A method based on samples to capture user needs for generalisation

Frédéric Hubert and Anne Ruas
COGIT Laboratory IGN
2 avenue Pasteur 94165 Saint Mandé Cedex France
frederic.hubert@ign.fr ; anne.ruas@ign.fr

Abstract

For few years, new models and systems of generalisation have been proposed by the research community. These models allow to produce a new representation of geographical information according to generalisation constraints. One of the missing point is to be able to tune these systems according to different user needs. In the present paper, we propose an implemented system made of web interface and specific modelling and engine that proposes to a user a set of already generalised objects used to identify its needs. The principle is to create a sample data base that contains the information related to the constraints used to obtain a large set of different types of generalisation. These samples are proposed to the user that chooses the solutions that look like what he is looking for. Behind the interface, the system analyses the answers to gradually converge towards the optimal final parameter values, if possible. To allow more flexibility, natural language has been added for the first step of the query and to allow some qualitative reaction of the user (such as 'bigger' or 'with less details'). As a first try the work has been limited to building generalisation but the data modelling has been conceived in a generic way to allow the extension to other types of objects.

1- Context :

Since few years, new models of generalisation are proposed by the research community. Among others we can list three approaches that have recently been proposed by the research community :

1. to design new algorithms that would model generalisation as a constraint solving problem [Harrie et al, 02]. Constraints of proximity, of size are represented at the co-ordinates level. Classical mathematical methods such as least square one are used to displace in an iterative way the position of each point in order to respect constraints,
2. to try a set of combination of operators and to choose the solution that best satisfies a set of constraints. The theoretical number of trials depends on the set of objects and on the possible combination of operators. Simulated annealing technique is used by [Ware et al, 03] to reduce the quantity of trials and to choose if not the best at least a good solution,
3. to try to replace the cartographer analytical capacity by spatial analysis tools that would guide step by step the process [Barrault et al, 01]

All of these recent methods are based on constraints that are used to control the process. These constraints represent the user requirements in a way the system can manage to reach a quality close to these requirements. In the present state, existing systems are limited to experts only, and more specifically to the persons that conceived these systems. Generally, the parameters of the system (i.e. the specification) are set after experimentation and a bit of chance. The next step to reach is to create a link between such systems and a non expert user. As a first step, these non expert users can be people from production or merely another researcher that did not conceived the system.

Whatever the model (present or future) for generalisation, the first essential task is to find a convenient way to let the user introduce his requirement and to transform it into an information used during the process. It means first that some interfaces should be conceived for that. It means also that the information captured through the interface should be well incorporated into the system. The aim of this paper is to present a Ph.D. work which aims at providing an interface to acquire user needs for generalisation purposes. As a first try, the application has been limited to the specification of building generalisation only. The generalisation system used is the AGENT package. Consequently the parameter to set are the values contained within the building constraints functions.
As a first comment, we remain the fact that these parameters are not the parameters of the generalisation algorithms, even if some algorithms use the constraints values to find 'how much' to generalise. As an example, the algorithm of displacement uses the value of the constraint of proximity to displace the object. In the same way, the algorithm of emphasising uses the constraint of size to know how much it should emphasise an object. In the following, the expression 'parameter values' refers to the values contained at the constraint level.

To facilitate the understanding we name **CARTABLE** the system developed to acquire user needs. CARTABLE does not include the system of generalisation but exchanges information with it.

2- The principles

2.1. Choice by examples

The principle of the proposed method is rather simple. Instead of asking a user to directly find the constraint value (such as minimum_building_size = 300m_) we propose to a user a set of non generalised (named the basic samples) and a set of already generalised objects (named the generalised samples). Then we ask him to select for each basic sample, the generalised solution that is the closest to what he is looking for as a final result. The samples are presented on a screen as images and the user clicks on the one he prefers [Hubert, 02].

These samples are generated by the system that will be used to realise the final generalisation (in our study we used the AGENT package). In such a way we are sure that the system is able to do what it is proposing. To go further, it also shows the limit of the possible final solution. The system shows 'what it can do' and the user has to chose the solution the closest. Implicitly it states that the current system can not do better than what is proposed.

As a result, when a user select a sample, the system interprets the parameter values associated with the selected sample. It means that CARTABLE has a representation of the information associated to each sample. This information is hidden to the user but is necessary for the detection of the optimal values. This information is represented in a specific data base named the Sample Data Base, presented in section 3.

2.2. The problem of coherence and convergence

In order to work properly, the proposed method has to overcome three difficulties: one is due to generalisation, the other one is due to the space of possibility; the last one is due to the user himself.

The first point is that the result of generalisation depends not only on the parameter values but also on the characteristics of each object. Let us take the example of the size constraint : if a sample before generalisation has a size bigger than the minimum size required by the user, the user will ask for no change, it just means that the minimum size is smaller than the size of this building. As a contrary, if the sample has a basic size smaller than the required threshold, the user selects a generalised building with the required size final value. Moreover, a building is defined by a set of criteria : its size, its shape, its granularity (the size of its smallest shape), and generalisation aims at finding a 'good balance' that would respect the constraints in the best way. The result of a generalisation, however it is done, always depends on the characteristics of the proposed samples. It means that we need to find a way to overcome this difficulty by proposing a set of different basic samples. If this solution helps to identify the optimal parameter values it also creates problems as we will have to be able to compare the choices of the user up to an optimal solution

The second point is related to the space of possibility. The space of possibility represents all the possible combinations of parameter values (as there is always more than one parameter to fit). Let us say we have four parameters to fit. If each parameter can take for example five values, it means that we have to propose 625 solutions which is 1/ not possible on the screen and 2/ not acceptable for the user. It means that we have to propose a strategy (or several) to gradually localise the final solution space by means of several interaction with the user. The chosen principle is to begin the negotiation with very different solutions and to try to identify the area - in the space of possibility - which is the most frequently selected. This method is significant only if a cluster of values is detected which means that :

- a minimum set of samples should be proposed to identify a cluster and to propose the next set of samples focused around this new search area
- the system should work in a statistical way to overcome the difficulty pointed out previously : some samples will not be very significant due to the characteristic of the initial samples. It means that the system should accept to have some selected samples outside its search area

Figure 1 illustrates different samples with the same parameter values.

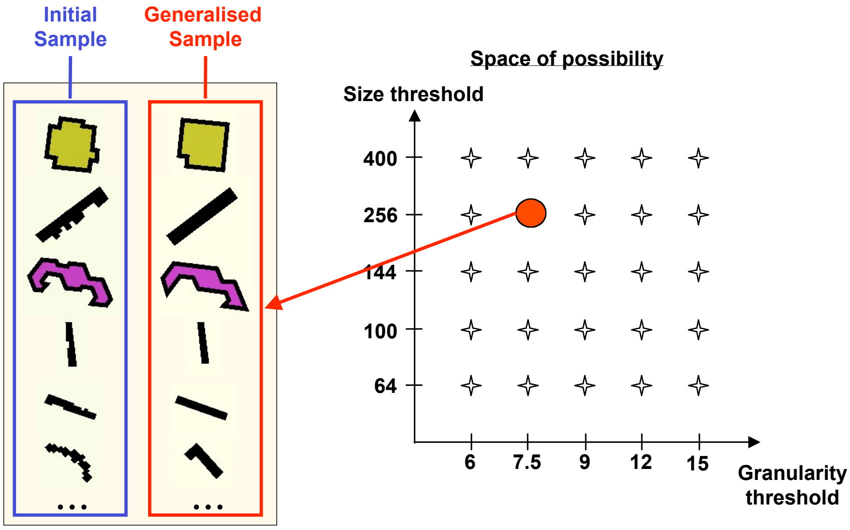

Figure 1: Different generalisation for the same parameter values

The last point is related to the user : If the user answers in a non coherent way or if its wishes are outside the proposed solution, the system will never be able to converge. It means that the system needs a method to identify its non convergence and should notify it to the user.

3- The modelling and engine behind

In order to propose solutions and to converge, CARTABLE needs to represent and to manage two types of information :

- the samples : the basic and generalised samples
- the interactions : the succession of choices and rejects to gradually try to converge towards to final solution space.

The samples data base is basically structured in the following way :

- For each type of object (road, building, etc.), a set of samples are defined (we worked with 33 samples for buildings). Each type (e.g. buildings) holds a set of relevant criteria (named the *Descriptive-Criterion*).
- each non generalised sample is an object of *Basic-smpl*
  - it is described by : its image, a set of characteristic before generalisation - such as size, shape, granularity. Each characteristic (e.g. the size) is an object of the class named the *initial-criterion*
  - it is associated to a set of generalised objects.
- Each generalised object is an object of *Gen-smpl.* It is described by : its image, a set of characteristics that has been used to govern its generalisation - the active criteria - and the quality of the final generalisation (the severity in AGENT).

As a simplified example, a basic building sample named B_4-5 is described by three objects *Initial-criterion* : one named 'size' with value 239 ; one named 'granularity' with value 1.3 and another named 'width' with value 4.1. One of its generalised solution B_4-5-41 is described by three objects *Active-criterion* : one named 'size' with value 300 ; one named 'granularity' with value 4 and another is named 'width' with value 6. The proposed modelling is voluntary generic in order to use the same modelling and engine whatever the name of the geographical object class (building or roads or urban block) and their related characteristics.

This information is structured in a data base presented in figure 2 with UML formalism.

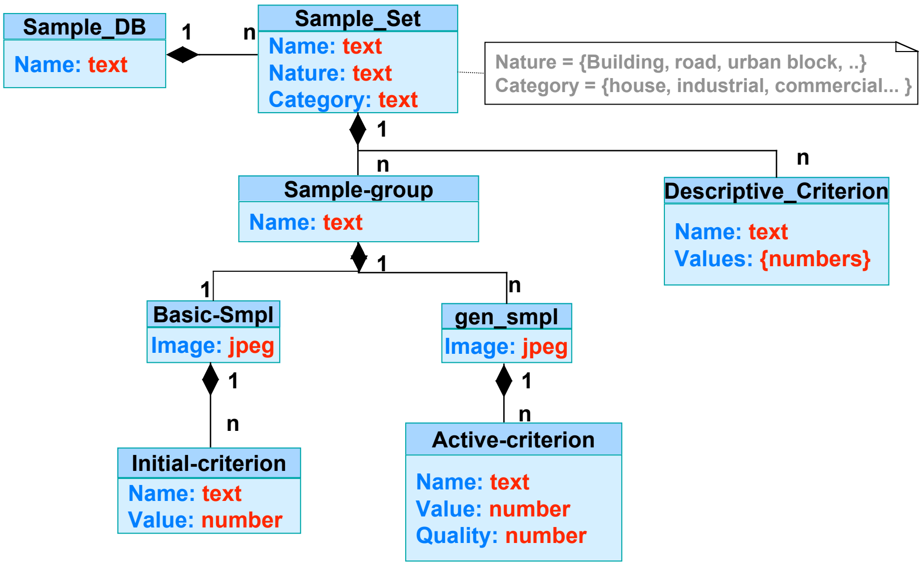

Figure 2 : the sample data base modelling

Moreover, as state previously, in order to ensure convergence, the system needs to record :

1- the previous proposed generalised samples
2- the reaction of the user (choice and reject)
3- the area on which it is converging at the current state.

Associated with this information, CARTABLE needs an algorithm to identify the solution space :

- at the beginning of the process, the algorithm is just looking for a hull around the selected samples (figure 3). We use the link between each sample and the values of these parameters to represent them in space of possibility.
- then it looks for a cluster, as dense as possible to identify the 'best solution space' (figure 3).

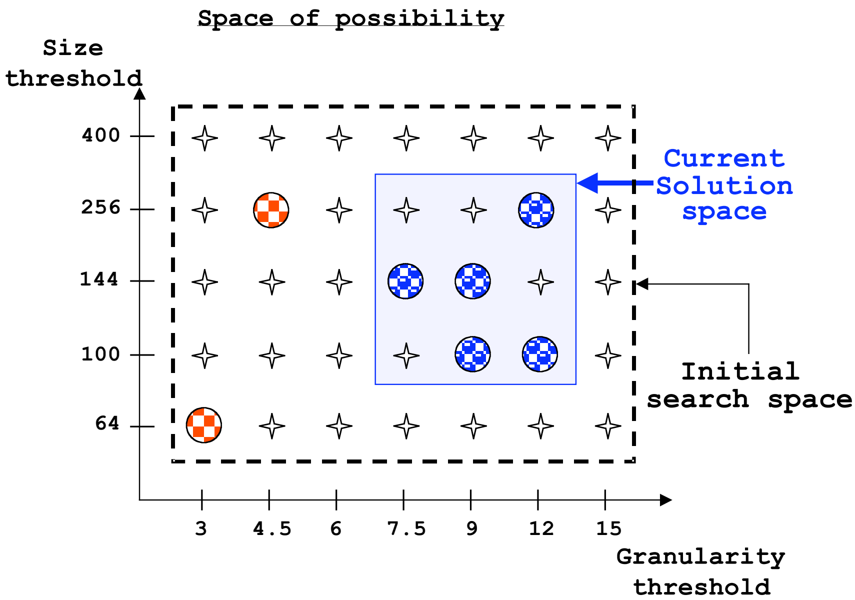

Figure 3: analysing results to identify the next area

At the end of the process, the system can detect :

- a single solution : the selected samples were very coherent one to another
- an area : the selected samples were close one to another but are not equal,
- no area at all : either the system has no solution for the user or the user made incoherent requests.

Whatever the case, the system should present its final solution to the user, again by means of samples if it managed to find at least an area. After a final validation from the user, the system can send to the generalisation package the appropriate parameter values captured for this specific generalisation.

4- The use of natural language to try to go further

In order to improve intuitiveness, we introduce natural language to facilitate the expression of the requests. The sentences from the users are validated when the system is able to interpret them. In our work, the system "understands" only sentences about geographic information. We distinguish three cases for which natural language seams relevant : (1) the initial intervention, (2) the reaction on proposed samples and (3) the positive and negative critics.

To introduce the dialogue with the system, user should express his need by using natural language. This introduction to the dialogue by user corresponds to the initial intervention. For a generalisation process, the user can write the sentence such as : "*I want to generalise buildings*". The system interprets the sentence, identifies the first goal of the user and proposes some samples of buildings to the user.

The second use of natural language is the user reactions on the proposed samples. In this case, user selects one or more samples (images) and adds a commentary. This commentary specifies a particular need regarding to the selected samples. For example, the user can write : "*I select the sample S, but I want fewer details*". The system interprets this sentence and identifies a reaction. Relating to a generalisation process, the natural language is useful to facilitate the expression of critics without having knowledge on parameters of the generalisation process. It is more intuitive for the user to express his request regarding to a sample than to use specific graphic object to modify all parameters. Moreover, the system should have specific knowledge on the link between natural language expression like "*fewer details*" and samples. *Few details* is thus understood as 'bigger value for granularity'.

The last use of natural language is a positive or a negative critics. The user can express his feeling on proposed samples or the negotiation. For example, a positive critic can be "*Ok, I like that*". A negative critic can be "*I don't understand*".

5- The interface

The Web user interface offers to the user the means to interact with the CARTABLE system and to visualise images and texts. The means of communication to the user in order to express his needs are natural language and samples. The Web interface is divided in three areas (figure 4) : (1) a dialogue box, (2) a result area and (3) "common ground" area.

The dialogue box (1) allows the dialogue between a user and the CARTABLE system in order to facilitate the negotiation about the determination of the parameter setting of generalisation process. User interacts by writing sentences in natural language. These sentences relate the different kinds of request detailed in section 4. Moreover, the dialogue box is used by system to response to the user. It can display questions and explanations. This box is a Java applet connected to our system with RMI (Remote Method Invocation).

The result area (2) is used to display intermediate and final results. This area presents mainly a set of images relating to basic and generalised samples. We associate at each image a text and a graphic objet (a check box). The text allows to differentiate each proposed sample by an identifier. The graphic object is used by the user to interact with images. Different kinds of graphic objects are included in this area like checkboxes, radio buttons, buttons and so on. The result area is created by writing pages with HTML and Javascript languages. The connection with our system is established with Javascript socket.

The "common ground" area (3) displays previous agreements between the user and the system. The "common ground" corresponds to knowledge resulting from the inter-understanding between the user and the system during the current session [Nicolle et al., 99]. Different kinds of interaction are able to the user. He can consult previous interaction, come back previous step or ask information about a previous step. It also allows us to control the 'current solution space' to control convergence.

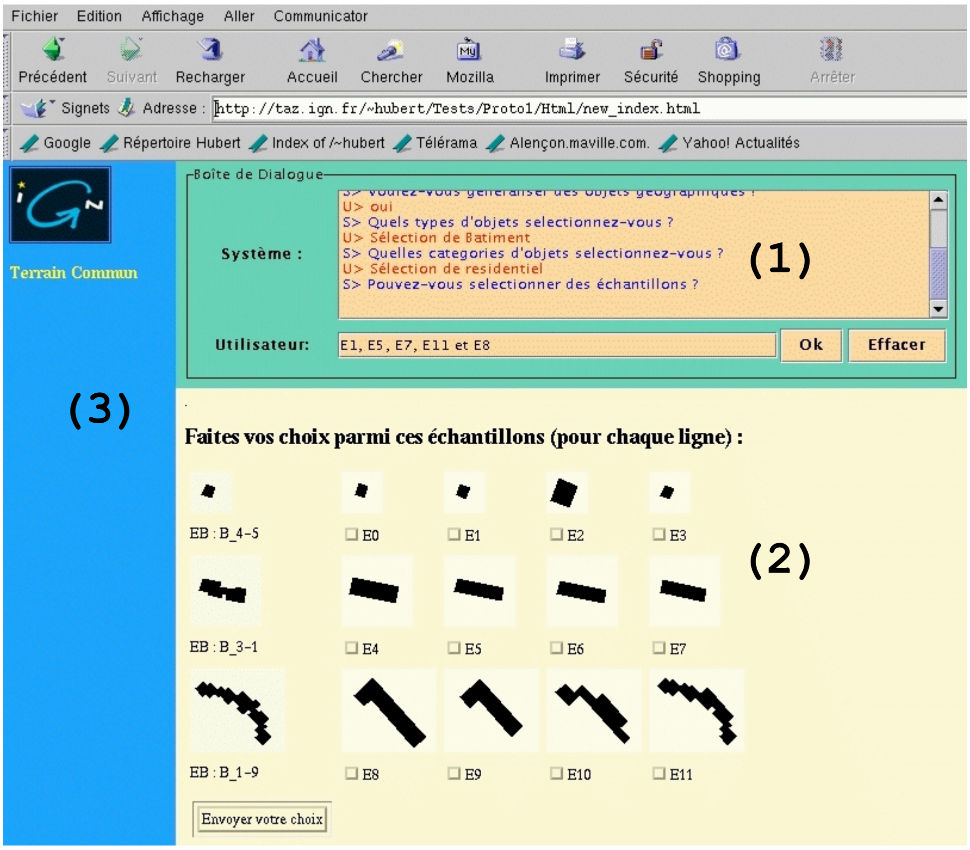

Figure 4: the Web user interface.

6- Conclusion and perspectives

This paper presents a current research that has the objective of proposing an interface that allows a user to specify its generalisation needs by means of samples and natural language. A large part of the work has been devoted to the conception of a flexible web interface, the samples data base modelling, the integration and the adaptation of a natural language package and the modelling of user interaction. Current research is focusing on the improvement of the search strategy in order to ensure a convergence besides the uncertainty due to the samples, the process of generalisation and the user.

This research is really a first try to reduce the distance between systems for experts (and by experts) and common user. The systems conceived by our research community have a very short life time because they depend too much on the specific and deep understanding of a very limited number of persons. To save them, it is necessary to bridge the gap between them and the user knowledge. This domain of research is complex because no one is able to precisely understand and define user needs. However, we believe that step by step we can propose methods, tools, ideas that would reduce this distance to make our research more useful because accessible.

References:

Barrault M., N. Regnauld, C. Duchêne, K. Haire, C. Baeijs, Y. Demazeau, P. Hardy, W. Mackaness, A. Ruas, R. Weibel, Integrating multi-agent, object oriented and algorithmic techniques for improved automated map generalization, *Proc. of 20th International Cartographic Conference,* Beijing, China, 2001, 2110-2116.

Harrie L.and T. Sarjakoski, Simultaneous graphic generalization of vector data sets. *GeoInformatica* 6(3), 2002, 233-262.

Hubert F., Map samples to help gi users specify their needs, *10th International Symposium on Spatial Data Handling*, D.E. Richardson and P. van Oosterom (eds.), Ottawa, Canada, 2002, 533-543.

Nicolle A. and V. Saint-Dizier de Almeida, Vers un modèle des interactions langagières. In: *Analyse et Simulation de Conversations : De la théorie des actes de discours aux systèmes multiagents*, Moulin B., Delisle S. and Chaib-draa B. (eds), L'interdisciplinaire, Lyon, 1999, 133-169.

Ware J.M., C.B. .Jones and N. Thomas, Automated cartographic map generalization with multiple operators: a simulated annealing approach, to appear in *The International Journal of Geographical Information Systems*.2003.