A MULTI-AGENT SYSTEM FOR GENERALIZATION OF FEATURES DEFINED BY ISOBATHS IN NAUTICAL CHART CONSTRUCTION

ZHANG XUNRUO

M.Phil
The Hong Kong Polytechnic University
2013

The Hong Kong Polytechnic University
Department of Land Surveying & Geo-Informatics

A Multi-Agent System for Generalization of Features Defined by Isobaths in Nautical Chart Construction

ZHANG Xunruo

A thesis submitted in partial fulfilment of the requirements for
the degree of Master of Philosophy

August 2012

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge of belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the test.

(Singed)

(Name of Student) ZHANG Xunruo

Abstract

Nautical charts provide a schematic representation of the seafloor and are used by navigators to plan their routes and to identify navigation hazards. Nautical charts are official documents issued by national hydrographic offices which are in charge of collecting the data and producing the charts. One of the most important stages in the production process is the chart generalization. It consists in abstracting the chart to emphasize important items and increase its legibility. The cartographer has not only to adapt the amount of information to the scale of the chart but also to select the information according to the types of features on the seabed and their relevance for navigation. Methods usually applied for contour generalization on topographic maps cannot be applied on isobaths directly because they do not consider the type of terrain features characterized by the isobaths. Therefore, a strategy is needed for the generalization of isobaths that fits nautical chart requirements.

This dissertation focuses on a new generalization approach where features formed by groups of isobaths are identified and classified in a hierarchical structure based on their inclusion and elevation. Generalization constraints are defined according to the type of feature (pit or peak) and specific generalization operators are defined and applied according to the constraints.

After exposing the specificities of nautical charts and reviewing the main techniques developed in line generalization, a list of generalization operators that apply to isobaths delineating features is defined. Aggregation and enlargement operations are based on a snake model where a system is stable when its energy is minimal. Violating a cartographic constraint adds energy to the line which has to be deformed in order to find a stable position.

`

In order to automate the application of generalization operations, a strategy based on a Multi Agent System (MAS) is developed. Each feature and each isobath is represented by an agent. Each feature agent is able to evaluate its situation with regard to its own attributes and other agents. Based on this evaluation, a feature can set up different plans of action corresponding to the generalization operators that apply to the type of feature. An evaluation of the different plans is performed and the one minimizing constraint violations is selected.

The method, combining the snake model and the MAS is applied on a set of isobaths provided by the French Hydrographic Office. Results are discussed and conclusions and directions for future works are presented in a last section.

Acknowledgements

I would like to express my gratitude to my supervisor Dr. Eric Guilbert, for his guidance, help and encouragement throughout the whole project.

I would like to express my gratitude to my co-supervisor Li Zhilin

I would like to thanks my friends and my parents for their encouragements during the project work

Table of Contents

1. INTRODUCTION 1

1.1. THE NAUTICAL CHART 1

1.2. NAUTICAL CHART GENERALIZATION 2

1.3. OBJECTIVES AND SIGNIFICANCE 5

1.4. ORGANIZATION OF THE THESIS 6

A REVIEW OF GENERALIZATION METHODS RELATED TO ISOBATHS 8

1.5. INTRODUCTION 8

1.6. TERRAIN GENERALIZATION 9

1.7. LINE GENERALIZATION 11

1.8. GENERALIZATION BASED ON MULTI AGENT SYSTEMS (MAS) 24

1.9. SUMMARY 33

A NEW METHOD FOR GENERALIZING UNDERSEA FEATURES 35

1.10. INTRODUCTION 35

1.11. PRINCIPLE OF THE METHOD 36

1.12. DEFINITION OF GENERALIZATION OPERATORS BASED ON A SNAKE MODEL 44

1.13. EXPERIMENTAL EVALUATION OF THE OPERATORS 48

1.14. SUMMARY 60

AUTOMATIC FEATURE GENERALIZATION USING A MULTI-AGENT SYSTEM 62

1.15. INTRODUCTION 62

1.16. THE MAS MODEL 63

1.17. APPLICATION TO FEATURE GENERALIZATION 72

1.18. SUMMARY 79

CONCLUSION 81

1.19. SUMMARY 81

1.20. DIRECTIONS FOR FURTHER INVESTIGATIONS 83

REFERENCES 85

1. Introduction

1.1. The nautical chart

A topographic map is a kind of map providing a schematic representation of the surface of the earth. It includes both natural and man-made features which are selected according to the scale and purpose of the map. A nautical chart is one specific kind of topographic map. The main usage of nautical chart is for navigation. It must provide an accurate and up-to-date visual representation of the seafloor and the surrounding land for navigators to plan their route and evaluate their position safely. Elements of coastal areas that are useful for navigation such as the coastline, terrain features, lighthouse and docks are portrayed.

In water areas, non-natural objects such as buoys, wrecks but also traffic separation schemes are shown to assist navigation. However, the most important issue of a nautical chart is the portrayal of the seafloor morphology. The seafloor is described by soundings and isobaths (or bathymetric lines). A sounding is a spot showing the sea depth at an exact location. Isobaths are contour lines joining points of equal depth. Soundings are obtained during hydrographic surveys (nowadays using multi-beams echo soundings or, in shallow areas, from lidar and satellite images) and stored into a bathymetric database. Isobaths are built from the soundings by interpolation. Soundings defined by exact position and depth provide highly accurate information about the morphology while isobaths "*enable the mariner to form a better mental image of the shape of the ocean bottom*" by providing a statistical surface (NOAA, 1997, p. 4-11). Isobaths play a slightly different role than elevation lines on topographic maps. They are used to "*illustrate shallow areas, shoals and banks, irregular bottoms, navigable channels and passages, and deeps*" (NOAA, 1997, p.

4-11). A main reason is that the mariner on a ship cannot see the seafloor and can only rely on the information provided by the chart and by positioning equipments on board. The vertical interval between isobaths on a chart is therefore shorter in shallow areas where more relevant terrain features for navigation are located.

1.2. Nautical chart generalization

Construction of a nautical chart is done by the cartographer who selects the data from the database and arranges them to produce a legible map. All the data stored in the database cannot be portrayed on the map. The cartographer has to select them according to their relevance to navigation and adapt them to the scale of the chart. This specific stage in the chart construction process is the generalization.

Map generalization is a process of deriving maps at smaller scale or lower resolution from large scale or high resolution (Melih, 2002). There are three kinds of generalization: object generalization, model generalization and cartographic generalization. Object generalization is the process of abstracting an original database from resource data (Weibel and Dutton, 1999). *The main objective of model generalization is controlled data reduction for various purposes* (Melih, 2002). When a thematic map is built from source data or derived from a larger scale map, most data in the original database or map are made redundant: they are not relevant and can be omitted or too detailed and can be simplified according to the purpose of the map. Model generalization is therefore the process of selecting and adapting objects from the database according to the amount of information needed while preserving their semantic meaning. Operations required in model generalization of often relevant to database management and include selection/omission, aggregation and typification.

Cartographic generalization consists in increasing the legibility of the map (Melih, 2002). Unlike model generalization, cartographic generalization focuses more on visualization and aesthetic aspect. The semantic meaning and relationships between objects are not modified. Objects are kept according to the scale of the map. Geometric features are smoothed or exaggerated according to their relevance.

The development of efficient methods for terrain generalization on topographic maps has been a research focus for many years. However, most research focused on cartographic generalization. Existing methods relate to terrain model simplification, spot height selection and contour interpolation and smoothing. These methods are used to generalize the topography at a smaller scale with respect to legibility constraints but with limited consideration for terrain morphology or constraints specific to the map purpose. As a consequence, they are not suitable for nautical chart generalization where the cartographer's prime objective is not to produce a seafloor representation as close as possible to reality but to report bathymetric information that are relevant to the mariner with respect of strong cartographic constraints.

The most important constraint imposed on nautical charts is that they must guarantee safety of navigation. That means that navigational hazards such as reefs much be clearly emphasized on the map and that fairways which are safe for navigation should be portrayed. Most of all, when generalizing the bathymetry, the cartographer must make sure that the depth reported on the chart is never deeper than the real depth. As a consequence, the bathymetry presented on the final chart is a schematic representation providing a higher envelope of the seafloor.

Constraints imposed on sounding selection and isobath generalization depend on the kind of terrain they portray. For example, a sounding characterizing a reef must be maintained as its removal would bring safety issue while a sounding marking a pit may be removed. Similar considerations also apply to isobaths. As shown in Figure 1.1, an isobath representing an undersea peak must be maintained while an isobath representing a pit can be removed if too small or does not mark a relevant navigation passage or a berth.

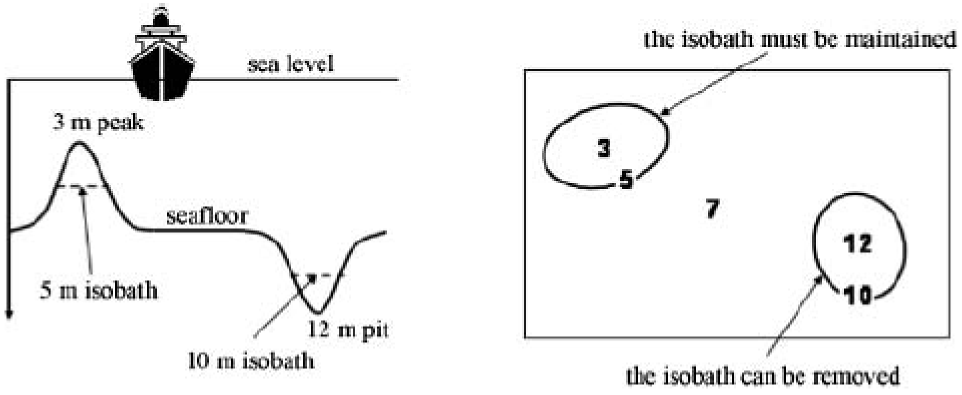


Figure 1.1 Safety of Nautical Chart (Guilbert and Saux, 2008)

Similarly, displacement and smoothing of isobaths are restricted by the safety constraint: the shape of an isobath can be modified only by pushing the line towards greater depth (Figure 1.2). A direct consequence is traditional line simplification algorithms (Douglas and Peucker, 1973; Li and Openshaw, 1993) cannot apply. It forbids also interpolating new isobaths from existing isobaths (Li and Sui, 2000) or interpolating isobaths from a simplified terrain model.

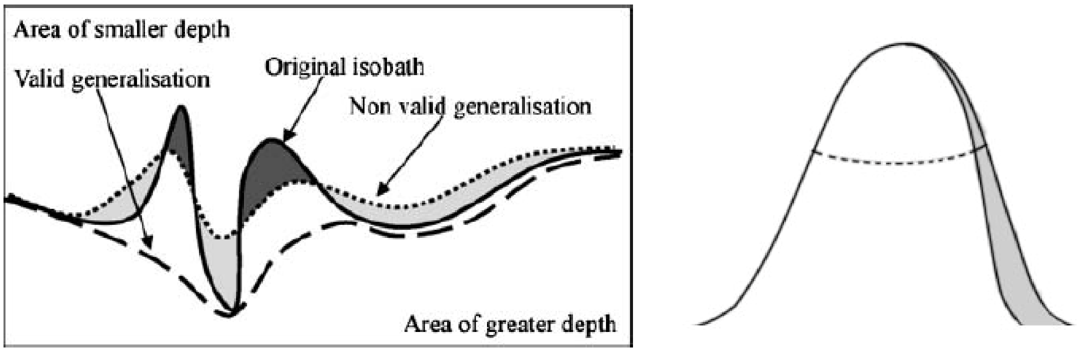


Figure 1.2 the safety constraint: the shape of an isobath can be modified only by pushing the line towards greater depth (Guilbert and Saux, 2008)

1.3. Objectives and significance

While generalization of terrain on topographic maps is mostly automated, most generalization processes in nautical charts require human intervention due to its purpose. Specific sounding selection methods were developed (Oraas, 1975; Zoraster and Bayer, 1993) and are commonly used by hydrographic offices to trim down the number of soundings but isobath generalization is still an issue. Specific operators for a single isobath displacement and smoothing with respect to the safety constraints were developed (Guilbert et al., 2006; Guilbert and Saux, 2008). However, the choice of an operator applied to an isobath depends of its context and the type of feature it models. Indeed, as stated by Robinson et al. (1995), landscape features are depicted by sets of contour lines and not the lines themselves. A logical approach would be therefore to characterize the features modeled by the isobaths and make use of this knowledge to select and apply the operators.

Such an approach in isobath generalization is relevant to model generalization as isobaths are selected and adapted according to the meaning carried on the map. The objective of this project is therefore to develop a model for the automatic identification and generalization of submarine features formed by isobaths. Other elements of the chart such as soundings and coastline are not considered. The original contribution of this thesis is at two levels. As mentioned before, automation of nautical chart production still stays behind other types of maps due its specificity and the lack of methods. Therefore, a first contribution is to provide a novel strategy that helps automating the generalization process. From a more theoretic point of view, most research done on terrain generalization is concerned by cartographic generalization. The second contribution is to address model generalization of terrain where terrain features are defined as map elements with their own characteristics and operators.

To achieve this goal, several questions need to be answered. First, features must be characterized from the isobaths and organized in a data structure. The structure is used to compute topologic and spatial relationships between features and isobaths. Second, a set of generalization operators that apply to features must be defined. The focus is on model generalization and a set of operators selecting features and correcting conflicts between features is introduced. Finally, a strategy that combines these operators is established so that the process can be automated.

For that purpose, a Multi-Agent System (MAS) is introduced. MAS has been successfully used in topographic map for road, building and road generalization (Duchêne et al., 2001, Gaffuri, 2006). In this thesis, both features and isobaths are considered as agents at different levels which can exchange information in order to form a plan of actions and evaluate the result of a generalization.

1.4. Organization of the thesis

The next chapter will provide a review of existing generalization approaches together with works done on isobath generalization in the context of nautical charts. It will also discuss previous research on the application of MAS to map generalization.

In chapter three, submarine feature is introduced as a new cartographic object formed by isobaths. The feature tree, a topologic data structure defining features and their topologic relationship is defined. Finally, a list of feature generalization operators is developed. These operators can be discrete (deletion) or continuous (enlargement). Modifications brought by the operators to the feature tree are discussed and energy constraints applied to the snake model are detailed and their performances are discussed.

In a fourth chapter, a new generalization strategy based on a MAS is presented. Different types of agents are defined. Plans of actions are set according to the type of features. Selection and validation of a plan is based on evaluation tools which estimate if cartographic constraints are respected or not. The method is applied on a set of isobaths extracted from a bathymetric database provided by the French hydrographic office. Results are discussed from a cartographer's perspective stating the potentials and limitations of the method for possible integration in nautical chart production line.

The last chapter gives a summary and conclusions of the work done and its results. Limitations of the current approach and plans for further investigations are discussed including the integration of other objects and constraints for further automation of the generalization process and the application of the model to other relevant problems in cartography and beyond.

2. A review of generalization methods related to isobaths

2.1. Introduction

This chapter reviews existing approaches that relate to topographic map generalization and more specifically to line objects. Generalization requires the application of different types of operators which need to be combined and applied to different objects. Generalization is a subjective process and although rules to observe are clearly known, most of them are informal and the final result depends on the cartographer's knowledge and experience. Automating the generalization process consisted on one side in developing some techniques to perform specific operations and on the other side in developing some approaches modeling the whole process. A topographic map provides a representation of the relief and of natural and manmade features. Relief can be seen as a continuous phenomenon and modeled by a field function while features such as buildings, roads and rivers are perceived and modeled as objects on the map. Therefore terrain generalization may be perceived as a mathematical process where the objective is to obtain an optimal representation of the terrain according to an error threshold which also preserves the terrain morphology. As isobaths provide a representation of the seafloor, terrain generalization techniques can also be considered for bathymetric generalization. However, the objective of a nautical chart is not to provide an accurate depiction of the seafloor as much as it is to provide meaningful information related to navigation and therefore generalizing a nautical chart consists also in selecting submarine features according to their relevance to the navigator, whether they represent a danger or a safe route.

In order to automate isobaths generalization, two approaches can therefore be considered. Isobaths can be generalized by simplifying a terrain model (grid or triangulated network) built from the lines or by directly generalizing the isobaths which are seen as objects that can be selected or modified. A review of different approaches applied to terrain and contour generalization is therefore presented in the next sections. Section 2.2 focuses on terrain generalization and the different approaches to the problem. Section 2.3 is concerned with constraint-based modeling for line generalization with special attention given to existing algorithms developed for isobaths. Section 2.4 introduces agent modeling and reviews multi agent system models as a tool used to model a generalization strategy. Finally, limitations of existing works are discussed and directions are given for the development of a model for isobath generalization.

2.2. Terrain generalization

2.2.1. DTM generalization

Possible approaches for terrain model generalization are discussed by Weibel (1992). The author identifies three types of method and proposes a conceptual strategy for DTM generalization for topographic maps.

The generalization process is divided into five steps: structure recognition, process recognition, process modeling, process execution, and data display. Structure recognition identifies the specific cartographic objects, spatial relationships and measures of importance. Process recognition identifies the generalization processes (operators). Process modeling compiles rules. Then process execution does the generalization and data display draws the transformed map.

Three types of generalization methods can be applied. First are global filtering methods which apply a low-pass filter to the whole terrain. The approach is based on resampling and filtering methods as used in image processing for denoising and is not adaptive to the structure of the terrain. Selective filtering generalizes the map by eliminating non-significant points from a grid or a triangulated network. Methods were not developed only for terrain simplification but also for hydrological and geological purposes. Two approaches are considered (Chen et al., 2012): the first one selects very important points (VIP) based on a distance or error threshold (Peucker and Douglas, 1975), second is based on the extraction of feature points and lines obtained from the drainage system. Chen et al. (2012) compares three terrain generalization based on drainage extraction. Such approach can also apply to spot height selection where feature points are selected however these methods do not consider cartographic constraints such as the distribution of the points on the map or their importance for the map reader. These methods are also limited to selection operations. Nautical chart generalization requires not only the selection of important soundings but also the deformation of the terrain to emphasize or caricature certain submarine features and require the use of individual operators.

Such methods are referred as heuristic generalization and form the third approach identified by Weibel (1992). They consist in developing individual generalization operators that apply to terrain elements in order to mimic manual techniques. Such methods are for example applied to contour lines for smoothing but should also apply to landforms. As such, landforms should be addressable as objects and each class of landform should have its own dedicated types of operators that are applied according to a specific purpose. Such technique received limited consideration as it requires a formal definition of landforms and such classification is still an open problem (Smith and Mark, 2003). On the other side, significant amount of work was done in simplifying and correcting contours for topographic maps.

2.2.2. Contour generalization

Two approaches are usually considered for contour generalization (Li and Sui, 2000). The first approach consists in generating a DTM surface from the contours, generalizing the terrain and extracting the new contours from the generalized surface. It belongs therefore to selective filtering approaches. Other methods process the contours directly. They are developed to perform specific operations such as segment removal (Mackaness and Steven, 2006), smoothing (Irigoyen et al., 2009, Lopes et al., 2011) and simplification (Li and Sui, 2000, Gökgöz, 2005, Matuk et al., 2006).

Li and Sui (2000) algorithm is based on the SVO (smallest visible object) principle to simplify the contours. The method can also extract contours at a new vertical interval by interpolation however such operation is not allowed for isobath generalization. Gökgöz (2005), Mackaness and Steven (2006) and Irigoyen et al. (2009) follow a condition-action approach (Harrie and Weibel, 2007) where contour points are displaced or deleted based on shape, slope or distance criteria. Matuk et al. (2006) first builds the terrain skeleton from the contours and prunes the skeleton to simplify the contours. Gökgöz (2005) and Matuk et al. (2006) simplification methods are followed by a smoothing process based on spline interpolation. Irigoyen et al. (2009) make use of a neural network to satisfy different generalization criteria.

These methods were designed for cartographic generalization of contours on topographic maps. They are concerned with the legibility and aesthetic of the map and its topological correctness but the map purpose is not considered. They cannot be applied directly to nautical chart generalization as the safety constraint is not considered. Furthermore, generalization must be done according to the type of feature recognized on the seafloor. These features can be caricatured or aggregated and so require the definition of other operators. Therefore, an automatic process needs to be able to identify the features and select and perform one or several generalization operations. Combining methods based on condition-action approach leads to difficulties as rules of several operations must be explicit and each possible situation must be considered. The next sections discuss other existing work on line generalization that may apply to contour line and existing strategy for automating the generalization process.

2.3. Line generalization

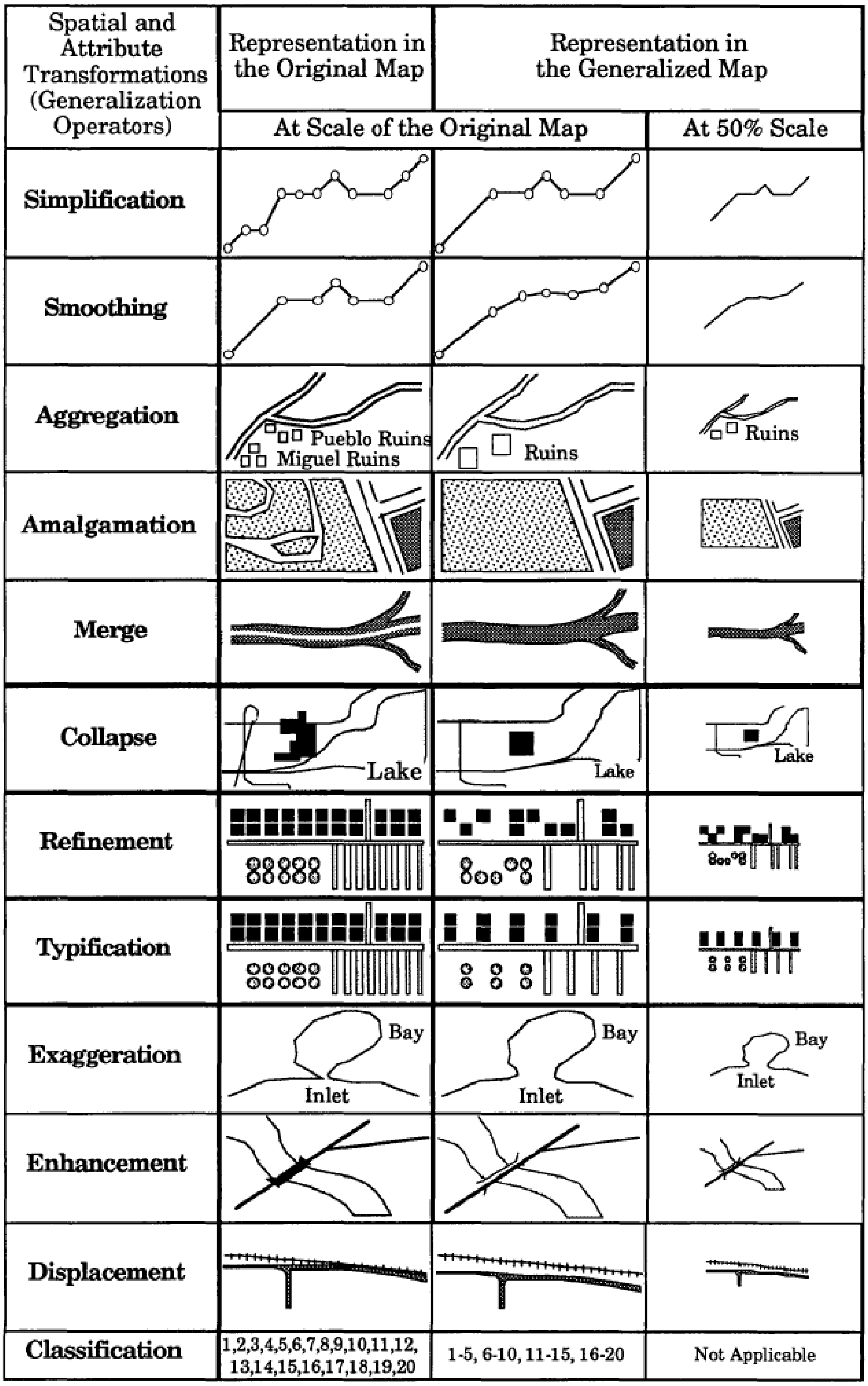
2.3.1. Generalization operators for line generalization

As mentioned in the introduction, the map generalization process is divided into object generalization, model generalization where data are selected based on their relevance, and cartographic generalization concerned with the aesthetics and legibility of the map. The process consists in applying different operations in order to satisfy cartographic rules. The result depends of the choice of operations and the order in which they are applied. Operators can be specific to either model or cartographic generalization or can be used in both. For example, an object can be removed because it is not relevant for the map purpose or because it is too small to be legible. Therefore it belongs to both types of generalization.

McMaster and Shea (1988) listed 12 operations for generalization (Table 2.1). The difference between aggregation and amalgamation is that, in aggregation, a group of small objects is merged while in amalgamation, a group of small objects is merged and their outline is maintained. Refinement and typification are similar to each other. When there are cluster of small and similar objects, refinement operator omits the small and unimportant ones. Typification uses pattern and symbol instead of cluster of objects.

Among operators listed in table 2.1, simplification, smoothing, displacement, exaggeration, aggregation and refinement (by deleting some isobaths) are applicable to isobaths and are used in cartographic generalization. However the selection of an operator and its application depend of the kind of conflict to be corrected and the meaning conveyed by the isobaths, which can be either emphasized or diminished. That requires the definition of a strategy on when to trigger and how to control an operation that is set up through the modeling of the generalization process.

Table 2.1 List of generalization operations (McMaster and Shea, 1988)



Spatial and Attribute Transformations (Generalization Operators)	Representation in the Original Map	Representation in the Generalized Map	
	At Scale of the Original Map		At 50% Scale
Simplification			
Smoothing			
Aggregation	Pueblo Ruins Miguel Ruins	Ruins	Ruins
Amalgamation			
Merge			
Collapse	Lake	Lake	Lake
Refinement			
Typification			
Exaggeration	Bay Inlet	Bay Inlet	Bay Inlet
Enhancement			
Displacement			
Classification	1,2,3,4,5,6,7,8,9,10,11,12, 13,14,15,16,17,18,19,20	1-5, 6-10, 11-15, 16-20	Not Applicable

Three main approaches were developed to model the generalization process (Harrie and Weibel, 2007): condition-action modeling and human interaction modeling based on rules and expert systems and, more recently, constraint-based modeling. Generalization being a subjective task, it is difficult in the first two models to formalize all rules exhaustively and automate the process. Different rules or rules applied in different orders would lead to different generalization results. Complex situations can also lead to contradictions that cannot be handled or that require external intervention. In the last model, conditions to be satisfied by the map are expressed as constraints and an optimal solution minimizing constraint violation is sought.

Constraints state what should be obtained. Each constraint can be measured to estimate how important a violation is. The approach gives more flexibility as the objective is not to satisfy all the constraints but to find a compromise that minimizes the violations. Beard (1991) classified constraints into application, graphic, structural and procedural constraints. Following this classification, Guilbert and Saux (2008) identify the following constraints for isobath generalization:
The application constraint of safety
The graphical constraint of legibility
The structural constraint of geomorphology
The procedural constraint of consistency

As specified in the introduction, the safety constraint is specific to nautical charts and cannot be violated. The depth reported on the chart should be equal to or shallower than the real depth. That means that an isobath characterizing a peak cannot be removed and if an isobath is modified, it must be pushed towards greater depth.

The legibility constraint defines the minimum distance between two isobaths or two segments of an isobath on a chart. It also defines a minimum area for closed isobaths to be portrayed on the chart. A closed isobath always modeling a peak or a pit, the shallowest or deepest point of the feature must be marked by a sounding. Therefore the minimum area is defined by the area required to mark the sounding. A last constraint related to legibility is the line smoothness as the isobath shape should be adapted to the scale. Due to the safety constraint, a specific operator needs to be defined.

Structural constraints seek to preserve the patterns formed by map objects. On a topographic map, the terrain morphology should be maintained. Slope and roughness characterizing the geomorphology should be preserved and important terrain features must be emphasized. Consistency constraints are logical rules about the objects that need to be respected when performing an operation. For example, when aggregating two lines, one line must be closed and both lines should be at the same depth.

All constraints do not have the same importance. The legibility and safety constraints are strong constraints that the cartographer must respect while the structural constraint is a weak constraint. The seafloor morphology needs to be maintained as it provides the mariner valuable information about the environment to the condition that it does not affect the legibility and that greater prominence is given to shallow features. As a consequence, a nautical chart usually provides a more schematic representation than other topographic maps.

Once constraints are defined, generalization is performed by applying different operators in order to minimize constraint violations. Operations can be classified according to three main modeling techniques (Harrie and Weibel, 2007): combinatorial optimization, continuous optimization and agent modeling. Combinatorial optimization methods are more relevant to discrete operators while the second technique is used to develop continuous operations (Gaffuri, 2006). Agent modeling allows combination of both discrete and continuous operations defined by other techniques into one model.

Typical techniques used in line generalization are optimization techniques providing continuous deformations to lines and are mainly used for smoothing and displacement. Sester (2000) made use least square adjustment (LSA) to displace roads and rivers. However, constraints must be linearized and it is hard to control the direction of displacement. Guilbert et al. (2004) used cable network to solve distance conflicts between isobaths. However, it is difficult to control the location and smoothness of final line. Burghardt and Meier (1997) and Burghardt (2005) introduced the snake model for road displacement and smoothing providing a better control of the constraints. The next section describes the snake model in details and presents different approaches, including methods by (Guilbert et al., 2006) and (Guilbert and Saux, 2008) which are applicable to isobaths.

2.3.2. Line generalization using the snake model

The snake (or active contour) model is an optimization model based on the principle that a physical system is at its equilibrium position when its total energy is minimum. The model was introduced by Kass et al. (1987) in image processing for delineating object boundaries. A line is seen as a system with an internal energy part which represents the intrinsic features of the line and an external energy which represents external constraints applied to the line (Equation 2.1). The external energy definition depends on the application. The internal energy controls the shape of the line and is defined by the first and second derivatives. They represent the tensile force and the pressure force and are controlled by two parameters α and β (Equation 2.2). An optimal solution is achieved by minimizing the total energy (Equation 2.1). The solution is computed by an iterative method so that during the deformation, the line is attracted toward areas of low external energy but has to remain smooth at the same time.

$$E_{tot} = \int_a^b \left[E\mathrm{int}\left(f(t)\right) + Eext\left(f(t)\right)\right]dt \quad \text{(Equation 2.1)}$$

$$E_{\mathrm{int}} = \frac{1}{2}\left(\alpha\left|f'(t)\right|^2 + \beta\left|f''(t)\right|^2\right) \quad \text{(Equation 2.2)}$$

2.3.3. Snakes for line displacement

The snake model was introduced in cartographic generalization by Burghardt and Meier (1997) for line displacement. The internal energy is defined as:

$$E_{\mathrm{int}} = \frac{1}{2}\left(\alpha \,|\, d'(s) \,|^2 + \beta \,|\, d''(s) \,|^2\right) \text{(Equation 2.4)}$$

where s is the curvilinear abscissa and $d(s) = f(s) - f^{\,0}(s)$ is the displacement of f from its original position $f^{\,0}$. In this model, the snake is not the line itself but its displacement. The functions d' and d'' are the first and second derivatives of the displacement. The parameters α and β control the tension and bending of the displacement. The original line has a null internal energy. Keeping a low internal energy means that the displacement is homogeneous and the shape of the line is preserved.

The external energy is defined as:

$$Eext(v_i) = \begin{cases} 1 - (a/h) : a < h \\ 0 : a \geq h \end{cases} \quad \text{(Equation 2.3)}$$

where a is the distance between two lines and h is a minimum distance tolerance given by the user. As this external energy is defined, when the distance between the lines is greater than the tolerance, there is no conflict and the external energy is zero. If the distance is smaller, the external energy is positive. The line will move away from the other line to reduce the external energy. At the same time, the internal energy tends to preserve the shape of the line by having a homogeneous displacement.

Minimization of the total energy requires a discretization of the equations. Burghardt and Meier (1997) use finite differences to build a matrix system that can be solved. After transformation to Eulerian equations, the original formulas are decomposed into two independent functions for the x-direction and y-direction.

$$\frac{\partial Eext}{\partial x} - \frac{dEx_s}{ds} + \frac{d^2 Ex_{ss}}{ds^2} = 0 \quad \text{(Equation 2.5)}$$

$$\frac{\partial Eext}{\partial y} - \frac{dEy_s}{ds} + \frac{d^2 Ey_{ss}}{ds^2} = 0 \quad \text{(Equation 2.6)}$$

In both equations, the subscripts s and ss indicate the first and second derivatives. After discretizing with finite differences, the internal energy part of each point Pi of the line is written:

$$0 = (E_x^i, E_y^i) + c\omega_{i-2} + b\omega_{i-1} + a\omega_i + b\omega_{i+1} + c\omega_{i+2} \quad \text{(Equation 2.7)}$$

where $\omega = (x - x_0, y - y_0)$ and (E_x^i, E_y^i) is the gradient of external energy, i.e. the external force that is applied to the point.

Coefficients a, b and c are defined as $a = 2\alpha + 4\beta, b = -\alpha - 4\beta, c = \beta$ and equation 2.7 can be written in matrix presentation:

$$A(x_t - x_0) = E_{extx} \quad \text{(Equation 2.8)}$$

$$A(y_t - y_0) = E_{exty} \quad \text{(Equation 2.9)}$$

where A is:

$$A = \begin{pmatrix} a & b & c & 0 & \ldots\ldots & & \\ b & a & b & c & 0 & \ldots\ldots & \\ c & b & a & b & c & 0 & \ldots\ldots \\ 0 & c & b & a & b & c & 0\ldots \\ 0 & \ldots\ldots & c & b & a & b & c \\ 0 & \ldots\ldots & & c & b & a & b \\ 0 & \ldots\ldots & & & c & b & a \end{pmatrix} \quad \text{(Equation 2.10)}$$

In A, each row corresponds to a point. The size of A is equal to the number of point of the line.

The system is solved iteratively. A parameter γ is introduced to relax the system and avoid large displacement to a non local minimum.

$$(A+\gamma I)\left(x^{t}-x^{0}\right)=x^{t-1}-x^{0}-Eextx\left(x^{t-1},y^{t-1}\right) \quad \text{(Equation 2.11)}$$

$$(A+\gamma I)\left(y^{t}-y^{0}\right)=y^{t-1}-y^{0}-Eexty\left(x^{t-1},y^{t-1}\right) \quad \text{(Equation 2.12)}$$

where t is the iteration number and can be seen as a time component, I is the identity matrix of same size as A, $x0$ and $y0$ are coordinates of points of the original line, xt and yt are new positions of these points and xt-1, yt-1 are coordinate of last displacement. In equation 2.11 and 2.12, only xt and yt are unknown value. Therefore, these equations can be solved.

2.3.4. Displacement of a set of lines

Bader (2001) extended the model by considering correction of distance conflicts within a set of lines such as a road network. Conflicts between segments of the same line are treated and junctions between lines are preserved. The internal energy is still defined by the line displacement. Given a set of lines L1, L2, …, Ln with Li and Lj two adjacent lines as shown on figure 2.2, the external force applied to a point P of Lj is defined as follows:

$$\boldsymbol{f}_P^{(t)} = \sum_{\substack{i=1,\dots,n \\ i\neq j}} \begin{cases} \frac{\boldsymbol{v}_i}{|\boldsymbol{v}_i|}\left(r_{ij}-\min(|\boldsymbol{v}_i|, r_{ij})\right), & \text{if } (P \rightsquigarrow P+\boldsymbol{v}_i > 2r_{ij}) \\ \boldsymbol{0}, & \text{otherwise} \end{cases}$$

Where r_{ij} is the required minimum distance between Li and Lj and v_i is the vector from point P on Lj to the closest point on Li. $\frac{v_i}{|v_i|}$ is the unit vector of same direction as v_i.

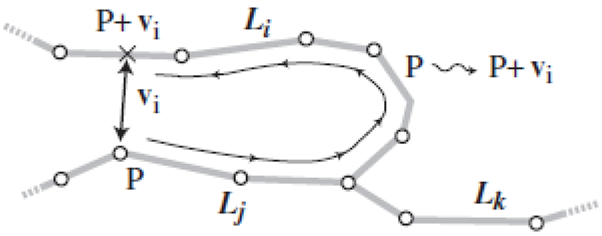


Figure 2.2 Terms used for the definition of external energy (Bader, 2001)

Junctions are preserved either by fixing the points or by fixing the directions at the junction. The system can be solved either by solving a system for each line successively, meaning that lines are displaced in a sequence or by building a global system where all the lines are modified at the same time. The second approach is much more demanding in terms of computation but provides a global solution while the first approach acts by solving local conflicts separately and the result may depend on the order in which lines are processed. Bader (2001) makes use of a finite element method to solve the system.

2.3.5. Snakes for line smoothing

Burghardt (2005) introduced snake model for line smoothing. In smoothing, the internal energy is used for simplify lines. The difference between displacement and smoothing in the snake model is that in the first case, the snake is the displacement while in smoothing, it is the line itself. As smoothing is applied to a single line with no consideration for other objects, no external energy is defined. Taking the same notations as in section 2.3.1, the following matrix system is solved:

$$(A+\gamma I)x^t = \gamma x^{t-1}$$

$$(A+\gamma I)y^t = \gamma y^{t-1}$$

In order to adapt the algorithm to the shape of the line, Burghardt (2005) also segments the line in segments of similar sinuosity and applies α and β coefficients which are related to the sinuosity of each segment.

2.3.6. Snakes for isobath generalization

Guilbert et al. (2006) proposed a snake model for displacement which respects the safety constraint. Isobaths are modeled with cubic B-spline curves, providing an intrinsically smooth representation. Each curve is a parametric curve $f(t)$ interpolating a given set of points Pi such that Pi = $f(\zeta i)$ where ζi are a set of parameters. The second derivative is replaced by the curvature defined as

$$\kappa(t) = \frac{\det(d'(t), d''(t))}{\left\| d'(t) \right\|^3}$$

where $d(t)$ is the line displacement and det is the determinant.

As a result, the internal energy is described as

$$E_{\mathrm{int}} = \frac{1}{2}\left(\alpha(t)\left| d'(t) \right|^2 + \beta(t)\left| \kappa(t) \right|^2 \right) .$$

If two lines are in conflict, only the deeper line is displaced by pushing it away from the other line. Computing the external energy first requires the computation of the minimum displacement $d_{\min}$ to apply to the deeper line. The external energy depends of the amount of deformation required to reach this minimum displacement and is given by:

$$E_{ext}(d(\zeta_j^1)) = \begin{cases} \dfrac{\left\| d_{\min}(\zeta_j^1) - d(\zeta_j^1) \right\|^2}{\varepsilon_{vis}^2} & \text{if } d(\zeta_j^1) < d_{\min}(\zeta_j^1) \\ 0 & \text{otherwise} \end{cases}$$

The coefficient $\varepsilon_{\mathrm{vis}}$ is the minimum distance to satisfy the legibility constraint. If no conflict occurs, both internal and external energies are zero, and the curve is not modified; else, an external energy is added with respect to $d_{\min}$ and the snake is modified until the displacement is greater than $d_{\min}$ (Guilbert et al., 2006). The system of equations is solved by a gradient descent algorithm. The method avoids linearizing the curvature however it is computationally more intensive and, although spline curves provide a better representation, spline interpolation is also intensive and less robust than a polygonal line representation.

Guilbert and Lin (2006) proposed a smoothing technique based on snakes that satisfy the safety constraint. The algorithm also applies to B-spline curves and as in (Burghardt, 2005), the snake is the line itself. The safety constraint imposes that the line is smoothed by pushing it towards one side (Figure 2.1). This is achieved first by defining an external energy that applies to any point falling on the wrong side and second by fixing some critical points of the line which act as attractors.

The external energy is defined by equation 2.3 where $d(P, L)$ is the distance between point P of the line and L, the original line position. Constant ε_{dist} corresponds to the minimal distance that defines visual intersections and is used here to normalize the energies.

$$E_{ext}(P) = \begin{cases} \frac{d(P,L)^2}{\varepsilon_{dist}^2} & \text{if } P \text{ is on the wrong side} \\ 0 & \text{otherwise} \end{cases}$$

(Equation 2.3)

Minimizing the snake energy means that a compromise solution may be found between the internal and the external energy: if the curve is smoothed enough, a minimum solution can be reached with a non-zero external energy and the safety constraint is not respected. The problem is solved by fixing some critical points which are leaning toward the deeper side so that the smoothing forces the points to move toward these critical points to guaranty they are moving to the right direction.

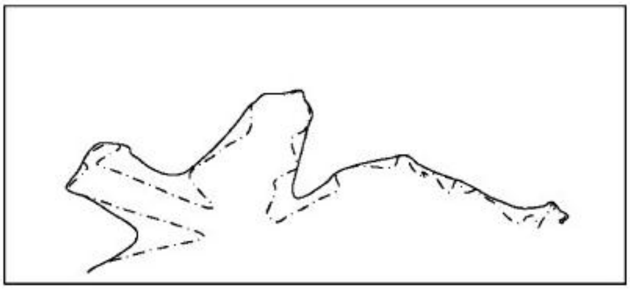


Figure 2.1 smoothing by keeping the line on one side (Guilbert and Saux, 2008). Dashed line: original line. Plain line: smoothed line.

Guilbert and Saux (2008) extended the method to combine both smoothing and displacement. In order to take into account the distance conflicts created by neighboring lines, the line L of equation 2.3 is defined by a boundary line which delineates the admissible area where the isobath must be. The method also takes into account distance conflicts that can occur between segments of a same curve. Depending on the orientation of the displacement, such conflict is automatically corrected by smoothing the curve or the curve is severed at the bottleneck. The method is applied to the generalization of a set of isobaths. Isobaths are generalized starting from the shallow ones towards the deeper ones so that displacements can be propagated. One limitation of the method is that the user has to indicate for each line where the deeper side is and which operator should be applied.

Only the methods in this last section are applicable to isobath generalization as others cannot respect the safety constraint. The last technique was applied to sets of isobaths successfully however it still suffers limitations for automatic processing. The user has to select the operator (deletion, aggregation, smoothing) and indicate which side is the safe side. Discrete operations cannot be achieved with this approach and a strategy where operators can be selected according to cartographic constraints including the safety and the terrain morphology shall be developed in order to move toward automation. As mentioned in the previous section, agent modeling allows integration of different operators. Therefore, the next section reviews and discusses generalization methods based on Multi-Agent Systems.

2.4. Generalization based on Multi Agent Systems (MAS)

2.4.1. Multi agent systems

Multi agent system (MAS) is a branch of Artificial Intelligent (AI) issued from Knowledge-based AI and Behavior AI (Schumacher, 2001). Environment and agent are two components in MAS. The environment describes the situation of all agents and other objects in system. Slightly different definitions of agents can be found in the literature. Weiss (1992, p2) defines agents as "*autonomous, computational entities that can be viewed as perceiving their environment through sensors and acting upon their environment through effectors*". In (Beer, 1995, 72:173-215), an agent is "*An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future*".

From both definitions, one important characteristic of an agent is its autonomy. An agent receives information from other related agents for example neighboring agents. An agent has a subjective view (Schumacher, 2001). It controls itself to choose its actions depending on the received information but cannot control other agents through its actions (Schumacher, 2001).

The relationship between intelligent agents and their environment is shown in Figure 2.3. The agent first receives effects of other agents via a sensor. Then it apperceives the situation of surrounding agents for judgment. State is a part of function which can record the situation of agents. This part is used for choosing the best result. Then control algorithm part tries from an action list after it senses the environment and chooses appropriate actions with comparing the result in state part. Finally, it sends its situation to other agents via effectors (Weiss, 1999).

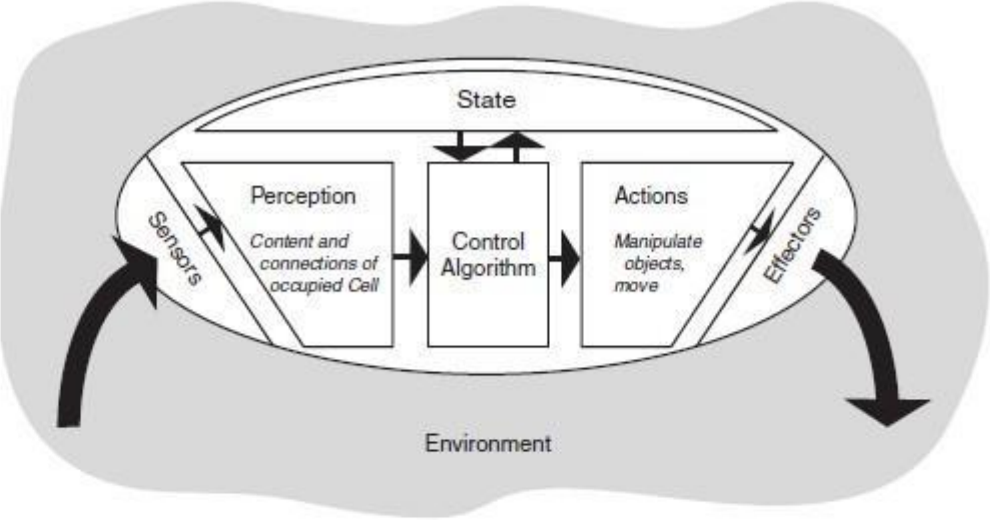


Figure 2.3 Architecture of an intelligent agent (Schumacher, 2001)

2.4.2. Life cycle and level of perception of an agent

MAS was first applied to generalization by Baeijs et al. (1996) who introduced the SIGMA model (SemI automated Generalization using Multi-Agent system). The system is used to generalizing river, road and building. In this system, they designed a group of scopes to control agents' actions. First is perception scope which is suit for all kind of agents for perceiving the environment. Second is class scope. This scope contains all agents belong to the same geographical class. The corresponding identical action scope is used for changing dynamically the current legend. The third scope is object scope which classifies agents into different group with their geographical object. Its corresponding action scope is used to calculate the proportional following. The proximity scope contains all the agents which distance is closed to the resolution which is defined by user. The corresponding action scope is performing the interactions between agents. The last scope is group scope. This scope groups agents with same ovjective. The corresponding action scope is used to maintain the groups.

They used reactive agents which do not contain a state model to evaluate the result and react when conditions are triggered. They cannot judge which actions are better.

Ruas (1999) proposed a more intelligent system for generalization by introducing the agent life cycle to coordinate constraints and actions and divided agents into three levels to coordinate their relationships. Objects on a map can be organized as single objects, as belonging to one group of objects or as several groups of objects. Operators such as selective omission, displacement processed on one object may need to analyze the situation of an object within a group of objects. It requires a group of objects to control itself and coordinates the objects within.

The three levels of agents are the macro, meso and micro levels (Ruas, 1999). Macro agents contain a population of objects with different objects for example the one macro agent is road and the other is houses. Meso agents are groups of objects such as city blocks. Micro agents are single objects such as buildings and roads. Macro agents contain meso agents and meso agents contain micro agents. For example, the set of all roads form a macro agent, all buildings form another macro agent, a group of buildings forms a meso agent and one building is a micro agent.

The life cycle of agent was introduced by Duchêne et al in 2010.

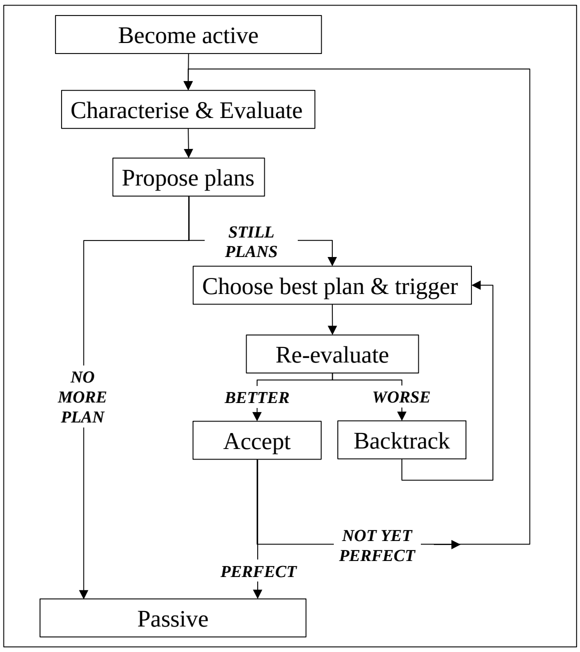


Figure 2.4 Generic life cycle of an agent (Duchêne et al, 2001)

Figure 2.4 shows the basic rule of behavior for agent. The agents evaluate the environment first. Then make one or several plans for its actions. Then the system will re evaluate the environment with the changed situation and choose the best plan for resolving problem. This rule does not describe the relationship between different level agents (meso agent and micro agent). Duchêne et al (2001) also gave the life cycle of a meso agent in their paper (figure 2.5).

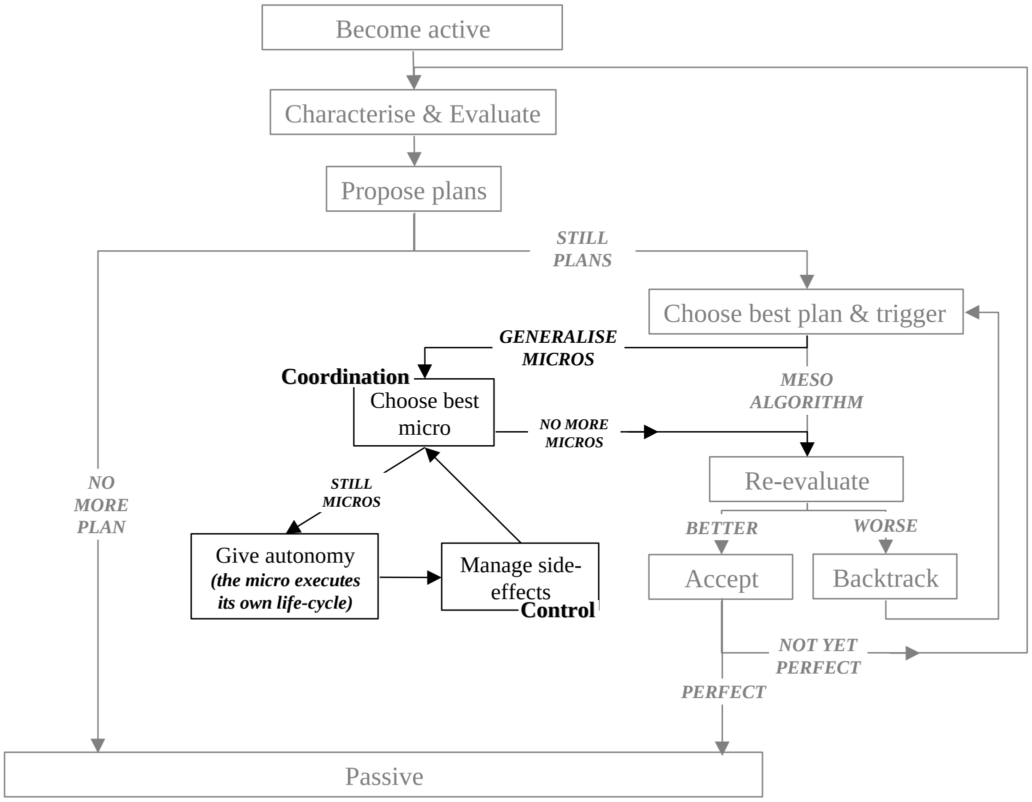


Figure 2.5 Life cycle of meso agent (Duchêne et al, 2001)

The relationship between macro and meso agent is similar with meso and micro agent. Therefore, the rule of macro agent resembles that of meso agents.

2.4.3. Evaluation of an agent action

Galanda (2003) proposed an evaluation method to evaluate constraint violations of an agent. When performing a plan, constraints are evaluated and a score is given according to the importance of the constraint and the violation. The sum of violations is computed for the plan. The smaller the sum, the better the plan. If the violation is larger than the current score, the situation is worse and the plan is rejected. The agent record sum of violations for all plans and chooses the one with smallest value.

Figure 2.6 illustrates an example of this method (Galanda, 2003). State0 is the initial situation of the agent (a group of light gray objects). From state3 in figure 2.6, the violation of constraint C is large and generalization is required. Agent makes plans state1 to state4 for satisfying constraints. After comparison, state4 has the minimum sum violation. Therefore the agent will choose this plan.

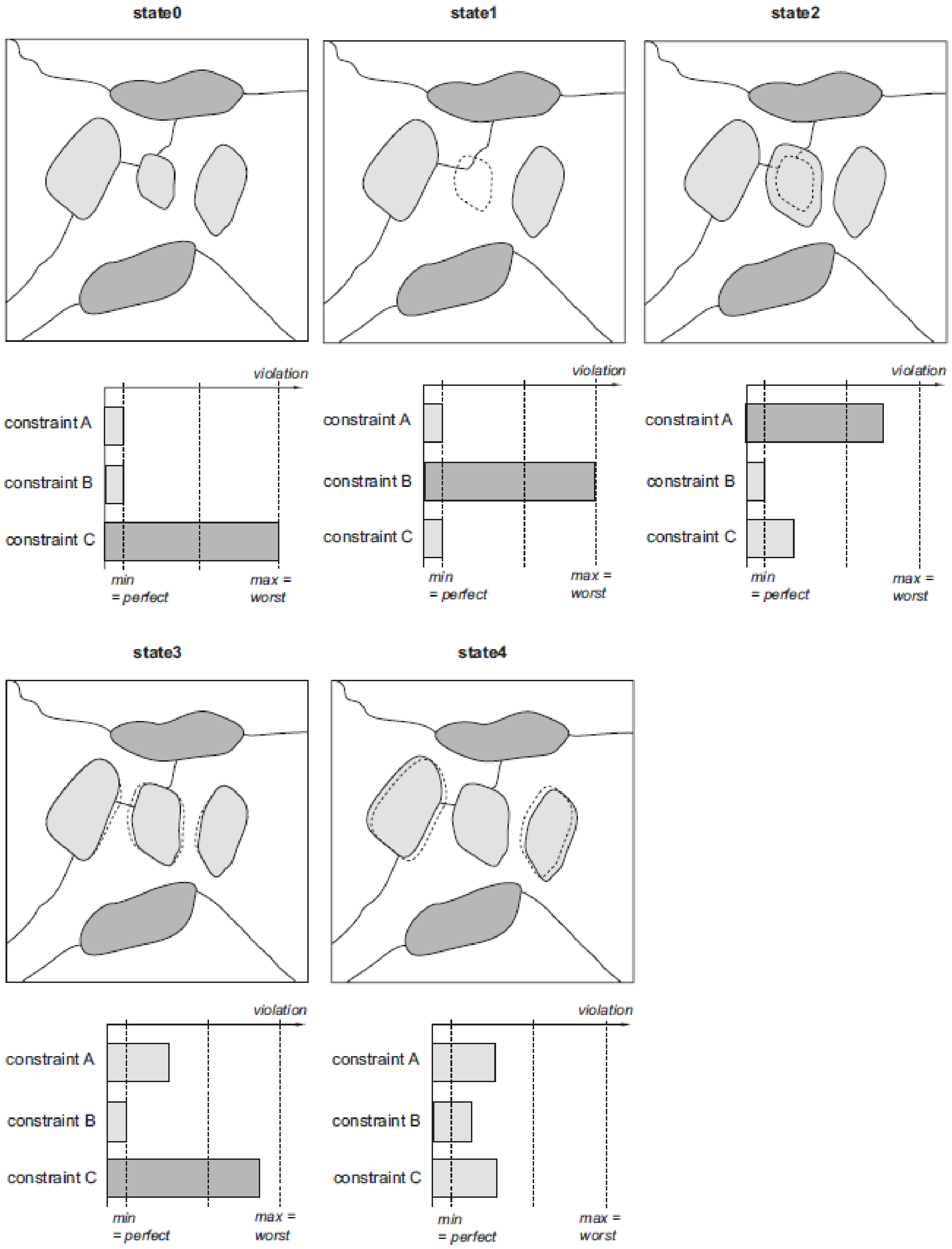


Figure 2.6 Different states and corresponding constraint satisfaction (Galanda, 2003)

The advantage of this method is that it can quantify the effect of each constraint so that the agent can find the balance between different constraints. For example, the legibility constraint tends to deform or move objects while the structural constraint tends to preserve their shape. The effects of these two constraints on an object are contradictory. Which constraint is more important depends on the application. This method allows the system to evaluate the plan by different environment.

The problem of this model of life cycle is that this method does not solve the problem if no plan is accepted. The plans made by an agent is divided into better (accept) and worse (backtrack). Worse plans are not considered. However, if all the plans are worse, how the agent decides its actions is not described.

Ruas and Duchêne (2007) proposed the CartACom model where they defined Relational Constraint and Speech Acts in agent communication. The relational constraint describes the relationship between two agents. Generally, one kind of agent has its own constraints. But the relational constraint belongs to both related agents. The relational constraint is defined in two parts. First part describes the state aimed by both agents (rhombus in Figure 2.7). Second part is "analysis and management" part which describes constraints separately for each constraint (squares in Figure 2.6). This part links with the first part constraint and the agent.

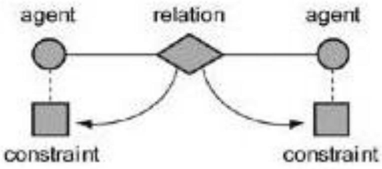


Figure 2.7 relational constraint between two agents (Ruas and Duchêne, 2007)

Speech Acts allow agents to "ask" other agents to act as described by Duchêne (2003). She separates communication between agents into two types:

request for action: an agent asks another one to perform an action; this one either accepts and performs the action, or refuses;

information transmission: an agent informs another one of a fact.

In such an approach, if an agent cannot perform a plan to solve a conflict, it can require another agent to perform an action or pass information for the agent to solve the conflict.

2.4.4. MAS for terrain modeling

Gaffuri (2006) also used several levels of agents to satisfy generalization problem. This project is interesting because it focuses on a terrain model which is similar with our project. In this model, Gaffuri used a digital terrain model (DTM) to represent surface information. He decomposed the DTM as points, segments, angles and triangles as sub-micro agents.

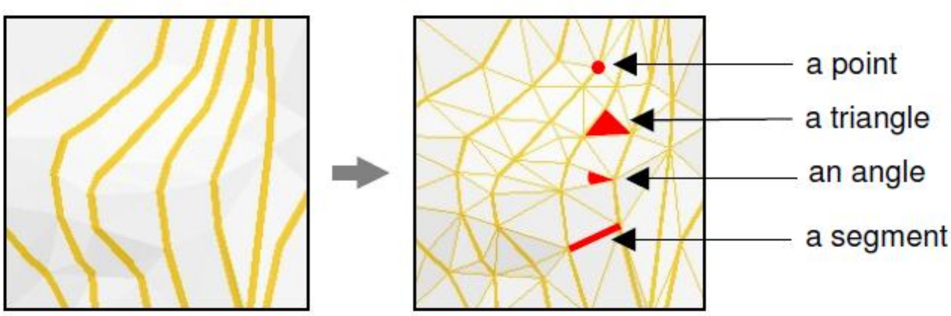


Figure 2.8 Decomposition of DTM to sub-micro agents (Gaffuri, 2006)

Gaffuri designed elastic constraints, meaning these constraints can be modified by different applications, to move and maintain the agents. Two groups of constraints are proposed. First group of constraints is used to maintain the position and shape of agents. Gaffuri listed 8 constraints in this group, for example, point position preservation and triangle area preservation (Figure 2.9).

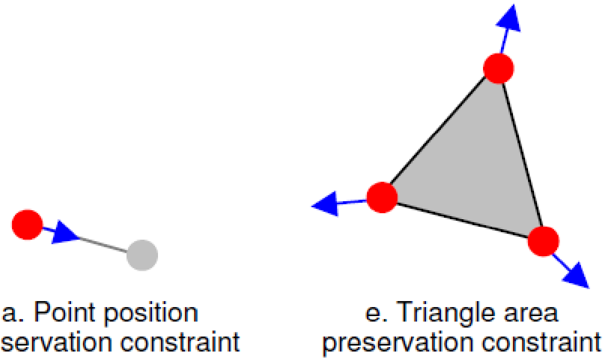


Figure 2.9 Example of preservation constraint (Gaffuri, 2006). The red points present current state. The blue arrow shows the constraint affected on points. They point to the initial state (grey points) to preserve the original shape of map.

The other group of constraints is added on different agents. The purpose of them is to move the agents. This kind of constraints influences agents as separated forces. For example, the point's minimum distance constraint maintains a minimum distance between two point agents. If two points are too close, this constraint will force them to move into different directions.

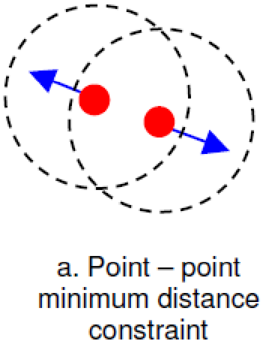


Figure 2.10 Example of movement constraint (Gaffuri, 2006)

That means, before using this model for processing contour line, we should create triangulated irregular networks (TIN). It is interesting that how this model performs without using TIN structure. The idea of decomposed can be used in our project. Instead of decomposing TIN, we can decompose the line into point and segment for control the shape of line and solve some problem comes from snake model such as self-intersection.

2.5. Summary

Automation of the isobath generalization process requires the definition of appropriate generalization operators and the definition of a strategy for the selection and application of these operators with respect to the chart constraints. In this chapter, the first section introduced a list of common operations performed in cartographic generalization and presented more specifically the constraints to be observed when generalizing isobaths on a nautical chart. These constraints are of four types. The first two, legibility and safety, are strong constraints that cannot be violated while the structural constraint, a weaker constraint, is used to evaluate how much of the terrain morphology has been preserved during the process. The last procedural constraint establishes procedures to follow for proper application of operations. Expressing generalization rules and conflicts as constraints, generalization is performed by finding a balance minimizing constraint violations. Main techniques used are continuous optimization modeling, including the snake model, performing continuous deformations and agent modeling which allow combination of different types of operations.

Section 2.3 introduced existing works done on line generalization based on the snake model. They include smoothing and displacement operations. Among them, works by Guilbert et al. (2006) and Guilbert and Saux (2008) are applicable to isobaths as they can take into consideration both legibility and safety constraints while earlier works only consider legibility. However they apply to individual lines and cannot be used for processing a set of line without interaction from the user due to the lack of topologic information. Bader (2001) presents a method that generalizes a set of lines by displacement. Although structural and legibility constraints are considered, its interest is limited as isobath generalization requires the application of other operators.

Section 2.4 reviews several agent models defined for cartographic generalization. In a MAS model, each cartographic object is considered an agent which can evaluate its situation in a given environment and perform its own actions. In the models presented by Ruas (1999) and Duchêne et al. (2001), agents are organized at different levels and can exchange information. Agents can establish plans of action according to their situation, evaluate each of them and apply the most appropriate. More specifically, MAS was applied to terrain generalization by Gaffuri (2006) where TIN elements were micro agents under spatial constraints.

Current work on isobath generalization is limited to the definition of smoothing and displacement operations. Due to the safety constraints, other work related to topographic maps is not transposable to nautical charts. A main reason is that selection and application of operators depends of the type of feature portrayed on the chart. Taking into account information about submarine features means that a first step is required to identify the features from the set of isobaths and integrate them in the generalization process. As a feature is portrayed not by a single isobath but by a group of isobaths, the semantic information related to a feature may not be held by each isobath of the feature but by a feature object formed by a group of isobaths. In such an approach, generalizing the seafloor would be achieved by generalizing the features. As features are processed not only according to their shape but mostly according to their relevance to the navigator, such an approach would be more relevant to model generalization rather than cartographic generalization and can be achieved by defining a new generalization process and operations that apply to features. These operations can involve discrete and continuous operations that are ultimately applied to isobaths. Therefore, the process can logically be modeled by combining a MAS with a snake model. Such model is introduced in the next chapter.

3. A new method for generalizing undersea features

3.1. Introduction

This chapter presents a new approach for isobath generalization. The objective is to generalize isobaths according to the topographic meaning they carry on the map so that the safety constraint can be integrated in view of an automated processing. A single isobath does not hold much topographic information on its own as it is merely a line connecting points of the same depth. The navigator can interpret the topography by looking at the depths and spatial relationships within a group of isobaths: the difference of depth and the distance between adjacent isobaths provide information about the direction of a slope and its steepness; a closed isobath contained by a deeper isobath models a peak. Therefore the generalization process is feature-driven meaning that isobaths are generalized by defining and applying operators to the features.

By focusing on feature generalization, the approach developed in this chapter belongs to model generalization as isobaths are selected or modified according to their topographic meaning and relevance for navigation. Features are mostly useful to express constraints related to safety and terrain morphology. Although legibility conflicts are not taken into account in model generalization, the legibility constraint is still considered to avoid creating new conflicts during the process.

The first section of this chapter presents the principle of the feature-driven approach. Undersea features are defined from the set of isobaths and the feature tree, a data structure storing topological relationships between features is introduced. Then operators required for feature generalization are discussed.
Second section presents the deformation model based on the snake model defining the

generalization operations. Constraints are expressed as energy terms that compose the external energy of the snake.

Third section presents some experimental results when applying the new algorithms on sample data. Influence of shape parameters of the snake is assessed and possible changes to the feature tree resulting from the operations are described.

3.2. Principle of the method

3.2.1. Identification of undersea features

According to the International Hydrographic Organization (IHO, 2008), an undersea feature is "*a part of the ocean floor or seabed that has measurable relief or is delimited by relief*". Given a set of isobaths modeling the seafloor, remarkable undersea features are formed by groups of isobaths that are higher (peaks) or lower (pits) than their surrounding.

As shown on Figure 3.1, a feature is a peak if it is formed by a set of isobaths where the isobath on the boundary is deeper than all isobaths inside. Respectively, a pit is a feature formed by a set of isobaths where the isobath on the boundary is shallower than the inner isobaths. Such a definition has the benefit of characterizing features at different levels of representation. For example, features shown in white on the contour map of Figure 3.1 can obviously be defined as three peaks (features C, E and F) and one pit (feature G). In the same way, the feature B in dark grey on the contour map also forms a feature and the contour at its base is lower than all the contours inside, meaning that the terrain within the boundary contour of B is higher than the terrain outside B. Therefore, B is also a peak.

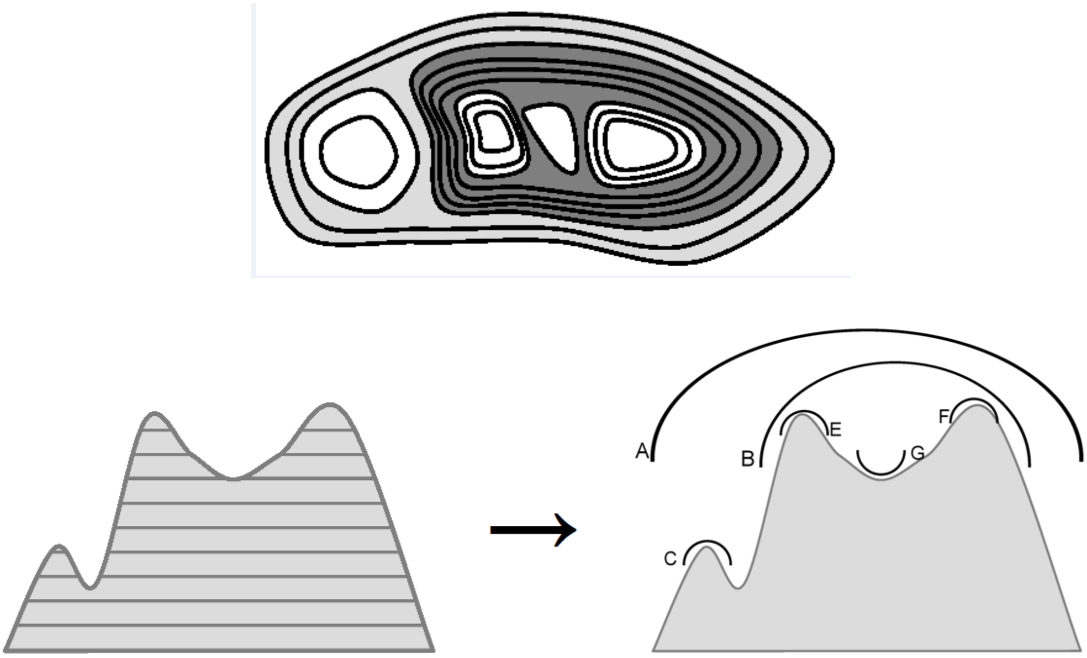


Figure 3.1 peak and pit in feature

3.2.2. The feature tree

Morphometric features from a contour map are usually extracted by building the contour tree based on inclusion relationships between the contours (Cronin, 1995), the largest contour containing all other contours being the root of the tree (right tree structure in Figure 3.2). Peaks and pits can be identified from the contour tree (Kweon and Kanade, 1995) but only features containing leaves of the contour tree are considered as shown in Figure 3.2. Representation of features at different levels as in Figure 3.1 is not possible and feature attributes are not stored in the tree. The feature tree structure is defined similarly to the contour tree. A feature is defined by a contour, the feature boundary, and all the contours contained by this contour. Each node of the feature tree can therefore store attributes about the feature and a contour can belong to several features.

Each contour does not define a feature boundary. In (Kweon, 1995), a branch forming a peak or a pit is defined by the largest series of contours having no more than one descendant. In the feature tree, a feature is always defined by the largest possible boundary contour. For example, in Figure 3.3, features E and F contain 3 contours while feature G contains only one contour which is also its boundary. The algorithm also checks for the existence of changes of slope in a branch of contours. These cases occur in volcano type features where a branch of contours defines a pit contained by a peak. The elevation of a feature can also be estimated from the feature tree by the elevation difference between the highest and lowest contour.

The feature tree also provides a multiple level representation. In Figure 3.3, the peak A contains two peaks B and C and peak B contains two peaks E and F and one pit G. A third kind of feature may appear in the feature tree which is a mixed feature, that is, a feature whose boundary contour is neither the highest or the lowest. Such feature always contains some descendants which can be described as peaks and pits.

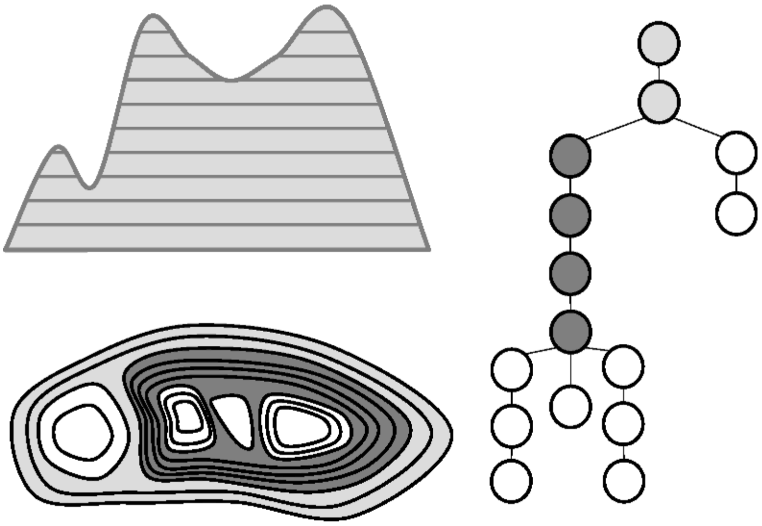


Figure 3.2: Contour tree corresponding to a contour map. Branches with white nodes are peaks or pits.

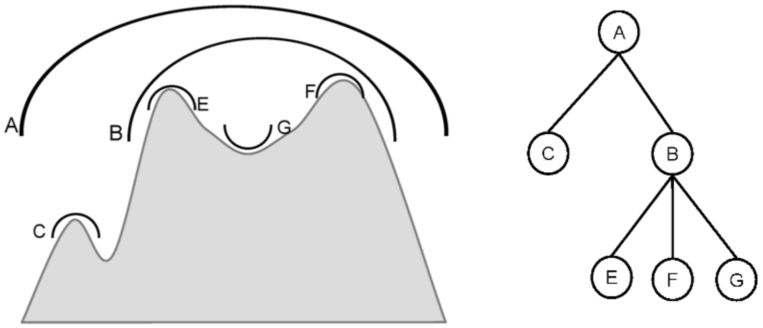


Figure 3.3: Feature representation and its corresponding feature tree built from Figure 3.2 contour tree.

The feature tree is built from the contour tree by creating an initial feature tree with as many features as isobaths. The feature tree is simplified by deleting spurious features. A feature is spurious if it is a single child and there is no change of slope between the boundary contour and its parent. The algorithm constructing the feature tree from a contour tree is given below.

```
Input: a contour tree C
Output: a feature tree F
Begin
  // create a feature for each contour
  For each contour c of C
    // create a feature c whose boundary is c
    Feature f = new Feature(c)
    featureList.pushback(f)
  End For
  // create the feature tree from the list of features
  Feature Tree F = new FeatureTree(featureList)
  // remove spurious features
  For each feature f of F
    If f.brothers is empty then
      c=f.boundary
      If c.depth!=c.parent.depth
        If not (c.child.size==land c.depth!=c.child(1).depth)
          F.remove(f)
        End If
      End If
    End If
  End For
```

```
End
```

The last step consists in classifying features into pits and peaks. This is done by comparing the boundary contour with its descendants. If the boundary is deeper than all its descendants, the feature is a peak; if the boundary is shallower than all its descendants, the feature is a pit; otherwise, the feature is of unknown type.

3.2.3. Feature generalization operators

Considering undersea features as map objects, specific generalization operations can be introduced. Among the list of operators presented in Figure 2.1, only those that can be applied with respect to nautical chart constraints are applicable. For example, displacing a feature is not applicable as parts of the terrain would appear deeper than they really are. The following operations can be performed on features:
Aggregation of two features forming a larger feature;
Enlargement of a feature
Selection/deletion
Smoothing by smoothing or displacing inner isobaths
Deformation of the boundary to correct legibility conflicts with neighboring features

All operations do not apply to both types of features. According to safety constraints, peaks cannot be removed or reduced while pits cannot be aggregated as the area between the pits would become deeper. Among these operations, smoothing and deformation are applied for correction of legibility conflicts and do not impact the meaning of the features. As the focus of this thesis is on object generalization, only the three other operations are considered. Enlargement is relevant to cartographic generalization but need to be applied on peaks which are selected but are not large enough to appear on the chart and the algorithm is required to aggregate isobaths.
Table 3.1 summarizes the operators and the features on which they apply.

Table 3.1 operators for different types of features

Feature type	Operators
Pit feature	Deletion
Peak feature	Enlargement
	aggregation

The deletion operator removes too small pits from the map. The minimum area is defined by a threshold value ε_{area}. The value is defined by the area of the boundary isobath. Deleting a feature means that the terrain element it portrays is removed and so all the contours forming the feature are deleted (Figure 3.4). If the feature contained over features, these features are also deleted.

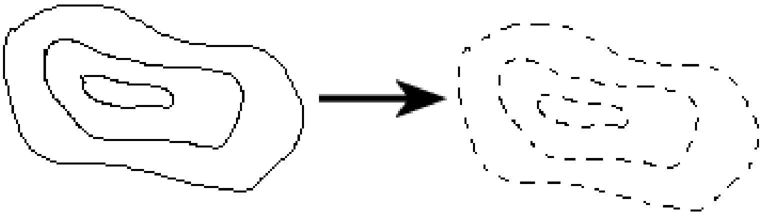


Figure 3.4 Feature deletion

The enlargement operator is applied to peaks whose area is smaller than ε_{area}. As the objective is to deal with the feature area and not with legibility conflicts between isobaths inside the feature, only the boundary isobaths is enlarged. Enlargement will be performed by using the snake model.

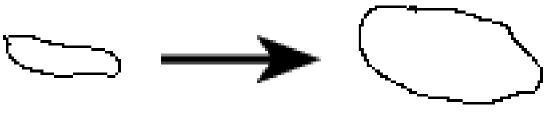


Figure 3.5 enlargement operator

The aggregation operator aggregates two peaks into one peak. It is performed by merging the boundary isobaths of each feature, which must be at the same depth. The result is a new peak whose boundary isobaths is the merged isobath. The method for aggregation is separated into two steps. First, one boundary is enlarged towards the other one. After two features overlap, the boundaries are merged into one curve. The first step is similar to the enlargement operator and based on the same snake model. The difference lies only in the definition of the external energy term reflecting the different constraints.

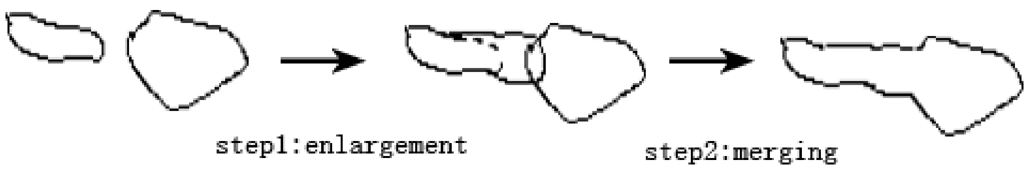


Figure 3.6 Aggregation

Because the deletion and aggregation remove or modify features, the structures of feature tree and the contour tree are modified. After aggregation, the new feature is added to the feature tree while a feature that was aggregated is not removed from the tree if its boundary contour has a single descendant with no change of slope. In that case, the descendant becomes the new boundary contour. Features that were modified or added are checked against their parents and descendants in the feature tree. If a feature is redundant because it is a single child of the same type as its parent, it is removed from the feature tree.

For example, in Figure 3.7, features B and C of Figure 3.3 are aggregated resulting in a new crosshatched feature. Features B and C are not removed from the feature tree but are redefined with a new boundary contour. The newly created feature is not added to the feature tree because it is the only child of A. Therefore, the feature tree appears unmodified but the definition of A, B and C is modified as the contour tree was updated. If features B and C are aggregated once more, the feature tree is modified (Figure 3.8): feature C disappears as there is no descendant to the boundary contour and feature B, being a peak like feature A, is also removed from the tree.

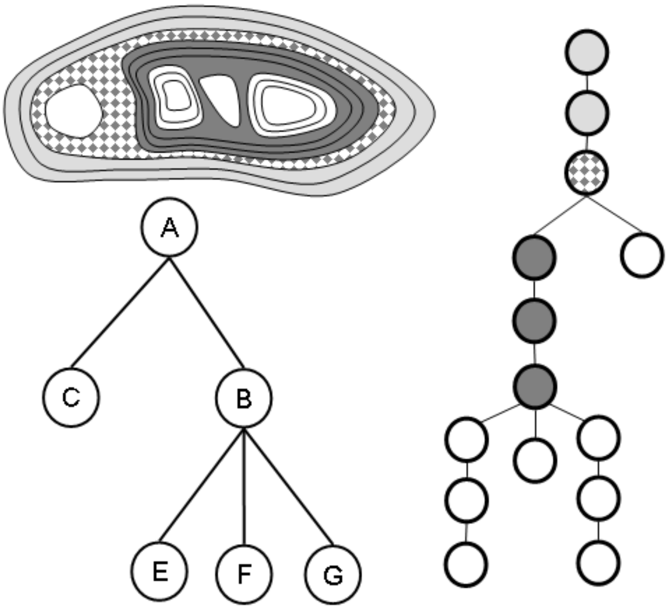


Figure 3.7 Aggregation of features B and C without feature removal

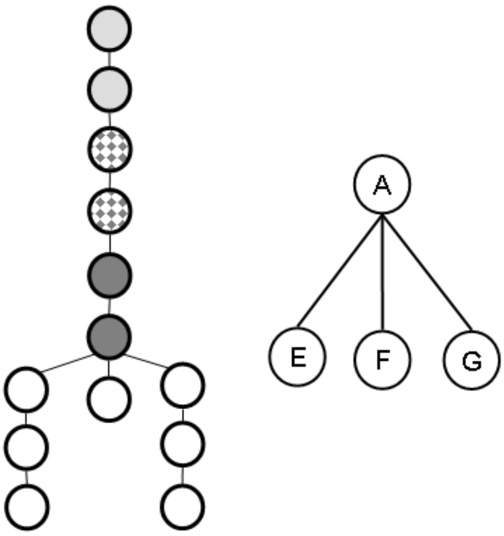


Figure 3.8 Aggregation of features B and C with removal of the features

3.3. Definition of generalization operators based on a snake model

As mentioned above, both enlargement and aggregation are based on the snake model. This section introduces the definition of the model including the internal and external energy for both operators. Because both two operators are moving lines, the definition of the internal energy is similar to Burghardt and Meier (1997) and corresponds to the line displacement.

3.3.1. Enlargement

Enlargement is performed when the area of a peak is too small. The operation must guaranty a minimal area but must guaranty the feature is not too narrow to contain a sounding. Therefore, enlarging the feature is not done simply by dilating the feature in all directions (Figure 3.9 bottom) but by expanding the feature in too narrow areas (Figure 3.9, top).

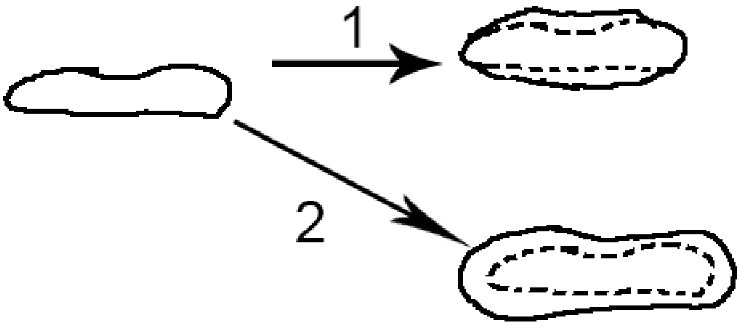


Figure 3.9 way to enlargement

The idea is therefore to define the external energy according to the distance between each point of the curve and the sounding. As in this project, soundings are not available, the center of the feature is considered. If the distance is big enough, then the external energy is equal to 0 otherwise the energy is the squared difference between the tolerance and the actual distance. If the feature is contained by a larger feature, enlargement may create distance conflict with the parent isobaths of the boundary such as figure 3.10 shows. As a result, it adds a condition to control the external energy if the feature is close to its parent feature to avoid the overlap situation.

The detail is shown as:

$$Eext = \begin{cases} \left\| distTolerance - d(f(s),c) \right\|^2 & d(f(s),c) < distTolerance \\ 0 & \text{when} \quad d(f(s),c) \geq distTolerance \\ 0 & d(f(s),p) < distToleranceToParent \end{cases}$$

where distTolerance is given tolerance. d(f(s),c) is the distance from point f(s) to curve center. The d(f(s),p) is the minimum distance between the point to parent curve, distToleranceForParent is a tolerance given by the user. The idea is if the distance from moving point to parent is smaller than a value, the energy will be 0 and will not push the point.

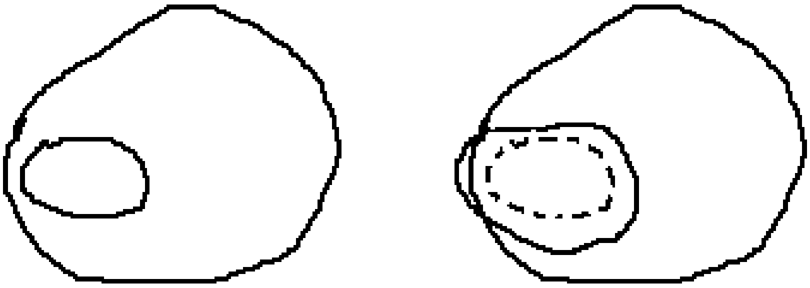


Figure 3.10 conflict with parent feature

3.3.2. Enlargement for merging

This operator is used when two peaks are close enough to be aggregated. The idea is to enlarge one feature until they overlap and then to merge their boundaries. The enlarging step uses the snake model. Because the objective is to make the features overlap, the external energy definition is different from the enlargement operation as one isobath is attracted by the other. Only a part of the isobath that is facing the other feature needs to be deformed. Therefore, points which are further than a given aggregation distance are fixed. The external energy applies to points for which the legibility constraint is not observed. The aggregation distance is larger than the legibility distance so that points can be moved to maintain the shape of the line. Same with enlargement, it also considers the overlapping with parent feature problem. The external energy is defined as follow:

$$Eext = \begin{cases} \left\| distTolerance - d(f(s), g) \right\|^2 & d(f(s), g) < distTolerance \\ 0 & \text{if} \quad d(f(s), g) \geq distTolerance \\ 0 & d(f(s), p) < distTolerance \end{cases}$$

where $d(f(s),g)$ is the shortest distance from this point to the curve g. $d(f(s),p)$ is the minimum distance between the point to parent curve.

Following this definition, the direction of displacement may violate the safety constraint: aggregating two features means that the space between the features is deeper than the space inside the features. If the force pushes a point inside the feature, it means that the point is moved towards a shallower position. For example, in Figure 3.11, p_0 is the original position of a point which moves to p_1 at the next iteration. The direction of the displacement points towards the inside of the feature and conflicts with the safety constraint. Therefore, if the force is directed towards the interior of the feature, i.e. the vector is oriented inside the angle formed by the segments joining at Pi, the point is locked and removed from the matrix system. As figure 3.12 shows, P1 and P2 are two points of P. The vector on P1 and P2 is the direction of the force. The displacement of P1 is valid but the displacement of P2 is not correct.

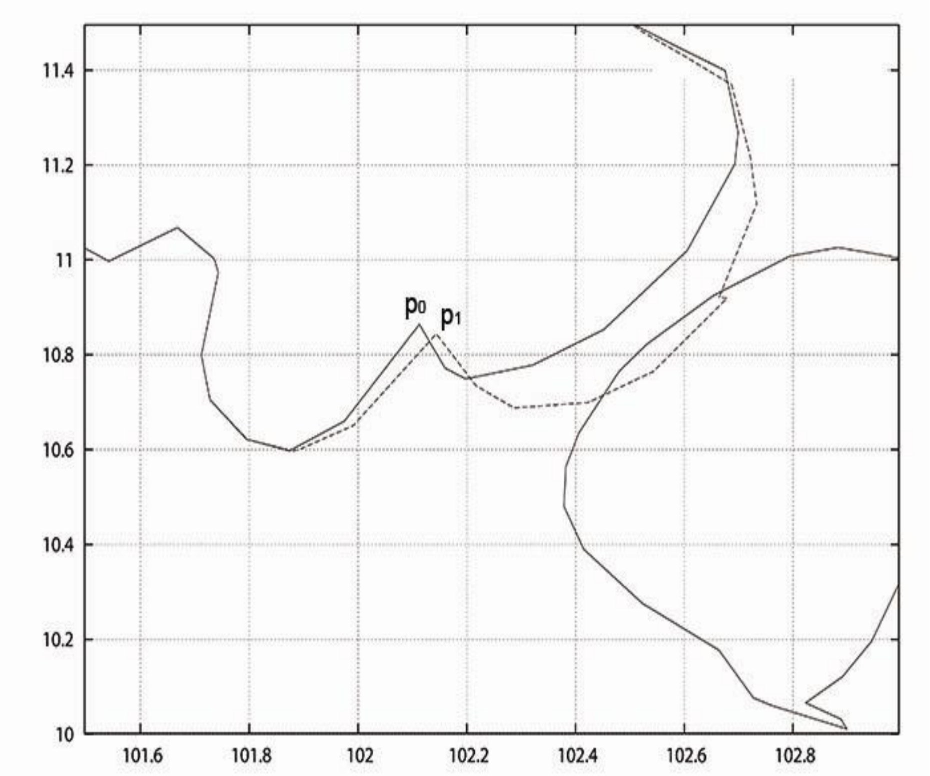


Figure 3.11 The displacement of p_0 to p_1 is not valid

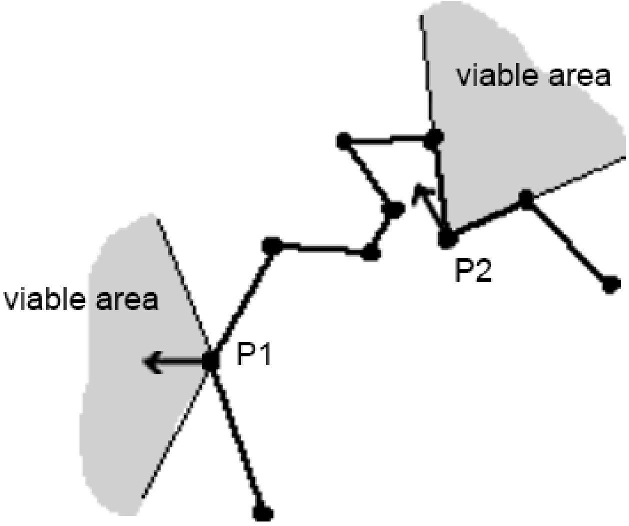


Figure 3.12 Displacement towards deeper side. P1 force is valid but P2 force is not valid. For each point of the isobath, the force vector must lie in the grey wedge formed by adjacent segments.

Features are merged when their boundaries overlap. If the overlapping area is small, the angles at the junctions may be too sharp and although the result fits the needs for feature selection, it is not aesthetically acceptable. One way to solve this problem is to set a threshold angle. If the junction angle is greater than this tolerance, they can be merged otherwise the process keeps iterating. However, in the snake model, controlling the junction angle is difficult and lacks of reliability as the junction angle depends on the shape of the curves. The other way is to consider that this issue is not relevant to model generalization but to cartographic generalization where the new isobath will be smoothed.

3.4. Experimental evaluation of the operators

This section shows some results for enlargement and aggregation operators. It will discuss the results of different values of parameters in the snake model. The system of equations defining the snake energy is discredited using finite differences as Burghardt and Meier (1997). For enlargement, the terrain deformation is quantified by the difference of curvature before and after the line transformed. For merging, the terrain deformation is quantified by the area difference between the resulting and original feature for enlargement and by the area difference between the aggregated feature and the original features for aggregation.

Table 3.2 quantify the displacement

	quantify the displacement
Enlargement	Area(enlarged curve) – area(original curve)
Aggregation	Area(merged curve) – area(original curve1) – area(original curve2)

3.4.1. Enlargement

This section discusses results obtained with the enlargement operator and more specifically discusses the influence of parameters α, β and γ. Parameters α and β are shape parameters, γ controls the step done for an iteration. Besides the snake parameters, other variables are the legibility constraints: minimum distance between two features and the minimum area for a feature. In this model, the area is controlled by the distance from the center of the feature and the distance to neighboring features including the parent is considered. These constraints are fixed as they refer to constraints which are common to all types of maps: the legibility distance corresponds to the minimum distance between two lines for the reader to distinguish them added with the thickness of the pen stroke and the area tolerance corresponds to the minimum space required to mark a sounding inside the feature. In the figures, the solid line is the original line and the dashed line is the result. Other lines are neighboring feature boundaries. The area tolerance is expressed by the minimum distance to the center of the feature. As soundings are not given, it assumes that the sounding is located at the center of the feature however this location can be replaced by the sounding position if available.

Two examples are shown in Figures 3.13 to 3.18. The feature in the upper part of the figures stands away from other features while the feature in the lower part is close to another isobath belonging to the parent feature, creating a distance conflict. The distance to center tolerance is set at 0.2 cm. The distance tolerance to the parent is equal to 0.05 cm. Results measured for different sets of parameters are summarized in Tables 3.3 and 3.4. Shape preservation is estimated by the curvature norm of the displacement. A smaller norm corresponds to a more homogeneous displacement of the curve and so a better preservation of the original shape.

A first test on figure 3.13 was done setting both shape parameters to 0. That means that deformation was done without considering the internal energy. In comparison with figure 3.14 where both shape parameters were non zero, it appears clearly that the feature is distorted and of a lower quality.

Figures 3.15 to 3.16 show results with different shape parameters. Setting a high value of α minimizes the tension, leading to a shorter curve while a higher value of β reduces the snake curvature and smoothes the displacement by propagating the deformation on a larger portion of the line, leading to a larger displacement. Best results are obtained when both parameters α and β are set to high values.

Finally, the impact of γ is shown in Figure 3.17 and Figure 3.18. Setting a small g means that deformations at each step are bigger. The process requires a smaller number of iteration to converge but the enlargement is more difficult to control as the line may jump to some other local minimum.

Generally, from table 3.3 and 3.4, setting higher parameters gives lower curvature value which means in this situation the curves are better preserved.

Table 3.3 Area difference for top feature

	Area (cm2)	Area diff (cm2)	Curvature
Original feature	0.0023	0	
α=20, β=20, γ=8	0.0534	0.0512	0.0579
α=20, β=0, γ=8	0.0548	0.0526	0.0658
α=0, β=20, γ=8	0.0686	0.0664	0.0729
α=0, β=0, γ=8	0.0703	0.0680	0.2080
α=20, β=20, γ=4	0.0817	0.0794	0.0713

Table 3.4 Area difference for bottom feature

	Area (cm2)	Area diff (cm2)	Curvature
Original feature	0.0079	0	
α=20, β=20, γ=8	0.0410	0.0331	0.0467
α=20, β=0, γ=8	0.0407	0.0328	0.0602
α=0, β=20, γ=8	0.0481	0.0402	0.0727
α=0, β=0, γ=8	0.0486	0.0407	0.1874
α=20, β=20, γ=4	0.0658	0.0579	0.0538

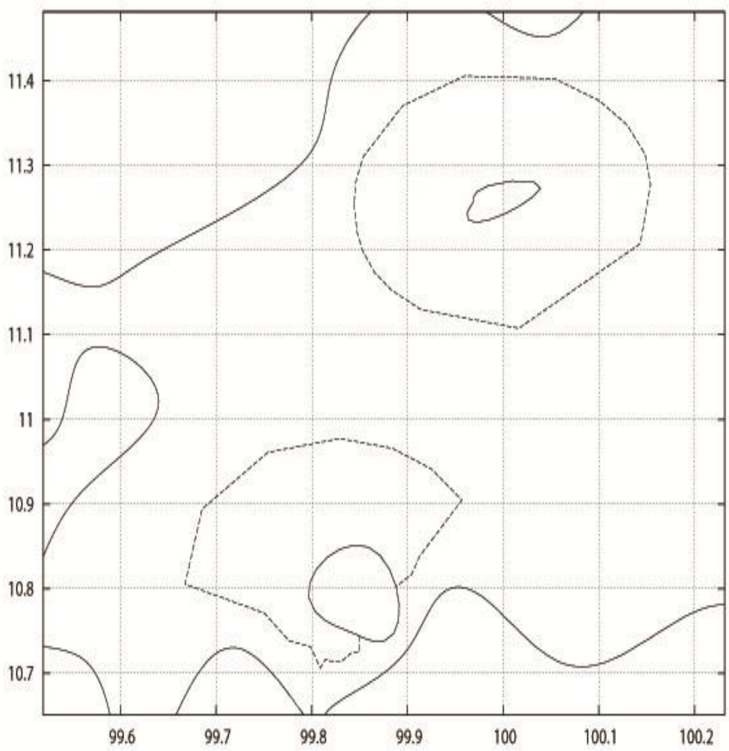


Figure 3.13 Enlargement: α=0, β=0, γ=8

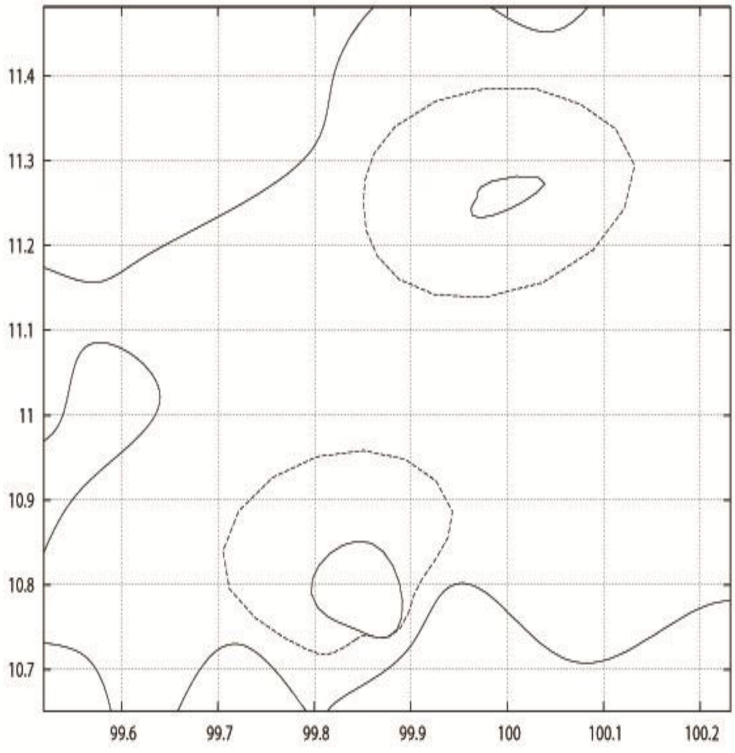


Figure 3.14 Enlargement: α=20, β=20, γ=8

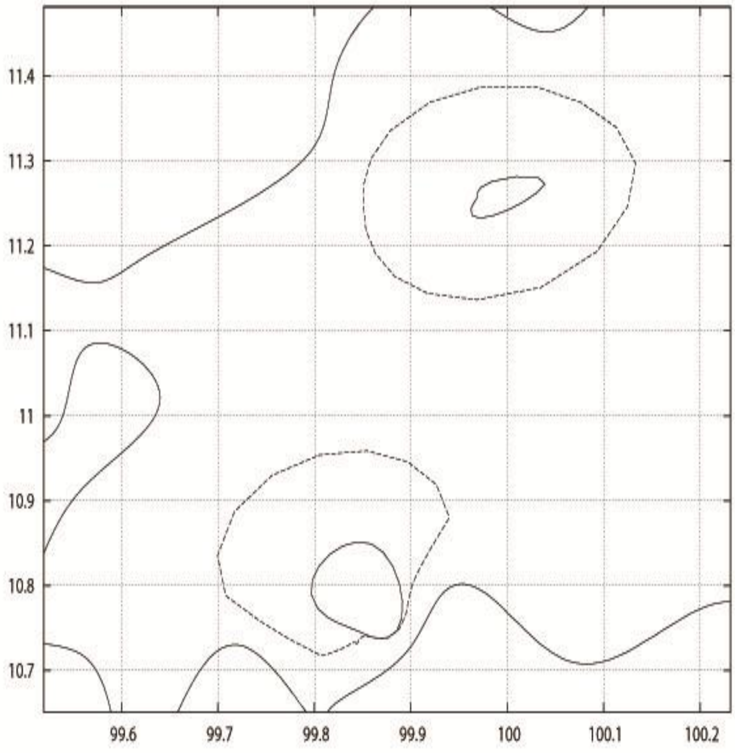


Figure 3.15 Enlargement: α=20, β=0, γ=8

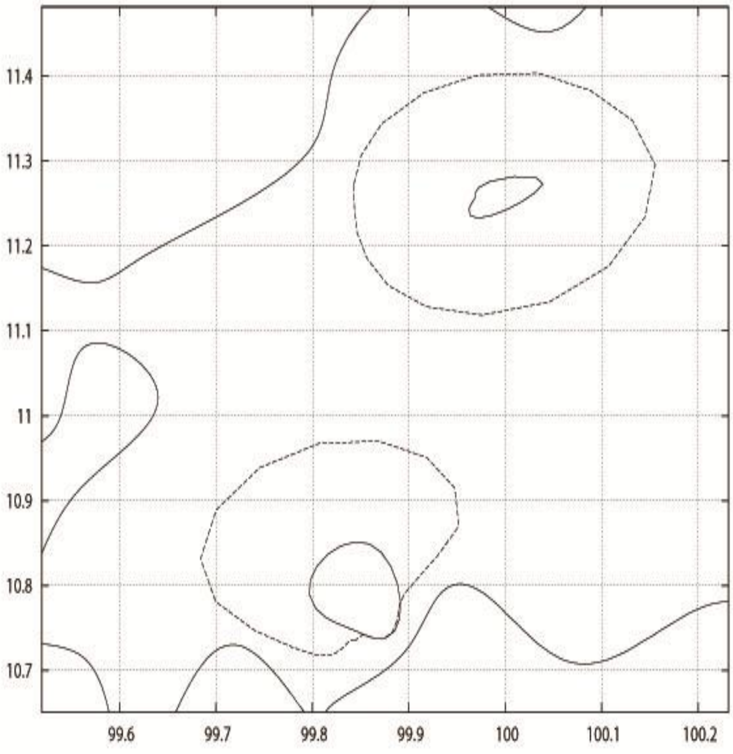


Figure 3.16 Enlargement: α=0, β=20, γ=8

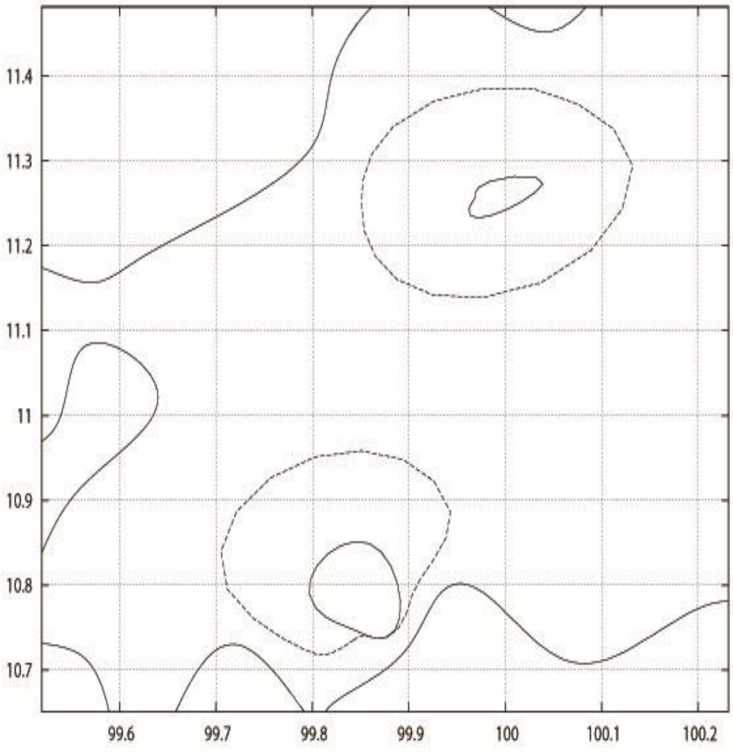


Figure 3.17 Enlargement: α=20, β=20, γ=8

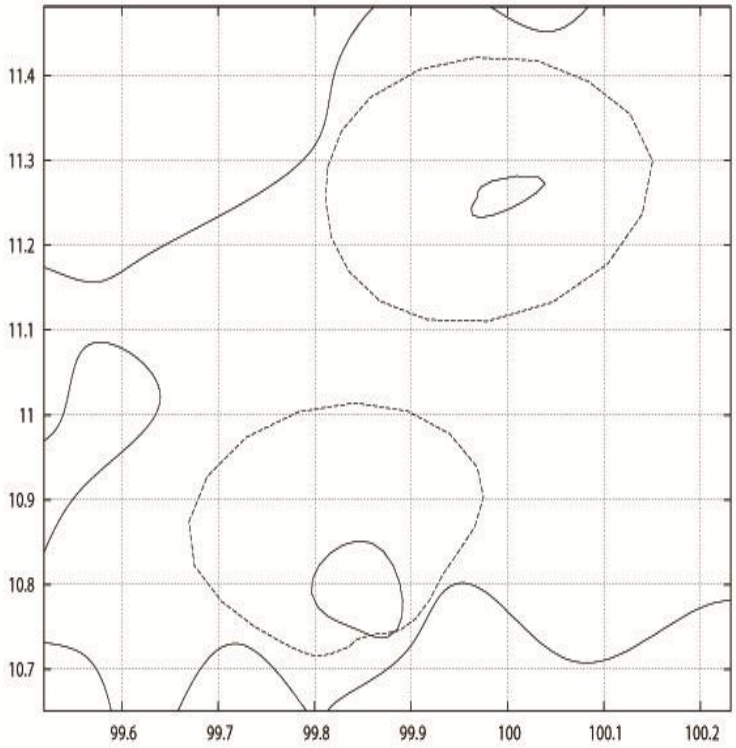


Figure 3.18 Enlargement: α=20, β=20, γ=4

3.4.2. Aggregation

On top of the previous parameters, another parameter for aggregation is the aggregation distance. Points whose distance to fixed curve is smaller than this value are included in the snake model. Points whose shortest distance is smaller than the legibility distance, have a non-zero external energy. In order to compare the results, constraints are set as follows: legibility distance = 0.2 and aggregation distance = 0.4. In the following figures, original lines are shown by solid lines and the aggregated line is shown by a dashed line.

Figure 3.19 and 3.20 show the impacts of parameters α and β. In figure 3.19 both α and β are equal to 0. No shape constraint is imposed and the line moves quickly to the other line to have a minimum energy. With non-null α and β, the curve shape is maintained. Figures 3.21 and 3.22 show the results with different α and β values. Table 3.4 shows the area difference obtained in each case. It can be seen that parameters have limited influence as results are quite similar. Figures 3.23 and 3.24 show the impact of γ. From the figure, a smaller γ provides a larger deformation at each step.

Table 3.4 Area difference between aggregated feature and original features

	Area (cm^2)	Area diff (cm^2)
Original feature	0.4939 and 2.1616	0
α=0, β=0, γ=2	2.6935	0.0380
α=10, β=0, γ=2	2.6951	0.0396
α=0, β=10, γ=2	2.6934	0.0379
α=10, β=10, γ=2	2.6951	0.0396
α=10, β=10, γ=5	2.6866	0.0312

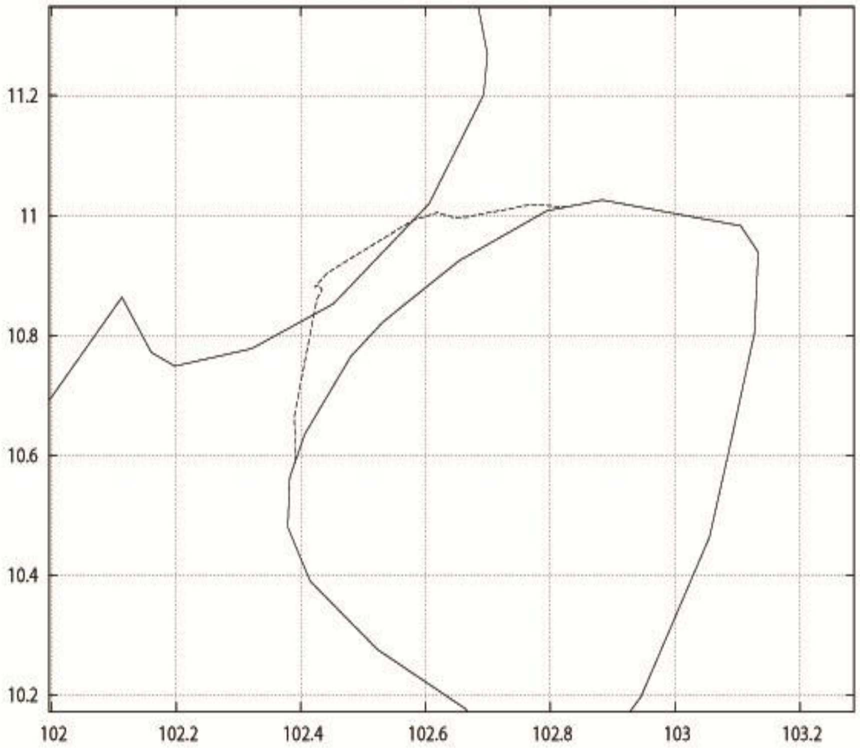


Figure 3.19 aggregation α=0, β=0, γ=2

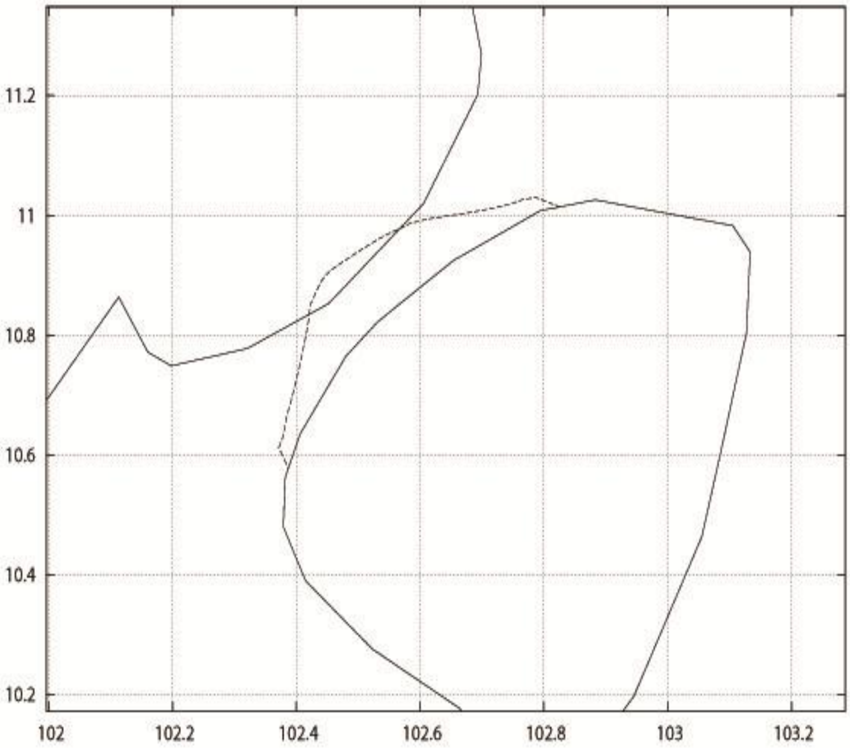


Figure 3.20 aggregation α=10, β=10, γ=2

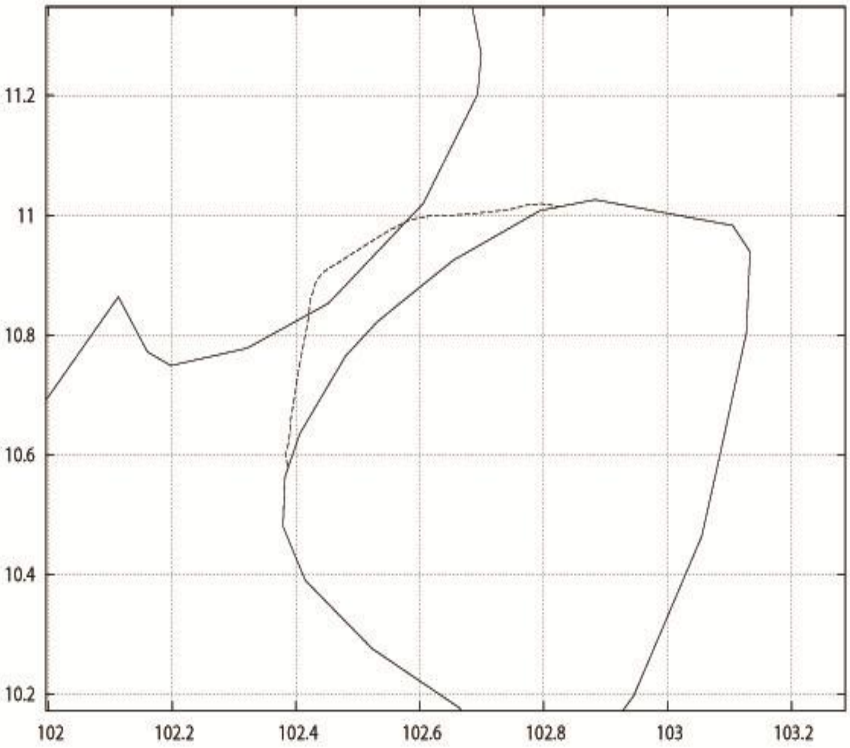


Figure 3.21 aggregation α=0, β=10, γ=2

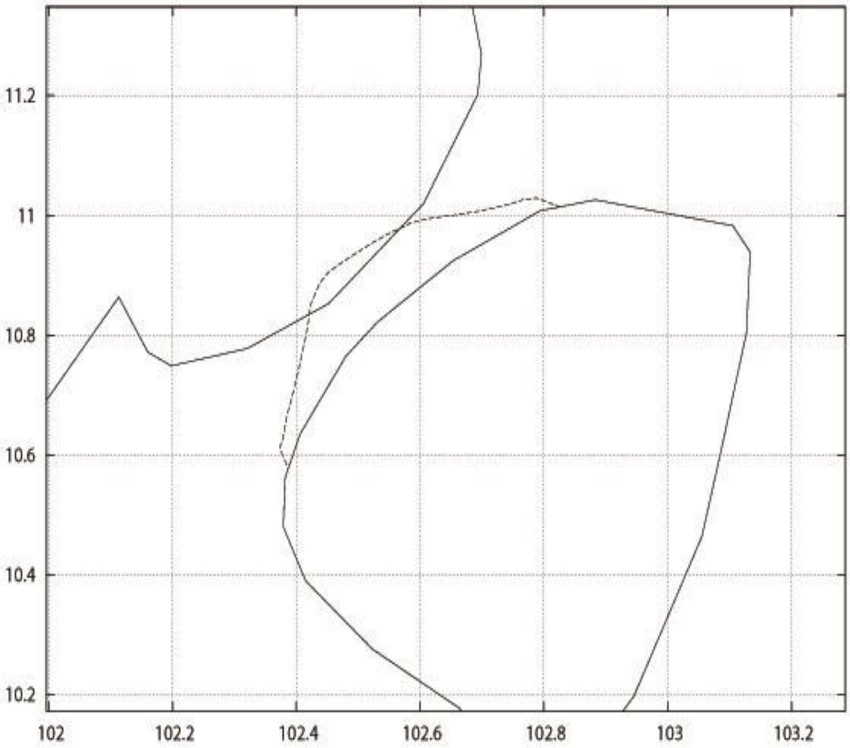


Figure 3.22 aggregation α=10, β=0, γ=2

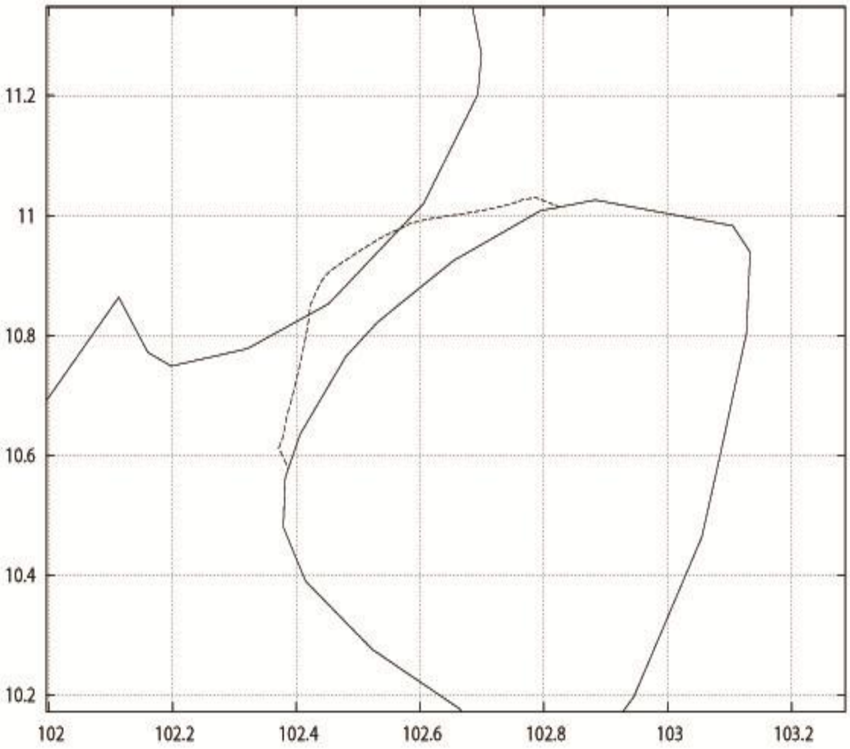


Figure 3.23 aggregation α=10, β=10, γ=2

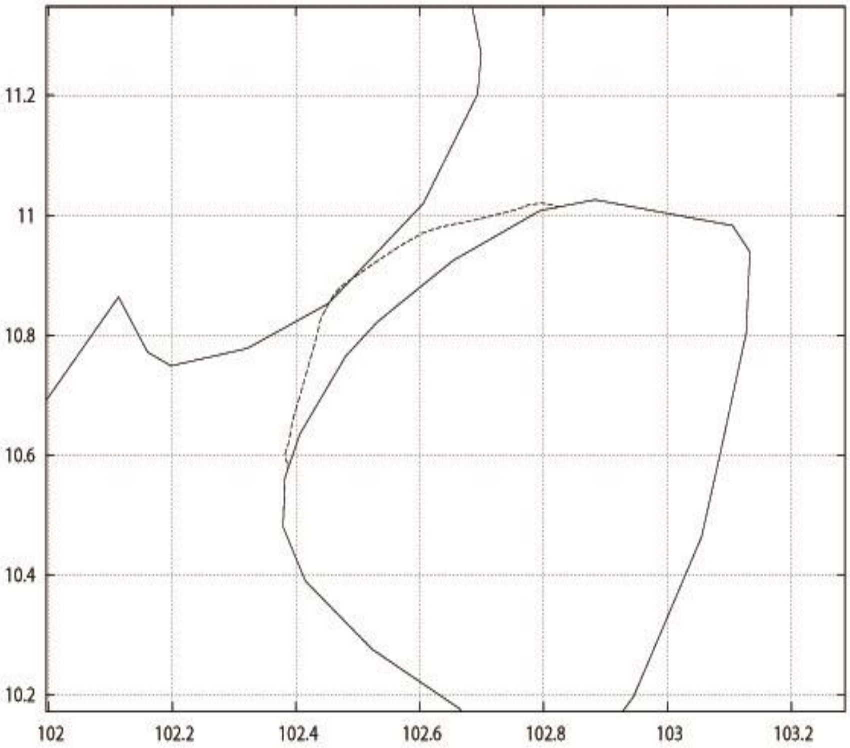
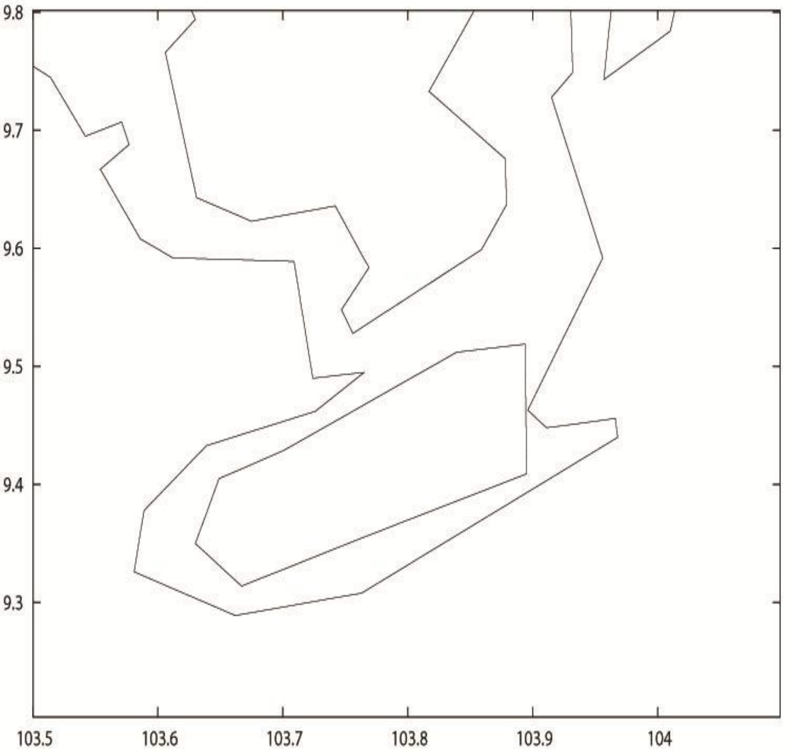
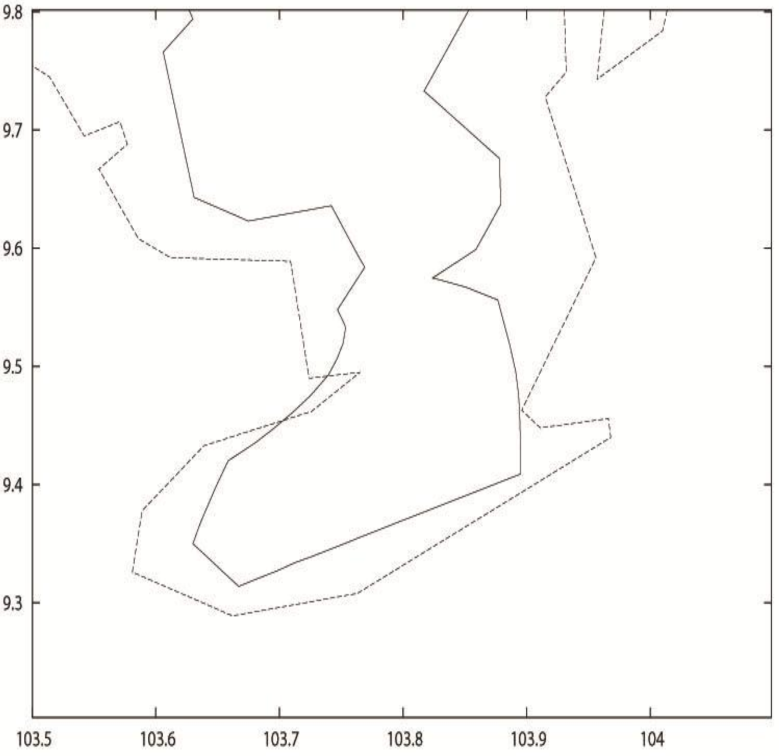


Figure 3.24 aggregation $\alpha=10$, $\beta=10$, $\gamma=5$

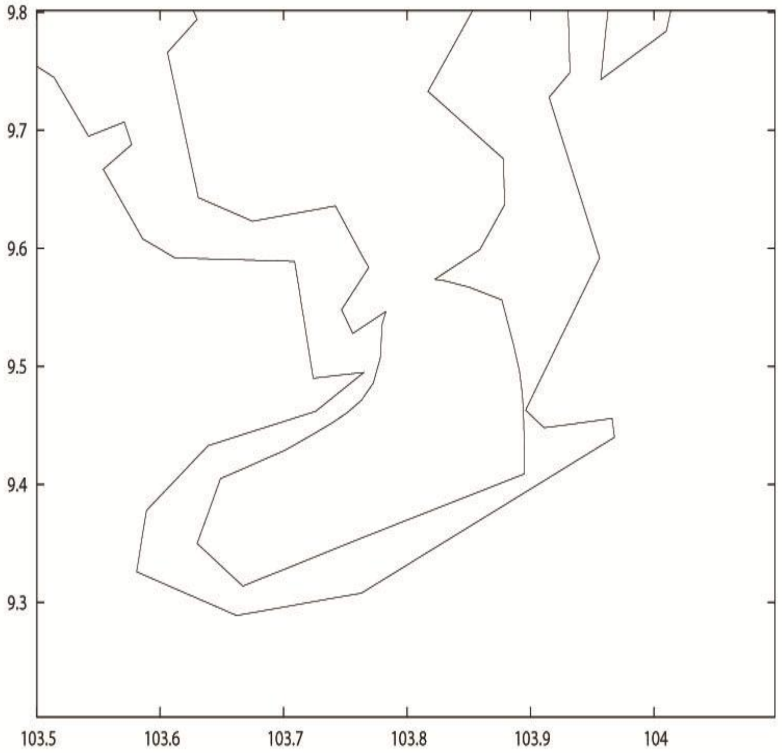
One interest of performing a smooth deformation for feature aggregation is that conflicts with neighboring features can be included in the process. Figures 3.25, 3.26 and 3.27 show an example where the distance to the parent feature is not considered during the deformation. In Figure 3.26, the merged curve overlaps with its parent feature, violating the topological constraint. Adding an external energy when the curve is too close to another feature can avoid this situation. Figure 3.27 is the results obtained when setting such energy term.



Feature 3.25 original map for merging



Feature 3.26 merging features without consider parent



Feature 3.27 merging features with consider parent

3.5. Summary

This chapter introduced a new approach for isobath generalization that takes into consideration the undersea features portrayed on the chart. The approach consists first in identifying the features and second in applying specific generalization algorithms defined for the features.

Section 3.2 gives the definition of the two types of features considered, the peak and the pit. These features are characterized only based on the topologic relationships between the contours and are organized in a feature tree. Generalization operators are defined for selecting the features. As the constraints considered refer to the meaning carried by the features, only model generalization is performed. Three operators are defined: selection/omission, enlargement and aggregation. Because of the safety constraint, operators only apply to one type of feature: omission applies to pits while enlargement and aggregation apply to peaks.

Both enlargement and aggregation operators are introduced in section 3.3 and rely on the snake model which performs continuous deformations. Although continuous deformations are usually used for cartographic generalization, they are considered in model generalization because peaks must always be selected and therefore must be enlarged or aggregated if too close or in conflict with another peak. The objective is to perform a smooth deformation of the isobath defining the boundary of the feature until it is large enough or it overlaps with another feature so that both can be merged. As no simplification is performed, the internal energy of the snake is defined by the derivatives of the isobath displacement to preserve the shape. A different external energy is associated to each operator. Enlargement is done by pushing the line away from the center of the feature and the process is stopped when the feature is sufficiently large while in aggregation, the line is attracted by the other feature until both features overlap. In both case, another constraint is added to prevent distance conflicts with other features.

Experimental results are presented in section 3.4. The snake model is tested on different sets of features and the results are assessed for different sets of parameters and tolerances. The other relevant parameter for aggregation is the angle formed by the intersecting isobaths. The angle influences the shape of the merged isobath however the line still needs to be smoothed as the result is not visually acceptable for a cartographer, although the safety and structure constraints are satisfied.

The last part of this research is concerned with the automation of the process. The data structure and the operations applied to the features being defined. The next chapter presents the agent based model that coordinates the process.

4. Automatic feature generalization using a multi-agent system

4.1. Introduction

This chapter presents a strategy for the automatic selection of generalization operators applied to features. The main objective of the method is to select the features that should appear on the chart based on their relevance. The criteria for selection are the type of feature and the size: peaks are more important than pits and selected features are emphasized if needed. As cartographic constraints are not considered, legibility conflicts that occur on one isobath or between two isobaths are not corrected. The strategy follows Ruas (1999)'s model presented in section 2.4. Features are able to evaluate their environment and make plans of actions.

The next section describes the two types of agents, modeling feature and isobath objects. Each active agent goes through a specific life cycle where it evaluates its environment and prepares a list of plans of action. Each action's result is evaluated and the best action is selected.

In section 4.3, the method is applied to a real case study on a set of isobaths extracted from the SHOM (the French Hydrographic and Oceanographic Service) bathymetric database. Examples of different plans are presented and discussed and results are analyzed from a cartographer's perspective.

4.2. The MAS model

4.2.1. Life cycle of feature agents

In order to perform generalization on a set of features, each feature is defined as an autonomous agent who can take its own decisions. Following the approach developed by Ruas (1999), map objects are defined as two different types of agents: meso agents who are able to take up decision by observing their global environment and micro agents which are individual elements of the map working at a local level with a limited perception of their surroundings. In the context of nautical charts, features formed by groups of isobaths are meso agents while isobaths are the micro agents.

Each type of agent has its own constraints, evaluation methods detecting conflicts and operators performing generalization actions. Isobaths, as micro agents, only have knowledge about the neighboring isobaths from the contour tree. They can evaluate distance conflicts with other isobaths. They can also perform some deformation action via the snake model. However they are not able to evaluate their situation and trigger an action as they lack of information about the terrain morphology.

Decisions are made at the meso level. Each feature agent goes through a series of steps, the life cycle, summarized in Figure 4.1. The feature first evaluates if generalization must be performed by communicating with the contour agent that defines its boundary. The feature passes information about neighboring features and admissible direction for deformation. The contour checks if any area or distance conflict occurs and returns the result to the feature. If some constraint is violated, a list of possible actions is set depending on the feature type. For a peak, possible actions are enlargement and aggregation. For a pit, the only possible action is deletion. The feature agent first triggers continuous operations. The feature calls again the isobath which minimizes its energy. During the process, the isobath checks if any other conflict occurs. For example, when enlarging a peak, it may move closer to another peak. In that case, the action is rolled back and the next action in the plan, aggregation, is performed. Once continuous operation is completed, the data structure is updated if required (indicated as discrete operation in Figure 4.1). During the process, topologic and safety constraints are always maintained as any operation that would violates these constraints would be rejected. Indeed, although existing distance conflicts between isobaths are not corrected, legibility is still considered in order to avoid the creation of new conflicts.

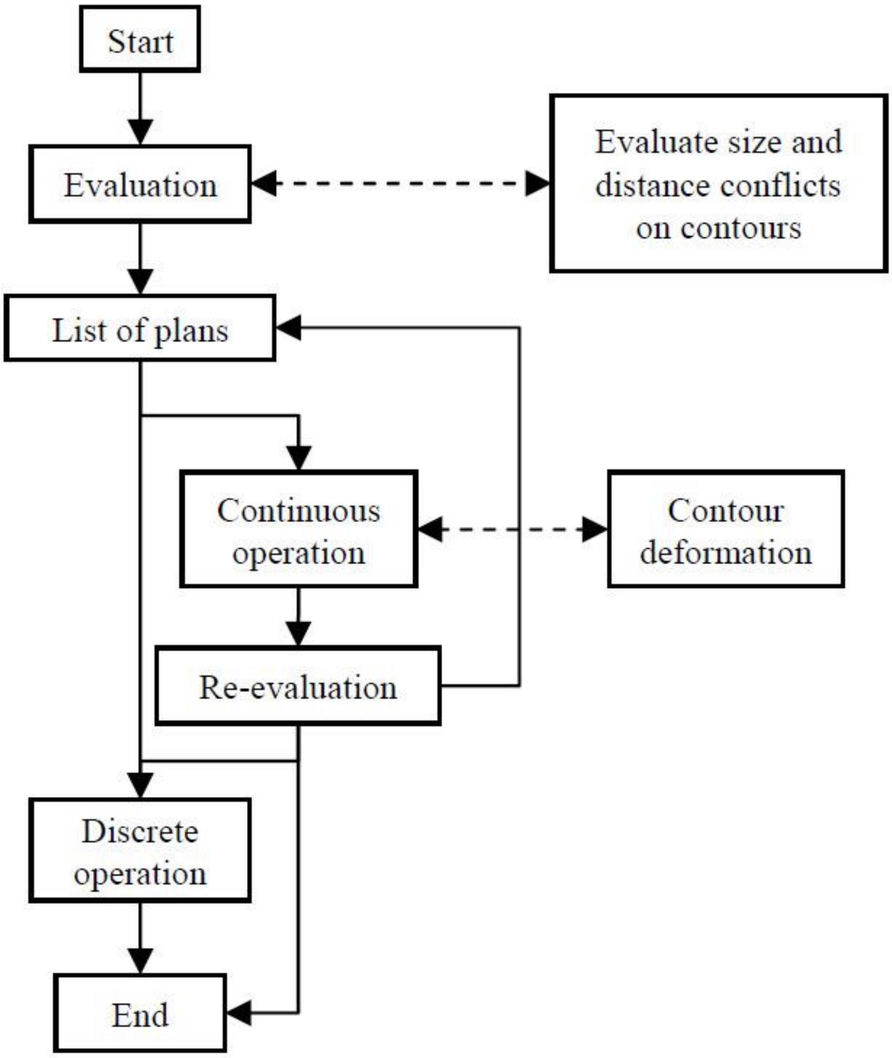


Figure 4.1 Life cycle of a feature agent

In multi agent systems, agents can run and execute actions in parallel. An agent can perform actions on other agents with the condition that they are not active. For example, if an agent evaluates a conflict with another agent who is already undergoing an enlargement, the evaluation will not be correct. Some actions also modify the structure of the feature tree by adding or removing features. If several agents modify the tree at the same or if a feature is deleted while exchanging with another feature, some exception is raised and the operations are interrupted, creating inconsistencies. In order to avoid this problem, agents are not activated all at the same time but in a sequence. Indeed two agent features which are included in two different features can be activate at the same time as there is no risk they interact but features sharing the same parent must be activated separately.

4.2.2. Evaluation of the environment

As mentioned above, in the life cycle, the feature agent first perceives its environment to decide which features can be selected. Existence of conflicts and the need to modify the features is determined by evaluating different cartographic constraints which are estimated based on feature attributes and threshold values set by the user. The feature attribute is the type of feature (peak or pit) and the constraints considered are, as introduced in chapter 3, the legibility distance with neighboring features and the feature area.

Controlling the environment involves two components of the agent. The perception component performs the evaluation by computing area and distances to neighbors as recorded by the feature tree. The state component stores the agent situation. The situation is evaluated as soon as the agent becomes active so that its original state is recorded before any generalization is done: the agent calculates and stores its area, compares it to the area constraint and records the area state. The similar step goes then to records the distance to the agent's parent and neighbors. Feature parent is treated separately from feature adjacent features. A conflict between two adjacent features involve the feature boundaries while a conflict between a feature and its parent does not involve the parent's boundary feature but an inner isobath whose generalization is not considered and, as mentioned in chapter 3, the distance between the isobaths is simply preserved so that no new conflict is created. The pseudo code is:

```
Input: a feature agent F
Output: state of feature agent F
Begin
  // check area state
  area = F.getArea()
  If area < area tolerance
    If F.type == PIT
      F.areaState = SMALLPIT
    Else
      F.areaState = SMALLPEAK
    End If
  Else
    F.areaState = NOAREACONLICT
  End If

  // check distance conflict with adjacent features

  Dist=F.distance(F0)
  For adjacent feature F0 to Fi
    Temp=F.distance(Fi)
    If (temp<dist)
      Dist=temp;
      Nearest Brother=I;
    End If
  End For

    If F.distance(Nearest Brother) < legibility distance
      If F.type == PIT
        F.distanceState = CLOSEPIT
      Else
        F.distanceState = CLOSEPEAK
      End If
    Else
      F.distanceState = NOCONFLICT
    End If
```

```
  //check distance conflict with parent feature
  If F.distance(F.parent) < distance tolerance
    If F.type == PIT
      F.parentState = CLOSEPIT
    Else
      F.parentState = CLOSEPEAK
    End If
  Else
    F.parentState = NOCONFLICT
  End If
End
```

During the evaluation, the feature calls up the isobath agent when performing area and distance computation. As the isobath does not have access to topographic information, the feature communicates the isobaths to be checked. Apart from defining the existing conflicts and storing the original feature state, the state component can also store the state of a feature after generalization is performed so that different situations of a feature can be compared.

4.2.3. Plans of action

After evaluating its situation, the agent knows which cartographic constraints are violated. Based on the situation described by the state component, the agent prepares different strategies combining different generalization operators. Each strategy corresponds to a plan defined by one operation or by a sequence combining different generalization operators.

The agent may have one or several plans depending on how it conflicts with cartographic constraints. Strategies are set according to the type of feature with respect to the safety constraint. The list of possible plans is summarized in Table 4.1. If the feature represents a pit and it conflicts with too close features, it will delete itself. Conflicts with the parent feature are not considered in the planning as they only affect the constraints set in the enlargement and aggregation operations by imposing fixed points. Only the case where the peak is too small and has a close enough neighbor will trigger different plans (enlargement, aggregation and enlargement plus aggregation). Each plan is run in a sequence and the different results are stored in the state component. The results are then compared with the existing situation in order to select the best score.

Table 4.1 List of possible plans of action

Feature type	Conflict	Plan
Peak	SMALLPEAK	Enlargement
	CLOSEPEAK	Aggregation
	CLOSEPEAK and SMALLPEAK	Enlargement
		Aggregation
		Enlargement Aggregation
Pit	SMALLPIT or CLOSEPIT	Omission

Algorithm SETPLAN

```
Input: Feature F,
Begin
F.planList.clean
if F.areaState == SMALLPEAK
  if F.brotherState != CLOSEPEAK
    F.planList.add(Enlargment)
  Else if f.brotherState == CLOSEPEAK
    F.planList.add(Aggregation)
    F.PlanList.add(Enlargment)
    F.planList.add(Enlargement, Aggregation)
  End If
Else if F.areaState == SMALLPIT
  F.planList.add(Deletion)
Else if
  F.planList.add(Empty plan)
End if

If F.brotherState == CLOSEPEAK and F.areaState != SMALLPEAK
  F0= F.First Brother
  Dist=F.distance(F0)
  For each brother feature F0 to Fi
```

```
      Temp = F.distance(Fi)
      If (temp<dist)
         Dist = temp;
         Nearest Brother = I;
      End If
   End For
   If Nearest Brother.Depth == F.Depth
      depthSame = true;
   End if
   if F.closed == true or Nearest brother == true
      oneCurveClosed = true;
   End if
   if depthSame = true and oneCurveClosed = true
      F.planList.add(aggregation)
   else
      F.planList.add(Empty plan)
   End If
Else If f.brotherState==CLOSEPIT
   F.planList.add(Deletion)
Else If
   F.planList.add(Empty plan)
END
```

4.2.4. Evaluation of the plans

Comparing the results of each plan requires an evaluation technique quantifying the amount of change brought to the chart. At this stage, the different results are stored within the state component. Some operators such as deletion and aggregation will change the structure of the feature tree which will impact on other plans. Therefore, even if only one plan is performed, the result is still evaluated to check if it is valid before updating the current feature situation.

Transformations are evaluated at two stages. First, after a continuous deformation, the agent checks if the distance and safety constraints were preserved. If not, for example, if while enlarging a boundary, the feature collides with another feature, the plan is automatically rejected and the next plan in the list is performed. Transformations are also evaluated after discrete operation is performed. At this level, both safety and legibility constraints are satisfied and the objective is to look for the solution that better preserves the seafloor morphology. Compared with isobaths, the emphasis in feature generalization is on the preservation of the terrain structure. Furthermore, omission and aggregation cannot be evaluated on the isobaths. Therefore, rather than focusing on line deformation or smoothness measurement, the variation of area brought to the feature is considered as a more relevant indicator. The variation of area for the enlargement is simply defined by the area difference between the original feature and the enlarged feature. For the aggregation, the variation is given by the difference between the area of the aggregated feature and the areas of both features before aggregation.

The plan with the smallest difference is then chosen as the best solution and the new solution becomes the current state. If some conflict still exists, because the feature has not reached the minimum area yet or some distance conflict still occurs, the agent goes again through the life cycle and a new set of plans is performed is performed. If a conflict cannot be solved after a given number of times, the process is stopped anyway. The issue occurs only in the case of a peak that cannot be enlarged because it is surrounded by an isobath belonging to its parent feature.

Algorithm TRY PLAN

```
Input Feature F, F.state 0 to n corresponding plan list 0 to n
Begin:
Feature FClone = F.Clone
For plan P of F.PlanList from 0 to n
  For each action A of P
    Switch (A):
      Empty Plan:
        F.state(n).setNewState
        F.state(n).CalculatedResult(F)
      Enlargment:
        FClone.enlargement
        F.state(n).setNewState
        F.state(n).CalculatedResult(FClone, F)
```

```
        Deletion:
          FClone.deletion
          F.state(n).setNewState
          F.state(n).CalculatedResult(f)
        Aggregation:
          If (FClone.merged(nearest brother)=success)
            F.state(n).setNewState
            F.state(n).calculatedResult(F,nearest brother, FClone)
          Ene If
      End switch
    End For
End For
END
```

Algorithm Plan Evaluation

```
Input:F, F.state 0 to n corresponding plan list 0 to n
Begin:
temp=F.state(0).result
LinkedList<int> planUseful=new LinkedList
//planUseful records the plan which can satisfy cartographic constraint
For F.state from 0 to n
  if F.state(n).areaState!=conflict and state(n).brotherState!=
conflict
    planUseful.add(n)
  End if
  if F.state(n).result<temp
    temp= F.state(n).result
    bestPlan=n
  End IF
End For

If planUseful.size!=0
  if planUseful contains bestPlan
    doPlan(bestPlan)
  else
    temp= F.state(0).result
    bestPlan=planUseful(0)
    For planUseful from 0 to j
      if F.state(planUseful (j)).result<temp
        temp= F.state(planUseful(j)).result
        bestPlan=planUseful(j)
      End If
```

```
    End For
    doPlan(bestPlan)
  End If
Else
  doPlan(bestPlan)
End If
END
```

4.3. Application to feature generalization

4.3.1. The case study

In this section, the algorithm is applied to a set of features formed by isobaths. Data were provided by the French hydrographic office (SHOM). Figure 4.2 is the original map. Different lines adapt with different snake model parameters. As a result, the system runs twice with different sets of parameters. Parameters setting are given in Table 4.2. Figure 4.3 gives the result of the whole map generalization process.

Table 4.2 Parameters of case study

Aggregation	
α	20
β	20
γ	8
Legibility distance	0.2
Aggregation distance	0.2
Max iteration times	30

Enlargement	
α	20
β	20
γ	8
Legibility distance	0.1
Area legibility distance (legibility distance^2*Pi)	0.031

Parent legibility distance	0.05
Max iteration times	100

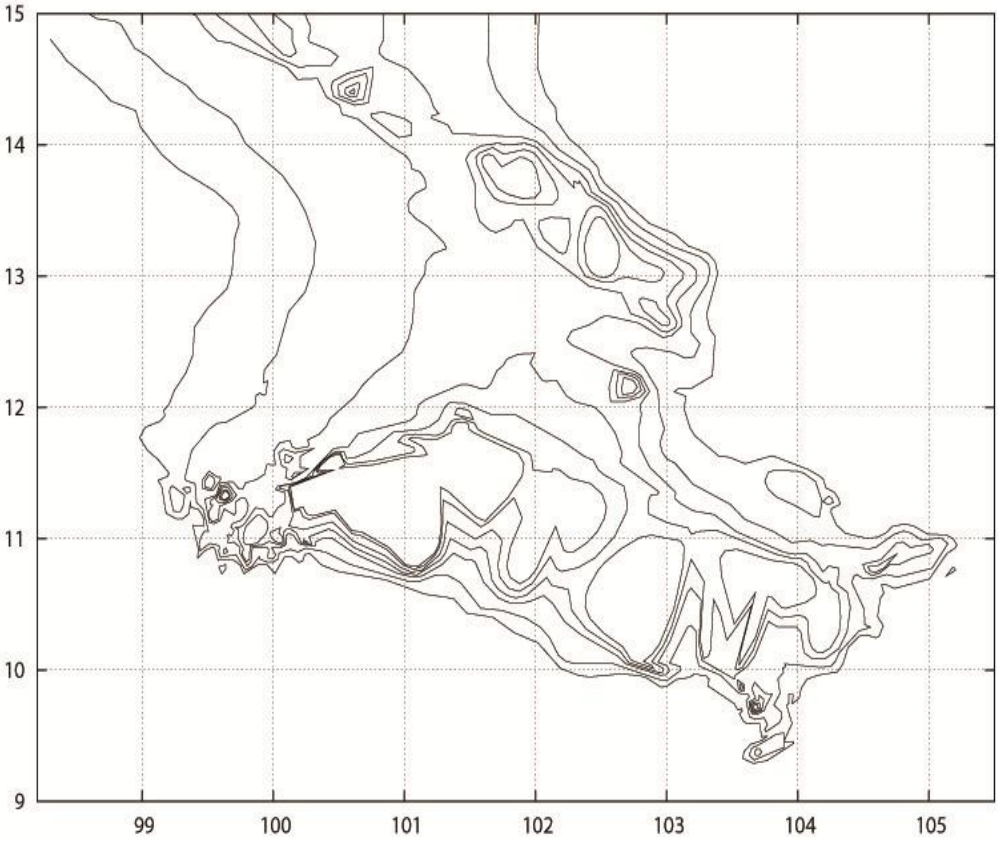


Figure 4.2 Original map

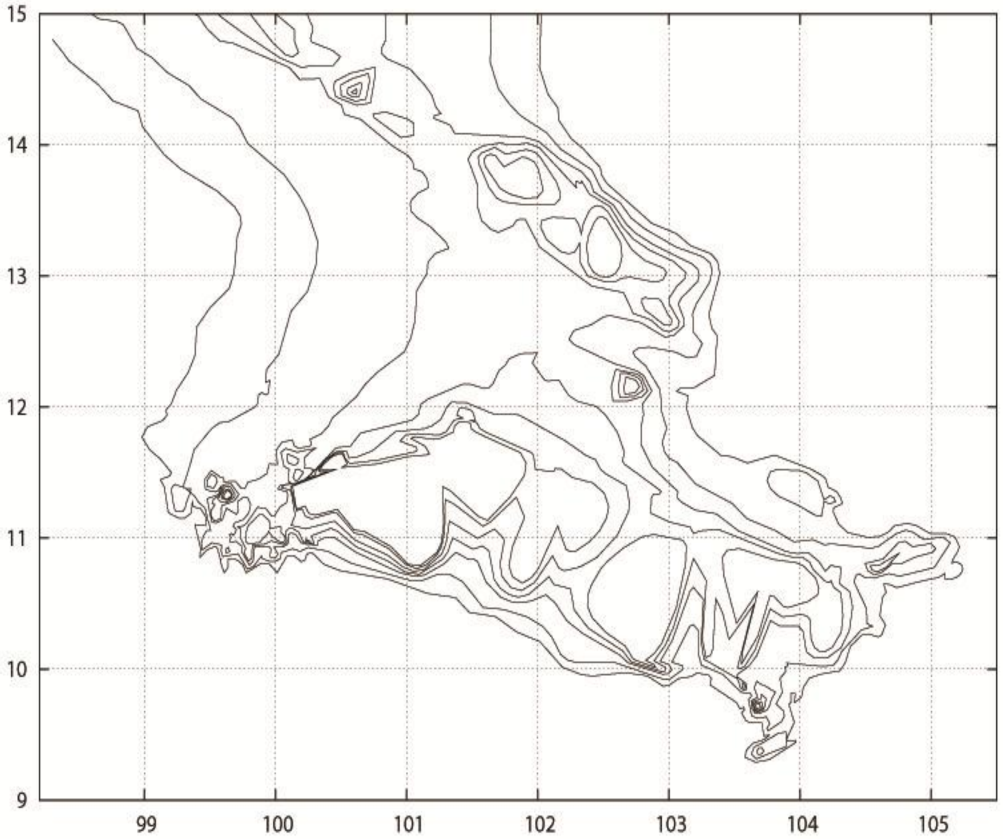


Figure 4.3 Generalized map

During the process, the feature tree structure is modified because of the merges. The feature tree structures of original map and generalized map are shown in Figures 4.4 and 4.5. Grey nodes are pit features and black nodes are peak features.

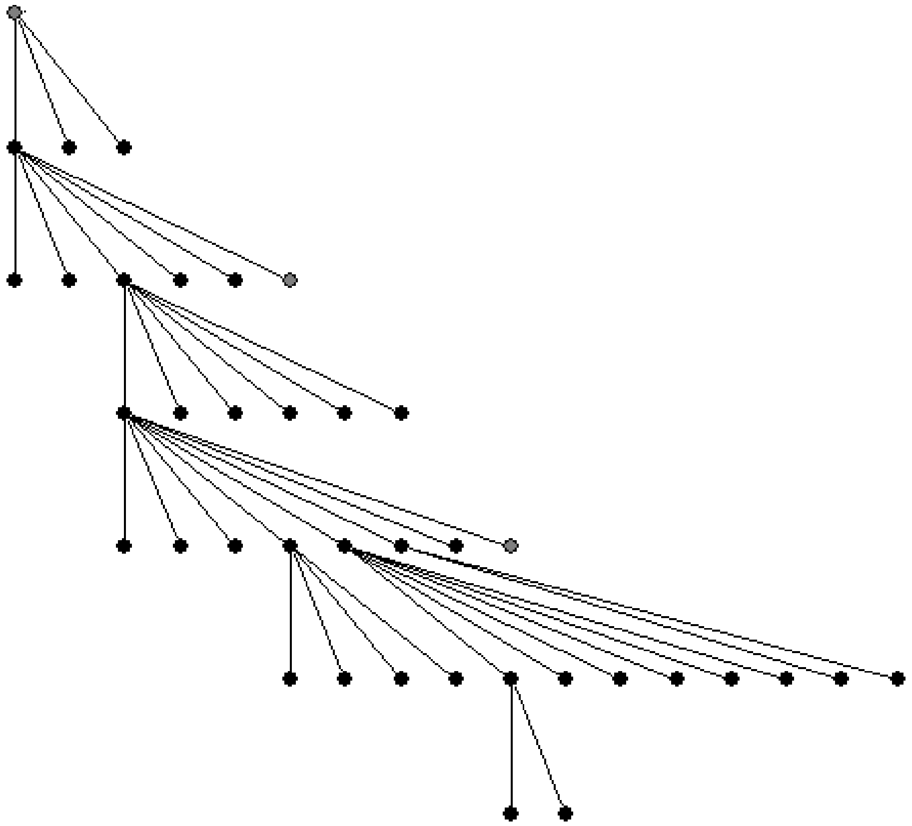


Figure 4.4 original feature tree

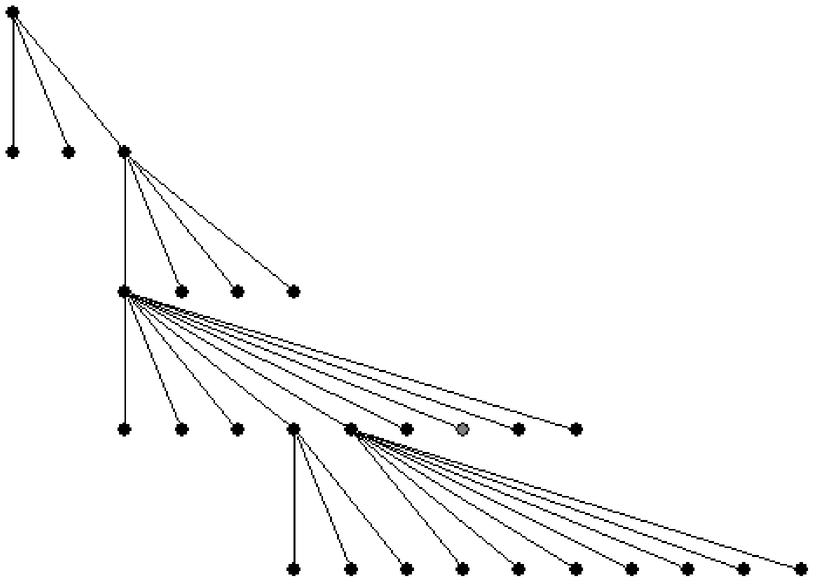


Figure 4.5 generalized feature tree

4.3.2. Discussion

Feature selection satisfies the constraints set beforehand, including the safety constraint. Depending on the tolerances set, too small pits are removed and peaks can be enlarged or aggregated.

Figure 4.6 and figure 4.7 give some details of the generalization result. In figure 4.6, small features are merged with their neighbors in order to increase their area. Some merges occur. Part A in figure 4.6 is interesting. There was a small pit feature on the original map. It is first removed and then the other two features beside it are merged together. In part B, the small feature cannot merge with another feature. It is because it is too close to its parent and there is not enough room for it to enlarge. One way to satisfy this problem is to add more communication part in the system. It may allow the parent feature to move for getting more space for the small feature to enlarge.

In figure 4.7, the feature in the right enlarged itself as it is too far from other feature for aggregation. The right feature was too small and too close to the features on the left. If a feature is in conflict with two features, the conflict with the closest feature is treated first. Among the three possible plans, enlargement followed by aggregation was selected. Once the first conflict was solved, the second conflict was treated and aggregation was performed. In such a case involving three features, only one feature agent can be active at a time otherwise one feature may try to perform an action with a feature that no longer exists.

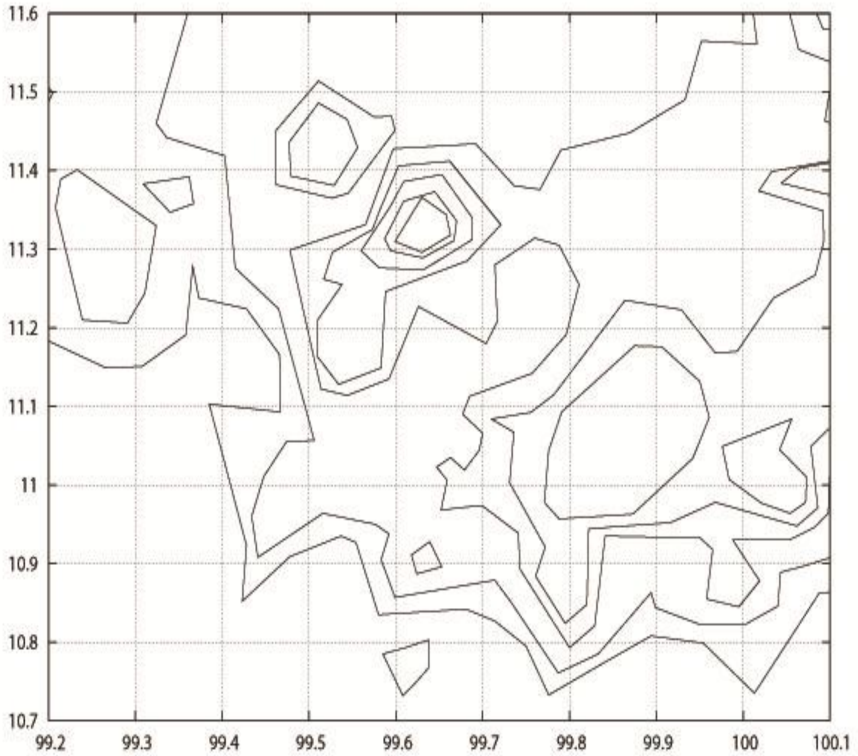


Figure 4.6 a) Detail of map: feature aggregation original map

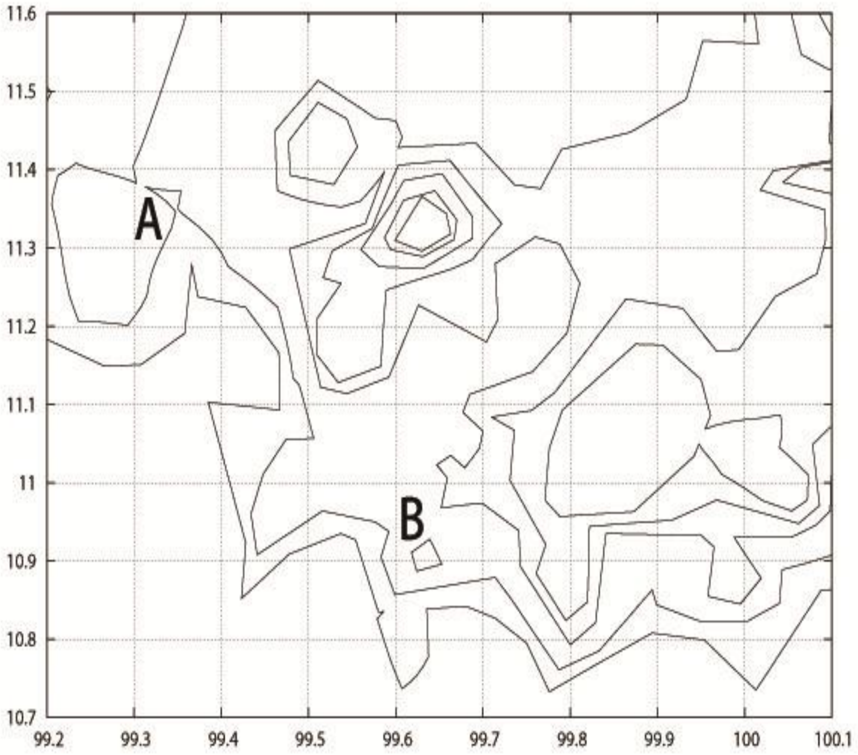


Figure 4.6 b) Detail of map: feature aggregation generalized map

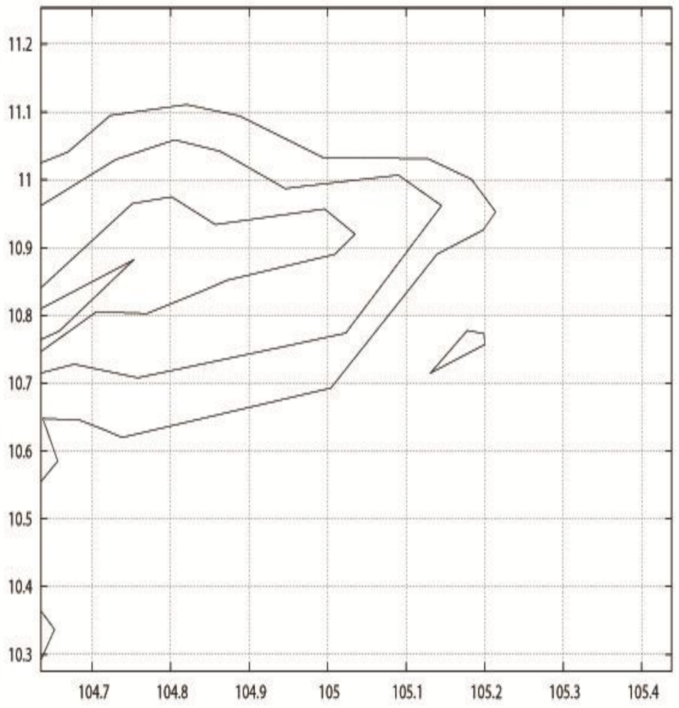


Figure 4.7 a) detail map of selection of plans (original map)

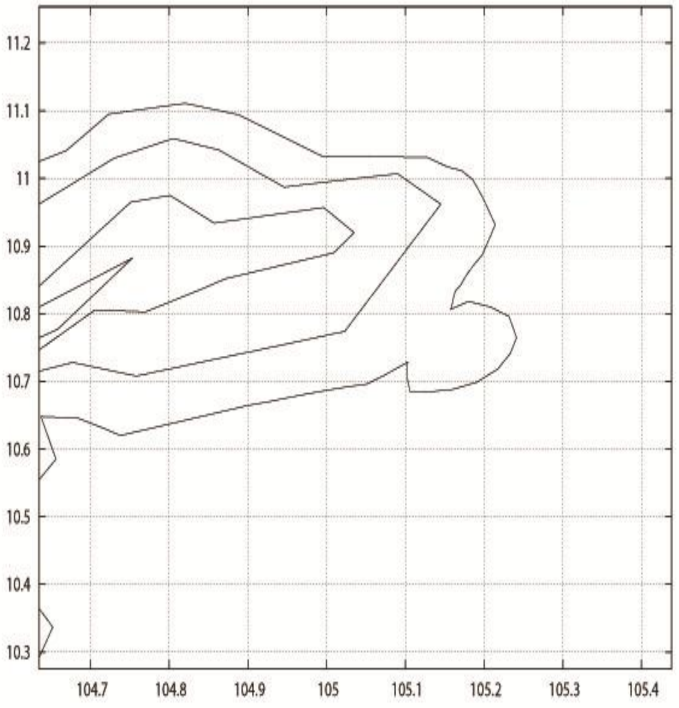


Figure 4.7 b) detail map of selection of plans (generalized map)

Aggregation allows selection of small peaks however the shape of the aggregated feature is not smooth enough to provide a fair line acceptable by a cartographer. A further smoothing operation needs to be performed. Therefore, a limitation to the quality of the results is that legibility constraints are not fully considered, apart from the distance between neighboring features. There is also no corresponding plan to satisfy the legibility conflict if a feature is close to its parent. One reason is the lack of communication between features. If one feature needs to be enlarged and is located too close to its parent, it could ask its parent to displace part of the conflicting isobath to achieve a cartographic constraint.

4.4. Summary

A multi agent system was developed to coordinate generalization operations. Features and isobaths are organized at two different levels. Feature agents are able to take a decision based on their environment and to trigger an action while isobaths at a micro level perform actions based on the information they receive from the features and return a result. Only three operators, omission, aggregation and enlargement, are considered. Feature agents are able to make plans, evaluate them and pick the best result. Evaluation of the result is done by measuring the area variation before and after generalization.

Specific attention was given to the coordination of the feature agents. During the process, agents can be involved in different processes at the same time. For example, a feature can be evaluating its distance with another feature which is being aggregated. Therefore, feature agents are activated in a sequence by starting from the feature tree root and moving down to successive levels. In such a way, agents at the same level are evolving in an environment delineated by their parent features which are fixed.

The method was applied to the generalization of a set of isobaths extracted from a bathymetric database. Only feature selection is performed and cartographic generalization constraints of legibility and aesthetics are not considered, although the distance constraint is used to avoid the creation of new conflicts. A consequence is that results obtained do not represent valid generalization yet. For example, an isobath obtained by aggregation shows too sharp angles at the junction and therefore will not be considered as a good generalization by the cartographer as the line is not smooth enough.

Communication between feature agents is also limited and if a feature cannot correct a conflict by itself, it cannot send a message to another agent for action. This issue needs to be taken into consideration for cartographic generalization as if a feature does not have enough space for enlargement or smoothing, it currently cannot ask a neighboring feature to propagate the enlargement or to delete an isobath or a segment.

5. Conclusion

5.1. Summary

This thesis focused on the generalization of isobaths on the nautical chart and took into consideration the morphologic information carried by the isobaths. Nautical charts need to ensure navigation safety and therefore submarine features must be emphasized if they represent a danger for navigation or a safe route. This specific constraint also imposes that the depth reported on the chart is never bigger than the real depth. These constraints are unique to nautical charts and require dedicated methods for generalization.

Chapter 2 gives a review of different approaches for line generalization. Among them, the most recent are constraint based models. The process provides more flexibility than the others as rules are not explicitly expressed. A series of constraints to be respected are expressed and the model finds an optimal solution minimizing constraint violations. Constraints that apply to nautical chart are mainly the safety constraint, specific to nautical chart, the legibility constraint related with the aspect and density of objects presented on the map and the geomorphologic constraint stating that main topographic characteristics of the seafloor shall be preserved.

Among existing constraint-based methods, specific attention is given to snake based methods which apply to lines and to agent-based methods which can integrate different techniques in order to automate a generalization process combining different operators and different types of objects. The snake is an energy minimization method where energies express constraints applied to the line. Snake models were developed for line smoothing and displacement. Specific methods were developed for isobaths, defining an external energy expressing the safety constraint. These methods apply for generalization of an individual isobath without taking into account the topographic information. In agent modeling, map objects are agents that can evaluate their environment and make their own decision. Agents can be defined at different levels and can be formed by individual objects such as a road or a building or can model a group of objects. Multi-agent systems are used to coordinate the actions of different agents and can combine both discrete and continuous operations, helping automation of the generalization process.

Chapter 3 introduced a new approach for isobath generalization. The original contribution is that generalization is performed by considering undersea features formed by groups of isobaths as cartographic objects. Features are extracted based on topologic relationships between isobaths and are organized in a feature tree. Two types of features are identified, peaks and pits, based on isobath depths and inclusion relationships. Then, three feature generalization operators are defined. The first, deletion, consists simply in removing all the isobaths forming a feature and applies only to pits. The two others, enlargement and aggregation, apply only to peaks and rely on the snake model to modify the isobath forming the boundary of the feature in order to either reach a minimum feature area or to merge the boundary with an adjacent boundary isobath for feature aggregation. Result of the operation depends on the values of different parameters defining the constraints and controlling the deformation. Results are evaluated and guidelines regarding constraint tolerances and shape parameters are established.

Chapter 4 presented the multi-agent system that was developed. Features and isobaths are organized in two different levels where features at the meso level are able to evaluate constraints. If a conflict occurs and a generalization must be done, the feature establishes a list of possible plans and evaluates each plan. A plan to be valid needs to satisfy the safety and legibility constraints. If two plans are valid, the plan which preserves the topography the best is selected and performed. If a continuous operation has to be performed on the boundary isobath, the feature passes the information to the isobath at the micro level to perform the operation and evaluates the result. Finally, the model is applied to a set of isobaths extracted from a bathymetric database. Choices made by the agents are discussed and results are discussed against the different tolerances set by the user.

The method provides valid results satisfying the legibility and safety constraints. Consideration of undersea features is the most original contribution of this thesis. It First, a feature tree providing further topographic knowledge is built, enriching the bathymetric database where features are considered as elements of the map. The second contribution of the thesis is the development of a new model for the isobath generalization process. The model is based on the generalization of features so that the process is driven by the information as perceived on the chart instead of focusing on local spatial conflicts as often done in cartographic generalization.

5.2. Directions for further investigations

The work presented in this thesis forms the foundation of a tool for nautical chart generalization and still suffers several limitations providing directions for short term work. Although the system achieves its objectives, it is still limited to a small number of operations and the results are not of a good enough quality for chart production. The main limitation is that cartographic constraints related to the aesthetic quality of the map are not considered. An isobath obtained after aggregation needs to be smoothed by a specific operator to produce an acceptable result. Distance conflicts that occur between features are not always processed, although they are considered during the enlargement and aggregation operations, so that the legibility constraint is not fully satisfied. Therefore, an immediate direction is to take into account all kinds of conflicts between features by adding smoothing and displacement operators. Such operators were defined by Guilbert and Saux (2008) and could be included as plans of action.

Furthermore, in the MAS, only feature conflicts are considered. Conflicts between isobaths at the micro level are not considered. Most operators correcting isobath conflicts are already defined for feature boundaries but in the current system, isobaths perform actions only based on the information they receive from the features. Dealing with conflicts at micro level means that the model should be extended so that micro agents are autonomous and can also evaluate their environment. A model similar to Duchêne (2003)'s model may be considered where an evaluation method specific to nautical chart constraints must be developed. In order to trigger isobath level agent, more communication protocol should be added in MAS. The communication part can also resolve the problem in Figure 4.6 part B.

Finally, an interest of the MAS approach is that the existing framework can be extended to include other bathymetric soundings and most specifically, soundings. Soundings can first be included as passive objects and included in the definition of legibility and topographic constraints but, on a long term, the objective is to be able to integrate them as active agents. Soundings can be seen as micro level agents enriching the definition of the features. Furthermore, other types of terrain features such as ridge lines or break of slopes may be identified from the soundings and added as agents at the meso level. An ontology of undersea features is presented by Yan et al. (2012) and can be used as a foundation for the classification of features at the meso level.

6. References

Bader M., 2001, Energy minimization methods for feature displacement in map generalization, PhD dissertation, University of Zurich

Baeijs C., Demazeau Y., and Alvares L., 1996, SIGMA: Application of Multi-Agent Systems to Cartographic Generalization. In: 7th European Workshop on Modeling Autonomous Agents in a Multi-Agent World, MAAMAW'96. Springer-Verlag, Eindhoven, the Netherlands.

Beard K., 1991, Constraints on rule formation. In Map Generalization, B. Buttenfield and R.B. McMaster (Eds), pp. 121–135 (London: Longman).

Beer R.D. 1995, A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, 72:173-215, 1995

Burghardt D. and Meier S., 1997, Cartographic displacement using the snakes concept, Semantic Modeling for the Acquisition of Topographic Information From Images and Maps, W. Forstner and L. Plumer (Eds), pp.51-71 (Basel: Birkhauser Verlag).

Burghardt D., 2005, Controlled Line Smoothing by Snakes, Geoinformatica 9:3, 237-252

Chan Y., Wilson. J.P., Zhu Q. and Chou Q., 2012, Comparison of drainage-constrained methods for DEM generalization, Computers & Geosciences 48 (2012) 41–49

Douglas D.H. and Peucker T.K., 1973, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10: 112-122.

Duchêne C., Barrault M. and Haire K., 2001, Road network generalization: A Multi-Agent approach, in Proceedings 20th International Cartographic Conference, Beijing China, 2166-2177

Duchêne C., 2003, Automated map generalization using communicating agents, Proceedings of the 21st International Cartographic Conference (ICC), Cartographic Renaissance, Durban, South Africa , August 10-16, pp.160-169, CD-ROM

Gaffuri J., 2006, How to merge optimization and agent-based techniques in a single generalization model?, Workshop of the ICA Commission on Map Generalisation and Multiple Representation, June 25th 2006

Galanda M., 2003: Automated Polygon Generalization in a Multi Agent System. Ph.D. thesis, Department of Geogra-phy, University of Zurich

Gökgöz T., 2005, Generalization of Contours Using Deviation Angles and Error Bands. The Cartographic journal 42(2):145–156

Guilbert E., Saux E., and Daniel M., 2004, Combining Geometrical and Mechanical Displacements for Suppressing Intersections in Cartographic Generalization, WSCG'2004, February 2-6, 2004, Plzen, Czech Republic. Copyright UNION Agency – Science Press

Guilbert E., Saux E. and Daniel M., 2006, Conflict removal between B-spline curves for isobathymetric line generalization using a snake model. Cartography and Geographic nformation Science, 33(1), pp. 37–52.

Guilbert E and Saux E., 2008, Cartographic generalization of lines based on a B-spline snake model, International Journal of Geographical Information Science, Vol.22, No. 8, August 2008, pp. 847-870

Harrie L. and Weibel R., 2007, Modelling the overall process of generalisation. In: Generalisation of Geographic Information: Cartographic Modelling and Applications, W.A. Mackaness, A. Ruas and L.T. Sarjakoski (eds.) (Amsterdam: Elsevier Science), pp. 67–87

IHO, 2008 International Hydrographic Organization. *Standardization of undersea feature names*. International Hydrographic Bureau, Monaco, 4th edition

Irigoyen J., Martin M., Rodriguez J., 2009, A smoothing algorithm for contour lines by means of triangulation, The Cartographic Journal 46(3):262–267

Kass M., Witkin A., and Terzopoulos D., 1987, Snakes: Active contour models, Proceddings of the First International Conference on Computer Vision, IEEE Comput. Soc. Press, pp. 259-268

Kweon I.S. and Kanade T., 1994, Extracting topographic terrain features from elevation maps. CVGIP: Image Understanding, 59(2):171–182

Li Z. and Openshaw S., 1993, A natural principle for the objective generalisation of digital maps. *Cartography and Geographic Information Systems*, 20: 19-29

Li Z. and Sui H., 2000. An integrated technique for automated generalization of contour maps. *The Cartographic Journal*, 37(1): 29–37

Li Z., "Algorithmic foundation of multi-scale spatial representation", 1st ed., CRC Press: Boca Raton, 2007.

Lopes J., Catalão J., Ruas A., 2011, Contour line generalization by means of artificial intelli-gence techniques. ICA CO-441, 15 pages

Mackaness W., Steven M., 2006, An Algorithm for Localised Contour Removal over Steep Terrain. The Cartographic Journal, 43(2): 144–156

McMaster R.B, 1992, Generalization in digital cartography, Washington D.C: Association of American Geographers.

Matuk K., Gold C. and Li Z., 2006, Skeleton Based Contour Line Generalization. In Riedl, Kainz, Elmes (eds) Progress in Spatial Data Handling pp. 643–658, Springer Berlin Heidelberg

Melih B., 2002, Model Generalization in GIS, Yildiz Technical University, Department of Geodetic and Photorammetric Engineering

Peucker T.K. and Douglas D.H., 1975, Detection of surface-specific points by local parallel pro-cessing of discrete terrain elevation data. Computer Graphics and Image Processing 4(4):375–387

NOAA, 1997, U.S. Department of Commerce, National Oceanic and Atmospheric Administration: Nautical Chart User's Manual

Oraas S.R., 1975, Automated Sounding Selection, *International Hydrographic Review*, 52(2):103–115.

Robinson A.H., Morrison J.L., Muehrcke P.C., Kimerling A.J. and Guptill S.C., 1995, *Elements of Cartography*, 6th edition, Wiley

Ruas A., 1999, The roles of meso objects for generalization, in proceedings 9th symposium on Spatial Data Handling, Beijing, China, 3b50-3b63.

Ruas A. and Duchêne C., 2007, A Prototype Generalisation System Based on the Multi-agent System Paradigm, Generalisation of Geographic Information: Cartographic Modelling and Applications, W. A. Mackaness, A. Ruas and L. T. Sarjakoski (editor), Chapter 14, pp 269-284

Schumacher M., 2001, Objective coordination in Multi-Agent System Engineering Design and Implementation

Sester. M., 2000, Generalization Based on Least Squares Adjustment, International Archives of Photogrammetry and Remote Sensing. Vol. XXXIII, Part B4. Amsterdam 2000.

Smith B, Mark D.M., 2003, Do Mountains Exist? Towards an Ontology of Landforms. Environment and Planning B: Planning and Design 30(3):411–427

Shea K.S. and McMaster R.B., 1988, Cartographic Generalization in a Digital Environment A Framework for implementation in a Geographic Information system. Proceddings, GIS/LIS'88,Vol 1, 240-249

Weibel R., 1992, Models and Experiments for Adaptive Computer-Assisted Terrain Generalization. *Cartography and Geographic Information Systems* 19(3):133–153

Weibel R. and Dutton G., 1999, Generalising spatial data and dealing with multiple representations, in *Geographical Information Systems: Principles, Techniques, Management and Applications*, 2nd edition, Longley, P., Goodchild, M. F., Maguire, D. J. and Rhind, D. W. editors, Wiley

Weiss G., 1999, Multiagent Systems A Modern Approach to Distributed Artificial Intelligence, The MIT Press, Cambridge

Yan J., Guilbert E. and Saux E., 2012, A conceptual model for submarine feature description and generalisation in nautical chart production, in *Proceedings of the 20th International Conference on Geoinformatics*, Hong Kong.

Zoroaster S. and Bayer S., 1993, Automated Cartographic Sounding Selection, *International Hydrographic Review*, 69(1).