Belief Revision of GIS Systems: The Results of REV!GIS

Salem Benferhat[3], Jonathan Bennaim[4], Robert Jeansoulin[2], Mahat Khelfallah[2], Sylvain Lagrue[3], Odile Papini[1], Nic Wilson[5], and Eric Würbel[1]

[1] LSIS-CNRS, université du Sud Toulon -Var. BP 132. 83957 La Garde Cedex France
papini, wurbel@univ-tln.fr
[2] LSIS-CNRS, CMI technopôle de Château Gombert. 13353., Marseille cedex 13. France
jeansoulin, mahat@cmi.uni-mrs.fr
[3] CRIL-CNRS, université d'Artois. Rue Jean Souvraz. 62307 Lens Cedex. France
benferhat, lagrue@cril.univ-artois.fr
[4] LIF-CNRS, CMI technopôle de Château Gombert. 13353., Marseille cedex 13. France
jbennaim@lif.univ-mrs.fr
[5] University College Cork. Cork. Ireland
n.wilson@4C.ucc.ie

Abstract. This paper presents a synthesis of works performed on the practical tractability of revision on geographic information within the european REV!GIS project[1]. It surveys different representations of the revision problem as well as different implementations of the adopted statgey: Removed Set Revision (RSR). A comparison of the representation formalisms is provided, a formal and an experimental comparison is conducted on the various implementations on real scale applications in the context of GIS.

1 Introduction

One of the aim of the REV!GIS project was to investigate how artificial intelligence tools could be used to perform revision in the case of spatially referenced information. Within this project different formalisms have been proposed for representing geographic information with a special focus on practical tractability for symbolic change operations. The present paper provides a synthesis of works done during the project. It presents a comparison and a discussion on the different symbolic formalisms to represent geographic information as well as on the various implementations of the Removed Set Revision (RSR) experimented on real scale applications.

[1] This work was supported by European Community project IST-1999-14189 REVIGIS.

L. Godo (Ed.): ECSQARU 2005, LNAI 3571, pp. 452–464, 2005.

The paper is organized as follows. After a short reminder on the specificity of geographic information in Section 2 and on RSR in Section 3, we survey and discuss in Section 4 different symbolic representations of the revision problem on geographic information. For each representation, we briefly recall in Section 5 adjustments of existing strategies, taking advantage of the nature of geographic information, to perform revision. The results of the implementations of the different revision approaches are discussed in Section 6 before concluding in Section 7.

2 Geographic Information

Geographic information is made of numerous items gathered from different sources (instruments, surveys, images), and recorded as values on some specified domain, after a theory-based interpretation. Therefore, all these items can be qualified by some confidence level, depending on the supposed fitness of the interpretative process, for the actual situation, or some preference which expresses the subjective vision of what the world should be. These items of information are uncertain, incomplete or inaccurate, and they can conflict with each other. Hence they may require corrective operations: revision, update or fusion according to the context.

In terms of formal representation, the huge amount of data raises tractability problems. For instance, a small problem involving a hundred spatial regions, with ten attributes defined on finite domains of low cardinality, is represented by about one hundred thousand propositional clauses. Another problem is that what is observed differs from the variables expected in the model built by the user: we need to apply (uncertain) inference rules, for deriving such variables from several observed variables, and inconsistency can result. The size and the variety of the data seem to prevent any reasonable implementation of belief change operations when reasoning with geographic information.

Three main considerations can help us. First, the information relies on space where everything get situated, overlaps or coexists according to definite relations, topologic, metric or temporal: these constraints can reduce the size of the problem. Second, in case of inconsistency, the conflicts are local, and their detection and resolution can be carried out over restricted parts of the data set. Finally, the spatial relations translate into a particular syntax, which allows us to adapt existing algorithms into faster versions. In this work we consider the case, particular but very frequent, where the information is linked to non ubiquitous spatial locations, (1) either distinct, as a set of non overlapping spots, (2) or elements of a space partition (full coverage, no intersection). This limits the topology to only three relations: same, different, adjacent. The case of partially overlapping zones, or with undetermined limits, should be treated in a separate work, for further integration. We use the general term of parcel for refering to such locations, and we use the capital letters A, B, C etc. to denote them. Throughout the paper, the following two examples, extracted from real scale applications, will be used for illustration and comparison.

Example 1. Flooding application. The aim is to assess the water height in a flooded valley, which is segmented into parcels. We assess a minimum/maximum interval of water height for each parcel, where this height can be considered as constant. We have two sources of information (aside from the geographical layout): (1) a set of hydraulic relations between neighbouring parcels; (2) a set of initial assessments of minimal and/or maximal submersion heights for some parcels, for more details see [13] and [18]. For the illustration, we consider only 3 parcels, and 2 flow relations: from A to B and from A to C. The sampled observation domain is $D = \{1,\ 2,\ 3,\ 4\}$, and the assessments are $\{A : 2,\ B : 3,\ C : 4\}$, for the maximum submersion height and $\{A : 1,\ B : 1,\ C : 3\}$, for the minimum submersion height.

Example 2. Best location problem. The aim is to find the best location(s) for building a construction according to some constraints, which can be partially or totally ordered [14]. We consider 3 parcels, and 3 constraints: (C_1) to be near a fire hydrant, the domain for the distance being $D_d = \{very\ close,\ close,\ far,\ very\ \ far\}$, ($C_2$) to be far from a street intersection, same domain D_d, (C_3) to be built-free, in the domain $D_b = \{yes,\ no,\ may\ be\}$.

3 Removed Set Revision

We briefly recall RSR. We first transform the initial set of formulas in CNF for dealing with clauses. Let K and A be finite sets of clauses, the method focuses on the minimal subsets of clauses to remove from K, called *removed sets* [12], in order to restore consistency of $K \cup A$. More formally:

Definition 1. *Let K and A be two sets of clauses such that $K \cup A$ is inconsistent. R a subset of clauses of $K \cup A$, is a* removed set *iff (i) $R \subseteq K$; (ii) $(K \cup A) \backslash R$ is consistent; iii) $\forall R' \subseteq K$, if $(K \cup A) \backslash R'$ is consistent then $\mid R \mid < \mid R' \mid$*[2]*.*

Definition 2. *Let K, K' and A be finite sets of clauses.* $K \circ_R A = \{K' \cup A, \text{such that } K' = K \backslash R, \text{ where } R \text{ is a removed set}\}$.

The removed sets define a family of revision operations $\circ_R$ which satisfy the AGM postulates and it can be checked that if R is a removed set then $(K \cup A) \backslash R$ is a so-called cardinality-based maximal consistent subbase of $(K \cup A)$ [2], [6] [10].

4 Representing Revision on Geographic Information

4.1 Propositional Clausal Form Representation

The most basic representation is the propositional clausal form. Representing geographic information with propositional calculus takes advantage of the simplicity of expression of this language and, from a computational point of view,

[2] $\mid R \mid$ denotes the number of clauses of R.

takes advantage of the decidability of this logic. Moreover most of the change operations defined in the area of knowledge representation are defined in propositional calculus.

The nature of geographic information knowledge leads to a special propositional clausal form representation. Any proposition refers to some phenomenon linked to one parcel, and we use the propositional variables a^k to denote the propositions concerning a phenomenon k which are linked to the parcel A. The arbitrary rank k varies within a finite set.

For measurable observations, the propositions represent the numerical response of some phenomenon against a finite, sampled, domain of values $D = \{v_1, \ldots, v_n\}$, the notation becomes a_i^k, $1 \leq i \leq n$, to remind us about the phenomenon (ranked by k) and the questioned value (phenomenon $k = v_i$). Therefore, the representation of a particular set of observations is encoded by the n-ary clause $a_1^k \vee a_2^k \vee \cdots \vee a_n^k$ and $n(n-1)/2$ binary mutual exclusive clauses: $\neg a_i^k \vee \neg a_j^k$, $1 \leq i \leq n$ and $j > i$. The binary relations between measures, for instance a simple inequality, a linear equation, or a more complex mathematical formula, can be represented by couples of forbidden values which are encoded in binary negative clauses. From now on, S^O, S^D and S^C denote the set of clauses representing the observations, the domain and the relations respectively.

Revision of a Set of Clauses by a Set of Clauses. The revision problem amounts to revising the set of clauses S^O by the set of clauses $S^D \cup S^C$.

Example 3. In the flooding application, for each parcel A, B and C, we define the propositional variables a^+ and a^-, b^+ and b^-, c^+ and c^- for maximal and minimal submersion height respectively[3]. These variables are defined on a domain $D = \{1, 2, 3, 4\}$. The set of clauses S^D representing the finite domain consists, for each variable, in one 4-ary clause and 6 binary negatives clauses. For instance, the clauses corresponding to the variable a^+ are the 4-ary clause $a_1^+ \vee a_2^+ \vee a_3^+ \vee a_4^+$ and the 6 binary negatives clauses $\neg a_i^+ \vee \neg a_j^+$, $i \in D$, $i \neq j$. The set of clauses representing the observations is $S^O = \{a_1^-, a_2^+, b_1^-, b_3^+, c_3^-, c_4^+\}$. The set of clauses S^C representing the flow relations between parcels is the set of forbidden couples for the inequalities representing the flow relations. For example the relation $a^+ \geq b^+$ is represented by the clauses $\neg a_1^+ \vee \neg b_2^+, \neg a_1^+ \vee \neg b_3^+, \neg a_1^+ \vee \neg b_4^+$, $\neg a_2^+ \vee \neg b_3^+, \neg a_2^+ \vee \neg b_4^+, \neg a_3^+ \vee \neg b_4^+$. The revision problem amounts to revising the set of clauses S^O by the set of clauses $S^D \cup S^C$.

Translation Into a Satisfiability Problem. We use the transformation proposed by De Kleer for ATMS [9]. Each clause c of S^O is replaced by the formula $\phi_c \rightarrow c$, where ϕ_c is a new variable, called hypothesis variable. If ϕ_c is assigned true then $\phi_c \rightarrow c$ is true iff c is true, this enforces c. On contrast if ϕ_c is assigned false then $\phi_c \rightarrow c$ is true whatever the truth value of c, the clause c is ignored. Let us denote $\mathcal{H}(S^O)$ the transformed set. The revision problem corresponds to the satisfiability of the set of clauses $\mathcal{H}(S^O) \cup (S^D \cup S^C)$ with some conditions on hypothesis variables ϕ_c according to the revision method, for instance minimizing the number of falsified hypothesis variables ϕ_c.

[3] For a better understanding we denote a^+ instead of a^1 and a^- instead of a^2.

Translation Into a ROBDD. A set of clauses can be compactly encoded in a Reduced Ordered Binary Decision Diagram (ROBDD), which is a labeled acyclic directed graph [4]. Using the transformation $\mathcal{H}$ defined above, the revision problem amounts to find the shortest path in the ROBDD corresponding to the set of clauses $\mathcal{H}(S^O) \cup (S^D \cup S^C)$ as described in [17].

4.2 Logic Programming Representation

Standard Logic Programming. In a standard logic programming approach (PROLOG or DATALOG), the observations are represented by facts, and relations between observations are represented by facts and rules. Inconstency rules have to be explicitely provided. The revision problem amounts to defining rules involving the facts representing the observations to solve the inconsistencies.

Logic Programming with Answer Set Semantics. In this approach we directly translate the revision problem into a logic program with anwser set semantics (ASP) [3]. This translation is suitable for Removed Set Revision. Firstly, for each clause c of S^O, we introduce a new atom in V the set of atoms ocurring in $S^O \cup (S^D \cup S^C)$. We then construct a logic progam $P_{S^O \cup (S^D \cup S^C)}$ whose anwser sets correspond to subsets R of S^O such that $(S^O \cup (S^D \cup S^C)) \backslash R$ is consistent. This construction stems from the enumeration of interpretations of V and a progressive elimination of interpretations. For more details see [3].

4.3 Constraint Satisfaction Problem Representation

Let $\mathcal{X}$ be a set of variables and $\mathcal{D}$ a set of sampled, hence discrete domains. The observations and the relations are encoded by the following CSPs denoted by $\mathcal{P}_O = \{\mathcal{X}, \mathcal{D}, \mathcal{C}_O, \mathcal{R}_O\}$ and $\mathcal{P}_C = \{\mathcal{X}, \mathcal{D}, \mathcal{C}_C, \mathcal{R}_C\}$ respectively where $\mathcal{C}_O$ and $\mathcal{C}_C$ are the constraints on the variables and $\mathcal{R}_O$ and $\mathcal{R}_C$ the relations [17].

4.4 Linear Constraint Representation

When the variable domain is continuous and the relations between the variables can be represented by linear constraints, another representation stems from the Logic of Linear Constraints (LLC) [15]. Within this framework, a variable X_i is associated with each parcel i. The measures and observations are given as intervals $[l_i, u_i]$ of possible values for variables X_i, where l_i and u_i are real scalars. The set of measures and observations is represented by a set $\mathcal{L}^O$ of linear constraints of the form $X_i \geq l_i$, called the *lower bound* constraints or constraints of the form $X_i \leq u_i$, called the *upper bound* constraints. The set of relations between variables is represented by a set $\mathcal{L}^C$ of linear constraints of the form $X_i - X_j \geq a_{ij}$ called *rules*, where a_{ij} is a real scalar. If $D = [L, U]$ is the domain of variables, then the variable domain represented by the set of linear constraints $\mathcal{L}^D$ consists, for each parcel i, in the constraint $L \leq X_i \leq U$. If the set of constraints $\mathcal{L}^O \cup \mathcal{L}^C \cup \mathcal{L}^D$ is inconsistent, the revision amounts to identifying constraints of $\mathcal{L}^O$ whose removal is sufficient to restore the consistency.

In the following section we present how revision is performed according to the different representations.

5 Performing Revision on Geographic Information

Revision in the framework of geogaphic information has been performed according to RSR strategy with suitable adjustments in order to take advantage of the spatial knowledge representation [7].

5.1 Removed Set Revision Using Hitting Sets (REM Algorithm)

The direct computation of removed sets consists in removing a clause from each element of the collection of minimal inconsistent subsets of $S^O \cup S^D \cup S^C$ without listing all elements of this collection. This strategy stems from the notion of minimal hitting set which is a minimal set of clauses that intersects with each minimal inconsistent subset. R is a removed set iff it is a minimal hitting set of the collection $\mathcal{I}\left(S^O \cup S^D \cup S^C\right)$ of the inconsistent subsets of $S^O \cup S^D \cup S^C$. This is described in [18] and [19].

5.2 Removed Set Revision as a SAT Problem

Using the represensation proposed in 4.1 the Removed Set Revision of S^O by $S^D \cup S^C$ corresponds to the problem of looking for a model of the set of clauses $\mathcal{H}(S^O) \cup (S^D \cup S^C)$ which minimizes the number of falsified hypothesis variables ϕ_c. This leads to the definition of a preference relation between interpretations stemming from the number of hypothesis variables they falsify, denoted by H_{S^O}-preference. Let M be a model of $\mathcal{H}(S^O) \cup (S^D \cup S^C)$ generated by a removed set R, then R is a removed set iff M is a H_{S^O}-preferred model of $\mathcal{H}(S^O) \cup (S^D \cup S^C)$ [3]. Performing Removed Set Revision amounts to looking for the H_{S^O}-preferred model of $\mathcal{H}(S^O) \cup (S^D \cup S^C)$. This can be achieved using a SAT-solver. In order to compare different implementations of Removed Set Revision we used the SAT-solver MiniSat[5].

5.3 Removed Set Revision with ROBDD

As shown in section 4, we can build a ROBDD representing $\mathcal{H}(S^O) \cup (S^D \cup S^C)$. In this context, minimizing the number of clauses to remove from S^O amounts to minimizing the number of hypothesis variables ϕ_c assigned false, see [19].

5.4 Revision in the Framework of Constraint Satisfaction Problems

In section 4, we described how to represent geographic information using the CSP framework. In this context, a revision situation arises when the problem $\mathcal{P}_{O \cup C} = \{\mathcal{X}, \mathcal{D}, \mathcal{C}_{O \cup C}, \mathcal{R}_{O \cup C}\}$ has no solution (we say that $\mathcal{P}_{O \cup C}$ is *overconstrained*), that is, there is no affectation of the variables which simultaneously satisfies all the constraints. This a static aspect of CSP, which is a limitation of the use of CSP in real situations [1]. This situation can be mainly addressed by two kind of approaches, Partial CSP (PCSP) and Flexible CSP (FCSP).

5.5 Prioritized Removed Set Revision with ASP

We now present the Prioritized Removed Set Revision (PRSR) which generalizes the Removed Set Revision to the case of prioritized belief bases. Let S^O be a prioritized finite set of clauses, where S^O is partitioned into n strata, i. e. $S^O = S_1^O \cup \ldots \cup S_n^O$, such that clauses in S_i^O have the same level of priority and are more prioritary than the ones in S_j^O where $j > i$. S_1^O contains the clauses which are the most prioritary beliefs in S^O, and S_n^O contains the ones which are the least prioritary in S^O [2].

When S^O is prioritized in order to restore consistency the principle of minimal change stems from removing the minimum number of clauses from S_1^O, then the minimum number of clauses in S_2^O, and so on. We generalize the notion of removed set in order to perform Removed Sets Revision with prioritized sets of clauses[4]. This generalization first requires the introduction of a preference relation between subsets of S^O and leads the definition of prioritized removed sets detailed in [2]. This definition of removed sets generalizes the definition 1. We directly translate the revision problem into an a logic program with anwser set semantics. We build a logic program denoted by $P_{S^O \cup (S^D \cup S^C)}$ such that the anwser sets of $P_{S^O \cup (S^D \cup S^C)}$ correspond to removed sets of $S^O \cup ((S^D \cup S^C)$. We then define a preference relation between anwser sets stemming from the preference relation between subsets of S^O and we establish the correspondence between prioritized removed sets and preferred answer sets. The computation of Prioritized Removed Sets Revision is based on the adaptation of the smodels system. This is achieved in two steps. The first step, Prio, is an adaptation of smodels [11] system which computes the set of subsets of literals of R_{S^O} which lead to preferred anwser sets and which minimize the number of clauses to remove from each stratum. The second step, Rens, computes the prioritized removed sets of $S^O \cup (S^D \cup S^C)$ stratum by stratum [3].

5.6 Revision in the Framework of Logic of Linear Constraints

In this approach we revise the set of bound constraints $\mathcal{L}^O$ by the set of rules $\mathcal{L}^C$. The revision method consists in first checking the consistency of the set of constraints $\mathcal{L}^O \cup \mathcal{L}^C$, this is performed by propagation of upper and lower bound constraints. In case of inconsistency, we have to identify the best subset(s), in terms of cardinality, of bound constraints $\mathcal{L}^O$ whose removal is sufficient to restore consistency. This is achieved by assigning each bound constraints in conflict $X_i \leq u_i$ (resp. $X_j \geq l_j$) a propositional variable U_i (resp. L_j) and to look for the models of $\neg(\bigvee_{i,j}(U_i \wedge L_j))$. For more details see [16].

6 Comparison

6.1 Comparison Between the Different Representations

We need to design a comparison framework suitable for geographic information. As specified in table 1, a first classification stems from the different levels of

[4] When there is no stratification PRSR amounts to RSR.

Table 1. Comparison between representation formalisms

available information	epistemic state	representation formalism	logic
unordered information	belief set	propositional representation, ROBDD, SAT, ASP,	PL
		PROLOG	FL
partially ordered information	partial pre-order	propositional representation + partially ordered information	PL
totally ordered information	total pre-order	propositional representation + totally ordered information	PL
		propositional representation + quality	PL
	total order	Flexible CSP	HL
	dense total order	LLC	HL

representation of the epistemic states, depending on the available information on the relations between observations. Another classification can be made according to the different levels of the underlying logical formalisms, propositional logic (PL), first order logic (FL) or high order logics (HL).

The propositional logic involves a huge amount of propositional variables and clauses, though it takes advantage of the existing algorithms for revision in the propositional case, of possible translation into SAT problem and of compact representation with ROBDD approaches. The inconsistency is not explicit but comes out from the resolution of the satisfaction. The main drawback is the loss of the structure of the initial problem. However representing the quality of data, with, for example, a total pre-order on propositional variables allows us to give again a certain structure to the representation. Consequently, this reduces the search space.

The standard logic programming approach is very close to natural languages and directly representable in relational database. However the difficulties are twofold. Inconsistency rules have to be explicited, but these rules depend on the problem and there is no general formulation. Besides, revision rules have to be defined, these rules also depend on the problem and on the strategy used to solve the revision problem, the formulation of such rules is, in general, very difficult.

On contrast, the propositional clausal representation of the problem can be translated into a normal logic program with anwser set semantics (ASP) stemming from the used revision strategy, like the proposed translation for Removed Set revision [3]. This is not suprising, since there is an equivalence between revision and non-monotonic inference. The inference relation used in standard logic programming is a monotonic inference relation whereas normal logic programming with anwser set semantics uses non-monotonic inference.

The CSP representation provides a compact representation since it involves a smaller number of variables. Moreover, this representation is more expressive, because the relations capture part of the stucture of the problem. In the example from the flooding application, when dealing with 3 parcels there are 6 variables while there are 24 variables for the clausal representation and the set of relations given in intension, expresses the flow. Since standard CSP uses monotonic inference, Flexible or Fuzzy CSP is suitable for representing revision, however the relaxation of constraints may modify a lot of conflicts. The minimality of change takes the form of minimality in terms of optimization and compromises the principle of minimal change in terms of minimal change of explicit beliefs.

The LLC formalism also provides a compact representation, since it uses real valued variables. The domain consisting in the real numbers is continuous and dense, and given in intension as well as the relations that express in a very natural way the structure of the problem. In the flooding application, when dealing with 3 parcels, there are 3 real variables and the relations are $A \geq B$ and $A \geq C$, which is a very natural and simple way for expressing the flow relations. However the LLC representation is not general, it is suitable for linear problems, but not for non-linear problems. This is not always the case when dealing with geographical information because we also have to deal with qualitative data defined on discrete domains, like shapes or colors, for example, or boolean data, and not every problem can be represented in terms of linear constraints as illustrated by example 2.

The expression of the revision problem is different in the different representations, however the revision problem is the same. The revision problem consists in identifying the conflicting observations to modify in order to restore consistency.

6.2 Comparison Between the Different Revision Approaches

The different approaches of revision presented in Section 5 can be classified according to the different levels of the underlying logical formalisms. In propositional approaches and first order logic representations the loss of structure put all conflicts at the same level whereas in higher order logic representations some conflicts can be solved by constraint propagation. This leads to a classification of the approaches into two categories. The first category encompasses all revision operations which concentrate on the detection of the conflicts between different sources of information. The second category contains all approaches which concentrate on the direct resolution of the conflicts by means of propagation mechanisms.

Comparison Between the Approaches Stemming from Conflict Detection. The first category (i.e. conflict identification) contains all approaches based on propositional logic. They perform RSR using the previously described representations. The main part of the work on these approaches had been to provide an adequate revision machinery in order to break down the complexity inherent to logical based reasoning. More precisely, the "complexity break down" work has been tackled using two different points of view. On one hand encoding the knowledge by means of propositional clauses and finding heuristics lowering the complexity of the satisfiability tests needed during the revision process. On the other hand using knowledge compilation techniques to perform all computationaly heavy tasks during a compilation phase, yet allowing us to work further on lighter representations of our knowledge.

Typically, compiled forms of the knowledge allows satisfiability test to be done with a worst case time complexity linear in the number variables or even constant. These approaches are summarized in the following table:

Approach	Type	Comments
RSR with REM	Clauses	hitting sets
RSR with SAT	Clauses	preferred models
RSR with ASP	Clauses	preferred models
PRSR with ASP	Clauses	strat., preferred models, the most efficient
RSR with ROBDD	Knowledge compilation	Compilation stage size problems

These approaches have been shown to be equivalent [19, 3], they provide the same removed sets, except, of course, in the PRSR case.

Experimental Comparison. All experimental comparison and measures have been presented in previously published work [19, 3]. We just recall here the main results. In [19] it has been shown that the REM algorithm described in subsection 5.1 which computes the removed sets by using a modification of Reiter's algorithm for the computation of minimal hitting set gives better results than the ROBDD approach. A comparison between the REM algorithm and the Rens algorithm which is an adaptation of the smodels system for RSR with ASP in [3] showed that the adaptation of the smodels system for RSR with ASP gave the best results. In [3], we compared the SAT approach which uses the efficient SAT-solver MiniSat and to the Rens algorithm which is an adaptation of the smodels system for RSR with ASP. This test showed that Rens gave the best results.

However, RSR with ASP can deal with 60 parcels with a reasonable running time (few minutes) but reaches a CPU time limit (10 hours) around 64 parcels. In the flooding application we have to deal with a block consisting of 120 parcels and the stratification is useful to deal with the whole area. A stratification of S_1 is induced from the geographic position of parcels. Parcels located in the upstream part of the valley are preferred to the parcels located in the downstream part of the valley. Using a stratification of S_1, we observed that Rens algorithm for PRSR with ASP can deal with the whole area with a reasonable running time [3].

Comparison Between the Approaches Stemming from Propagation. The second category of approaches is the "propagation" set of approaches. This category contains the original method used by CEMAGREF to solve the problem before we start our common work on this project. It is a purely numerical method, which tries to correct conflicting information as soon as it is discovered. The search space of the conflicts is reduced by using the upstream/downstream orientation of the flooded valley. The complexity of this method is very low (almost linear in the number of parcels).

The second method contained in this "conflict correction" category is based on the Logic of Linear Constraints (LLC) and a directed propagation algorithm, proposed in [8]. This approach is a logical framework for the original approach developped by the CEMAGREF and follows 2 steps. The first step, the conflict detection, stems from the propagation of the upper bounds or lower bounds constraints (worst case time complexity : $O(n^2)$, n being the # of parcels). In a second step, a logical formula is then constructed from the list of detected conflicts according to the process described in Section 5.6. Therefore the determination of the models of this formula which falsify the least number of literals of this formula corresponds to the determination of the subsets of constraints to revise. Since a Davis and Putnam procedure was used to compute the models, the complexity in the worst case is exponential in the number of detected conflicts. However, an experimental study has shown that revision using LLC is efficient because the number of detected conflicts is generally low. In the flooding application, for the whole area consisting of the 3 blocks, that is 200 parcels only 15 conflicts are detected and the algorithm provides 128 subsets of constraints to revise, each subset consisting in 10 constraints. The global running time for revision for LLC is around 2 seconds.

The main differences between LLC and CSP approaches are the following. FCSP approach deals with finite discrete domains which is not the case of LLC which deals with variables defined on continuous domains. The FCSP approach uses constraints defined with a degree of uncertainty (or degree of satisfaction) that allows us to represent uncertain data and preferences. On contrast, in LLC the constraints are satisfied or not. LLC follows the principle of minimal change in minimizing the number of constraints to revise, in a similar way as RSR while the minimality of change in the FCSP approach amounts to the min-max optimization (maximizing the degree of satisfaction of the less satisfied constraint).

Comparison Between Conflict Detection and Propagation. Directly comparing the two preceeding classes of approaches is rather difficult since they tackle the problem from different points of view. On one hand purely logical approaches concentrate on the detection of minimal sets of conflicts. On the other hand propagation approaches try to detect conflicts while solving the problem at the same time by the mean of constraint propagation.

The minimal change principle is not the same in the two classes of approaches as stated above.

The constraints propagation approaches provide best running times since they take into account the structure of the problem, while in the detection approaches the loss of structure of the in initial problem put all conflicts at the same level. However, the propagation approaches are not general, they are suitable only for linear problems, while non-linear problems can be dealt with detection approaches. By the way, we can list the pros and the cons of the two families of approaches:

	Conflict detection	Propagation
Pros	Focuses on the explanation of the conflicts suitable for non linear problems	Directly delivers a solution Low worst case time complexity (quadratic)
Cons	No numerical results (no refinment of initial assessments) High worst case complexity in the general case	Less general (bound to linear problems) Computation of minimal sets of conflicts is “ad-hoc” if we do not use a logical revision framework

7 Conclusion

We studied different representations of the revision problem on geographic information. We then discussed the advantages and the drawbacks of the different representations and we illustrated the revision problem by examples extracted from real scale applications. According to each representation, we then proposed adjustments of existing strategies, taking advantage of the nature of geographic information, to perform revision. We implemented the different revision approaches and we conducted an experimental study on the flooding application. The comparison between the different approaches leads to a classification into two classes of approaches, the propagation approaches which are not general but suitable and very efficient for linear revision problems and the logical approaches which are less efficient for linear revision problems, but more general and suitable for non-linear problems.

The problem of merging multiple sources of information is central in GIS. Since revision is a special case of fusion with two sources where one source is preferred to the other, it could be interesting to investigate how we could generalize the adjustments proposed for revision to fusion.

References

1. A. Bellicha and al. Autour du problème de satisfaction de contraintes. In *Actes des 5èmes journées nationales du PRC GDR Intelligence Artificielle*, pages 159–178, 1995.
2. S. Benferhat, Cayrol C, D. Dubois, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *Proc. of IJCAI'93*, pages 640–645, 1993.
3. J. Bennaim, S. Benferhat, O. Papini, and E. Würbel. An answer set programming approach of prioritized removed sets revision : Application to gis. In *Proc. of JELIA'04*, pages 604–616. LNAI, 2004.
4. R. E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on computers*, C-35(8):677–691, Aout 1986.
5. N. Eén and N. Sörensson. An Extensible SAT-solver. In *Proc. of ICTAS'03*, 2003.
6. De Kleer J. Using crude probability estimates to guide diagnosis. *Artificial Intelligence*, 45:381–392, 1990.
7. R. Jeansoulin and O. Papini. Révision et information spatiale. In *Le temps, l'espace et l'evolutif en sciences du traitement de l'information*, pages 293–304. Cépadues-éditions, 2000.
8. M. Khelfallah, N. Wilson, and R. Jeansoulin. Geographic Information Revision based on linear constraints. In *Tech. Rep., Annex 313.4 of report R313, REVIGIS project*, 2003.
9. J. De Kleer. An assumption-based TMS. *Artificial Intelligence*, 28:127–162, 1986.
10. Bernhard Nebel. Syntax based approaches to belief revision. In Peter Gärdenfors, editor, *Belief Revision*, pages 52–88. Cambridge University Press, UK, 1992.
11. I. Niemela and P. Simons. An implementation of stable model and well-founded semantics for normal logic programs. In *Proc. of LPNMR'97*, pages 420–429, 1997.
12. O. Papini. A complete revision function in propositionnal calculus. In B. Neumann, editor, *Proc. of ECAI'92*, pages 339–343. John Wiley and Sons. Ltd, 1992.

13. D. Raclot and C. Puech. Photographies aériennes et inondation : globalisation d'informations floues par un système de contraintes pour définir les niveaux d'eau en zone inondée. *Revue internationale de géomatique*, 8(1):191–206, 1998.
14. S.Lagrue, R. Devillers, and J-Y. Besqueut. Partially ordered preferences applied to the site location problem in urban planning. In *Proc. of DEXA'04*, 2004.
15. N. Wilson. The logic of linear constraints and its application to the flooding problem. Technical report, REVIGIS project report, 2002.
16. N. Wilson, M.Khelfallah, and R. Jeansoulin. Geographic information revision based on linear constraints. Technical report, REVIGIS project- Annex 313.4, 2003.
17. E. Wurbel, R. Jeansoulin, and O.Papini. Spatial information revision : A comparision between 3 approaches. In *Proc. of ECSQARU'2001*, pages 454–465. L NA I, 2143, Springer, 2001.
18. E. Würbel, R. Jeansoulin, and O. Papini. Revision : An application in the framework of gis. In *Proc. of KR'2000*, pages 505–516. Morgan Kaufmann, 2000.
19. E. Würbel, R. Jeansoulin, and O. Papini. Spatial information revision : A comparision between 3 approaches. In *Proc. of ECSQARU'2001*, number 2143 in LNAI, pages 454–465. Springer Verlag, 2001.