The introduction of automated generalisation procedures in map production systems requires that generalisation systems are capable of processing large amounts of map data in acceptable time and that cartographic quality is similar to traditional map products. With respect to these requirements, we examine two complementary approaches that should improve generalisation systems currently in use by national topographic mapping agencies. Our focus is particularly on self-evaluating systems, taking as an example those systems that build on the multi-agent paradigm. The first approach aims to improve the cartographic quality by utilising cartographic expert knowledge relating to spatial context. More specifically, we introduce expert rules for the selection of generalisation operations based on a classification of buildings into five urban structure types, including inner city, urban, suburban, rural, and industrial and commercial areas. The second approach aims to utilise machine learning techniques to extract heuristics that allow us to reduce the search space and hence the time in which a good cartographical solution is reached. Both approaches are tested individually and in combination for the generalisation of buildings from map scale 1:5000 to the target map scale of 1:25 000. Our experiments show improvements in terms of efficiency and effectiveness. We provide evidence that both approaches complement each other and that a combination of expert and machine learnt rules give better results than the individual approaches. Both approaches are sufficiently general to be applicable to other forms of self-evaluating, constraint-based systems than multi-agent systems, and to other feature classes than buildings. Problems have been identified resulting from difficulties to formalise cartographic quality by means of constraints for the control of the generalisation process.

Keywords: Map generalisation; building generalisation; data enrichment; machine learning; expert knowledge; self-evaluating systems; multi-agent systems

1. Introduction

The number of national mapping agencies (NMAs) that introduce automated map generalisation procedures into their map production workflows is steadily increasing (Stoter 2005). Conventional and automated map generalisation share the same basic objectives, which include fulfilling the intended map purpose and ensuring map legibility, taking into account user habits and principles of human visual perception.
In order to achieve these objectives for automated map generalisation it is necessary to transfer the cartographic knowledge into a machine understandable form. Armstrong (1991) has identified three types of cartographic knowledge: (a) geometric, (b) procedural and (c) structural knowledge. The geometric knowledge describes information on the size, shape and topology of map objects and can be obtained from measures applied on the objects’ geometry. Procedural knowledge describes rules for the selection of appropriate generalisation algorithms in the presence of cartographic conflicts. For instance, if a building is too small to be clearly legible on the reduced scale map (a conflict), then the building may be enlarged (algorithm A) or eliminated (algorithm B). Finally, structural knowledge covers the cartographic knowledge needed to identify which objects and ‘structures’ are important, for example, in terms of their cultural, economical or geomorphological meaning. Thus, the structural knowledge influences the decision whether the small building of the above example must be preserved (enlarge building) or whether it is unimportant for the map reader (eliminate building).

An analysis of the pertinent literature as well as discussions with map production experts in NMAs that we have conducted revealed that existing automated generalisation systems need to be fine tuned, that is, their knowledge refined, if they should be used (more) successfully in production lines. Hence, a specific focus of current map generalisation research is on the refinement of existing procedural knowledge that is currently applied in self-evaluating cartographic systems. In this article, we particularly consider self-evaluating systems that are used for the production of (static) topographic maps by NMAs (Mackaness et al. 2007). As an example for self-evaluating systems, we use systems that build on the multi-agent system (MAS) paradigm. In the context of such generalisation systems, we consider primarily two issues.

First, a high cartographic quality that is comparable to the quality of traditional maps can only be obtained with automated systems that treat every map object according to its context (SSC 2005). Hence, a topographic map production system should be able to account for spatial and semantic context information in the generalisation process. This requirement is exemplified by a recent call for tender by the Swiss NMA Swisstopo, who demanded the functionality to create and utilise generalisation zones, such as dense versus scattered settlement zones, where every topographic feature (e.g. building or road) is treated differently depending on the zone (or context) that it is located in.

To make the generalisation process context-aware, the structural knowledge needs to be extracted and stored in a process that has been termed data enrichment (Ruas and Plazanet 1996, Neun et al. 2004). Afterwards the procedural knowledge can be refined with respect to the enriched data and structural knowledge. A context-dependent treatment can then be realised for instance by inclusion of expert rules into the system.

A second problem that has been reported for MAS in cartography is that the generalisation process can be computationally very expensive. This can be attributed on the one hand to the computational complexity of the generalisation task (see Haunert and Wolff 2008) but on the other hand also to the iterative local search approach used for the selection of a generalisation algorithm in a particular map situation (see Section 2). Thus, acquiring and fine-tuning procedural knowledge, that is, rules for the selection of an appropriate generalisation algorithm, or an appropriate sequence of algorithms, should improve the efficiency of such systems.
The objective of this paper is to show two approaches with which the above-mentioned problems can be addressed for the example of the generalisation of single buildings. In order to counter the first problem, context-dependent generalisation, we will try to use expert rules for the generalisation of single buildings to improve the quality of the resulting map. The expert rules have been defined based on the work by Boffet (2001) as well as results of experiments previously conducted at the Institut Géographique National, France (see also Lecordix et al. 2006). The second issue, computational performance of MAS, will be addressed by procedural rules that are acquired with machine learning techniques. The rules inferred from machine learning allow reducing the search space of the MAS reasoning mechanism (or ‘agent engine’) by introducing procedural heuristics.

Apart from developing and testing the expert rules approach and the machine learning approach individually, we aim to test both approaches together. To the authors’ knowledge such an experiment is conducted for the first time, while reports on experiments with the individual approaches are sparse. We expect that both approaches are complementary, since we realise them in such a way that different components of the generalisation system are affected. Hence, our hypothesis is that the combination of both types of knowledge utilisation will help to improve the map generalisation process both in terms of efficiency (i.e. processing speed) and effectiveness (i.e. achieved quality). To test our hypothesis, we will specifically examine knowledge for the generalisation of individual buildings for a topographic base map (scale 1:25,000). We start from existing rules that only take into account the internal conflicts of a single building in order to decide which generalisation algorithm to try applying on it next. We evaluate our hypothesis regarding efficiency by considering the time required to generalise a building. The evaluation regarding the improvement of effectiveness will be based on visual assessment of the generalised buildings.

The remainder of this paper is structured as follows. Section 2 reviews agent-based systems currently used in map production, introduces the concept of constraints in MAS and explains how MAS work. Section 3 discusses limitations of generalisation systems and the approaches we propose to overcome these problems, including machine learning. Section 4 presents the design of the experiment conducted in this study, Section 5 the results and Section 6 the discussion of problems identified and possible improvements. We conclude in Section 7 by recalling the main achievements and outlining perspectives for further research.

2. The current approach for building generalisation

Automated map generalisation systems based on the MAS paradigm are currently in use or being introduced in topographic map production of NMAs such as IGN France, Ordnance Survey (UK) or KMS (Denmark). These systems consist of several logical components. In general, one can distinguish four components (Ruas and Plazanet 1996, Weibel and Dutton 1998): (1) constraints, (2) measures, (3) generalisation algorithms and (4) a mechanism that controls the generalisation process. The control mechanism is responsible for the decisionmaking, determining how to generalise by evaluating the constraints (1) and triggering the generalisation algorithms (3). In the following sub-sections, we will explain these four components with respect to building generalisation because we will later focus in the experimental part on the generalisation of buildings.
2.1 Constraints and algorithms for building generalisation

2.1.1 Constraints on building representation. A map should meet two basic requirements. First, the map should be designed to fulfil a specific purpose, and second, the map must be legible. In order to specify these requirements in more detail and define what a good map should look like, current approaches to map generalisation use a set of cartographic constraints, which can be understood as conditions to which the map should adhere (Weibel and Dutton 1998). The idea to introduce constraints for map generalisation has been proposed by Beard (1991), presumably inspired by the use of constraints in computer science. Here constraints are commonly used to restrict search problems. However, we note that the notion and the application of cartographic constraints and of constraints in search problems are slightly different (see e.g. Michalewicz and Fogel 2004). The constraint-based approach to automated map generalisation is now widely accepted as the standard approach for modelling the generalisation process (Harrie and Weibel 2007). Hence, we will also employ constraint-based modelling in this paper and describe the application in Section 2.1.3 and Section 2.2 below.

Several constraints have been found to be useful to describe the legibility of a map as described by Weibel and Dutton (1998) and by the AGENT Consortium (1998, 1999). With respect to buildings the legibility constraints identified focus exclusively on the geometrical aspects. In practical experiments during the ‘Nouvelle Carte de Base’ (NCDB) project (Lecordix et al. 2006) at IGN France, five constraints have been found to be useful (Figure 1): (C1) minimum building size, (C2) building outline granularity, (C3) wall squareness, (C4) minimum inner width and (C5) minimum distance between two buildings. Opposed to legibility constraints are preserving constraints. Such constraints are used to prevent strong changes resulting from generalisation actions activated by violating a previously listed constraint. With respect to a single building one can identify at least four of these constraints in the literature. The first constraint is intended to prevent strong changes of the building shape and is called concavity constraint (C6; Bard 2004). The second constraint (C7) is called positional accuracy and should prevent a building’s position

Figure 1. Constraints acting on buildings.
from being altered too much during building displacement. Such a displacement operation may be triggered due to a violation of the minimum distance constraint between buildings. The third constraint, denoted as conservation constraint, should prevent the elimination of important buildings (C8). This constraint demands a definition of importance on a structural or semantic level. For instance, hospital or school buildings are often considered as important types of building due to their unique function and their different geometrical characteristics compared to other buildings in a residential district. The fourth constraint is the building density preserving constraint (C9), which should ensure that the visually experienced density stays constant despite, for instance, building enlargement operations. Besides these four preserving constraints, other constraints have also been proposed for the generalisation of a group of buildings. An example is the alignment constraint (C10), which explicitly preserves building alignments (Gaffuri and Trévisan 2004). In the remainder of this paper, we will call the violation of a constraint a cartographic conflict.

2.1.2 Actions for building generalisation. Several actions can be activated if one of the previously listed constraints is violated. An extensive listing of actions, so-called generalisation operations, to meet the legibility constraints has been presented by McMaster and Shea (1992). For the generalisation of buildings, those actions focus either on the elimination or the geometrical transformation of buildings. We like to note here that in automated generalisation a particular cartographic operation, for example, building displacement, can be realised with different algorithms that utilise different solution approaches. A list of generalisation algorithms that deal with the above-mentioned constraints C1–C5 is given in Table 1.

A violation of preserving constraints does not necessarily result in an activation of a specific operation or algorithm. In operational generalisation systems such constraints can additionally trigger a recovery of the initial state prior to generalisation (termed backtracking) or flag the result as invalid.

2.1.3 Evaluating constraints. One property of the constraint-based approach to modelling the generalisation process is that several cartographic constraints can be defined and evaluated first, and only then will it be decided what action is triggered to solve a given cartographic conflict. This procedure is especially of value if

<table>
<thead>
<tr>
<th>Generalisation algorithm</th>
<th>Applicable to following constraints</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1: scale polygon</td>
<td>C1</td>
<td></td>
</tr>
<tr>
<td>A3: building wall squaring</td>
<td>C3</td>
<td>Regnauld et al. (1999)</td>
</tr>
<tr>
<td>A4: enlarge width locally</td>
<td>C4</td>
<td>Regnauld et al. (1999)</td>
</tr>
<tr>
<td>A5: simplify to rectangle</td>
<td>C2, C3, C4</td>
<td>AGENT Cons. (1999)</td>
</tr>
<tr>
<td>A6: enlarge to rectangle</td>
<td>C1, C2, C3, C4</td>
<td>AGENT Cons. (1999)</td>
</tr>
</tbody>
</table>
constraints are in conflict with each other such as C5 (minimum distance) and C7 (positional accuracy). To achieve a solution, every constraint proposes none, one or several actions to solve a given problem. After all existing constraints have been evaluated, a ranking of all proposed actions is established and finally the most promising action is triggered.

But how can we know whether a constraint is fulfilled or not? Every constraint is associated with a measure (identified as the second component of a generalisation system above). This measure returns a quantitative value for a geometrical or topological property of one or more map objects. This value is then mapped into a qualitative statement, the so-called constraint satisfaction, by comparing it to a reference value, for example, the minimum building size that corresponds to a particular target scale and map purpose. The qualitative statements can be expressed either as Boolean (true/false), integer scores (e.g. 1–5) or continuous floating-point scores (e.g. 1.0–5.0), using different mapping functions (Bard 2004). In our experiment, we will use continuous scores in the range of 1.0 = constraint violated to 10.0 = constraint fully satisfied.

2.2 Controlling generalisation with an agent model

Of the four components of a state-of-the-art generalisation system listed at the beginning of this section, we have explained constraints, measures and generalisation algorithms. What remains to be explained is the mechanism that controls the overall generalisation and that allows us to make associations between constraints, measures and generalisation algorithms (or operations) and to determine in which order the generalisation operations are carried out. This control mechanism is often termed inference engine. A variety of approaches for this purpose have been reported in the literature. A review of relevant techniques can be found in Harrie and Weibel (2007). The currently most promising techniques to solve this optimisation problem appear to be simulated annealing (Ware et al. 2003a), genetic algorithms (Ware et al. 2003b) and MAS (Ruas 1999, Barrault et al. 2001, Ruas and Duchêne 2007).

In our experiments, we will utilise the agent-based approach (based on the original work of Ruas 1999) and hence need to explain it in more detail regarding those elements that will be affected by our experiments. First, as we focus on building generalisation, every building will be modelled as an agent. In agent-based modelling, the term ‘plan’ is used to denote an action that can be executed by an agent to attain his ‘goals’. In automated map generalisation, such a plan consists of a generalisation algorithm plus parameter settings, and goals correspond to cartographic constraints. A building agent gets assigned those legibility constraints that it must fulfil. The processing principle of the generalisation of such a building agent, the so-called agent lifecycle, is shown in Figure 2. This process carries out a ‘trial-and-error’ approach that is known as local search method in computer science (Michalewicz and Fogel 2004) and which is essentially mimicking the work style of a cartographer. Every time a plan (i.e. a generalisation algorithm) is applied this is seen as one trial that results in a new agent state. In the last step of a single life cycle, the current state is evaluated and classified as ‘valid’, ‘invalid’ or ‘perfect’ state based on a so-called happiness value. For each state, the happiness is calculated as a weighted average of the constraint satisfaction values over all constraints. A state is usually considered as valid if the happiness or the satisfaction of the constraint that proposed the executed plan has improved compared to a previous state. For all other cases the
state is considered as invalid. A state is classified as perfect if all constraints are satisfied (no violation). In the latter case the generalisation of the building is terminated while for the other two cases the state is stored and the generalisation process continues with a new trial. For a valid state the process continues from the current state, whereas for an invalid state the system will return to a previous valid state and continue from there. Thus, invalid states are never considered as starting points for further actions, which may lead to the omission of good solutions that emanate from those. Finally, after all plans have been tried, the system will select the one state as generalisation solution that best fulfils all constraints (i.e. has the highest happiness value) of all stored states. The described trial-and-error generalisation process is depicted as a process tree for one building in Figure 3.

To conclude this section, we aim to discuss the parameters used for controlling the generalisation process in the agent model. The first parameter is called constraint importance and represents a weight that is assigned to each constraint type in order to calculate the overall happiness value of an agent as a weighted average. The second parameter, constraint priority, is used to determine in which order constraint violations should be solved. Defining an appropriate order is useful since the preceding solution of one conflict may involve an easier solution for a following conflict. For instance, if a minimum size conflict (C1) is solved first by building enlargement, then a previously detected minimum width conflict (C4) could have been solved at the same time. Finally, giving individual weights to plans is necessary to define which plan should be executed before another plan that has been proposed by the same constraint. As a rule of thumb preference should be given to plans that results in less (geometrical) changes. All three parameters are usually pre-defined by an expert but may also be set dynamically at runtime (Ruas 1999). For a more detailed introduction of multi-agent models
in cartographic generalisation, we refer to the review by Ruas and Duchêne (2007).

3. Ways to improve building generalisation

In the previous section, we have introduced an approach for the generalisation of buildings that is based on constraint modelling and the use of an MAS to control the generalisation process. We further listed the constraints that should ensure map legibility, (preserving) constraints that should avoid excessive changes and algorithms for resolving cartographic conflicts. In this section, we aim to introduce two approaches for the improvement of knowledge applied in the agent-based process control. After discussing deficiencies and possible improvements (Section 3.1), we will give a short overview of previous work (Section 3.2). Subsequently, we describe one approach for the refinement of procedural knowledge (Section 3.3) and one for the extraction of procedural knowledge by machine learning (Section 3.4).

3.1 Deficiencies of the current approach and possible improvements

As outlined in the introduction, we aim to improve the MAS-based generalisation approach in two respects. On the one hand we focus on making the system more computationally efficient while on the other hand we want to improve the effectiveness (i.e. the cartographic quality). Therefore it is necessary to analyse the current approach to generalisation process control.
The disadvantage of the local search approach is that it can be very extensive due to the combinatorial nature of our search problem. An analysis of the tree in Figure 3 reveals that the tree contains redundancies (the same sub-branch exists twice) and a number of the branches end with unsatisfactory and invalid solutions. Thus, in both cases, processing efforts could be reduced and computational efficiency increased if either (1) dynamic validity and termination criteria are applied or (2) the list of available plans and the selection of plans are better controlled. A dynamic, context-dependent setting is necessary since a static validity criterion, used for all buildings, may result in cases in which the best solution is missed (in Figure 3 solution B instead of A). We will provide solutions to both of the above cases. Dynamic validity and termination criteria will be set by rules that are derived with machine learning techniques. An improved control of the plan list and plan selection will be achieved with rules obtained inductively with machine learning techniques but also with deductive expert rules. In both cases, we will allow only plans to be proposed that are appropriate for the specific cartographic context of a particular building.

Improving the system in terms of cartographic effectiveness is also related to the proposal of plans adapted to the spatial and semantic context of map objects. But here we aim to obtain a cartographically more convincing solution that should theoretically correspond to a higher number of satisfied constraints and hence higher values of happiness. Thereby a smaller set of trials will probably be a side effect of the adaptation to the context. An improvement in effectiveness is possible by introduction of expert rules. Note that the use of expert rules can result in improvements that we are not able to evaluate quantitatively, since visually obvious improvements are not necessarily recognised by the quantitative measures used by our set of constraints. A gain in effectiveness is also possible with machine learnt rules (Taillandier 2007). However, this holds only if the rules are learnt from a comprehensive solution tree (i.e. using a weak validity criterion), while the reference generalisation system applies a strong validity criterion.

3.2 Related work to improve the performance of generalisation systems

The use of expert rules in cartographic systems has experienced a mixed history. In the heyday of ‘expert system’ technology in the late 1980s, a series of studies were reported using expert rules in map generalisation and related areas (Mackaness et al. 1986, Mackaness and Fisher 1987, Nickerson 1988, Doerschler and Freeman 1992, Schylberg 1993). These early activities were followed by a relatively long period of silence, which was due to the scarcity of formalised cartographic knowledge: If no expert rules exist, no expert system can be built. Hence, research turned to knowledge acquisition during the subsequent years, including the utilisation of machine learning techniques for the extraction of generalisation processing rules. After initial experiments by Weibel et al. (1995), Plazanet et al. (1998) developed a supervised learning approach for the selection of appropriate line generalisation algorithms, primarily for roads. This work was later extended by Mustière (2005). Mustière et al. (2000) and Ruas et al. (2006) evaluated learning-based methods for the extraction of rules for the generalisation of buildings. Of the above, only Ruas et al. (2006) focused on learning rules from mining the logs of a self-evaluating generalisation system to improve its efficiency. The other studies extracted rules from logging human expert interactions with interactive systems to improve the quality of automated
generalisation results. Our research continues the work by Ruas et al. (2006), but we use a different rule learning approach and directly integrate (i.e. test) the obtained rules within the MAS that was used.

Experiments with expert rules to guide the selection of generalisation algorithms in self-evaluating systems have rarely been reported so far, owing to the relatively recent introduction of such systems. Gaffuri and Trévisan (2004) used the classification of urban areas by Boffet (2000, 2001) into different zones (e.g. urban blocks, dense residential, scattered residential, industrial) to enable a contextual treatment of the blocks. Similar work was reported by Revell et al. (2006) to differently generalise buildings in urban and rural environments. Our approach differs from the above in that we differentiate between more types of situations. Also, we use a classification that is assigned as an attribute to each individual building, rather than to entire classes of buildings. That provides the capability of generalising a supermarket in a housing area differently than the surrounding residential houses.

3.3 Context analysis and building classification

The objective for the use of expert rules in the generalisation process is to ensure that a particular map object receives a context-dependent treatment. Hence the expert, that is, a cartographer, needs to identify first what objects need special treatment. Then, methods have to be defined that select those objects. Finally, the (expert) rules are set up, defining the generalisation procedure for these objects. In our case study, the expert generalisation rules will utilise higher order semantic concepts related to the urban fabric. Such concepts are implicitly (or latently) contained in the map data but are not explicitly coded and hence need to be extracted and made explicit with pattern recognition techniques. A condition for the usability of such higher order semantic concepts is that they can be related to existing cartographic map generalisation rules, while at the same time the concepts must be intuitive to understand for the person using the map. Based on an analysis of the generalisation literature (e.g. SSC 2005), the study of topographic maps and the study of maps for urban planning and education, Steiniger et al. (2008) identified five urban structure classes that are often used: (1) inner city buildings, (2) industrial and commercial buildings, (3) urban buildings, (4) suburban buildings and (5) rural buildings. To assign buildings to one of these classes, Steiniger et al. (2008) propose a supervised classification approach that establishes a mapping between geometrical properties of buildings and the above urban structure classes. We applied this classification approach implemented as a web-generalisation service by Neun et al. (2008). The result of the classification process for a data set of Zurich (Switzerland) is shown in Figure 4. As features for the classification the following geometric properties of the building geometry have been used: (I) area, (II) number of corners, (III) shape index, (IV) squareness, (V) elongation and (VI) the number of courtyards. Additionally, the following density measures: (VII) number of buildings within a 200 m buffer, (VIII) the ratio of building area within a 200 m buffer to their convex hull area and (IX) the ratio of building area within a 200 m buffer to the buffer area, are calculated for every building. The classifier used was a support vector machine with a radial basis function kernel (for details, see Steiniger et al. 2008). We reached an overall classification accuracy of 82% (kappa statistics: 0.73), meaning that there is a chance of 18 misclassified buildings out of 100 buildings.
Once classified, every building can be related to its (urban) context, represented by the urban structure type that it has been classified into. Based on this information, we are able to introduce expert rules in the generalisation system that trigger specific generalisation algorithms and algorithm sequences for each individual building (see below in Section 4.1.2). The set of expert rules is introduced and evaluated during the collection of plans, which happens in the agent life cycle step 2 of Figure 2. Here, the expert rules control the compilation of the list of plans, where plans are proposed by a violated constraint to the inference engine. For misclassified buildings, however, the list of generalisation algorithms proposed by the expert rules can be inappropriate. For instance, the set of algorithms for a suburban building that has been misclassified as an inner city building may not be suitable.

3.4 Learning rules with machine learning techniques

3.4.1 What we aim to learn. To improve the efficiency of the generalisation system, we would like to learn rules of the following structure: If (building_size < 200 m²) And (building_type = inner_city) Then (privilege building_elimination_plan). The advantage of such rules is that they are easy to interpret and subsequently also useful to evaluate existing knowledge of the generalisation system. In accordance with previous positive experiences reported in Taillandier (2007), we decided to focus on three different rule types. These three types should facilitate the following actions:

1) Choosing a branch (of the search tree) – The first rule type is called priority rule and helps to identify the constraint which should be solved next to obtain the best solution (in terms of a high happiness value) with as few generalisation trials as possible. Thus, this type of rule will try to minimise the number of tested solutions and subsequently the size of the search tree in Figure 3.

Figure 4. Classification results for the Zurich building data. The lighter areas mark the training data. Data courtesy of the City of Zurich, Geomatik + Vermessung, 16.10.2007.
Avoiding a branch – The second rule type is called validity rule and is used to identify situations in which it is likely that no acceptable (i.e. an invalid) solution is obtained if one proceeds with the current generalisation result. Such rules help to avoid unnecessary generalisation tries.

(3) Terminate process – Finally the third type of rules to be learnt are termination rules. They identify situations in which the generalisation process should be terminated prematurely since obtaining better solutions in terms of a higher happiness value is unlikely. Thus, the number of generalisation trials is limited by these rules.

3.4.2 General learning method. In the artificial intelligence and data mining community several rule learning approaches have been developed (Hand et al. 2001, Witten and Frank 2005). For our purposes we will use a supervised rule learning approach, which means that we will have to provide training samples to the learning algorithm. The approach follows the general three-phase learning scheme:

(1) Exploration step – This step consists in logging the actions of the generalisation process for a large number of geographical objects. During this phase, the process uses the procedural knowledge initially contained in the generalisation system. The logs record the whole information related to successes and failures of the various actions invoked by the system, and hence of the procedural knowledge it contains.

(2) Analysis step – This step is comprised of analysing the logs obtained during the previous phase and in deducing new knowledge from it. Thus, the training samples are selected from the database generated in the analysis step and afterwards the rules are learnt from these samples.

(3) Exploitation step – The third step involves testing the obtained rules with the generalisation system on a different data set. Hence, we can evaluate whether the learnt rules lead to an improvement of the generalisation process.

We will describe the two parts that are essential for our experiments, that is, the creation of the training samples and the generation of rules, in the next sub-sections. The results of the third step are presented in Section 5.

3.4.3 Training samples selection. For the rule learning it is necessary to define from which kind of data the rules should be learnt. In the case of supervised learning every sample of the training data must consist of a description vector that is used to define the condition, and a label that corresponds to the action. For our purposes, we will use the constraint satisfaction of every state and the building type as description vector. The labels will be defined according to the rule type that should be learnt (priority rules, validity rules and termination rules). Furthermore, we restrict the selection of training samples to those states that are directly related to the best (successful) path to the final solution. This definition allows obtaining a correct state characterisation and at the same time is not too complex for the learning procedure. In Figure 5, we give an example that shows which states of the generalisation process of one building are used for the training. The state $S_i$ in Figure 5 is a vector that contains the satisfaction values for squareness and concavity but also the type of the building class (rural, suburban, etc.). Obviously, however, the final set of training data used as input
for the rule learning algorithm will consist of samples obtained from several dozens of buildings, and not only from a single one.

3.4.4 Learning method used. The technique that we will use for learning the rules is exception-based learning (Witten and Frank 2005). In particular, we use a learning approach that consists of two components: the algorithm that generates/extracts the (exception) rules called IREP (incremental reduced error pruning; Fürnkranz and Widmer 1994) and a data structure for the management and application of the rules called RDR or RIDOR (ripple-down rules; Compton et al. 1991).

One common approach to learn rules is to generate a decision tree first, then to transform the tree into a set of rules, and finally simplify the rule set (Frank and Witten 1998). This approach is for instance used in the C4.5rules algorithm (Quinlan 1993). The other approach, used by IREP, applies a ‘separate-and-conquer’ strategy (for the difference to ‘divide-and-conquer’, see Fürnkranz and Widmer 1994). Here, the most powerful rule is determined, that is, the rule that covers most training examples, and afterwards all examples that are covered by this rule are deleted. This process is repeated if there are no positive examples left, or until the last rule found produces an unacceptably large error rate (Cohen 1995).

RIDOR is a data acquisition methodology that should ensure rule sets having minimal inter-rule interactions and simple maintenance (Gaines and Compton 1995). Developed for the management of rules that are provided by experts, the ripple down rule methodology can be used also to manage learned rule sets. In a RIDOR knowledge base the rules form a binary decision tree. A default rule is generated first and then the first rule delivered by IREP is used as exception. Subsequently follow-up rules from IREP are added as exceptions to the predecessor rule and a classification tree emerges. Figure 6 provides a real-world example for a single classification RDR (SCRDR) knowledge base that has been learnt to determine which generalisation constraint to solve next.

Choosing a rule learning approach for a particular problem is difficult. Using RIDOR to maintain the knowledge base of rules and not only the rule learner, enables us to learn a structured set of rules, which is easier to interpret than a mere...
4. Experiment

4.1 Experimental set-up

4.1.1 Target map scale and derived constraint settings. For the experimental part, we decided to focus on the generalisation of buildings for the base map scale of 1:25,000, starting off from data at a nominal scale of 1:5000 to 1:15,000 (cf. Section 4.1.3). Given that we exclusively want to deal with the generalisation of single buildings, focusing on such a large scale has the advantage that only a few buildings need to be eliminated (Müller 1990) and only few displacement operations due to overlaps between buildings and between buildings and roads are necessary. Thus, complex operations such as building typification for dense built-up areas are not considered and it is easier to evaluate the effects of the expert and learnt rules. Hence, the set of constraints that we applied involves only the following constraints for individual buildings: C1 – minimum size, C2 – granularity, C3 – squareness, C4 – minimum width and C6 – concavity. The minimum distance constraint is not applied for two reasons: On the one hand displacement operations can be executed after the previously mentioned constraints are satisfied. On the other hand, if the generalisation of one building is influenced by the generalisation of its neighbour buildings, it is harder to identify emerging knock-on conflicts due to geometry transformations. In Table 2 the constraints, their parameter settings and the plans (i.e. generalisation algorithms) are listed that are proposed if a constraint is violated.
Table 2. Constraints applied in the experiment for the scale change to 1:25,000 used for the reference building generalisation. NCDB, Lecordix et al. (2006). Settings are similar to the Nouvelle Carte de Base project.

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Constraint type</th>
<th>Priority (1: last ... 5: first)</th>
<th>Importance (1:low ... 5: high)</th>
<th>Default settings</th>
<th>Condition for plan proposal (with plan weight $w$)</th>
<th>Threshold values for 1:25,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum size (C1)</td>
<td>Legibility</td>
<td>5</td>
<td>5 (NCDB: 4)</td>
<td></td>
<td>A1: scale polygon</td>
<td>minSize: 80m$^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A6: enlarge to rectangle</td>
<td>IF $\text{area} \leq \text{delSize}$</td>
<td>delSize: 20m$^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A7: eliminate</td>
<td>THEN A7 ($w=1$)</td>
<td>medSize: 60m$^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ELSE IF ${\text{area} &gt; \text{medSize}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>THEN propose A1 ($w=2$) and</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A2 ($w=1$) and</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ELSE { A1 ($w=1$) and</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A2 ($w=2$) }</td>
<td></td>
</tr>
<tr>
<td>Granularity (C2)</td>
<td>Legibility</td>
<td>4</td>
<td>5</td>
<td></td>
<td>A2: simplify</td>
<td>minSize: 60m$^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A5: simplify to rectangle</td>
<td>delSize: 20m$^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>THEN propose A5 ($w=2$) and</td>
<td>medSize: 62.5m$^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A2 ($w=1$) and</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ELSE propose A5 ($w=1$) and</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A2 ($w=2$) and</td>
<td></td>
</tr>
<tr>
<td>Minimum width (C4)</td>
<td>Legibility</td>
<td>3 (NCDB: 4)</td>
<td>3 (NCDB: 5)</td>
<td></td>
<td>A4: enlarge width locally</td>
<td>minEdge: 60m$^2$</td>
</tr>
<tr>
<td>Concavity (C6)</td>
<td>Preservation</td>
<td>1</td>
<td>4</td>
<td></td>
<td>Proposes A3 ($w=1$) if</td>
<td>delta: 10$^\circ$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>violated</td>
<td>Tolerance: 0.5$^\circ$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Proposes A4 ($w=1$) if</td>
<td>minWidth: 6.5m</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>violated</td>
<td>Tolerance: 0.5 m</td>
</tr>
</tbody>
</table>

Settings are similar to the Nouvelle Carte de Base project.
We adopted the parameters and plans as developed by experts during the two projects AGENT (Barrault et al. 2001) and NCDB (Lecordix et al. 2006), with small modifications. The parameters (thresholds) listed in Table 2 have different meanings. For instance, \textit{minSize}, \textit{delSize} and \textit{medSize} are used to ensure a minimum building size (constraint C1). Based on a comparison with the thresholds \textit{delSize} and \textit{medSize}, it is decided whether a building will be eliminated or enlarged to meet the minimum size constraint \textit{(minSize)}. Tolerance values, such as the one for \textit{minEdge}, should avoid actions that result in small, unnecessary changes when the current value is close to the goal value. For instance, if the length of a building wall is 6.0 m, then the building is not generalised with a simplification algorithm, due to the length tolerance of 0.5 m (threshold \textit{minEdge} set to 6.25 m). Since we assume that the settings and proposed plans of the constraints C1, C2, C3 and C4 are intuitive to be understood, we will only explain the settings for the defensive constraint C6 (concavity).

Constraint C6 should ensure that geometric transformations applied to one building do not change the building shape in an unacceptable way. Therefore the ratio of the area of the original building to the area of its convex hull is computed, and the ratio values before and after generalisation are compared (Bard 2004). In Table 2 it can be seen that the constraint C7 has a low priority, a high importance and does not propose any plans. The value for the priority parameter is low since priority proposes no plans. If the change of the building outline is too strong it is desired that the solution be rejected, and either another plan applied or the building flagged for subsequent interactive generalisation. A rejection is achieved in that the high importance value combined with a low concavity constraint satisfaction will result in a lower happiness value for the building than before. The lower happiness value will then prevent that this state is selected as the best solution, since other states (even the initial state) should receive higher happiness values.

**4.1.2 Expert rules introduced.** The settings of Table 2 are used to obtain the reference generalisation results for the comparison with the results gained with expert rules and learnt rules. The learnt rules, which we introduce to the generalisation system, are presented in the results section (Section 5) since they are derived from an actual generalisation run with the settings given in Table 2. In contrast, the modifications of the settings and plans of Table 2 evolving from expert rules are conceptual and the outcome of knowledge elicitation from experts. The context-dependent expert rules, presented in Table 3, have been elaborated by the authors when comparing the results of an MAS-based generalisation system with NMA map specifications from IGN France and Swisstopo. With these rules we aim to realise the following cartographic considerations: Industrial and commercial buildings should not be squared since the building sizes tend to be large and are adapted to the previously existing infrastructure. This also affects the possibility to simplify buildings to a rectangle, since a representation as a block is on the one hand inadequate with respect to their often complex shape and on the other hand may result in overlaps with other infrastructure objects (roads and buildings). Similar considerations exist with respect to inner city blocks. Usually, the individual buildings forming a block adapt to the nature of the topography and the existing infrastructure. Particularly in (European) old towns where the urban fabric has been shaped over centuries straight shapes of building blocks are rather unusual, as can be seen in the Zurich data set presented below (Figure 7, old town on the lower left). An additional rule applied to the inner city buildings is to eliminate unimportant
Table 3. Expert rules accounting for the specificities of five urban context classes, applied to the settings of Table 2. The proposed changes are explained in Section 4.1.2.

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Industry and commercial</th>
<th>Inner city</th>
<th>Urban</th>
<th>Suburban</th>
<th>Rural</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum size (C1)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>1. Set <code>delSize</code> to <code>minSize</code></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>2. Don’t propose enlarge to rectangle (A6)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Granularity (C2)</td>
<td>Don’t propose simplify to rectangle (A5)</td>
<td>Don’t propose simplify to rectangle (A5)</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Squareness (C3)</td>
<td>Don’t propose squaring (A3)</td>
<td>Don’t propose squaring (A3)</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

1. Set weight of enlarge to rectangle (A6) higher than for scale polygon (A1)
2. Don’t propose eliminate (A7)
3. Set weight of simplify to rectangle (A5) higher than for simplify (A2)
small buildings to strictly retain free space for necessary building enlargement and displacement operations. Assuming that in suburban areas residential districts dominate, consisting of individual and rather small buildings, we propose to enforce the plan that simplifies small houses to rectangles instead of trying out time-consuming building wall-by-wall simplification that is followed by an enlargement operation. This assumption is also applied to buildings in the rural context. A second objective for rural buildings is to preserve even small buildings as far as possible, since they may be an important point for the map reader’s orientation, for example, if the map is used for hiking.

From the above considerations and Table 3 it can be obtained that no specific rules are introduced for urban buildings. Thus, they are handled with the modified reference settings of the NCDB project. To all other urban context classes we have
been able to assign specific rules. Hence, every urban context class except ‘urban buildings’ has its own, cartographically justified set of rules.

4.1.3 Test data, generalisation system and learning framework. For the experimental part we used two data sets. The first data set from Switzerland, AV-Light data provided by the City of Zurich, contains building data with a resolution corresponding to a 1:5000 map scale. The buildings have been classified according to the approach described in Steiniger et al. (2008); see also Section 3.3 and Figure 4. In a step preceding the generalisation buildings touching each other were merged into one building. Such a merge could be done as well with the agent approach, but then we would need to introduce so-called meso agents that control several single building agents together (see Ruas and Duchêne 2007). This would add an additional level of complexity and make the experimental set-up and evaluation more difficult than necessary.

The second data set contains buildings from the region of Orthez in France and has been extracted from the IGN BD-Topo® database. The French data have a resolution of about 1 m, corresponding to a map scale of roughly 1:15,000. The building data are pre-classified with a classification approach of Boffet (2001), which is better adapted to the French data than our generic approach. To utilise this existing context classification we applied a mapping between the – partially similar – concepts given in Table 4. For the French data a merge operation for touching buildings has been applied as well if the buildings were of similar function type.

For the generalisation of the buildings we used the commercial map generalisation system Radius Clarity™ by I Spatial (2007). This software has been developed from the prototype of the AGENT project (Barrault et al. 2001) and the inference machine can be adapted to use different search heuristics (i.e. so-called pruning strategies). We applied generalisation algorithms delivered with Radius Clarity™ to explore potential algorithmic deficiencies of the commercial system for further experiments. As mentioned previously the system does not explore the full tree of possible generalisations for a building in order to avoid redundant search (Figure 3). The generalisation of an object or situation in Radius Clarity™ is valid if (1) the constraint satisfaction of the constraint proposing the plan has improved and (2) at least one of the constraint satisfaction values for the current solution (state) has improved compared to every previously generated solution.

<table>
<thead>
<tr>
<th>French urban context classes after Boffet (2001)</th>
<th>Mappings to urban context classes used in Steiniger et al. (2008)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Centre Ville</td>
<td>Inner city</td>
<td></td>
</tr>
<tr>
<td>Divers</td>
<td>Urban</td>
<td></td>
</tr>
<tr>
<td>Lotissement</td>
<td>Suburban</td>
<td></td>
</tr>
<tr>
<td>Peri urbain</td>
<td>Urban</td>
<td></td>
</tr>
<tr>
<td>Unitaire</td>
<td>Rural</td>
<td></td>
</tr>
<tr>
<td>Activité</td>
<td>Industrial and commercial</td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Mapping of the urban context classes given for French BD-Topo® data to the classes used for the expert rules of Table 3.
4.2 Evaluation strategy

4.2.1 Effectiveness – cartographic quality. An evaluation of the improvement in cartographic quality due to the introduction of urban context rules (i.e. the expert rules) is accomplished by visual inspection, not in a quantitative manner, by use of the overall happiness. This decision was necessary since our first experiments showed that we are not yet able to sufficiently formalise cartographic quality in quantitative terms (see discussion section Section 6.2). An improvement in cartographic quality for the rules inferred by machine learning is not possible, since these rules are derived from analysing the values of constraint satisfaction for the reference generalisation. In other words, the learning system does not receive information on how a ‘perfect’ generalisation result should look, but only information about the best result achieved with the reference setup. Hence, in terms of cartographic quality, the machine learnt rules will simply reproduce the result of the reference generalisation, provided the rule learner works perfectly; but it can be expected that this result will be achieved faster (Section 4.2.2).

4.2.2 Efficiency – processing speed. To evaluate whether the efficiency increased when rules are applied we generated statistics for the generalisation process with respect to: (a) the number of generalisation trials for one building, (b) the necessary time to generalise a building and (c) the average happiness. Differentiating between the number of solutions and the processing time is useful because different generalisation algorithms require different amounts of time for computation. Hence, the same number of solutions does not necessarily result in the same execution time for the overall generalisation process. For instance, the simplify algorithm (A2) is a comparatively time-consuming algorithm since it generalises every building wall separately. If the algorithm is avoided the efficiency with respect to processing time will improve whereas the number of tried solutions can still remain the same.

5. Results

5.1 Rules learnt

In the machine learning part of the experiment we concentrated on obtaining rules from the analysis of the course of the happiness function obtained in previous runs of the MAS. This happiness function is calculated as a weighted average of the constraint satisfaction for every state of a building agent. For the learning process we used 288 buildings as training data from the Zurich sample data set, which contains 724 buildings (Figure 7). As described in Section 3.4.1 we learnt three types of rules: (1) priority rules, (2) validity rules and (3) termination rules. All rules learnt for the Zurich data are listed in Table 5. It can be seen that two of the termination rules (rules 7 and 8) not only account for the evaluation of a constraint satisfaction but also for urban structure types. Thus, a first hint is provided that the introduction of the urban structure concepts helped to better characterise the buildings, which in effect may increase generalisation efficiency.

5.2 Results for expert rules

Figure 7 shows the generalisation results for a part of the Zurich data. As previously mentioned we did not apply displacement operations to make the identification of problems with individual buildings easier. Thus, overlaps between buildings and roads are possible. A comparison of the result for the reference settings of Table 2
Table 5. Rules learnt from a test generalisation of buildings from the Zurich data set (Figure 7, middle).

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 1</td>
<td>IF ((\text{minimum size satisfaction} \leq 9.5)^*) THEN constraint to solve next = \text{minimum size}</td>
</tr>
<tr>
<td>Rule 2</td>
<td>IF ((\text{minimum size satisfaction} &gt; 9.5) \land (\text{granularity satisfaction} \leq 7.5)) THEN constraint to solve next = \text{granularity}</td>
</tr>
<tr>
<td>Rule 3</td>
<td>IF ((\text{minimum size satisfaction} &gt; 9.5) \land (\text{granularity satisfaction} &gt; 7.5)) AND ((\text{minimum width satisfaction} \leq 7.5)) THEN constraint to solve next = \text{squareness}</td>
</tr>
<tr>
<td>Rule 4</td>
<td>IF ((\text{minimum size satisfaction} &gt; 9.5) \land (\text{granularity satisfaction} &gt; 7.5)) AND ((\text{minimum width satisfaction} &lt; 7.5)) \land (\text{squareness satisfaction} &gt; 8.5) OR ((\text{squareness satisfaction} &gt; 8.5)) THEN constraint to solve next = \text{minimum width}</td>
</tr>
<tr>
<td>Rule 5</td>
<td>IF ((\text{squareness satisfaction} = 10)) \land (\text{concavity satisfaction} \leq 5)) THEN \text{invalid state}</td>
</tr>
</tbody>
</table>

Rules for automated setting of constraint priority

Rules for checking the validity of a transformation

Rules for terminating the generalisation of a building

A satisfaction value of 10 corresponds to a fully satisfied constraint.

*Constraint satisfaction values are measured in 0.5 intervals.

(Figure 7, middle) with the control settings that are derived from expert rules and dependent on urban context (Figure 7, bottom) shows that especially for inner city and industrial and commercial buildings the cartographic quality is preserved. More specifically with the reference settings a number of buildings are simplified to a rectangle resulting in overlaps with streets and nearby buildings. With the context rules, such cases are avoided. A disadvantage of the introduced rules is that in some cases too much detail of the buildings is retained (see dashed circle in Figure 7). Deficiencies of the building simplification algorithm that is triggered by the granularity constraint can be recognised as well. Often courtyards are not preserved although they would be large enough for visualisation (see black rectangle in Figure 7).

To evaluate improvements in the efficiency we prepared the diagrams given in Figure 8. One can obtain from the time diagram that we achieved a processing time reduction by approximately 15% for the Zurich data. Such a significant reduction does not appear for the Orthez dataset, however (approx. 1%). We see a reason for these different results for the two datasets in the proportion of contextually generalised buildings compared to the reference generalisation. For instance, the fraction of inner city and industrial buildings is 21% for Zurich and only 6% for Orthez.

5.3 Results for learnt rules

For the Zurich dataset we obtain a similar reduction in processing time as for the expert rules of about 14% (see Figure 8). In contrast to the experiment with expert
rules, however, the learnt rules achieve a considerable time reduction of 27% for the Orthez dataset. This improvement in terms of efficiency has also been determined by Taillandier (2007) with the rule learner C4.5. A negative effect with respect to cartographic quality appears for the application of the machine learnt rules, which can be seen in Figure 9. In some cases the learnt rules propose the simplification of buildings to rectangles although the loss of detail is not acceptable from a cartographic point of view. In Figure 8 one can see that the average happiness value slightly decreased when the learnt rules are applied. This may indicate that the rules

Figure 8. Statistics for the generalisation of the data for Zurich, Switzerland, and Orthez, France. The absolute generalisation times for both data sets are not comparable due to different memory needs and logging settings.

Figure 9. Generalisation result for a selection of the French BDTopo® data showing parts of the town of Orthez. Generalisation map scale is 1:25 000. Different grey tones correspond to the five urban classes (see Figure 7). Data reproduced by permission of IGN France.
learnt are not necessarily leading to the best solution obtained with the reference system, that is, the learnt rules are not perfect.

5.4 Results for a combined application of expert and learnt rules

With respect to improvements in efficiency we obtained very satisfying results: For the Zurich data we measured a reduction in processing time of about 45%, and for the Orthez data set a reduction of approximately 30% (Figure 8). The positive effects of an improvement in visual quality resulting from the expert rules are retained. However, the degradations for a few buildings that can be recognised in Figure 9 also stem from the learnt rules. Fortunately in most of these cases this happened for rural buildings only. Thus, such strong simplifications could be avoided by introducing a corresponding expert rule specifically for rural buildings. An additional experiment with such an expert rule that preferred a building wall-by-wall simplification over a transformation to a rectangle prevented the oversimplification.

6. Discussion

6.1 Summary of improvements

This study departed from the hypothesis that the effectiveness and the efficiency of MAS for cartographic generalisation can be improved by a combination of expert rules, obtained from knowledge elicitation from expert cartographers, and rules that were machine learnt through an analysis of previous system runs.

As the above results demonstrate we generally obtained the improvements that we were striving for with respect to the chosen improvement method, that is, rule type. Table 6 summarises the improvements and also lists the main problems identified. The particular strength of the expert rules is the improvement of the cartographic quality, but also a reduction in processing time could be achieved as a secondary effect, at least for the Zurich data. The effect of the application of machine learnt rules is that the efficiency could be improved by a reduction in processing time.

The combined application of both rules types, that is, expert and machine learnt rules, came out as expected: The two approaches indeed affect different system components, and hence are complementary. We obtained satisfying improvements with respect to both the generalisation quality and efficiency. Thereby we discovered that the time reduction, achieved for expert and for learnt rules, seems to add up for the combination of rules.

In conclusion, the hypothesis formulated in Section 1 has been verified by the experiments. The only unsatisfactory result is that for a few rural buildings, the learnt rules propose a cartographically inappropriate simplification to rectangle operation. This effect has been avoided by introducing a specific expert rule. Thus, we showed that it is possible to balance drawbacks of learnt rules with expert rules due to the different adaptation points in the generalisation process.

6.2 Problems identified

Apart from the positive results given above we also discovered problems (summarised in the last row of Table 6). As can be obtained from the bar charts in Figure 8, the reduction in processing time (desired) correlates with a reduction of the average happiness value (undesired). The degradation of the average agent
happiness is owing to two problems. One cause for this effect is that in some cases rules are learnt that guide the generalisation process not towards the solution that is indeed the cartographically best solution. We assume that this problem relates to the termination rules and validity rules learnt. The second reason for the degradation of agent satisfaction is that we are not able to sufficiently formalise the desired cartographic quality with the available set of cartographic constraints. This fact becomes evident when visually comparing the results obtained with the reference generalisation settings and the cartographically more appealing results obtained with expert rules (Figure 7). That is, sometimes the building agents ‘believe’ that their happiness has been degraded while in fact it should have improved, but the constraints and measures available to the agents cannot detect the improvement in cartographic quality.

To further analyse this problem we visualised the differences of the happiness values in Figure 10. It is noticeable that for the majority of the buildings the happiness has not changed (light grey fill), which is reasonable, since most buildings were classified as urban and we did not apply specific rules to urban buildings. However, for some industrial and commercial buildings and inner city buildings the happiness has decreased (grey fill to dark grey fill), even though visual inspection suggests that a better cartographic representation is achieved or subsequent overlaps

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Result: effectiveness</th>
<th>Result: efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expert rules</td>
<td>Expert rules influence the proposed plans and their ranking (plan weight) based on evaluation of additional information</td>
<td>Better visual appearance of generalised objects (buildings), that is, preservation of object details and avoidance of object overlaps</td>
<td>Time reduction of: (a) 15% for Zurich data set, (b) 1% for Orthez dataset</td>
</tr>
<tr>
<td>Machine learning rules</td>
<td>Learnt rules influence the choice of the next plan by</td>
<td>Cartographic quality is similar to reference. In a few cases too strong change of the agent’s geometry, that is, building conversion to rectangle</td>
<td>Time reduction of 15% for Zurich dataset</td>
</tr>
<tr>
<td>Combination of rules</td>
<td>- changing the constraint priority</td>
<td>Improved cartographic quality according to the expert rules, but also with the deficiencies introduced by the learnt rules (i.e. conversion to rectangle in rare cases). Deficiencies can be prevented by additional expert rule</td>
<td>Time reduction of: (a) 45% for Zurich dataset and (b) 30% for Orthez dataset</td>
</tr>
<tr>
<td>Problems identified</td>
<td>- Agent happiness decreases although visual quality is better</td>
<td>- Learnt rules do not always guide towards the cartographically best solution</td>
<td>- Sometimes unsatisfactory results of building simplification algorithm</td>
</tr>
</tbody>
</table>
of buildings and roads are prevented. The lower happiness value is most often caused by an unsuitable weighting of the preserving concavity constraint (C6) and the active squareness constraint (C3) in the calculation of the happiness. We would like to give hints for potential solutions, facilitating a future detailed analysis and further studies.

Generally excluding the squareness constraint from the calculation of agent happiness for these types of buildings is not recommended since some walls of the buildings may need to be made orthogonal (e.g. in cases where the wall is not parallel to a road) or more simplified. However, we believe that a local structural analysis and the development of a specialised squareness constraint and more sophisticated generalisation algorithms may solve the problem. With respect to the use of the concavity constraint one should consider to differently weight the concavity constraint and to introduce further shape-preserving constraints as presented by Bard (2004). Such constraints can, for instance, penalise solutions by reducing the happiness if a building is represented by a rectangle.

A problem that subsequently emerges from the lack of detail in constraint formalisation appears for the machine learning of rules. If we learn from the reference generalisation (regardless if with or without expert rules) and the learning process tries to optimise the happiness function, then rules can be obtained that cause cartographically inappropriate solutions. Thus, for the learning part, we must ensure that the highest happiness value always is associated with the generalisation solution that is indeed the best solution in terms of cartographic quality.

Finally, a note should be made with respect to the generalisation algorithms of the \textit{Radius Clarity TM} software used in the experiments. We found that the building

Figure 10. Comparison of the happiness values reached for generalisation without rules versus with expert rules for the Zurich data. The happiness for the generalisation solution with expert rules is lower, but cartographic quality is better. Data courtesy of the City of Zurich, Geomatik + Vermessung, 16.10.2007.
simplification algorithm (A2) sometimes returns inadequate solutions (Figure 7). Here we see a need to either try out other simplification algorithms, such as the algorithms presented by Haunert and Wolff (2008) and Damen et al. (2008), or to adapt the current algorithm to specific cases. If different algorithms and different parameter settings should be utilised, then further constraints need to be applied, which better describe the requirements of cartographic quality than the currently used set of constraints. This would require a ranking between the solutions of different simplification algorithms on a finer scale. A solution with multiple algorithms would also increase the need for machine learning rules and the introduction of expert rules, since every branch of the tree of possible solutions (Figure 4) will grow a new sub-branch for every new algorithm that is added if no heuristics are applied.

### 7. Conclusions and outlook

Automated procedures for the generalisation of topographic maps are increasingly integrated in the production lines of NMAs. Apart from the external (customer) requirement to deliver timely mapping data and maps also internal requirements are imposed on such automated methods to be feasible for the production environment. Two of these internal requirements are to produce maps with a cartographic quality close to that of traditional, handmade maps and the ability to process large amounts of geo-data in acceptable time. Our work aims to contribute to the improvement of automated building generalisation with respect to these internal requirements. In particular, we focus on self-evaluating systems based on the multi-agent paradigm. Note, however, that our methodology could also be adapted to other types of self-evaluation generalisation systems with an explicit inference mechanism. In order to make the generalisation process more effective, that is to obtain a better cartographic quality, we introduced – previously not utilised – cartographic expert knowledge. To this end the expert knowledge has been formalised in terms of different building generalisation rules for different urban context classes. We restricted ourselves to using generalisation algorithms that merely modify single buildings, that is, no algorithms were used that take the context of several buildings into account, such as displacement, typification or amalgamation. Nevertheless, we achieved a context-dependent generalisation for different urban structure types as it is recommended in the cartographic literature (e.g. SSC 2005) and demanded by mapping agencies. An improvement of efficiency of the generalisation process was achieved by the utilisation of machine learning techniques. Based on previous work at IGN France, we developed an approach to learn three types of rules that should guide the generalisation system faster to the ‘best cartographic generalisation solution. Both approaches, the one that introduces expert rules and the one that utilises learnt rules, were realised in a complementary fashion and tested together for the first time.

In the experimental part, we generalised two building data sets from a map scale of 1:5000 and 1:15,000 to 1:25,000. We obtained satisfactory results for the combination of expert and learnt rules with respect to a gain in effectiveness and improvements of cartographic quality. Moreover, we observed that a degradation of few rural buildings resulting from machine learnt rules could be balanced and prevented by introducing an additional expert rule.

During the experiments we also identified problems in the ability to formalise, by means of constraints, the cartographic requirements for graphical quality. We see
here clearly a need for future research to better define shape constraints and parameter settings to ensure that the happiness values computed on the basis of constraint satisfaction indeed express the cartographic quality as it is visually experienced by the map reader. This will help us to improve the generalisation results on the one hand and on the other hand the learning approach will return better rules for a well-defined objective function. Note that these benefits would accrue for any constraint-based generalisation, not only MAS.

Other issues for further research can be identified. In our experiments, we only considered constraints acting on individual buildings, which are largely sufficient for our target scale of 1:25 000. Thus, further studies should consider larger scale reduction factors, such as from 1:25 000 to 1:50 000 and 1:100 000. Here, we see even more potential to influence the control and selection of generalisation algorithms based on the urban context classes, since more topographic detail needs to be reduced (Müller 1990), and hence more contextual generalisation operations are necessary. Thereby one should not only focus on the generalisation of buildings but may also control the generalisation of other object classes, including roads. For instance, Edwardes and Regnauld (2000) outline an approach for the differentiated generalisation of roads in urban, inner city and rural areas.

Finally, as a further research objective one should try to include semantic information in the urban classification where it is available, as exemplified by Boffet (2001, 2000), who used information on industrial/commercial areas. This will help us to identify misclassified buildings and enable us to introduce specific generalisation rules for objects of particular interest such as hospitals.

Acknowledgements
The research reported in this paper was partially funded by the Swiss NSF through grant no. 20-101798, project DEGEN. We are grateful to Julien Gaffuri for helping us to get started with Clarity and Cécile Duchêne for discussions during the experiments. We would specifically like to acknowledge the fruitful exchange of ideas with the IGN Nouvelle Carte de Base team. The authors are also grateful for the comments from anonymous reviewers.

References

BOFFET, A., 2001, Methode de creation d’information multi-niveaux: pour la generalisation cartographique de l’urbain. Thesis (PhD), Université de Marne-la-Vallée [in French].


REVELL, P., REGNAULD, N. and THOM, S., 2006, Generalising and symbolising Ordnance Survey base scale data to create a prototype 1:50 000 scale vector map. *In: The 9th ICA workshop on progress in automated map generalization*. Portland, WA.


