



HAL
open science

CollaGen: Collaboration between Automatic Cartographic Generalisation Processes

Guillaume Touya, Cécile Duchêne

► **To cite this version:**

Guillaume Touya, Cécile Duchêne. CollaGen: Collaboration between Automatic Cartographic Generalisation Processes. 25th International Cartographic Conference, Jul 2011, Paris, France. pp.541-558, 10.1007/978-3-642-19143-5_30 . hal-02302032

HAL Id: hal-02302032

<https://hal.science/hal-02302032>

Submitted on 1 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

***CollaGen*: Collaboration between Automatic Cartographic Generalisation Processes**

Guillaume Touya, Cécile Duchêne

Laboratoire COGIT- IGN France,
73 avenue de Paris 94165 Saint-Mandé Cedex France

Abstract Cartographic generalisation seeks to summarise geographical information to produce legible maps at smaller scales. Past research led to the development of many automated cartographic generalisation processes, each one being more or less specialised to a particular problem: a landscape like urban areas, a data theme like land use, a cartographic conflict like linear symbol overlap or most of the time of mix of the three. This paper deals with the development of a model allowing collaborative generalisation i.e. the collaboration between automatic processes like these in order to tackle the generalisation of a complete map. CollaGen, our proposed model, allows to partition data in geographic spaces and to find to best suited process to generalise each space. The applications of a process on a space are automatically orchestrated. Interoperability between processes is managed thanks to formal constraints and side effects are monitored after each process application. Results from CollaGen prototype are shown and discussed.

1. Background and Objectives

Cartographic generalisation seeks to summarise geographic data to produce legible maps at smaller scales. The automation of cartographic generalisation would make the production of map series easier as well as it would allow quality on-demand mapping. The past twenty years of research in the generalisation domain have led to the development of many different and complementary automatic models and processes. (Barrault et al. 2001, Harrie and Sarjakoski 2002, Duchêne 2004, Bader et al. 2005, Haunert 2007) are a small sample of the available cartographic generalisation processes.

If so many processes have been developed over the years, it is due to the impossibility to solve the complex problem of generalisation with a single process. Indeed, every process is only completely relevant for a limited part of the generalisation problem. Some processes are well adapted to particular *landscapes*: AGENT (Ruas 1999, Barrault et al. 2001) is designed for urban generalisation while GAEL (Gaffuri 2007) may be specialised to deal with high relief landscapes. Moreover, some are only relevant for the generalisation of a specific data *theme*: (Haunert 2007) is dedicated to land use generalisation for instance. Added to that, some are relevant for solving a limited part of the cartographic *conflicts* resulting from scale change: for instance, the simulated annealing process of (Ware et al. 2003) is designed for solving proximity conflicts. Finally, some mix the three previous cases: the Elastic Beams (Bader et al. 2005) are relevant for road overlap conflicts (*theme* and *conflict*).

Either automatically producing map series or on-demand mapping requires to be able to generalise any landscape, data theme or solve any necessary kind of conflict, which is not possible using a single existing process. Rather than developing a new complete generalisation process, which seems a bit rash, the objectives of our research is to benefit from the existing processes and make them work together. We propose a new framework, *Collaborative Generalisation* (CG), to make processes collaborate to correctly generalise a entire map.

The second part of the paper describes the CG approach and the CollaGen model. The third part focuses on the results obtained with CollaGen. The fourth part draws some conclusions and proposes future plans.

2. Approach and Methods

2.1. The Collaborative Generalisation Framework

Automatic generalisation research first tried to answer to the questions “why, when and how to generalise?” (McMaster and Shea 1988, Brassel and Weibel 1988). Inspired from (Regnauld 2007) and (Duchêne and Gaffuri 2008), the CG framework we define aims at allowing an answer to the question *why*, *when* and *how* to apply *which* automatic process? Within this framework, automatic generalisation processes are applied on parts of space where they are expected to be efficient while side effects are likely managed at generalisation neighbourhood (Fig. 1).

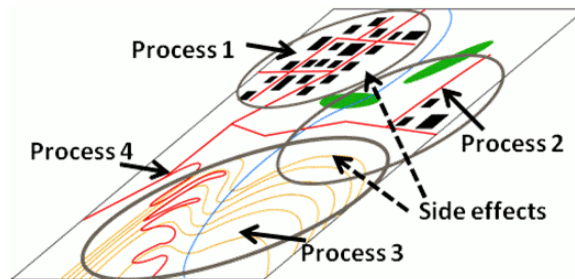


Fig. 1. The collaboration principle between generalisation processes. A process 1 is carried out on the town area, a process 2 on the rural area, and then a process 3 on the mountain area and finally a process 4 is carried out on the road network. Side effects are corrected at the neighbourhood (dashed arrows) of application spaces.

2.2. Overview of CG Framework Components

Generalising data within the CG framework brings about specific problems like process interoperability, treatment heterogeneity or side effects (Touya 2008). The framework function analysis lead to a six main components and three resources groups structure (Fig. 1.2). *Partitioning* builds the *geographic spaces* where the *available generalisation processes* can be applied. The *Translator* parameterises the processes. The *Registry* chooses the process to generalise a given space. The *Observation* provides online evaluation. *Side effects* are managed by the eponymous component. Finally, the *Scheduling Component* orchestrates the whole process.

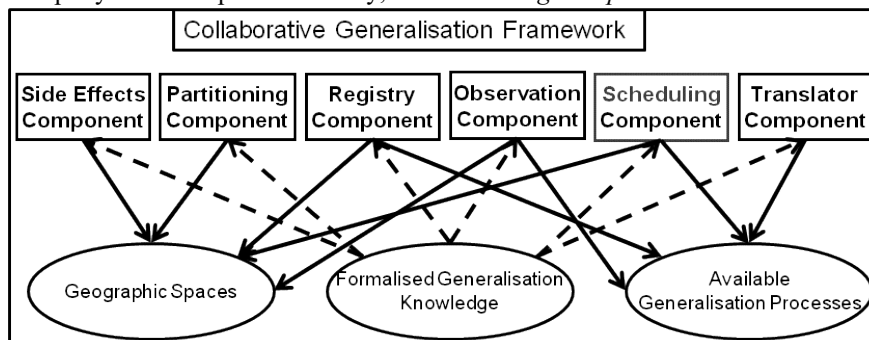


Fig. 1.2. The main Components (rectangles) and Resources (ellipses) of a Collaborative Generalisation framework and how the components act on the resources (plain arrows). Dashed arrows show that formal knowledge is used by each component.

Within the CG framework, we developed the CollaGen model that implements all the aspects of collaborative generalisation defined in Fig. 1.2. The next parts describe how each aspect is managed in CollaGen.

2.3. The CollaGen Model

2.3.1. Automatic Generalisation Processes

CG consists in making several available automatic processes collaborate to optimise the entire map generalisation. Thus, we consider as an available automatic generalisation process, any process that can be triggered on geographic data from the software platform the CG framework is developed on. A process is a computer program that automatically triggers generalisation operator (e.g. simplification, displacement...) sequences on geographic objects. For instance, in CollaGen, developed on a research platform (Renard et al. 2010), AGENT (Ruas 1999), CartACom (Duchêne 2004), GAEL (Gaffuri 2007), Least Squares (Harrie and Sarjakoski 2002) or road selection processes are implemented and thus available for collaboration. Processes published as web services (Regnauld 2007, Neun et al. 2008) could also be considered as available processes.

2.3.2. Geographic Spaces

We define a geographic space as a geographically meaningful extract of the data that can be a relevant input for a given generalisation process (Touya 2010). The use of geographic spaces in CollaGen is useful for both optimising the use of the existing generalisation processes and partitioning the data to avoid the processing of very large datasets. The geographic spaces (Fig. 3) can be areal (e.g. urban or rural area), thematic (e.g. road network or vegetation) or both areal and thematic (e.g. mountain roads).

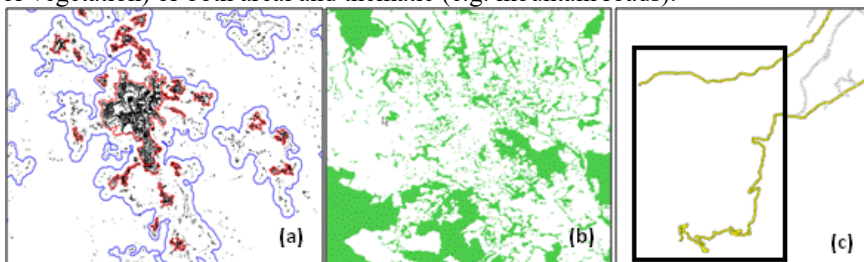


Fig. 3. (a) buildings in black, urban areas in red and rural areas in blue. (b) vegetation thematic space on the same area. (c) mountain roads space (in the rectangle).

It can be noticed that with such a definition, the geographic spaces do not form a mathematical partition as metric spaces can overlap and thematic spaces cross metric spaces. Spaces can be cut in several portions in order to keep small spaces and minimise processing time.

Moreover, some emerging spaces can be managed by CollaGen: they are sub-spaces where conflicts remain unsolved. During the generalisation of a space by a given process, the observation component can identify conflict clusters (close conflicting objects) that emerge as sub-spaces to be generalised by another process than the one processing the whole space (§2.3.9).

2.3.3. Formalised Knowledge in CollaGen

Formalised cartographic generalisation knowledge is necessary to allow process collaboration. The model designer (e.g. we are the CollaGen model designer) has to provide a *generalisation ontology* and *sequencing rules*; a process developer (the one that makes a new generalisation process available for collaboration) has to provide a *process description*; the user (the one that generalises data) has to provide *generalisation constraints* and *operation rules* (Fig. 4).

- The *generalisation ontology* is the support for sharing a common vocabulary in the collaboration model. It helps to express that map specification (“buildings smaller than 50 m² must be deleted”), a data type (“BATIMENT” class) or a process requirement (“agent_building” in AGENT process) deal with the same concept of ‘building’ for instance.
- The *sequencing rules* are guidelines for sequencing applications of processes on geographic spaces. As in the Global Master Plan of (Ruas and Plazanet 1996), it allows to formalise that ‘road selection’ should be triggered before ‘urban generalisation’ for instance, or that ‘urban areas’ should be generalised before ‘rural areas’.

- The *generalisation constraints* and the *operation rules* formalise the map specifications. The formal model summarises past research (Beard 1991, Stöter et al. 2007, Duchêne and Gaffuri 2008) and allows to express different constraints like ‘building area > 0.2 map mm²’, ‘building block density should be preserved’ or ‘very concave buildings should maintain initial concavity with 10% margin’.
- A *process description* formalises the capabilities of a generalisation process. As for web services composition, capabilities are described with pre and post conditions (Lutz 2007). In CollaGen, pre conditions are adapted spaces for the process and post conditions are a priori satisfied generalisations constraints and operation rules (after the process has been applied).

□ Provided by the model designer ■ Provided by the user □ Provided by the process developer

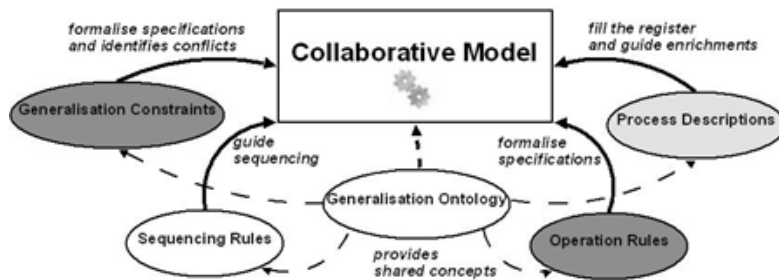


Fig. 4. A diagram of the 5 parts of formalised knowledge and their use in the collaborative model. The dashed arrows show that the Ontology provides shared concepts to every part.

(Touya et al. 2010) describes how each piece of formalised knowledge is modelled in CollaGen and how it can be acquired by a user, a process developer or the model designer.

2.3.4. The CollaGen Workflow

CollaGen proposes a workflow for CG to chain the components actions (Fig. 5).

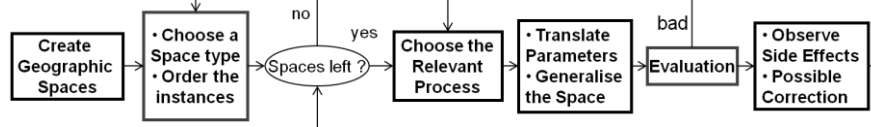


Fig. 5. Simplified view of the sequence of CollaGen generalisation (emergence mechanism is not included). Each box is the action of one component.

As a simplified example, let us imagine an area that is firstly partitioned (into two rural spaces, an urban space and the road network), and three processes (AGENT, CartACom and Beams). Then, the urban space is chosen and the registry chooses AGENT. The process is parameterised according to the constraints and generalised. Generalisation is evaluated as good and there is no side effect. Then, the next chosen space type is rural and the two instances are ordered. CartACom is chosen and generalises the two instances correctly without side effects. Finally, the Beams generalise the remaining space (road network) but generates side effects by overlapping buildings. Then, the side effects are corrected by a specific process (Least Squares here) and the CollaGen workflow is finished. An implemented version of this example is illustrated in Fig. 15.

2.3.5. Partitioning Component

The *partitioning component* is responsible for the creation of the geographic spaces as additional data (Touya 2010). Thus, the component has to be fed with spatial analysis algorithms able to outline the required spaces. For instance, algorithms to identify urban, suburban, rural, costal, mountain areas were implemented, among others.

2.3.6. Translator Component

The *translator component* provides three kinds of services to translate inputs and outputs of the processes in the language used to convey interoperability: the formal constraints and the ontology (Touya et al. 2010). First, the translator allows process interoperability making the constraints the only input and output of every process (Fig. 6). A translating function is provided for every process and transforms the constraints and operation rules into the specific parameters of the process (e.g. numeric thresholds for a river selection, specific constraints for

AGENT or CartACom, equations on coordinates for Least Squares or additional attributes for the Elastic Beams).

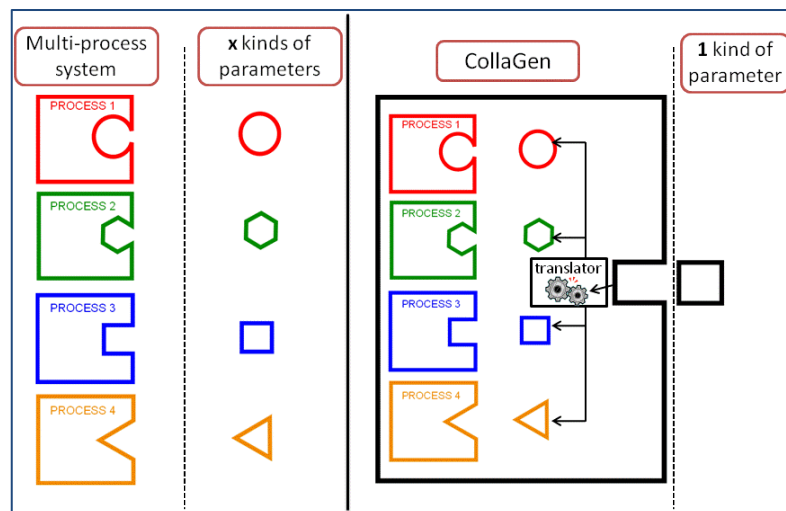


Fig. 6. CollaGen allows interoperability with the use of a single output, constraints and the translator.

The translator also serves as a *registration mapping* (Lemmens 2008) to tag the geographic data with the corresponding ontology concept: it makes the mapping between the “IGN_BUILDING” data class and the “building” ontology concept for instance (Touya et al. 2010). Finally, the translator allows to map the formal constraints with measurement algorithms that compute the constraint current value and satisfaction for a given object as in AGENT (Barrault et al. 2001). The use of such mechanism is detailed in §2.3.9.

2.3.7. Registry Component

The *registry* serves as yellow pages service to choose a relevant process to generalise a given space at a given time. To a request like ‘what are the best available processes to generalise rural space n°x?’ the registry responds with a list of applicable processes sorted by relevance (e.g. ‘CartACom, Least Squares, AGENT’).

To build the list, the registry first selects the relevant processes according to their description pre-condition: if the pre-condition, a list of space types with relevance rate (e.g. ‘urban area 4/5’) contains the request space type, the process is selected; then it is rated according to relevance rate.

In a second step, the processes are reordered according to the description post-condition, the list of a priori satisfied constraints after generalisation: a ratio is computed between the post-condition and the occurrences of constraints inside the space (Fig. 7). The ratio weights the relevance rate, reordering the processes.

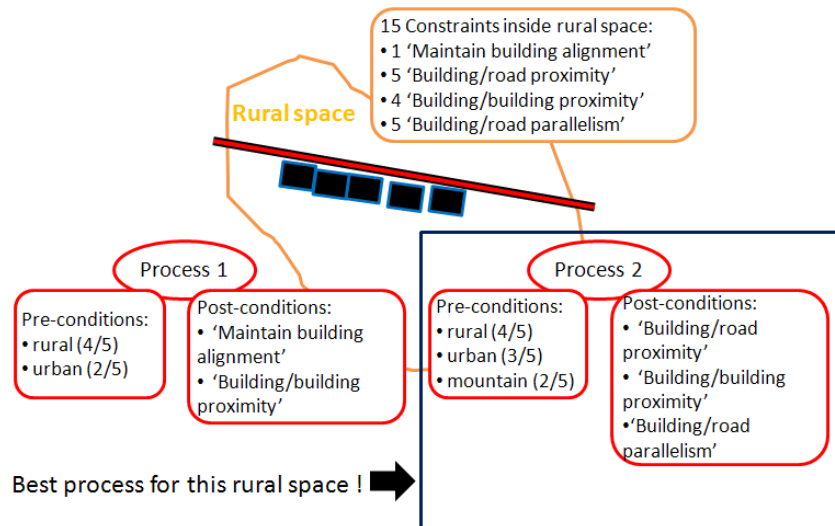


Fig. 7. Illustration of the registry response according to the constraints inside a given space: process 2 matches 14/15 constraints against 5/15 for process 1.

2.3.8. Scheduling Component

The *scheduling component* orchestrates the generalisation of spaces by processes. After every generalisation, it decides what to do next: it chooses the next type of geographic space to generalise and then orders the instances of this type. As in a *Global Master Plan* (Ruas and Plazanet 1996) but here rule-based, the space type (urban, rural...) is chosen according to the active *sequencing rules*. If it is not enough to choose, the space type whose instances have the highest conflict mean is selected first. Once the type chosen, the instances are ordered by conflict importance: the most conflicting ones are peeked (Fig. 8).

Generalisation is considered in CollaGen as a four step operation: geometry changes (e.g. area-to-point collapses), selection, cartographic and graphic generalisation (Harrie and Sarjakoski 2002). The processes are described as contributing to one or more of these steps and can be used only during the right steps. The scheduling component chains these steps according to the rules.

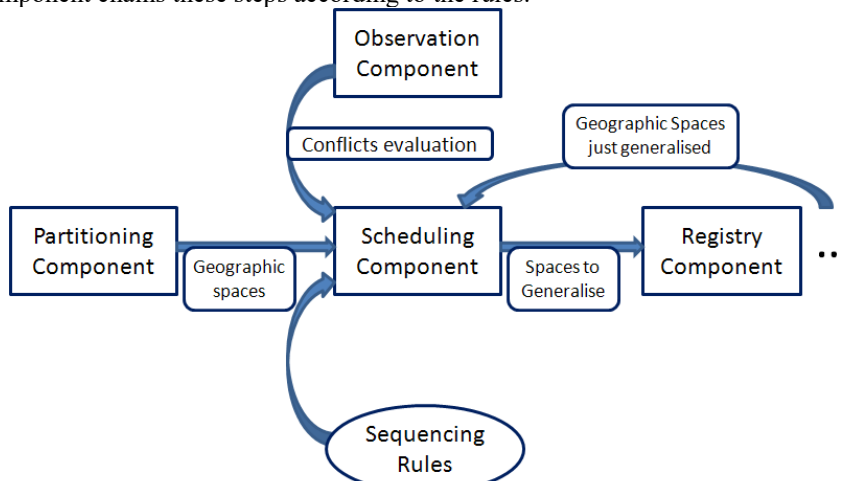


Fig. 8. Interactions of the scheduling component with other components and resources.

Moreover, the scheduling component provides state management to allow local and global corrections (Fig. 9). When a generalisation is badly evaluated, the scheduling may cancel it and go back to previous states of data. To allow this try/error mechanism, as in (Zhou et al. 2008), the initial state stores the attribute data while all the geometry states are kept linked to the generalisation pair (space/process) that led to the following step.

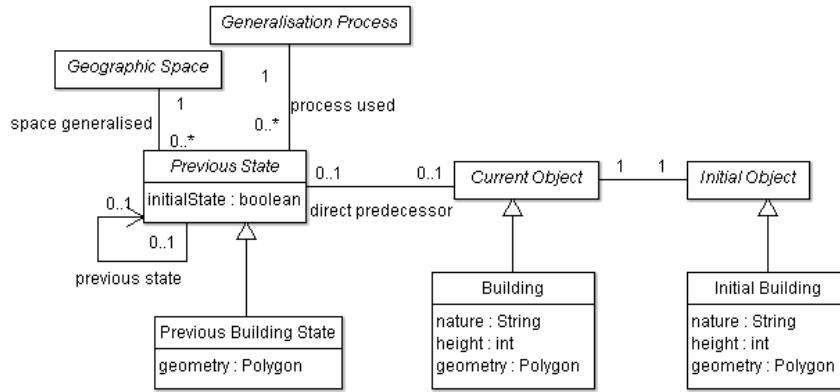


Fig. 9. UML class diagram of the state management system of the scheduling component.

2.3.9. Observation Component

Generalising a space with the best available process does not guarantee a complete success. Conflicts clusters may emerge during generalisation and the observation component allows to detect them as *conflicting areas* (Duchêne and Touya 2010). Therefore, the component observes the generalisations online and evaluates the progress. The observation component analyses the conflict areas, pauses the process when some are too big, extends the areas to subserve solving, triggers the registry component to propose a local solution and once the local conflicts are solved, resumes to the interrupted generalisation.

The online evaluation embedded in the emergence mechanism, as well as the global evaluation performed on a geographic space after generalisation rely on the constraints to guarantee homogeneous evaluation for every process. To enable this monitoring of the constraints against the data, *located constraints monitors* (LCM) are added on each object concerned by constraints as in (Barrault et al. 2001): if “granularity” and “size” constraints are defined on buildings, granularity and size LCMs are added for every building. LCMs are able to give the satisfaction of the constraint for the given building.

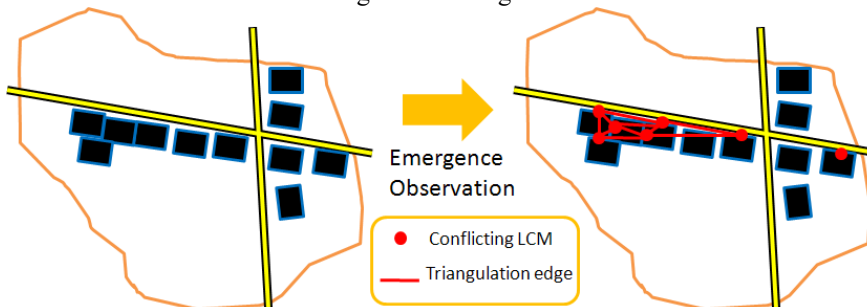


Fig. 10. Conflicting LCM are clustered by triangulation to observe ‘conflicting area’ emergence.

The located constraints are provided with a geometry (Fig. 10) that allows the spatial clustering necessary for the emergence mechanism. The geometry also allows to quantify the LCMs *inside* a geographic space so as to evaluate globally the space generalisation from the distribution of individual evaluation. Progress means less unsatisfied LCMs in the distribution while good generalisation means few very unsatisfied LCMs and lots of very satisfied LCMs.

2.3.10. Side Effects Component

Within the CG Framework, generalising a space may cause additional conflicts just outside the space: for instance, a building is moved too close to a building that was just outside (thus not managed by generalisation). We call such additional conflicts side effects. In order to detect and correct side effects, the neighbourhood of

each space is defined depending on the topological relation shared by two spaces (adjacency or overlap) (Touya 2010).

Following the principle of the LCM evaluation, the side effects management is based on *consistency* constraints located in each space's neighbourhood. Consistency constraints are kind of integrity constraints that guarantee the consistency of data before and after the generalisation of a space. Three types of consistency constraints can be identified: the *inter-space* relational constraints, the *non-existence* relational constraints and the *operation consistency* constraints.

The *inter-space* relational constraints are relational LCMs (LCM on a geographic relation between two objects) concerning an object inside the space and an object outside the space (Fig. 11). The figure "relative position relation" has a building in rural space 1 and the other in rural space 2. The "relative orientation relation" is an inter-space relation in the road network space point of view (building outside the space) but not in the rural space 2 point of view as the road is inside the space.

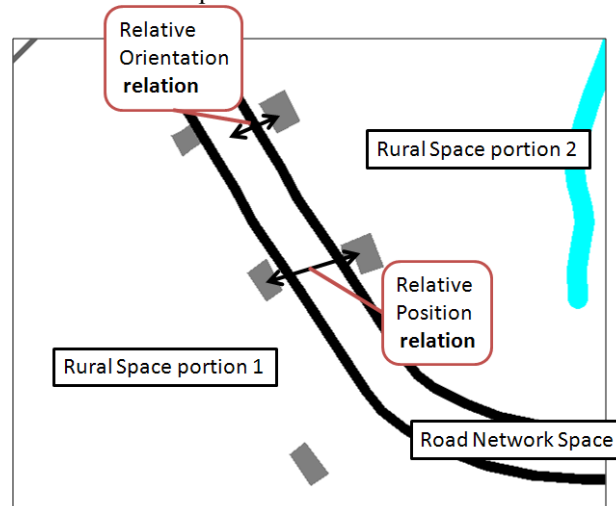


Fig. 11. Two examples of inter-space relations with two rural space portions separated by a road: a relative position relation that should be preserved after each rural space is generalised and relative orientation relation (shared by rural space 2 and the road network space) that should be preserved if the road network is generalised.

Then, *non-existence* relational constraints check that no additional *inter-space* relation has been created by generalisation.

The *operation consistency* constraints affect the intersection neighbourhood. If the intersecting space has already been generalised, these constraints check that the second generalisation is not inconsistent with the first one. For instance, if a building has been moved one way, it should not be moved the other way round. The operation consistency constraints are based on the previous states of the objects stored by the scheduling component.

Finally, if consistency constraints are violated, side effects correcting processes are triggered. Side effects correctors are monitored by consistency constraints as processes are monitored by constraints. GAEL (Gaffuri 2007), Elastic Beams (Bader et al. 2005) or Least Squares (Harrie and Sarjakoski 2002) can be used to correct side effects (Fig. 12) as well as diffusion processes (Legrand et al. 2005). If no balanced solution can be found, the component can arbitrate by choosing a solution (before or after 2nd generalisation) like a legislator (Ruas 2000) or by relaxing some less important constraints like a controller (Ruas 2000).

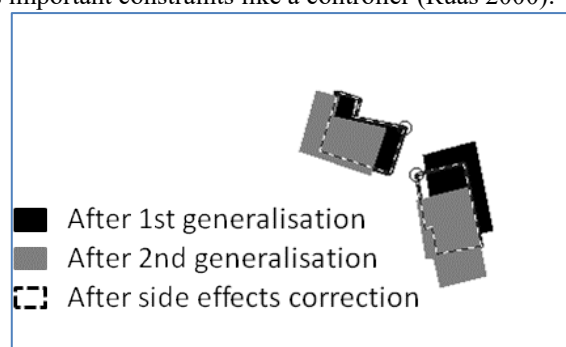


Fig. 12. Example of side effect correction by least squares adjustment : a balance between both generalisations.

3. Results

The CollaGen model is not fully implemented but some experiments were carried out on French topographic data (1:15k reference scale) to produce a 1:50k map on a large area containing heterogeneous landscapes (shore, city, country, mountain). A standard set of constraints and rules, extracted from NMA experience, is used as specification (80 constraints and rules). The available processes are the ones implemented on CartAGen platform (Renard et al. 2010): AGENT, CartACom, GAEL, Least Squares, Elastic Beams and network selection processes. Fig. 13 shows the steps of a rural area generalisation: the rural space is generalised by CartACom and the road network by the Beams generating a side effect conflict.



Fig. 13. (a) initial rural area. (b) rural space generalised by CartACom. (c) Roads generalised by the Beams with a side effect on the circled building.

Fig. 14 shows the registry answer to a rural space request: three processes match the pre-condition but CartACom is ranked first as it better matches the constraints (a 10 constraints caricatural sample is used) a priori (97% against 59 for Agent and 6% for the Least Squares).

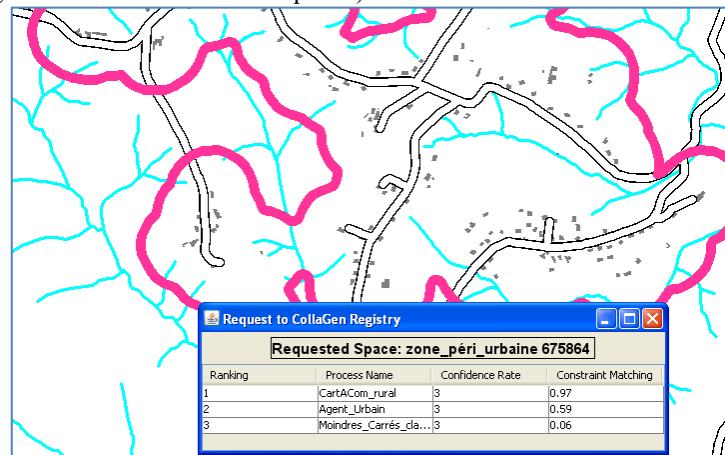


Fig. 14. The registry answer to a request from the delineated rural space. CartACom is ranked first as it better suits the constraints than Agent or the Least Squares processes.

Fig. 15 shows a CollaGen result (without side effect correction) on a small area with urban, suburban and rural spaces.

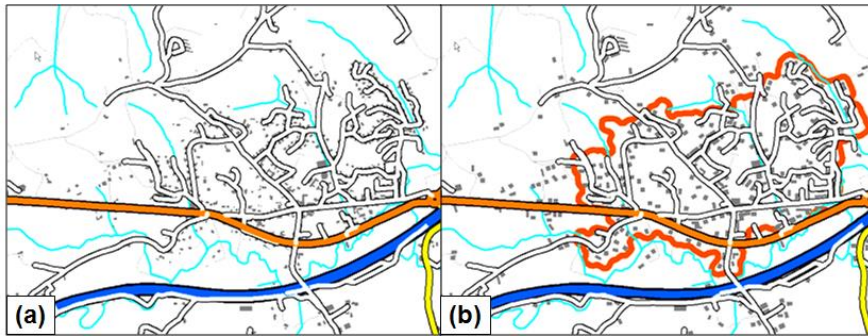


Fig. 15. (a) initial data. (b) urban space generalised by AGENT, then rural and rural spaces by CartACom and roads by the Beams (urban space is delineated with the thick outline).

Fig. 16 illustrates the importance of choosing the best process and the best order. The remaining conflicts in picture (2) show that the choices are not as good as pictures (3) and (4) ones. Moreover, the order of picture (4) is better than in picture (3) as it did not result in side effects. Thus, the sequencing component has to be careful in choosing the order!

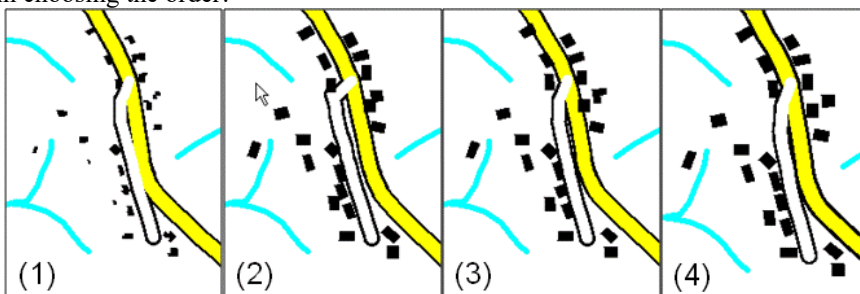


Fig. 16. Illustration of order and registry importance: (1) before generalisation. (2) generalisation with AGENT then Least Squares process. (3) generalisation with CartACom then the Beams. (4) generalisation with the beams then CartACom.

Finally, Fig. 17 shows observed emerging conflicting areas in a space too dense for CartACom and the conflict correction using Least Squares.



Fig. 17. Conflict emergence from CartACom generalisation. (a) ungeneralised data. (b) emerging conflict areas are detected after generalization. (c) zoom on a conflict area extended to better solve the conflict. (d) conflicts solved by least squares generalization.

In order to evaluate CollaGen contribution, we used it to generalise the benchmark dataset from EuroSDR generalisation state-of-the-art project (Stoter et al. 2009) and compared the results to the best ones obtained during the tests with CPT and Clarity™ software (Fig. 18). Although the seven processes used could be tuned to improve individual results, the comparison shows the pros of CG and CollaGen, particularly in the southwest suburban part that has to be generalised differently from the town area.

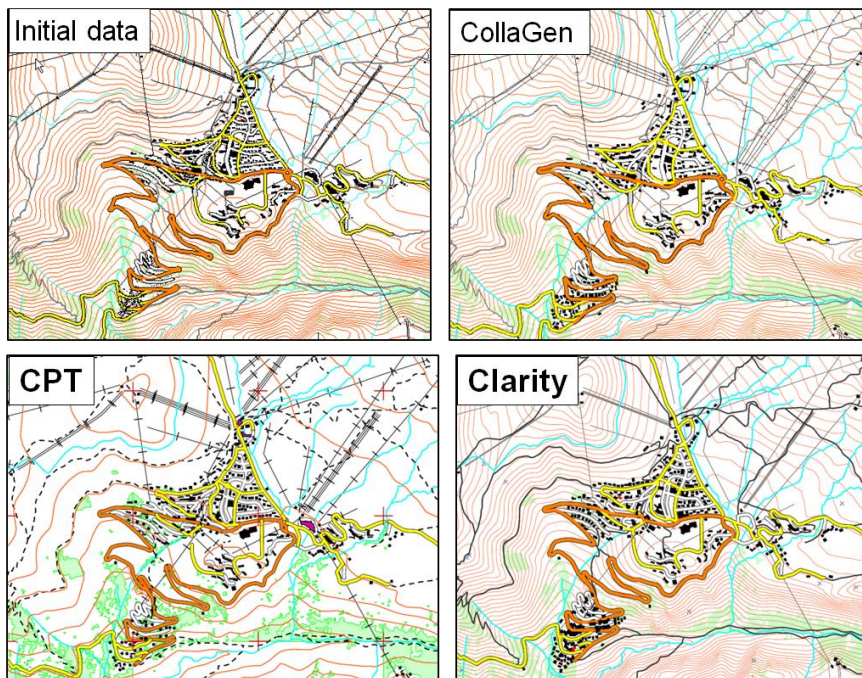


Fig. 18. A mountainous French dataset from EuroSDR tests (Stoter et al. 2009) generalized with CollaGen compared to the best results from the tests.

Conclusion and Future Plans

This paper introduced a new framework to perform automatic cartographic generalisation by making optimised use of past research processes: Collaborative Generalisation. Within this framework, the CollaGen model allows to sequence interoperable generalisations of geographic spaces by the relevant available process. Among CollaGen contributions, a formal constraints model, a generalisation domain ontology, online evaluation and side effects detection mechanisms can be noticed.

A lot can still be made to improve both CollaGen and the Collaborative Generalisation Framework. The side effects correction really need to be more investigated to know how far corrections can be made without undoing what was previously generalised. In-depth testing (different data and scale change) of CollaGen and each component is also necessary to identify remaining issues. Moreover, rather than being implemented on a platform, the available processes could be called as web services as proposed by (Regnauld 2007).

References

- Bader M, Barrault M, Weibel R (2005) Building displacement over a ductile truss. *International Journal of Geographical Information Science* 19 (8): 915-936
- Barrault M, Regnaud N, Duchêne C, Haire K, Baeijs C, Hardy P, Mackaness W, Ruas A, Weibel R (2001) Integrating multi-agent, object oriented and algorithmic techniques for improved automated map generalization. In: 20th International Cartographic Conference, Beijing
- Beard K (1991) Constraints on rule formation. In: Buttenfield B, McMaster R (eds) *Map Generalization*, pp. 121-135, Longman
- Burghardt D, Schmid S, Stöter J (2007) Investigations on cartographic constraint formalisation. In: 10th ICA Workshop on Generalisation and Multiple Representation, Moscow
- Brassel K E, Weibel R (1988) A review and conceptual framework of automated map generalization. *International Journal of Geographical Information Science* 2 (3): 229-244
- Duchêne C (2004) The cartacom model: a generalisation model for taking relational constraints into account. In: 7th ICA Workshop on Generalisation and Multiple Representation, Leicester, UK

- Duchêne C, Gaffuri J (2008) Combining Three Multi-agent Based Generalisation Models: AGENT, CartACom and GAEL. In: Ruas A, Gold C (eds) *Headway in Spatial Data Handling, 13th International Symposium on Spatial Data Handling, LNG&C*, pp. 277-296, Springer
- Duchêne C, Touya G (2010) Emergence de zones de conflits dans deux modèles de généralisation cartographique multi-agents. In: Occello M, Rejeb L (eds) *Systèmes Multi-Agents, Défis Sociétaux*, pp. 33-42, Cepaduès
- Gaffuri J (2007) Field deformation in an agent-based generalisation model: the gael model. In: Probst F, Kessler C (eds) *GI-days 2007 - young researches forum*, pp. 1-24, IfGIprints
- Harrie L, Sarjakoski T (2002) Simultaneous Graphic Generalization of Vector Data Sets. *Geoinformatica* 6(3): 233-261
- Hauert JH (2007) Optimization methods for area aggregation in land cover maps. In: *23rd International Cartographic Conference, Moscow, Russia*
- Legrand C, Duchêne C, Lecordix F (2005) Propagation of the displacements and deformations during a generalisation process. In: *Proceedings of International Cartographic Conference, La Coruna, Spain*
- Lemmens R (2008) Lost and found, the importance of modelling map content semantically. In: Peterson M (ed) *International Perspectives on Maps and the Internet*, pp. 377-396, Springer
- Lutz M (2007) Ontology-Based Descriptions for Semantic Discovery and Composition of Geoprocessing Services. *GeoInformatica* 11(1): 1-36
- McMaster R B, Shea K S (1988) Cartographic generalization in digital environment: A framework for implementation in a gis. In: *GIS/LIS'88*, pp. 240-249
- Neun M, Burghardt D, Weibel R (2008) Web service approaches for providing enriched data structures to generalisation operators. *International Journal of Geographical Information Science* 22 (2): 133-165
- Regnauld N (2007) Evolving from automating existing map production systems to producing maps on demand automatically. In: *ICA Workshop on Generalisation and Multiple Representation*
- Renard J, Gaffuri J, Duchêne C (2010) Capitalisation problem in research - example of a new platform for generalisation: Cartagen. In: *13th ICA Workshop on Generalisation and Multiple Representation, Zurich*
- Ruas A (2000) The roles of meso objects for generalisation. In: *Spatial Data Handling*, pp. 3b. 50-63
- Ruas A, Plazanet C (1996) Strategies for automated generalization. In: *7th International Symposium on Spatial Data Handling, Delft, Netherlands*, pp. 319-336
- Stoter J, Burghardt D, Duchêne C, Baella B, Bakker N, Blok C, Pla M, Regnauld N, Touya G, Schmid S (2009) Methodology for evaluating automated map generalization in commercial software. *Computers, Environment and Urban Systems* 33 (5): 311-324
- Touya G (2008) First Thoughts for the Orchestration of Generalisation Methods on Heterogeneous Landscapes. In: *11th ICA Workshop on Generalisation and Multiple Representation, Montpellier, France*
- Touya G (2010) Relevant Space Partitioning for Collaborative Generalisation. In: *13th ICA Workshop on Generalisation and Multiple Representation, Zurich*
- Touya G, Duchêne C, Ruas A (2010) Collaborative Generalisation: Formalisation of Cartographic Knowledge to Orchestrate Different Generalisation Processes. In: Fabrikant SI, Reichenbacher T, van Kreveld M, Schlieder C (eds) *GIScience 2010, LNCS 6292*, pp. 264-278, Springer
- Ware M J, Jones C B, Thomas N (2003) Automated map generalization with multiple operators: a simulated annealing approach. *International Journal of Geographical Information Science* 17 (8): 743-769.
- Zhou S, Regnauld N, Roensdorf C (2008) Towards a data model for update propagation in mr-dlm. In: *11th ICA Workshop on Generalisation and Multiple Representation, Montpellier, France*