THREE REUSE EXAMPLES OF A GENERIC DEFORMATION MODEL IN MAP GENERALISATION

Gaffuri Julien
Institut Géographique National – COGIT laboratory – Paris-Est University
73, avenue de Paris – 94165 Saint-Mandé Cedex – France
Julien.Gaffuri (at) ign.fr

Abstract

Many geographical data generalisation models have contributed to improve the automation of this complex simplification process. This paper focuses on the agent-based generalisation model GAEL, which provides a generic way to compute continuous object deformations. The principle of this deformation model is to make objects deformable by decomposing them into small constrained objects (segments, triangles, etc.). An object deformation is computed by balancing these constraints. Depending on the constraints choice, the object can be deformed in many different ways. The purpose of this paper is to illustrate how this model can be relatively easily reused for three new applications: building squaring, contour lines displacement, and buildings displacement. We then discuss the notions of genericity and easiness of reuse of generalisation models, and finally propose some methodical points helping to improve these important characteristics of generalisation models.

Introduction

Generalisation is the operation of simplification performed on geographical data when their scale of representation decreases. Because this operation is an important issue for data providers such as national mapping agencies, many research works have been carried out in order to automate it from vector databases. Part of this research deals with the design of generalisation models. A generalisation model is a general framework allowing orchestration of the generalisation process. Ideally, generalisation models have to be as generic as possible in order to be adapted to a wide set of generalisation cases. These cases depend on the initial data (nature of the objects, level of detail, etc.) and the state the target data should have, depending on the user needs (level of simplification, scale of representation, etc.). In most generalisation models, the user's needs are translated into constraints on the data (Beard 1991); these constraints allow a target data state description and are usually used to lead the generalisation process. The generalisation process tries to improve the constraints satisfaction. Among these generalisation models, we can make a distinction between two approaches. Some generalisation models consider the generalisation process as a *discrete* process: input data are transformed through the use of discrete operations performed locally and step by step on the data. For example, Ware & Jones (1998) propose the use of simulated annealing techniques to control the use of discrete transformation on the data. Ruas & Duchêne (2007) use multi-agent techniques to trigger discrete operations to solve local conflicts. Other generalisation models rather consider the generalisation process as a *continuous* process: input data are transformed using a single global deformation of the data. For example, Sester (2000) and Harrie (2001) propose models based on least-square adjustment to compute a deformation of the data. Many other works are based on the finite elements method (Højholt, 2000; Bader 2001).

In this paper, we focus on the GAEL generalisation model proposed by Gaffuri (2008). This agent-based generalisation model has been designed to manage field deformations during the objects discrete generalisation process of Ruas & Duchêne (2007). Among other functionalities, the GAEL model provides a generic way to compute object deformations during the generalisation process. This paper focuses on the genericity and reusability of this model. We present three applications of this generalisation model for specific generalisation operations, in order to give an overview of the genericity of this model and of the way it can be reused. We finally propose a discussion about the generalisation models genericity and reusability.

The GAEL model

The GAEL model (Generalisation based on Agents and ELasticity) has been designed to apply deformation of objects in map generalisation (Gaffuri 2008). Figures 1a, 1b and 1c give an example of such an operation: figure 1a represents an object in its initial state. After symbolisation (figure 1b), two cartographic conflicts are detected: the object overlaps with another gray one above it, and one part of the object becomes unreadable because it is too narrow. The object deformation represented on figure 1c allows these conflicts resolution: the object was deformed in order to avoid the overlapping and to enlarge its too narrow part. Let's now present the three main principles of the GAEL model allowing to get this result:

1. Object decomposition (figure 1d)**:** an object to deform is decomposed into small parts (points, segments, triangles, angles, etc.), called *submicro objects*. A submicro object is a small set of points, whose relative position can be constrained.

2. Use of constraints: a submicro object can bear some constraints on some of its characteristics. For example, segment length, triangle area, and point position can be constrained. Each constraint is characterised by an importance value, which represents how important it is to satisfy it. A submicro constraint is said *preservation constraint* when it constrains the submicro object not to change; it is said *change constraint* when it constrains the submicro object to change. These constraints are assigned to the submicro objects depending on the purpose of the operation. For our example, we choose to constrain the two overlapping points of the object to move away from the gray object (the others points are constrained to preserve their position) and the two triangles of the too narrow part to enlarge (the other triangles are constrained to preserve their area). In the object's initial state, preservation constraints are perfectly satisfied, while change constraints are violated (cf. figure 1e). The deformation is obtained by balancing these constraints violation.

3. Points displacement: in order to get a balance between preservation and change constraints, the object's points are displaced step by step. Each point is modelled as an agent: it tries to move autonomously until a balance position is reached. This balance position is a position which homogenises the submicro constraints violations (cf. figure 1f). The point constraints are the submicro constraints it is involved in. The balance is weighted using the constraints importance values. As a result, the object is deformed and the conflicts resolved (cf. figure 1c).

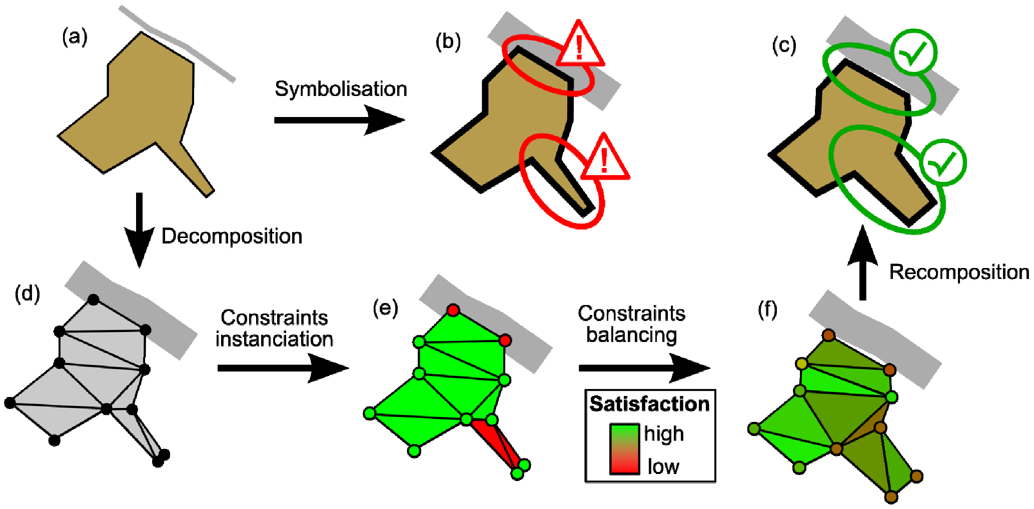

Figure 1. Deformations in the GAEL model

The GAEL model relies on the data model presented figure 2. Objects have the ability to become deformable through a *deform()* method. This method triggers a decomposition of the object into constrained submicro objects, and then a displacement of the points. Depending on the instantiated submicro objects and constraints, different deformation operations can be obtained. Several constraints have been developed: segment length, segment orientation, triangle area, triangle slope orientation, minimum distance between a point and a segment, etc. Thanks to this object-oriented modelling, submicro objects and constraints can potentially be reused for many different cases of application. New kinds of submicro objects and constraints, for specific needs, can be added too. The way to choose and conceive submicro constraints for a specific purpose is detailed in (Gaffuri, 2008).

The previously presented deformation model is used for field deformation in generalisation (Gaffuri, et al., 2008). It allows to ensure consistency between fields (relief, land-use cover, etc.) and geographic objects (buildings, roads, water courses, etc.) during the generalisation process of the geographic objects. In the following part, three other examples illustrate the way this model can be reused and adapted for other deformation operations in map generalisation. For each of them, we present the deformation operation we aim at automating, and then the submicro constraints used for that.

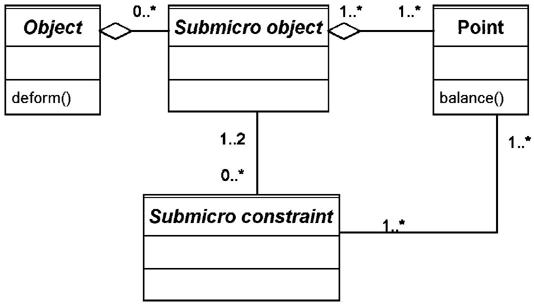

Figure 2. The UML data model

First application: building squaring

Buildings and many other man-made geographical objects usually have right angles. A well-known simplification operation in generalisation consists in *squaring* a building. This operation consists in transforming its almost-right angles into perfectly right angles (cf. figure 3a, 3b and 3c) by rotating the building walls. Airault (1996) proposes an algorithm to automate this operation. We present how the GAEL model is reused to compute this operation. The principle is to make each building deformable, and to constrain some of its segments orientation in order to constrain them to rotate, and so make the almost-right angles perfectly right.

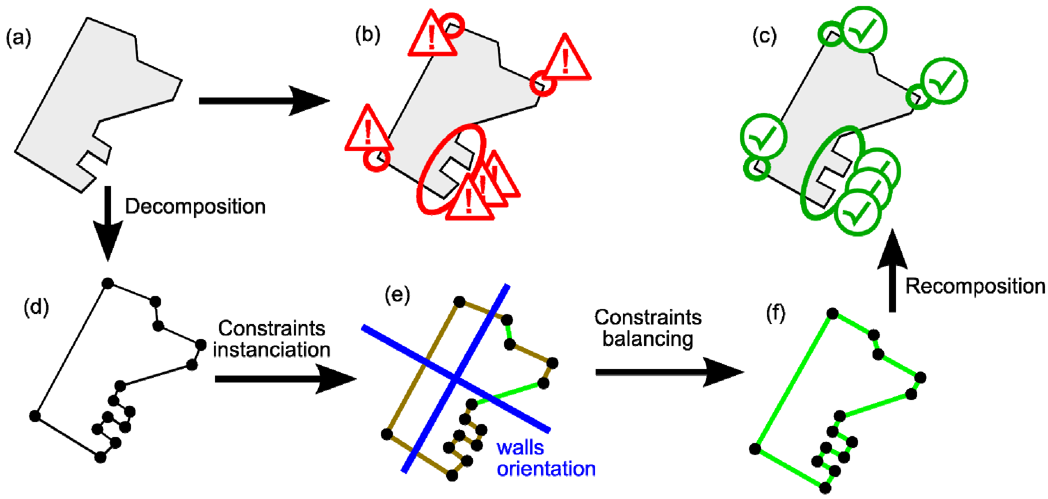

Figure 3. The building squaring operation

Each building becomes deformable by decomposing its contour into segments (cf. figure 3d). Segments lengths are constrained to be preserved. Segments orientations are constrained too, but not always to be preserved: they have to be constrained to rotate until a suitable goal orientation value is reached. This goal value is determined using the *walls orientation measure* proposed by Duchêne, et al. (2003). This measure gives a representative building walls mean orientation value, within [0, π/2[. This orientation can be represented by a cross on the buildings (cf. figure 4). The building walls' orientation value is used to constrain the segment orientations: the segments whose orientation modulo π/2 is closed (more or less a threshold parameter of 15°) to this value are constrained to rotate toward this orientation. The others segments orientations are constrained to be preserved (cf. figure 3e).

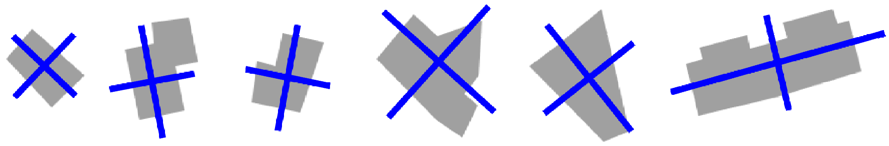

Figure 4. Orientation measure on some examples

In the initial state, when some angles are not perfectly right, some segment orientation constraints are not satisfied (cf. figure 3e). After activation, the points have moved in order to rotate the segments and give the target orientation value to them (cf. figure 3f). As a consequence, many non-right angles have become right (cf. figure 3c). Figure 5 gives some results of this operation on some others example buildings.

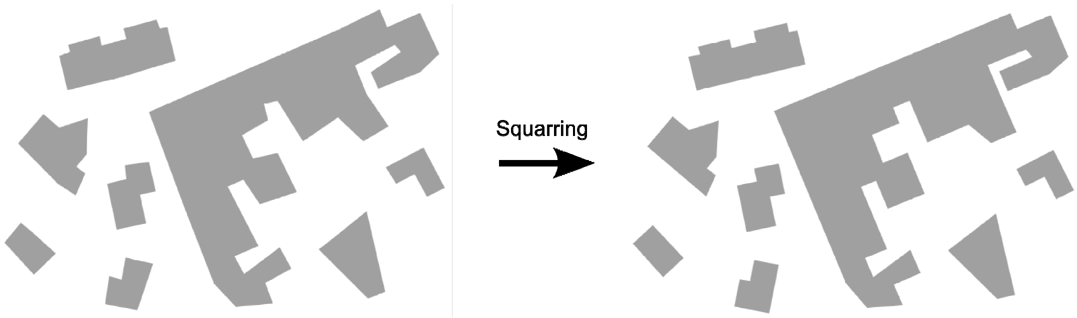

Figure 5. Results of the building squaring operation (with a parameter of 15°)

Second application: contours displacement

When the relief is represented using contour lines, these contour symbols may overlap (cf. figure 6a and 6b). This cartographic conflict often appears in steep areas, especially when the equidistance value is relatively low. A common generalisation operations used to solve this conflict consists in a light deformation of the contours in order to make the gap between consecutive contours big enough and preserve the general shape of the relief (Imhof, 1982). We present in this part how the GAEL model is used to automate this operation.

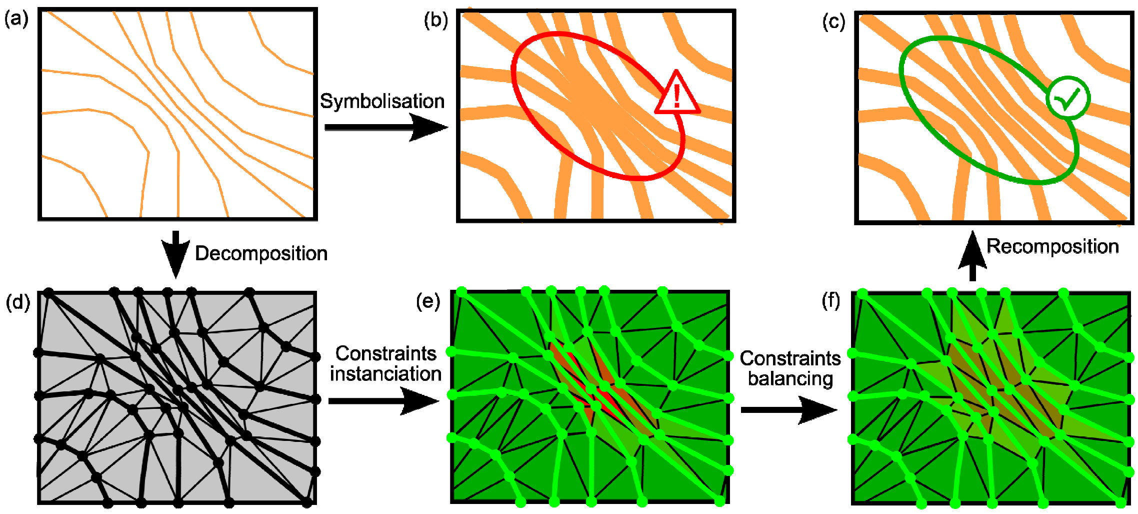

Figure 6. The contours displacement operation

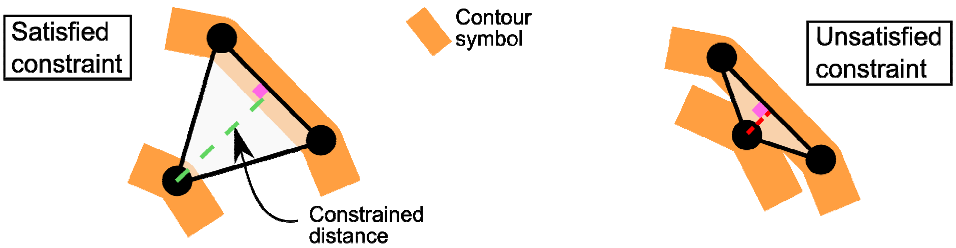

Figure 7. The point – contour triangle minimum distance constraint

The relief is first decomposed into a triangulation, using a Delaunay triangulation constrained by the contour geometries (cf. figure 6d). The submicro constraints used are the following: length and orientation preservation constraints are instantiated on segments composing the contours, area preservation constraints on the triangles, and position preservation constraints on the points. All these constraints are preservation constraints. In order to constrain the overlapping conflicts to be resolved, we use the change submicro constraint presented on figure 7: this constraint is a triangle constraint; its role is to constrain the distance between a contour point and a contour segment of a triangle to be far enough. The distance between them is constrained to be greater than the contour symbol width, plus the separation distance threshold (around 0.15mm).

In the initial state, triangle change constraints in steep areas are violated where the cartographic conflict occurs (cf. figure 6e). After an activation of the points, some points have been displaced and the constraints violation has been homogenised in the steep area (cf. figure 6f), and the conflict resolved (cf. figure 6c). Some further results are presented on figure 8. We notice that this operation is not always suitable, especially in really too steep areas; an equidistance value change, or a contour part deletion as proposed by Mackaness & Steven (2006) become more adapted for these too conflicted situations.

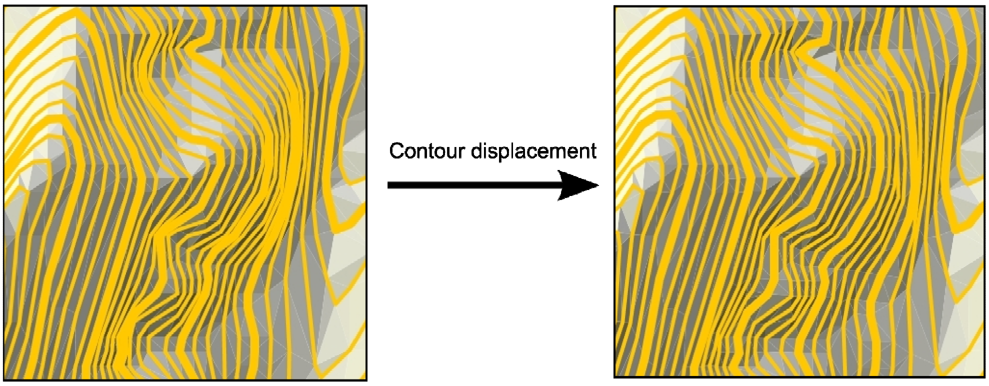

Figure 8. Result of the contours displacement operation

Third application: building displacement

The last presented application concerns the buildings displacement operation. The purpose of this well-known operation is to resolve overlapping conflicts in urban blocks by displacing buildings. Figures 9a, 9b and 9c give an example of such an operation. Many algorithms aiming at automating this operation have been proposed (Ruas, 1998; Ware & Jones, 1998; Bader, 2001; Ai, 2003).

The GAEL deformation model can be applied to this operation. The decomposition used is a Delaunay triangulation of the buildings centre points and the projections of these centre points on the roads (cf. figure 9d). This triangulated structure is the same as the one proposed by Ruas (1998). The submicro preservation constraints used for this application only concern the segment length and orientation – points composing the surrounding road are fixed (cf. figure 9e). In order to solve the conflicts, we use the segment change constraint described on figure 10: the principle is to constrain a segment linking two too closed or overlapping objects (either building or road) to lengthen to minimise the “collision” area (represented in red) between them. When this area is null, this constraint preserves the segment length.

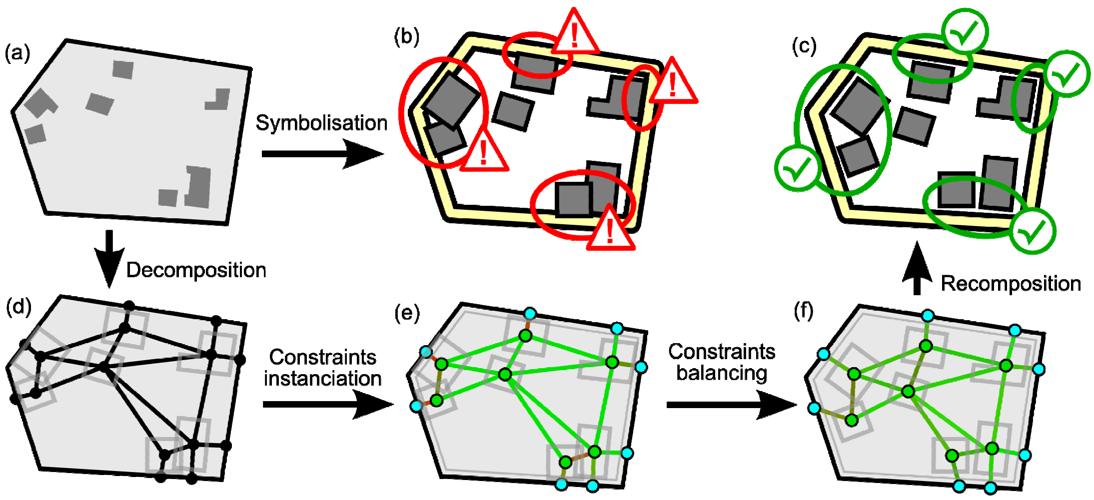

Figure 9. The building displacement operation

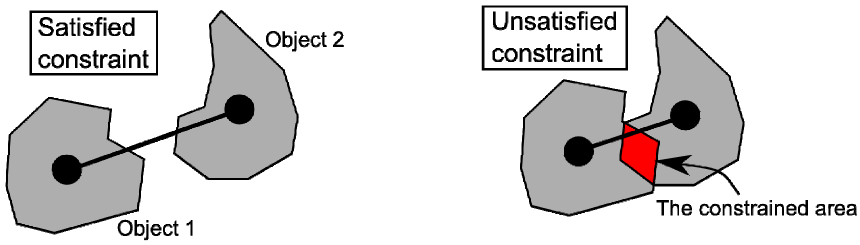

Figure 10. The segment non overlapping submicro constraint

The result after point's displacement given on figures 9f and 9c shows a resolution of the cartographic constraints. Note that the buildings in the centre of the block, even if they were not involved in any conflict, have been displaced; this displacement has allowed a preservation of their relative position with their neighbours. Figure 11 gives another result of this operation.

Figure 11. Result of the building displacement operation

Discussion and conclusion

In this paper, we have presented three applications of the GAEL generalisation model. We have shown how an easy reuse of existing components (submicro constraints) of the model has allowed tackling well-known generalisation issues. We have presented how the conception of two new submicro constraints (one triangle constraint for the contour displacement operation, and one segment constraint for the building displacement operation) allows for a wider use field of the presented model. This work allows an illustration of two important characteristics of generalisation models:

- **Genericity**: as a model, a generalisation model has to be applicable not only to a narrow set of different instances of the problem to solve, but to an as wide as possible one. For the case of generalisation, the set of generalisation problems depends on the type of input data and user needs. Because it is not possible to list and test exhaustively and precisely these possible applications, the genericity level of a generalisation model is almost impossible to measure objectively. Usually, the conception of a generalisation models is motivated by a specific problem to solve, and is supposed to be adaptable to a more or less wide set of cases presenting the same characteristics: generalisation models can be much more suitable for some specific types of data (polygonal, networks, small objects, topographic or thematic data, etc.), for specific scale changes (from 1:10k to 1:15k, from 1:10k to 1:250k, etc.), or for specific uses of the data (visually impaired maps, hiking maps, etc.). For example, Bader (2001) proposes the use of an elastic beams structure for network generalisation; this model has successfully been adapted to buildings block generalisation for small scale changes, and could also certainly be adapted for other cases of use. We argue that a completely generic system is impossible to conceive. Such a system would certainly face what we could call a "Swiss knife effect": this effect refers to the possibility for a tool to be able to do a lot of things, but not always in a perfect way, like a Swiss knife does. Usually, high level of genericity is a synonym of low performance. We rather think that the genericity of models could be improved not by conceiving always more and more complex models, but rather by trying to combine the advantages of already existing models (Duchêne & Gaffuri, 2008; Touya, 2008).

- **Easiness of use**: this paper presents generalisation model reuse examples. The easiness of reuse of a generalisation model is an important characteristic: if a model is too hard to reuse or adapt for new applications, its genericity is useless. We argue that an object/agent-oriented conception of the models helps improving its reusability. As presented in this paper, the independent conception of the generalisation model components allows to dispose of an easily reusable components library - submicro constraints have been reused for the new applications. As proposed by the AGENT project generalisation model (Ruas, 1999), we think the most useful components are algorithms, data enrichment methods, constraints and generalisation engines. All these components can be conceived more or less independently. A user of a generalisation system could only reuse them through a kind of "generalisation API" in order to build its own generalisation framework adapted to its map production flow line. Depending on its skills and knowledge of the model, he could have the possibility to conceive a missing component for his need, which could be reusable by others and enrich the components library.

Genericity and easiness of use are not always discussed in map generalisation models presentations. Some tests of generalisation softwares supervised by EuroSDR (Burghardt, et al. 2007) have been carried out in order to evaluate these softwares. Genericity was one of the tested characteristic of these softwares, while different kinds of data where used (urban, mountainous and costal data) - easiness of use too, because two kind of users (novice and expert) where involved in these tests. We think that this kind of work will allow an advanced study of the limits in term of genericity and easiness of use of the existing generalisation systems, in order to progress towards always more and more efficient data generalisation models.

References

S. Airault (1996). `De la base de données à la carte: une approche globale pour l'équarrissage de bâtiment'. *Revue Internationale de Géomatique* **6**(2-3):203-217.

M. Bader (2001). *Energy minimization methods for feature displacement in map generalisation*. Ph.D. thesis, Zurich university, Department of geography.

K. Beard (1991). `Constraints on rule formation'. In B. Buttenfield & R. Mcmaster (eds.), *Map generalisation: making rules for knowledge representation*, pp. 121-135. Longman Scientific and Technical.

D. Burghardt, et al. (2007). `Methodologies for the evaluation of generalised data derived with commercial available generalisation systems'. In *Workshop on Generalisation and Multiple Representation*.

C. Duchêne, et al. (2003). `Quantitative an qualitative description of building orientation'. In *fifth workshop on progress in automated map generalisation*, Paris, France.

C. Duchêne & J. Gaffuri (2008). `Combining Three Multi-agent Based Generalisation Models: AGENT, CartACom and GAEL'. In A. Ruas & C. Gold (eds.), *Headway in Spatial Data Handling*, pp. 277-296.

J. Gaffuri, et al. (2008). `Object-field relationships modelling in an agent-based generalisation model'. In *ICA workshop on generalisation and multiple representation*, Montpellier, France.

J. Gaffuri (2008). *Généralisation automatique pour la prise en compte de thèmes champs: le modèle GAEL*. PhD thesis in geographical information science, Paris-Est university.

L. Harrie (2001). *An optimisation approach to cartographic generalisation*. Ph.D. thesis, Lund university, Lund institute of technology, department of technology and society.

P. Højholt (2000). `Solving space conflicts in map generalisation: using a finite element method'. *Cartography and Geographic Information Science* **27**(1):65-74.

E. Imhof (1982). `Generalisation of contours'. In *cartographic relief representation*, chap. C, pp. 126-139. ESRI Press.

W. A. Mackaness & M. Steven (2006). `An Algorithm for Localised Contour Removal over Steep Terrain'. *The Cartographic Journal* **43**(2):144-156.

A. Ruas (1998). `A method for building displacement in automated map generalisation'. *International Journal of Geographical Information Science* **12**(8):789-803.

A. Ruas (1999). *Modèle de généralisation de données géographiques a base de contraintes et d'autonomie*. Ph.D. thesis, Marne la Vallée University, COGIT laboratory.

A. Ruas & C. Duchêne (2007). `A prototype generalisation system based on the multi-agent system paradigm'. In W. A. Mackaness, A. Ruas, & L. T. Sarjakoski (eds.), *Generalisation of Geographic information: cartographic modelling and applications*, chap. 14, pp. 269-284. Elsevier.

M. Sester (2000). `Generalization based on least square adjustment'. *International archives of photogrammetry and remote sensing* **23, part B4**:931-938.

G. Touya (2008). `First thoughts for the orchestration of generalisation methods on heterogeneous landscapes'. In *workshop on generalisation and multiple representation*, Montpellier, France.

M. J. Ware & C. B. Jones (1998). `Conflict Reduction in Map Generalization Using Iterative Improvement'. *GeoInformatica* **2**(4):383-407.