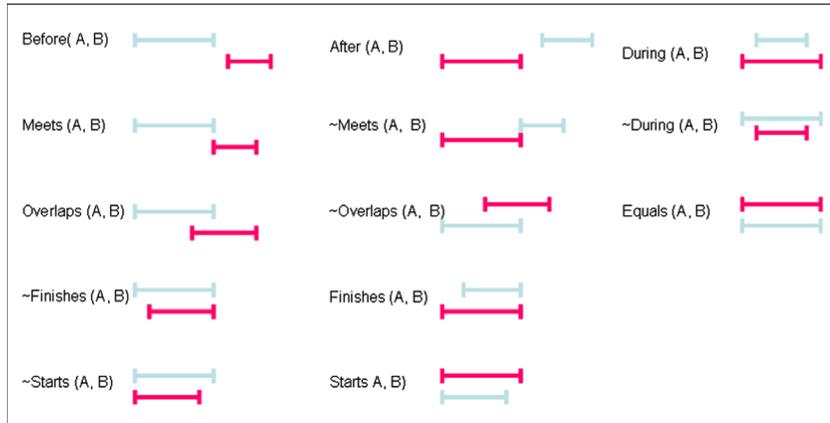


Fig. 1. The Allen topological relations.

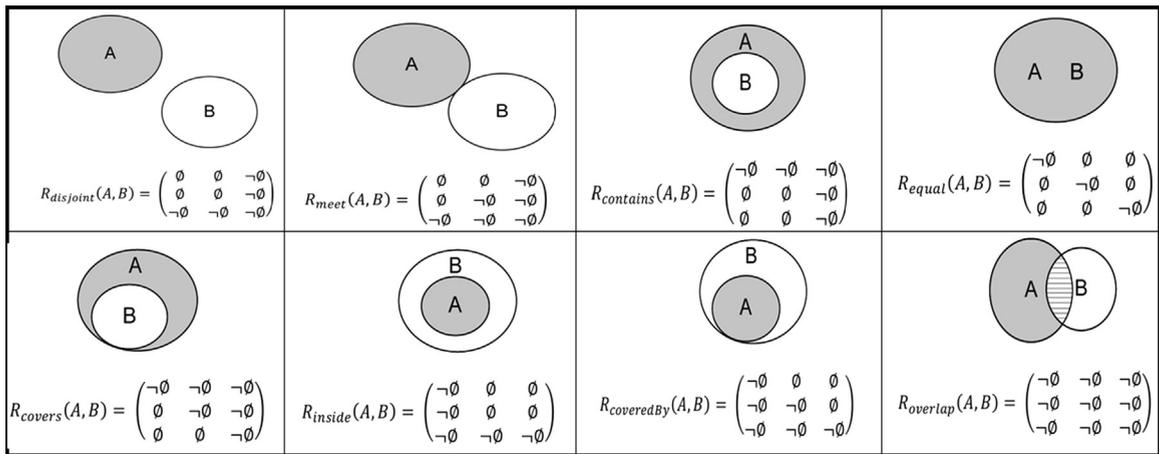


Fig. 2. The eight topological relations between two regions A and B.

4.1. Mathematical background

4.1.1. Definition of ribbon

We claim that ribbons may elegantly model rivers and roads (so-called linear objects): a ribbon can be loosely defined as a line or polyline with a width. Mathematically speaking, a ribbon is defined as a longish rectangle [9]. The ribbon has a skeleton which is its axis. See Fig. 5 for an example.

It is noted that the ribbons have width w , length l and longishness ratio r_l ($r_l=l/w$). The longishness ratio is supposed to be much greater than a positive value r_L so that $r_l > r_L$; a possible minimum value of this threshold r_L is 10.

Let us note Skeleton (R) is the axis of a ribbon. Remember that the ribbon can contain holes which can be useful for modeling islands in rivers.

In the sequel of this paper, to simplify the presentation, a ribbon will be represented by a longish rectangle. For instance, a motorway (see Fig. 6) can be described by several ribbons corresponding to several driving lanes, emergency lanes and one median.

4.1.2. Region

This feature may represent real objects, such as a building. We can define a region as a loose polygonal type. See Fig. 7 for an example, each region has an interior, boundary and exterior. Using these primitives, nine topological relationships can be formed by two regions called the 9-intersection model [6].

4.1.3. Basic theory

In this section, we give certain definitions of the intersection which will be used to formulate the mathematical description for each topological relationship between two ribbons or between a ribbon and a region:

Def 01 # the intersection:

If R^1 and R^2 are two ribbons, to define the intersection of $R^1 \cap R^2$, we have three cases:

- Point P (x,y).
- Line L ($y=ax+b$).
- Area A.

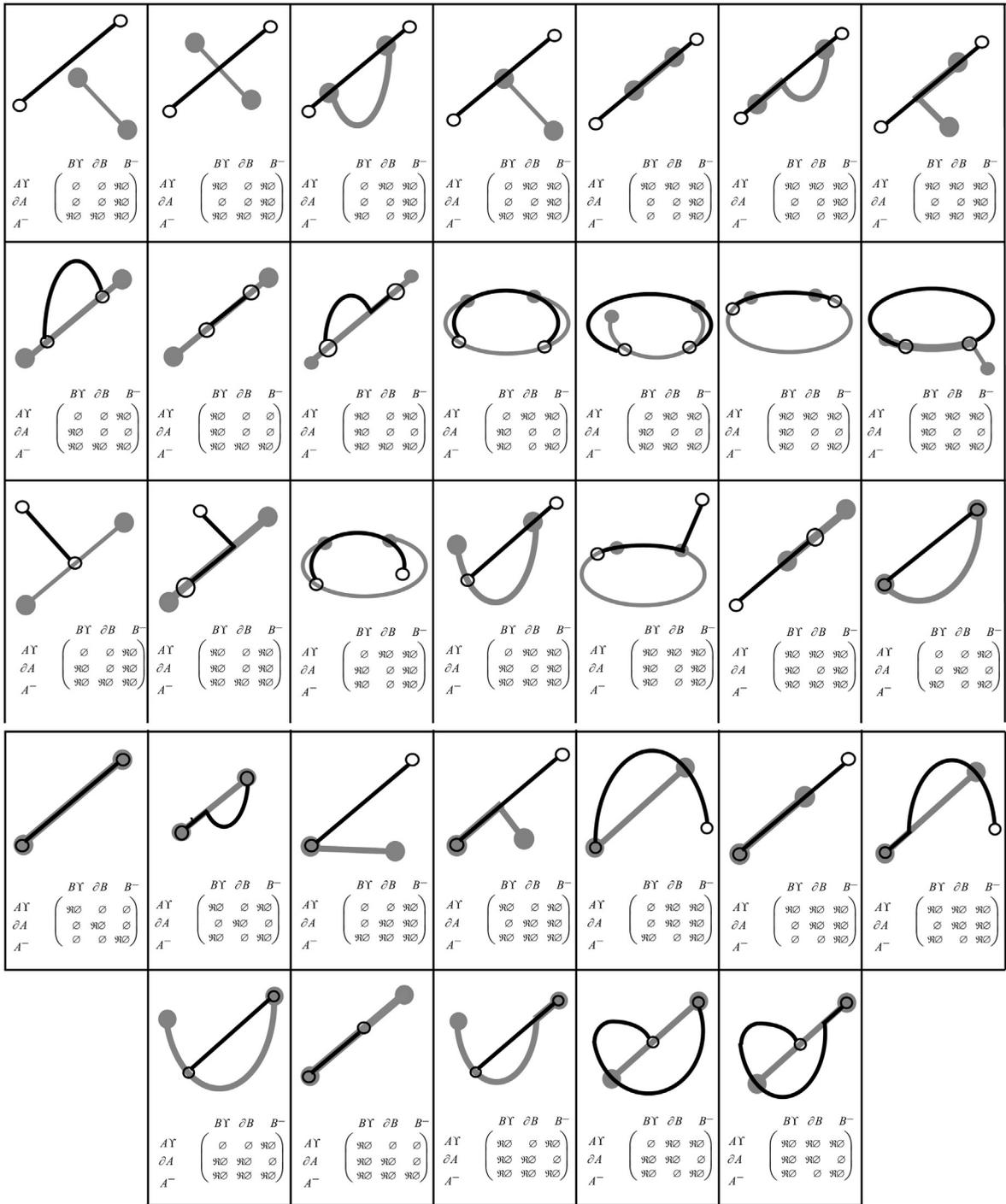


Fig. 3. Topological relations between two lines.

In other terms, this is an exclusive “belong to” defined as follows: $(P \oplus L \oplus A)$. Therefore, we can formulate it as $R^1 \cap R^2 = \{x/x \in (P \oplus L \oplus A)\}$

Def 02 # complement of the intersection:

Let there be two ribbons R^1 and R^2 . The **relative complement** of intersection $R^1 \cap R^2$ can be a set of points

belonging to R^1 or R^2 , but not to $R^1 \cap R^2$. Therefore, we can formally define the relative complement of intersection between ribbons as

$$CMP(R^1 \cap R^2) = \{x/x \in (R^1 \oplus R^2) \text{ et } x \notin (R^1 \cap R^2)\}$$

DISJOINT (48)			JOIN (40)			PART_OVLP (50)			CON TAIN (16)	BEL ^o ONG (16)

Fig. 4. The 169 types of spatial relations.

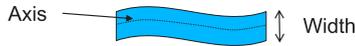


Fig. 5. Definition of ribbon.

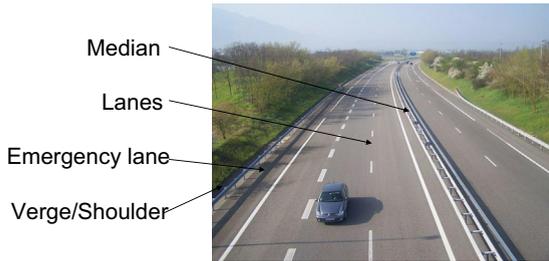


Fig. 6. Ribbon model applied to a motorway.

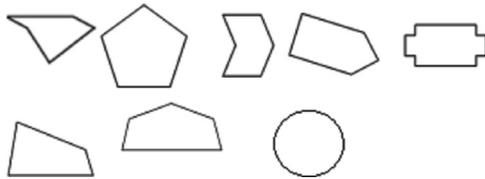


Fig. 7. Example of regions.

4.1.4. Downscaling process

In most cases the required representation scale does not, however, exist in geographical database, thus a derivation from the existing representation of the required representation is necessary. This process of adaptation and reduction of the representation content to a requested scale is called a downscaling process.

During the downscaling, the topological relationships can vary as the changes of objects geometry. We treat in this context, two principal objects: ribbons and regions. We can use the process as it is described in Ref. [9]

Step 0: original geographical features only modeled as areas and/or ribbons,

Step 1: as scale diminishes, small areas and ribbons will be generalized and possibly can coalesce,

Step 2: as scale continues to diminish, areas mutate to points and ribbons into lines (its skeleton), and

Step 3: as scale continues to diminish, points and lines can disappear.

4.1.5. Visual acuity applied to geographical objects

In the GIS, “Cartographic representation is linked to visual acuity” [9]. Thresholds must be defined. In classical cartography, the limit ranges from 1 mm to 0.1 mm. If one takes a road and a certain scale and if the transformation gives a width of more than 1 mm, this road is an area, between 1 mm and 0.1 mm it is a line, and if less than 0.1 mm the road disappears. The same reasoning is valid for cities or small countries such as Andorra, Liechtenstein, Monaco, etc.

In these cases, the “holes” in Italy or in France disappear cartographically.

In the sequel of this paper, sometimes some of those abbreviations will be used:

$Inters(R^1, R^2)$: represent the intersection between R^1 and R^2 ;

$Dist(R^1, R^2)$: is the distance between R^1 and R^2 ;

$Area(R^1 \cap R^2)$: represent the area of the intersection between R^1 and R^2 ;

$2Dmap(R^1, \sigma)$: is a function transforming a geographical object to some scale possibly with generalization, in the 2-dimension.

Thus, with the defined thresholds ε_i , ε_{lp} , we can formally get

- a. Disappearance of a geographical object (O) at scale σ :

$$\forall O \in \text{GeObject}, \forall \sigma \in \text{Scale} \wedge O_\sigma \\ = 2Dmap(O, \sigma) \wedge Area(O_\sigma) < (\varepsilon_{lp})^2 \Rightarrow O_\sigma = \phi.$$

- b. transformation of an area into a point (for instance, the centroid of the concerned object, for instance, taken as the center of the minimum bounding rectangle):

$$\forall O \in \text{GeObject}, \forall \sigma \in \text{Scale} \wedge O_\sigma \\ = 2Dmap(O, \sigma) \wedge (\varepsilon_i)^2 > Area(O_\sigma) \\ > (\varepsilon_{lp})^2 \Rightarrow O_\sigma = \text{Centroid}(O).$$

- c. Transformation of a ribbon R into a line (for instance, its skeleton):

$$\forall R \in \text{Ribbon}, \forall \sigma \in \text{Scale} \wedge R_\sigma \\ = 2Dmap(R, \sigma) \wedge \varepsilon_i > Width(R_\sigma) \\ > \varepsilon_{lp} \Rightarrow R = \text{Skel}(R).$$

Therefore, one can say that any spatial relation varies according to scale. As previously told, one says that a road runs along a sea; but in reality, in some place, the road does not run really along the water of the sea due to beaches, buildings, etc. At one scale, the road MEETS the sea (see Fig. 8a), but at another scale at some places, this is a DISJOINT relation (see Fig. 8b). Let us consider two geographical objects O^1 and O^2 and O_σ^1 and O_σ^2 their cartographic representations, for instance, the following assertion holds:

$$\forall O^1, O^2 \in \text{GeObject}, \forall \sigma \in \text{Scale} \wedge O_\sigma^1 = 2Dmap(O^1, \sigma) \wedge O_\sigma^2 \\ = 2Dmap(O^2, \sigma) \wedge \text{Disjo int}(O^1, O^2) \\ \wedge \text{Dist}(O^1, O^2) < \varepsilon_1 \Rightarrow \text{Meet}(O_\sigma^1, O_\sigma^2).$$

Similar assertions could be written for CONTAINS, OVERLAP relationships. In addition, two objects in the real world with a MEET relation can coalesce into a single one.

As a consequence, in reasoning what is true at one scale, can be wrong at another scale. So, any automatic system must be robust enough to deal with this issue.

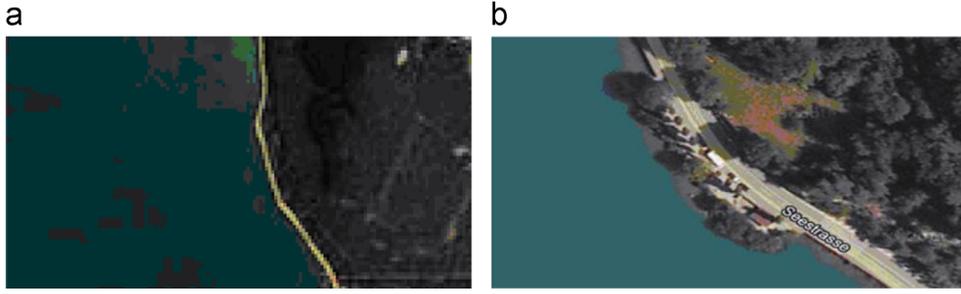


Fig. 8. According to scale, the road MEETS or not the sea.

Side-by-side	
End-to-end	
Fusion	
Splitting	

Fig. 9. Basic ribbon relations.

4.2. Ribbons–ribbons relations

In a recent paper of Laurini [9], ribbon relations were proposed to describe streets, roads and rivers. Four relations can be defined with ribbons as exemplified in Fig. 9, side-by-side, end-to-end, fusion (or merging) and splitting. For a real world feature (e.g., a road or a river), it can be modeled by a single composite ribbon, that is, a set of ribbons is linked by side-by-side and/or end-by-end relations. As the scale diminishes, ribbons will be reduced to lines, for instance, to their axes ($Axis(R)$). Thus, Laurini [9] has proposed ribbons and partially developed a model for ribbon relationships. In this work, we will complete and refine this model and we will define and classify more topological relationships between ribbons according to certain criteria, then a mathematical description will be given for each type. Thus, two ribbons can be disjoint or intersect. The disjunction is defined by a distance separating the two ribbons. The intersection between two ribbons can be point (0D), line (1D) or area (2D) according to certain criteria. In the following subsection, we will get formally the mathematical description for each topological relationship when we use thresholds and metric measurements; as area, distance, etc. let us present the most important relationships.

4.2.1. Disjoint relations

For disjoint relation between two ribbons $Disj(R^1, R^2)$, the first condition is the inexistence of an intersection between them. Fig. 10 shows five cases:

$$\forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R^1_\sigma = 2Dmap(R^1, \sigma)) \\ \wedge (R^2_\sigma = 2Dmap(R^2, \sigma))$$

$$\wedge Inters(R^1, R^2) = \phi \wedge (Dist(R^1, R^2) > \epsilon_{Ds}) \Rightarrow Disj(R^1_\sigma, R^2_\sigma).$$

4.2.2. Meeting relations

Two ribbons R^1 and R^2 are linked by a meeting relation $Meet(R^1, R^2)$ when

the intersection of two ribbons is $P(x, y) \vee L(y = ax + b)$, such as P is Point (0D) and L is Line (1D) (see Fig. 11).

$$\forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \\ \wedge (R^1_\sigma = 2Dmap(R^1, \sigma)) \wedge (R^2_\sigma = 2Dmap(R^2, \sigma)) \\ \wedge (Inters(R^1, R^2) = \{P \vee L\}) \wedge (Dist(R^1, R^2) = 0) \\ \Rightarrow Meet(R^1_\sigma, R^2_\sigma).$$

4.2.3. Merging relations

Two ribbons R^1 and R^2 are linked by a merging relation $Merge(R^1, R^2)$, if the intersection of these ribbons is an area. We obtain six cases, (see Fig. 12):

Formally, we can state:

$$\forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \\ \wedge (R^1_\sigma = 2Dmap(R^1, \sigma)) \wedge (R^2_\sigma = 2Dmap(R^2, \sigma)) \\ \wedge Inters(R^1, R^2) \neq \phi \\ \wedge (Area(R^1 \cap R^2) > \epsilon_{Mr}^2) \wedge (Area(CMP(R^1 \cap R^2)) = 0) \\ \Rightarrow Merge(R^1_\sigma, R^2_\sigma).$$

4.2.4. Crossing relations

This topological relationship is very important because 80% of spatial objects are polyline-type [31]. Common examples include road–road crossings and river–road crossings. For instance, see Fig. 13.

This relation is based on the area of the intersection between two ribbons R^1 and R^2 . For instance, a threshold ϵ_{Cr} can be given.

So, we have

$$\forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \\ \wedge (R^1_\sigma = 2Dmap(R^1, \sigma)) \wedge (R^2_\sigma = 2Dmap(R^2, \sigma)) \\ \wedge Inters(R^1, R^2) \neq \phi \\ \wedge (Area(R^1 \cap R^2) > \epsilon_{Cr}) \wedge (Area(CMP(R^1 \cap R^2)) > 0) \\ \Rightarrow Cross(R^1_\sigma, R^2_\sigma).$$

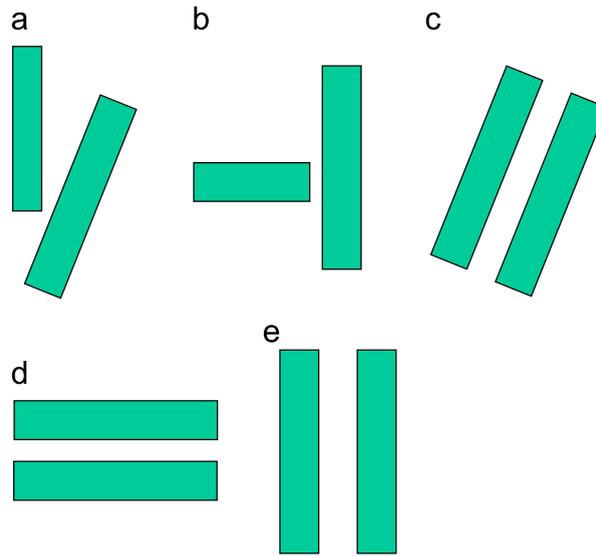


Fig. 10. Disjoint relations between two ribbons.

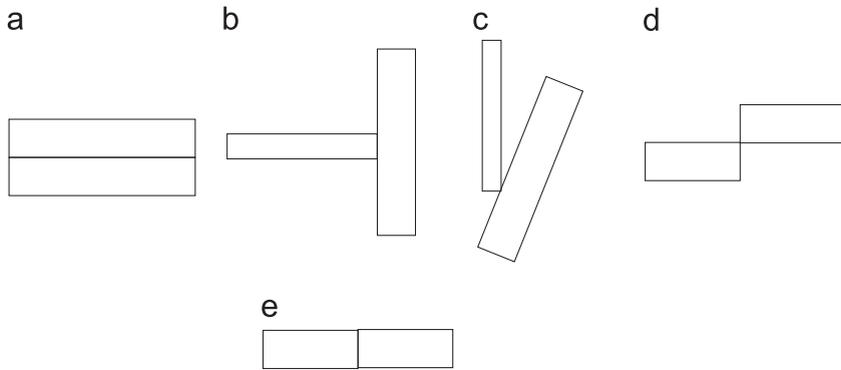


Fig. 11. Several cases for meeting from (b-d). Except (a) corresponding to a side-by-side and (e) to end-to-end.

4.3. Transformation of ribbons-ribbons relations

When downscaling, the transformation of topological relations can be applied. The topological relations between objects varied according to certain criteria, we present in the following subsection these transformations.

4.3.1. Transformation of disjoint to merge

This disjoint relation transformed into merging relation, when downscaling (see Fig. 14).

This process can be modeled as follows:

$$\begin{aligned} \forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma^1 = 2Dmap(R^1, \sigma)) \\ \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge Disj(R^1, R^2) \\ \wedge (Dist(R^1, R^2) < \varepsilon_{Dj}) \Rightarrow Merge(R_\sigma^1, R_\sigma^2). \end{aligned}$$

When a ribbon becomes very narrow, we apply this assertion:

$$\begin{aligned} \forall R \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma = 2Dmap(R, \sigma)) \\ \wedge (Width(R_\sigma) < \varepsilon_{lp}) \Rightarrow R_\sigma = \phi. \end{aligned}$$

4.3.2. Transformation of cross to merge

The crossing relation can transform into merging relation according to the area of complement of the intersection between the two ribbons: $Area(CMP(R^1 \cap R^2))$, (see Fig. 15).

The formal definition of this process is

$$\begin{aligned} \forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma^1 = 2Dmap(R^1, \sigma)) \\ \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge (Cross(R^1, R^2)) \\ \wedge (Area(CMP(R^1 \cap R^2)) < Area(R^1 \cap R^2)) \Rightarrow Merge(R_\sigma^1, R_\sigma^2). \end{aligned}$$

When a ribbon becomes very narrow, we apply this assertion:

$$\begin{aligned} \forall R \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma = 2Dmap(R, \sigma)) \\ \wedge (Width(R_\sigma) < \varepsilon_{lp}) \Rightarrow R_\sigma = \phi. \end{aligned}$$

4.3.3. Transformation of meet to merge

The transformation of meeting relation to merging relation was expressed by the following assertion

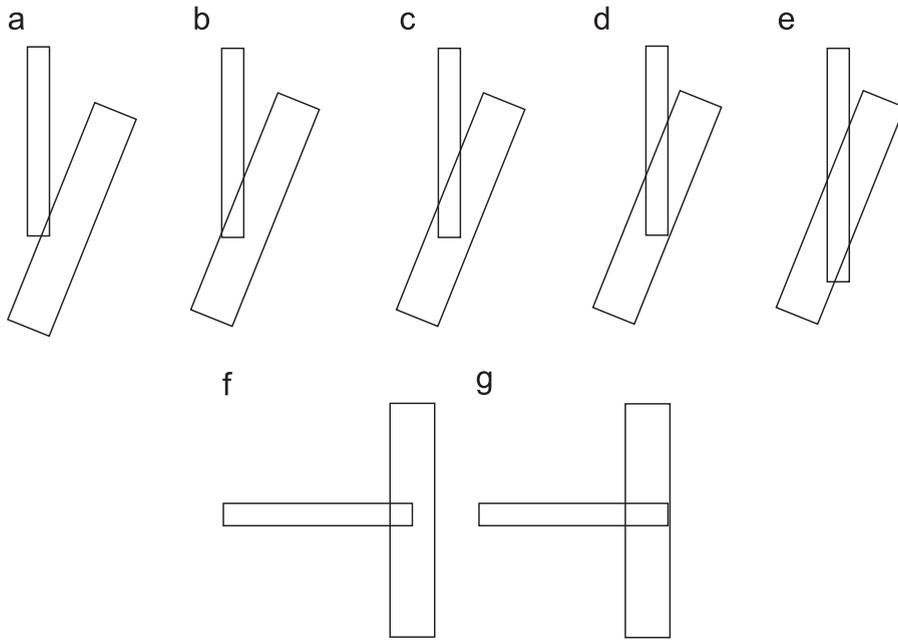


Fig. 12. Example of merging.

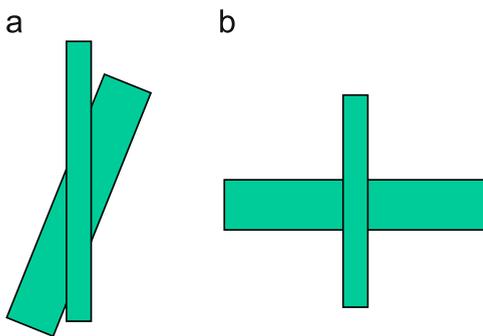


Fig. 13. Example of crossing.

(see Fig. 16):

$$\forall R^1, R^2 \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma^1 = 2Dmap(R^1, \sigma)) \wedge (R_\sigma^2 = 2Dmap(R^2, \sigma)) \wedge (meet(R^1, R^2)) \wedge (Area(R^1 \cap R^2) > \epsilon_{Mr}^2) \Rightarrow merge(R_\sigma^1, R_\sigma^2).$$

When a ribbon becomes very narrow, we apply this assertion:

$$\forall R \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma = 2Dmap(R, \sigma)) \wedge (Width(R_\sigma) < \epsilon_{ip}) \Rightarrow R_\sigma = \phi.$$

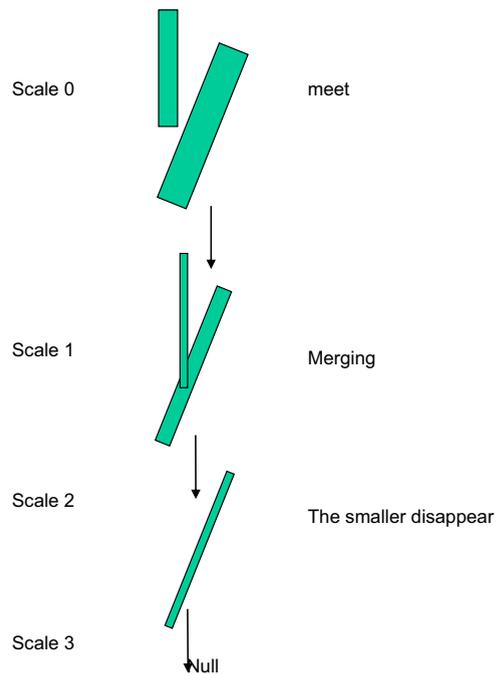


Fig. 14. Transformation of disjoint relation between two ribbons.

4.4. Ribbons–regions relations

In this section, we study the relations which can hold between ribbons and regions. To describe these relations, we can classify them into types, namely, disjoint, meet or touches, cross, cover (covered-by), contain (contained by), overlap and on-boundary, as shown in Fig. 17.

The common example in this case is when a road runs along the sea, what are exactly the spatial or geographical relations which are concerned? Sometimes, either the road touches the sea or a small beach is located between the road and the sea, etc. From a mathematical point of view, mostly there is a disjoint relation between the road and the sea whereas for people the relation is different.

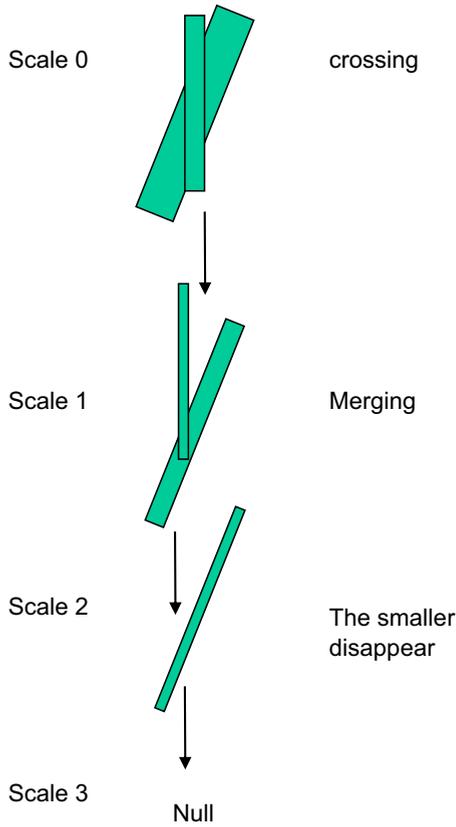


Fig. 15. Transformation of crossing relation between two ribbons.

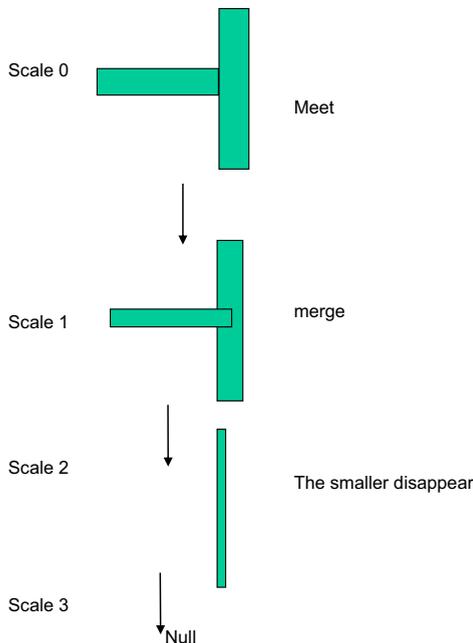


Fig. 16. Transformation of merge relation to meet.

In addition, when one is reading a map, according to scale, the topological relation can be different, disjoint or meet.

Also, the relations between regions and ribbons can also be varied according to the scale, for instance, the disjoint relation may be transformed into meet or on-boundary relations when downscaling.

4.5. Transformation of ribbons–regions relations

4.5.1. Transformation of disjoint to meet

The disjoint relation can transform into relation meet according to the following conditions (see Fig. 18):

The exact condition is that the distance between region and ribbons must be greater than the thresholds ϵ_{Ds} .

This transformation can be applied according to this assertion:

$$\begin{aligned} \forall R \in \text{Ribbon}, \forall G \in \text{Region}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma = 2Dmap(R, \sigma)) \\ \wedge (G_\sigma = 2Dmap(G, \sigma)) \wedge Disj(R, G) \\ \wedge (Dist(R, G) > \epsilon_{Ds}) \Rightarrow Meet(R_\sigma^1, R_\sigma^2). \end{aligned}$$

When a ribbon becomes very narrow, we apply this assertion:

$$\begin{aligned} \forall R \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \wedge (R_\sigma = 2Dmap(R, \sigma)) \\ \wedge (Width(R_\sigma) < \epsilon_1) \Rightarrow R_\sigma = \phi. \end{aligned}$$

The region can be eliminated if its area is too small to be well visible. Thus, in this case, the initial relation does not hold anymore.

$$\begin{aligned} \forall O \in \text{GeObject}, \forall \sigma \in \text{Scale} \wedge (O_\sigma = 2Dmap(O, \sigma)) \\ \wedge (Area(O_\sigma) < (\epsilon_{lp})^2) \Rightarrow O_\sigma = \phi. \end{aligned}$$

4.5.2. Transformation of contain to cover

The transformation of contain relation into cover relation was expressed by the following assertion (see Fig. 19):

$$\begin{aligned} \forall G \in \text{Region}, \forall R \in \text{Ribbon}, (\forall \sigma \in \text{Scale}) \\ \wedge (G_\sigma = 2Dmap(O, \sigma)) \\ \wedge (R = 2Dmap(R, \sigma)) \wedge (Contains(R, G)) \\ \wedge (Dist(R, G) < \epsilon_1) \Rightarrow Cover(O_\sigma^1, O_\sigma^2). \end{aligned}$$

But a smaller object can disappear or be eliminated if its area is too small to be well visible. So in this case, the initial relation does not hold anymore.

4.5.3. Generalized irregular tessellations when downscaling

By irregular tessellation (or tessellation), one means the total coverage of an area by sub-areas. For instance, the conterminous States in the USA form a tessellation to cover the whole country. Generally speaking, administrative subdivisions form tessellations, sometimes as hierarchical tessellations. Let us consider a domain D and several polygons P_i ; they form a tessellation if (see Fig. 20b):

- For any point p_k , if p_k belongs to D then there exists P_j , so that p_k belongs to P_j .
- For any p_k belonging to P_j , then p_k belongs to D .

A tessellation can also be described by Egenhofer relations applied to P_i and D , but in practical cases, due

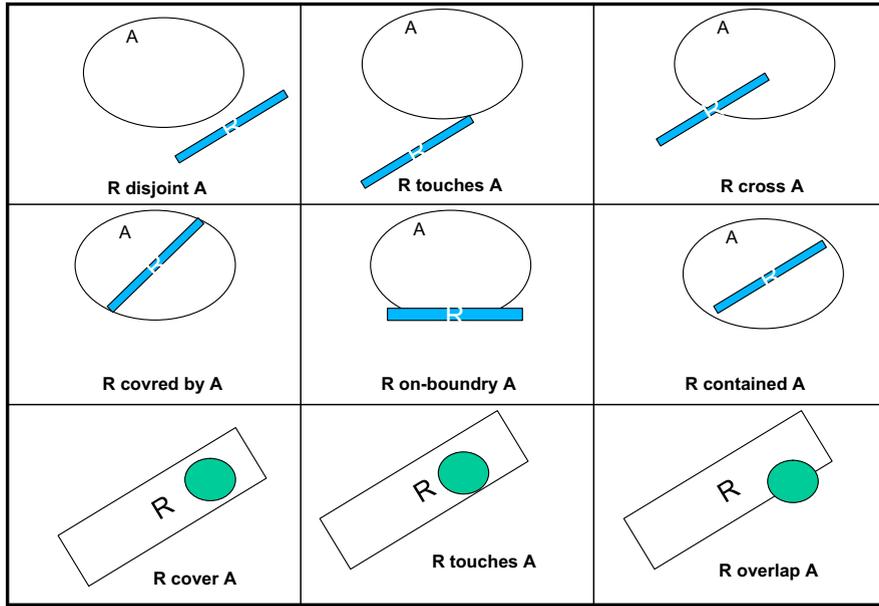


Fig. 17. Basic relations between regions and ribbons.

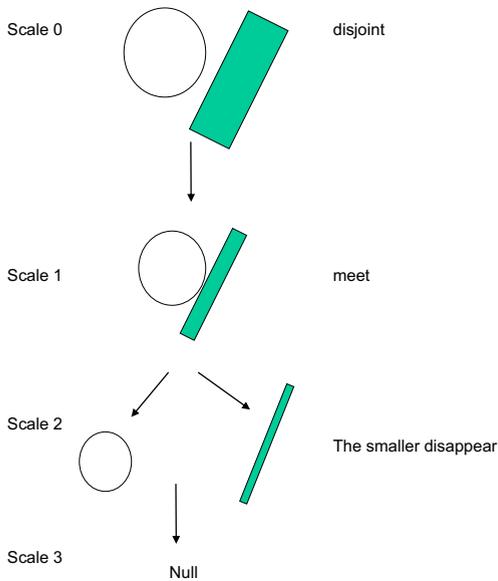


Fig. 18. Transformation of disjoint into meet.

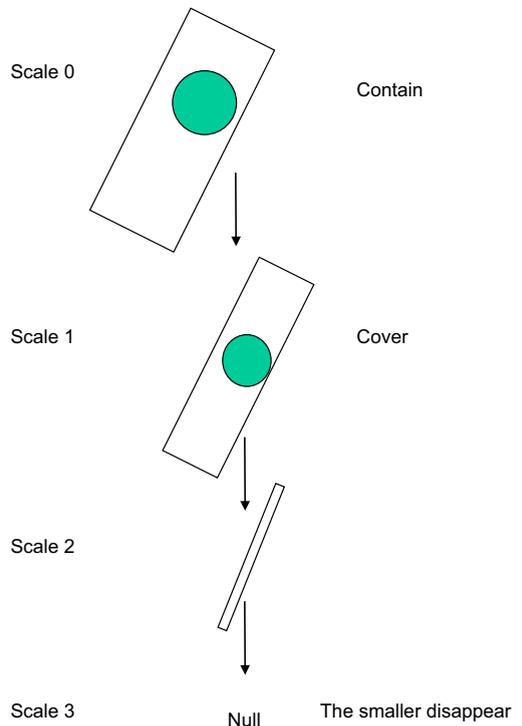


Fig. 19. Transformation of contain into cover.

to measurement errors, this definition must be relaxed in order to include sliver polygons (see Fig. 20a). Those errors are often very small, sometimes a few centimeters at scale 1. In other words, one has a tessellation from an administrative point of view, but not from a mathematical point of view.

When downscaling, those errors will be rapidly less than the threshold ϵ_{lp} so that the initial slivered or irregular tessellation will become a good-standing tessellation.

The situation becomes complex when roads or rivers traverse the tessellation, because we have to study all

topological relationships between tessellation and ribbons which represent the roads or rivers.

4.6. Chain of ribbons

A chain is defined by a set of ribbons linked by end-to-end relations. A ribbon chain may be closed; in this case, it

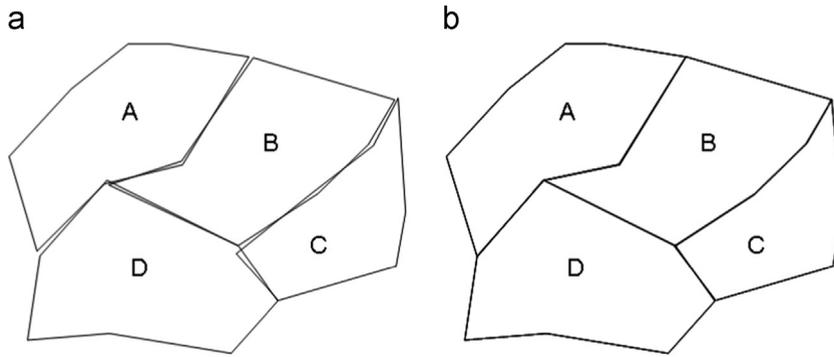


Fig. 20. A tessellation with sliver polygons and a good standing tessellation.

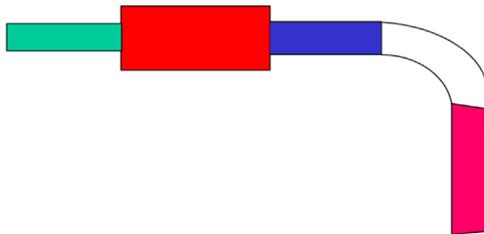


Fig. 21. Chain of ribbons.

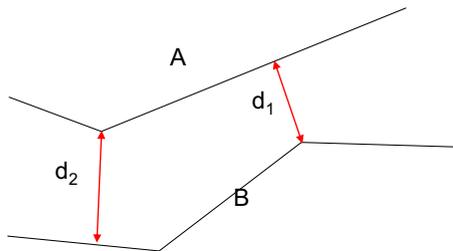


Fig. 22. The distance between two polylines.

constitutes a ribbon circuit. In general, since river's and road's widths are variable, they can be modeled by a chain of ribbons (see Fig. 21).

4.7. Distance between objects

We concentrated, in this work, especially on the distance between objects. Considering two objects A and B , what is the distance between them? An interesting definition is given by the Frechet distance which corresponds to the minimum leash between a dog and its owner, the dog walking on a line, and the owner in the other line as they walk without backtracking along their respective curves from one endpoint to the other. The definition is symmetrical with respect to the two curves (see Fig. 22) [11]. By noting a , a point of A , and b of B , the Frechet distance F is given as follows in which $dist$ is the Euclidean conventional distance:

$$F = \text{Max}(\text{Min}(\text{dist}(a, b)))$$

$a \in A \quad b \in B$

But in our case, one must consider two distances, let us say, the minimum and the maximum of the leash, so

giving:

$$d1 = \text{Min}(\text{Min}(\text{dist}(a, b)))$$

$a \in A \quad b \in B$

and

$$d2 = \text{Max}(\text{Min}(\text{dist}(a, b)))$$

$a \in A \quad b \in B$

The thresholds used in the mathematical assertions are defined from this distance. Then, the distance between two regions A and B is defined also as the Frechet distance between both boundaries. In this context, the algorithm defined in [11] is used.

4.8. Experimental analysis

The downscaling of a map implies that the topological relationships between spatial objects (ribbons and regions) should be transformed into other ones. To apply these transformations on the map, the following requirements should be satisfied: (1) the topological relationships between spatial objects should be defined, (2) a framework is required to derive the transformations of topological relationships, and (3) some metric measures and thresholds are taken to guide these transformations.

To analyze the variation of topological relationships using our mathematical framework, a prototype is developed. We implement three main functions

- $Dist(O^1, O^2)$: calculate the distance between two objects;
- $RelTOPO(O^1, O^2)$: define the relations holding between two objects;
- $TRansRel(Rel, Dist, threshold)$: apply the possible transformations using the mathematical assertions.

Using predefined thresholds, the prototype works as the following steps:

- Compute and store the topological relationships between the ribbons and/or regions, using the assertions developed in Sections 4.2 and 4.4.
- Apply the simplification operator of the generalization process.
- Use the assertions developed in Sections 4.3 and 4.5 to transform the possible topological relationships

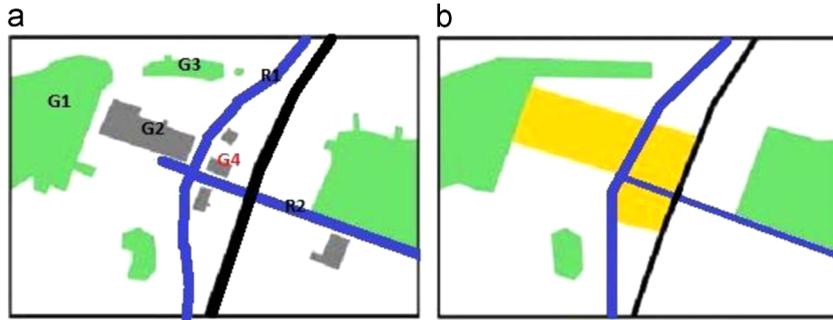


Fig. 23. Transformation of topological relationships. (a) before generalization, (b) after generalization.

Table 1

The variation of topological relationships.

Before generalization	After generalization
Disjoint(G^1, G^2)	Meet (G^1, G^2)
Cross(R^1, R^2)	Merge (R^1, R^2)
Disjoint(G^1, G^3)	Meet(G^1, G^3)
Disjoint (R^2, G^4)	Meet(R^2, G^4)

between the simplified objects (regions and ribbons).

The prototype can automatically detect the topological relationships between objects and transform them into other ones according to the mathematical assertions described in Sections 4.3 and 4.5.

Fig. 23 shows a real example; the River R^1 is crossed with another river R^2 and the buildings (G^1, G^2 and G^3) are disjoint. The topological relationships are as follows:

Disjoint(G^1, G^2).
 Cross(R^1, R^2).
 Disjoint(G^1, G^3).
 Disjoint (R^2, G^4).

Let us define two thresholds, ϵ_i for invisibility of objects and ϵ_{pl} for the reduction of objects (regions or ribbons) to point or line. In our implementation, we take $\epsilon_i = 0.1$ mm and $\epsilon_{pl} = 1$ mm. When downscaling, the rivers and the buildings are generalized and the topological relationships are transformed into other relationships. Table 1 illustrates these transformations of topological relationships

Fig. 24 shows another real example, consisting of several objects: rivers, coast, city. Thus, the French Riviera coast runs along the Mediterranean Sea, from Spain to Italy. There are three cities: Nice, Montpellier and Marseilles. The Rhone River is linked to the sea.

Meet (River, Coast).

When downscaling, the Mediterranean coast is generalized and the topological relation is transformed into Merge (River, Coast).

Since certain topological relations must be persistent, regardless of the scale of representation, those relations must hold. See, for instance, in Fig. 24 the Mediterranean Coast in the South of France: as the coast is generalized,

some harbors will be in the middle of the sea such as Nice, whereas others will be inside the country such as Marseilles and Montpellier; in addition, the confluence of the Rhone river will be badly positioned in the middle of the land. The constraints are as follows:

Covers (France, Nice).
 Covers (France, Marseilles).
 Covers (France, Montpellier).
 Covers (France, Rhone).

Another example of topological constraint when generalizing the Eastern French border is the case of Geneva which must hold outside France (see Fig. 25); the constraint is as follows:

Meet (France, Geneva).

In our implementation, we use the ribbons to represent the linear objects in order to verify the correctness of the concepts of the proposed framework. In this study, we present some examples to show the transformations of topological relationships when downscaling. The topological consistencies of the map are required when downscaling. However, traditional methods for maintaining consistencies of topological relationships are ineffective as they do not associate the shape simplification with the transformation of topological relationships. Thus, they cannot analyze the transformations of topological relationships; this makes them ineffective and weak to preserve topological consistencies in the map.

The framework presented in this paper consists of transforming the topological relationships into other ones in order to maintain the consistencies of topological relationships, thus, keep the high quality of the map when downscaling. Our collection of the cases we tested in the three previous examples corresponds to different topological relationships between spatial objects (ribbons or regions) as disjoint, meet, cross, merge and covers. They have been successfully tested and indicate the correctness of our concepts and the ability of our mathematical assertions to transform the topological relationships from any given map.

This study focuses only on the transformation of topological relationships when downscaling. The mathematical assertions of this framework can be integrated on any simplification algorithm provided by GIS as the algorithm presented in Ref. [15], but this is beyond the scope of this study. This work will be addressed in the future.

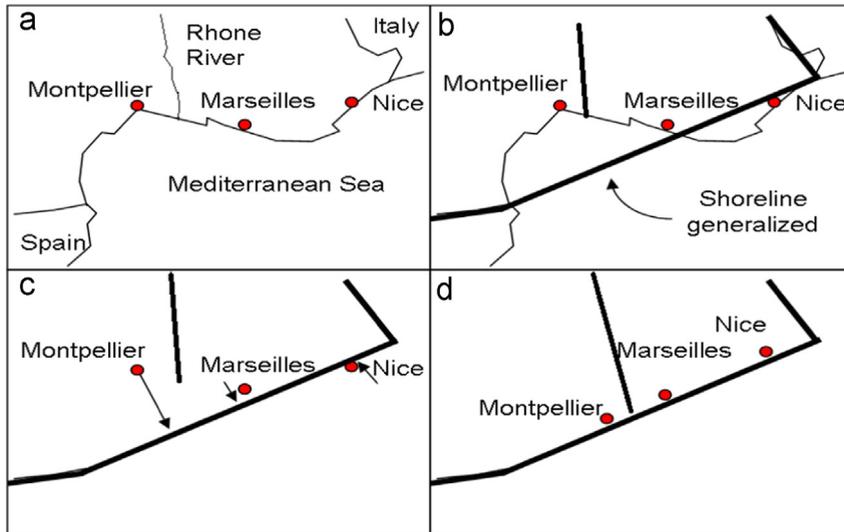


Fig. 24. Holding topological constraints for harbors in the Mediterranean Sea. (a) Before generalization. (b) Only the coastline is generalized. (c) Harbors must move. (d) After generalization (the meet relation transformed into merge).

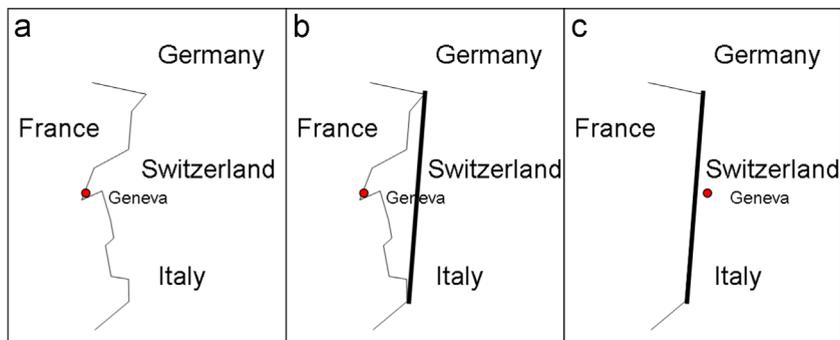


Fig. 25. Holding topological constraints for outside border cities.

5. Conclusion and future works

The concept of generalization was introduced into the GIS domain many years ago. Many propositions were given for modeling generalization but propositions do not exist which treat really the topological relations issues.

The application of the generalization operators may cause topological conflicts. To avoid these conflicts, topological conditions are used to generate the relationships in terms of meeting, overlapping, disjunction, and containment between map objects into others relationships. In this paper, we use these topological conditions to formulate some of mathematical frameworks which are composed of a set of assertions for treating the variety of topological relations according to the scale. We consider two principal types of objects: regions and ribbons. When downscaling, a spatial object, represented by area, can mutate into a point, or disappear; also a ribbon can mutate into a line, or disappear. These objects have topological relationships between them. So, each topological relation will also be generalized using the assertions given in mathematical framework for each situation. This framework was based on three principle

models of relations, Allen [5], Egenhofer [6] and Lee and Hsu [7,8].

This work can open various future works, such as:

- Integration of this topological model in on-the-fly web map generation.
- A map does not contain only the simple objects such as areas and ribbons, but there are also roundabouts and motorway interchanges that are complex objects. A complementary mathematical framework for this type of objects will be a future work.
- The assertions of the mathematical framework considered the geometries of object represented in the 2D domain, we would like to extend our work to deal with geometries of higher dimension, such as the 3D.

And finally, the foundations of a robust topology with newly presented concepts of ribbons, loose ribbons, chain of ribbons and their particular relations can be more useful to help solve real problems in geographical reasoning and in territorial intelligence. Indeed we need to design a

theory which must be robust against measurement errors and downscaling.

References

- [1] L. Harrie, The constraint method for solving spatial conflicts in cartographic generalization, *Cartogr. Geogr. Inf. Sci.* 26 (1) (1999) 55–69.
- [2] M. Egenhofer, R. Franzosa, Point–set topological spatial relations, *Int. J. Geogr. Inf. Syst.* 5 (2) (1991) 161–174.
- [3] E. Clementini, P. Di Felice, P. Van Oosterom, A Small, Set of formal topological relationships suitable for end–user interaction, in: B.C. Abeland (Ed.), *Advances in Spatial Databases, Lecture Notes in Computer Science*, vol. 692, Springer, 1993, pp. 277–295.
- [4] S. Winter, A. Frank, Topology in raster and vector representation, *Geoinformatica* 4 (1) (2000) 35–65.
- [5] J. Allen, Time and time again: the many ways to represent time, *Int. J. Intell. Syst.* 6 (1991) 341–355.
- [6] M. Egenhofer, J. Herring, A Mathematical Framework for the Definition of Topological Relationships, in: *Proceedings of the 4th International Symposium on Spatial Data Handling*, 1990, pp. 803–813.
- [7] Y. Lee, J. Hsu, Spatial reasoning and similarity retrieval of images using 2D C-string knowledge representation, *Pattern Recognit.* 25 (3) (1992) 305–318.
- [8] Y. Lee, J. Hsu, 2D C-string: a new spatial knowledge representation for image database systems, *Pattern Recognit.* 23 (10) (1990) 1077–1087.
- [9] R. Laurini, A conceptual framework for geographic knowledge engineering, *J. Vis. Lang. Comput.* 25 (1) (2014) 2–19.
- [10] M. Egenhofer, Topology and Reasoning: Reasoning about Binary Topological Relations., in: O. Gunther, H.-J. Schek (Eds.), *Proceedings of the Second Symposium on Large Spatial Databases, LNCS in Advances in Spatial Databases*, Springer-Verlag, 1991, pp. 141–160.
- [11] H. Alt, M. Godau, Computing the Fréchet distance between two polygonal curves, *Int. J. Comput. Geom. Appl.* 5 (1995) 75–91.
- [12] ICA, *Multilingual Dictionary of Technical Terms in Cartography*, Steiner, Wiesbaden, 1973.
- [13] B. McMaster, S. Shea, *Generalization in Digital Cartography*, Association of American Geographers, Washington D.C, 1992.
- [14] J. Müller, Z. Wang, Area-patch generalization: a competitive approach, *Cartogr. J.* 29 (2) (1992) 137–144.
- [15] D. Douglas, T. Peucker, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature, *Can Cartogr.* 10 (2) (1973) 112–122.
- [16] M. Galanda, R. Weibel, An agent-based framework for polygonal subdivision generalization, in: D. Richardson, P. van Oosterom (Eds.), *Advances in Spatial Data Handling, 10th International Symposium on Spatial Data Handling*, Springer, Berlin Heidelberg, 2002, pp. 121–136.
- [17] A. Ruas, C. Plazanet, Strategies for automated generalization, in: *Proceedings of the 7th International Symposium on Spatial Data Handling, Advances in GIS Research*, vol. II, 1996, pp. 1–6.
- [18] L. Harrie, *An Optimisation Approach to Cartographic Generalisation* (Ph.D. thesis), Lund University, Suède, 2001.
- [19] T. Sarjakoski, T. Kilpelainen, Holistic cartographic generalization by least squares adjustment for large data sets, in: *Proceedings of 19th International Cartographic Conference*, Ottawa, Canada, 1999, pp. 1091–1098.
- [20] M. Sester, Generalization based on least squares adjustment, *Int. Arch. Photogramm. Remote Sens.* 33 (B4/3) (2000) 931–938.
- [21] M. Ware, N. Thomas, C. Jones, Resolving Graphic Conflicts in Scale Reduced Maps: Refining, the Simulated Annealing Technique, in: *Proceedings of the 5th ICA Workshop on progress in automated map generalization*, Paris, 2003.
- [22] D. Wilson, M. Ware, A. Ware, Reducing Graphic Conflict in Scale Reduced Maps Using A Genetic Algorithm. *ICA Map Generalization Workshop*, Paris, 2003.
- [23] A. Ruas, *Modèle de généralisation de données géographiques à base de contraintes et d'autonomie* (Ph.D. thesis), University of Marne-la-Vallée, 1999.
- [24] C. Duchêne, Automated Map Generalization Using communicating Agents, in: *proceedings of the 21st International Conference of Cartography (ACI/ICA)*, Durban, Afrique du Sud, 2003, pp. 160–169.
- [25] A. Ruas, C. Duchêne, A prototype generalisation system based on the multi-agent system paradigm, in: W.A. Mackaness, A. Ruas, L.T. Sarjakoski (Eds.), *Generalisation of Geographic Information: Cartographic Modelling and Applications*, Elsevier Ltd., 2007, pp. 269–284. (Chapter 14).
- [26] N. Sabo, *Généralisation et des patrons géométriques pour la création des objets auto-généralisants (SGO) afin d'améliorer la généralisation cartographique à la volée* (Ph.D. thesis), Université de laval, Québec, 2007.
- [27] B. Lejdel, O. Kazar, Genetic agent for optimizing generalization process of spatial data, *Int. J. Digital Inf. Wireless Commun. (IJDIWC)* 1 (3) (2012) 729–737.
- [28] E. Clementini, J. Sharma, M. Egenhofer, Modeling topological spatial relations: strategies for query processing, *J. Comput. Graph.* 18 (6) (1994) 815–822.
- [29] J. Stoter, D. Burghardt, C. Duchêne, B. Baella, N. Bakker, C. Blok, M. Pla, N. Regnaud, G. Touya, S. Schmid, Methodology for evaluating automated map generalization in commercial software, *Comput., Environ. Urban Syst.* 33 (5) (2009) 311–324.
- [30] R. Thom, C. Zeeman, D. Postle, *The Cusp of Catastrophe*, Maps of the Mind, Charles Hampden-Turner; Collier Books, New York, 1981.
- [31] C. Plazanet, *Enrichissement des bases de données géographiques: analyse de la géométrie des objets linéaires pour la généralisation cartographique (application au routes)* (Ph.D. thesis), University of Marne-la-Vallée, 1996.