A LOCAL SEARCH APPROACH TO SOLVE INCOMPLETE FUZZY CSPs

Mirco Gelain, Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable
Department of Pure and Applied Mathematics, University of Padova, 35121 Padova, Italy

Toby Walsh
NICTA and UNSW, Sydney, Australia

Keywords: Local search, Preferences, Incompleteness, Elicitation.

Abstract: We consider fuzzy constraint problems where some of the preferences may be unspecified. This models, for example, settings where agents are distributed and have privacy issues, or where there is an ongoing preference elicitation process. In this context, we study how to find an optimal solution without having to wait for all the preferences. In particular, we define local search algorithms that interleave search and preference elicitation, with the goal to find a solution which is "necessarily optimal", that is, optimal no matter what the missing data are, while asking the user to reveal as few preferences as possible. While in the past this problem has been tackled with a branch & bound approach, which was guaranteed to find a solution with this property, we now want to see whether a local search approach can solve such problems optimally, or obtain a good quality solution, with fewer resources. At each step, our local search algorithm moves from the current solution to a new one, which differs in the value of a variable. The variable to reassign and its new value are chosen so to maximize the quality of the next solution. To compute this, we elicit some of the missing preferences in the neighbor solutions. Experimental results on randomly generated fuzzy CSPs with missing preferences show that our local search approach is promising, both in terms of percentage of elicited preferences and scaling properties.

1 INTRODUCTION

Constraint programming (Rossi et al., 2006) is a powerful paradigm for solving scheduling, planning, and resource allocation problems. A problem is represented by a set of variables, each with a domain of values, and a set of constraints. A solution is an assignment of values to the variables which satisfies all constraints and which optionally maximizes/minimizes an objective function. Soft constraints (Bistarelli et al., 1997) are a way to model optimization problems by allowing for several levels of satisfiability, modeled by the use of preference or cost values that represent how much we like a certain way to instantiate the variables of a constraint. Incomplete fuzzy constraint problems (IFCSPs) (Gelain et al., 2010a; Gelain et al., 2007; Gelain et al., 2008) can model situations in which some preferences are missing before solving starts. In this scenario, we aim at finding a solution which is necessarily optimal, that is, optimal no matter what the missing preferences are, while asking the user to reveal as few preferences as possible.

While in the past this problem has been tackled with a branch & bound approach, which was guaranteed to find a solution with this property, we now show that a local search method can solve such problems optimally, or obtain a good quality solution, with fewer resources. Our local search algorithm starts from a randomly chosen assignment to all the variables and, at each step, moves to the most promising assignment in its neighborhood, obtained by changing the value of one variable. The variable to reassign and its new value are chosen so to maximize the quality of the new assignment. To do this, we elicit some of the preferences missing in the neighbor assignments. The algorithm stops when a given limit on the number of steps is reached. Experimental results on randomly generated fuzzy CSPs with missing preferences show that our local search approach is promising, in terms of percentage of elicited preferences, solution quality,

Gelain M., Pini M., Rossi F., Brent Venable K. and Walsh T..
A LOCAL SEARCH APPROACH TO SOLVE INCOMPLETE FUZZY CSPs.
DOI: 10.5220/0003174505820585
In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence* (ICAART-2011), pages 582-585
ISBN: 978-989-8425-40-9

and scaling properties.

Incompleteness in soft constraints has also been considered in (Fargier et al., 1996; Pini et al., 2010). However, incomplete knowledge in these contexts is represented by the presence of uncontrollable variables, i.e., variables such that the agent has not the power to fix their value, and not by the presence of missing preferences.

A preliminary version of some results of the paper is contained in (Gelain et al., 2010b)[1].

2 BACKGROUND

We now give some basic notions about incomplete fuzzy constraint problems and local search.

Incomplete Fuzzy Problems. Incomplete Fuzzy constraint problems (IFCSPs) (Gelain et al., 2010a) extend Fuzzy Constraint Problems (FCSPs) (Bistarelli et al., 1997) to deal with partial information. In FCSPs preference values are between 0 and 1, the preference of a solution is the minimum preference value contributed by the constraints, and the optimal solutions are those with the highest value.

More formally, given a set of variables V with finite domain D, an *incomplete fuzzy constraint* is a pair $\langle idef, con\rangle$ where $con \subseteq V$ is the scope of the constraint and $idef : D^{|con|} \longrightarrow ([0,1] \cup \{?\})$ is the preference function of the constraint associating to each tuple of assignments to the variables in *con* either a preference value that belongs to the set $[0,1]$, or ?. All tuples mapped into ? by $idef$ are called *incomplete tuples*, meaning that their preference is unspecified.

Given an assignment s to all the variables of an IFCSP P, $pref(P,s)$ is the preference of s in P. More precisely, it is defined as $pref(P,s) = min_{<idef,con>\in C | idef(s_{\downarrow con})\neq ?} idef(s_{\downarrow con})$. In words, the preference of an assignment s in an IFCSP is obtained by taking the minimum of the known preferences associated to the projections of the assignment, that is, of the appropriated sub-tuples in the constraints.

In the fuzzy context, a complete assignment of values to all the variables is an optimal solution if its preference is maximal. This optimality notion is generalized to IFCSPs via the notion of *necessarily optimal solutions*, that is, complete assignments which are maximal no matter the value of the unknown preferences.

[1]Research partially supported by the Italian MIUR PRIN project 20089M932N: "Innovative and multidisciplinary approaches for constraint and preference reasoning".

In (Gelain et al., 2010a) several algorithms are proposed to find a necessarily optimal solution of an IFCSP. All these algorithms follow a branch and bound schema where search is interleaved with elicitation. Elicitation is needed since the given problem may have an empty set of necessarily optimal solutions. By eliciting more preferences, this set eventually becomes non-empty. Several elicitation strategies are considered in (Gelain et al., 2010a) in the attempt to elicit as little as possible before finding a necessarily optimal solution.

Local Search. Local search (Hoos and Stutzle, 2004) is one of the fundamental paradigms for solving computationally hard combinatorial problems. Given a problem instance, the basic idea underlying local search is to start from an initial search position in the space of all possible assignments (typically a randomly or heuristically generated assignment, which may be infeasible, sub-optimal or incomplete), and to improve iteratively this assignment by means of minor modifications. At each *search step* we move to a new assignment selected from a *local neighborhood*, chosen via a heuristic evaluation function. This process is iterated until a *termination criterion* is satisfied. The termination criterion is usually the fact that a solution is found or that a predetermined number of steps is reached. To ensure that the search process does not stagnate, most local search methods make use of random moves: at every step, with a certain probability a random move is performed rather than the usual move to the best neighbor. Another way to prevent local search from locking in a local minima is Tabu search (Glover and Laguna, 1997) that uses a short term memory to prevent the search from returning to recently visited assignments for a specified amount of steps.

A local search approach has been defined in (Aglanda et al., 2004) for soft constraints. Here we adapt it to deal with incompleteness.

3 LOCAL SEARCH ON IFCSPs

We will now present our local search algorithm for IFCSPs that interleaves elicitation with search. We basically follow the same algorithm as in (Codognet and Diaz, 2001), except for the following.

To start, we randomly generate an assignment of all the variables. To assess the quality of such an assignment, we compute its preference. However, since some missing preferences may be involved in the chosen assignment, we ask the user to reveal them.

In each step, when a variable is chosen, its local preference is computed by setting all the missing preferences to the preference value 1. To choose the new value for the selected variable, we compute the preferences of the assignments obtained by choosing the other values for this variable. Since some preference values may be missing, in computing the preference of a new assignment we just consider the preferences which are known at the current point. We then choose the value which is associated to the best new assignment. If two values are associated to assignments with the same preference, we choose the one associated to the assignment with the smaller number of incomplete tuples. In this way, we aim at moving to a new assignment which is better than the current one and has the fewest missing preferences.

Since the new assignment, say s', could have incomplete tuples, we ask the user to reveal enough of this data to compute the actual preference of s'. We call ALL the elicitation strategy that elicits all the missing preferences associated to the tuples obtained projecting s' on the constraints, and we call WORST the elicitation strategy that asks the user to reveal only the worst preference among the missing ones, if it is less than the worst known preference. This is enough to compute the actual preference of s' since the preference of an assignment coincides with the worst preference in its constraints.

As in many classical local search algorithms, to avoid stagnation in local minima, we employ tabu search and random moves. Our algorithm has two parameters: p, which is the probability of a random move, and t, which is the tabu tenure. When we have to choose a variable to re-assign, the variable is either randomly chosen with probability p or, with probability (1-p) and we perform the procedure described above. Also, if no improving move is possible, i.e., all new assignments in the neighborhood are worse than or equal to the current one, then the chosen variable is marked as tabu and not used for t steps.

During search, the algorithm maintains the best solution found so far, which is returned when the maximum number of allowed steps is exceeded. We will show later that, even when the returned solution is a necessarily optimal solution, its quality is not very far from that of the necessarily optimal solutions.

4 EXPERIMENTAL RESULTS

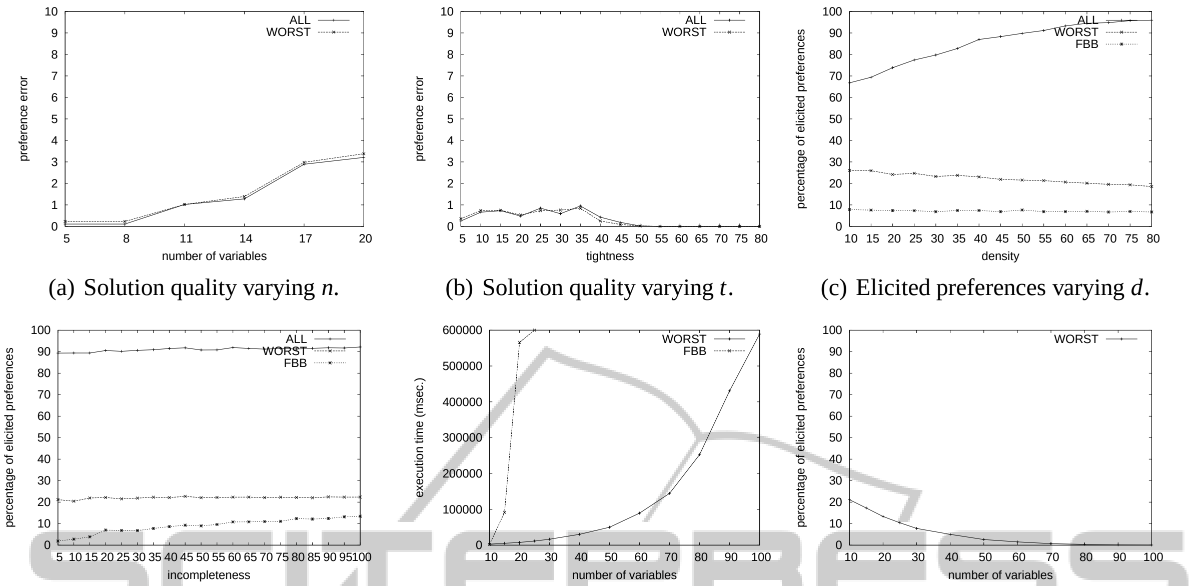
The test sets for IFCSPs are created using a generator that has the following parameters: n: number of variables; m: cardinality of the variable domains; d: density, i.e., the percentage of binary constraints present in the problem w.r.t. the total number of possible binary constraints that can be defined on n variables; t: tightness, i.e., the percentage of tuples with preference 0 in each constraint and in each domain w.r.t. the total number of tuples; i: incompleteness, i.e., the percentage of incomplete tuples (i.e., tuples with preference ?) in each constraint and in each domain. Our experiments measure the percentage of elicited preferences (over all the missing preferences), the solution quality (as the normalized distance from the quality of necessarily optimal solutions), and the execution time, as the generation parameters vary.

We first considered the quality of the returned solution. To do this, we computed the distance between the preference of the returned solution and that of the necessarily optimal solution returned by algorithm FBB (which stands for fuzzy branch and bound) which is one of the best algorithms in (Gelain et al., 2010a). In (Gelain et al., 2010a), this algorithm corresponds to the one called DPI.WORST.BRANCH. Such a distance is measured as the percentage over the whole range of preference values. For example, if the preference of the solution returned is 0.4 and the one of the solution given by FBB is 0.5, the preference error reported is 10%. A higher error denotes a lower solution quality.

Figures 1(a) and 1(b) show the preference error when the number of variables and tightness vary (please notice that the y-axis ranges from 0% to 10%). We can see that the error is always very small and its maximum value is 3.5% when we consider problems with 20 variables. In most of the other cases, it is below 1.5%. We also can notice that the solution quality is practically the same for both elicitation strategies.

If we look at the percentage of elicited preferences (Figures 1(c) and 1(d)), we can see that the WORST strategy elicits always less preferences than ALL, eliciting only 20% of incomplete preferences in most of the cases. The FBB algorithm elicits about half as many preferences as WORST. Thus, with 10 variables, FBB is better than our local search approach, since it guarantees to find a necessarily optimal solution while eliciting a smaller number of preferences.

We also tested the WORST strategies varying the number of variables from 10 to 100. In Figure 1(f) we show how the elicitation varies up to 100 variables. It is easy to notice that with more than 70 variables the percentage of elicited preferences decreases. This is because the probability of a complete assignment with a 0 preference arises (since density remains the same). Moreover, we can see how the local search algorithms can scale better than the branch and bound approach. In Figure 1(e) the FBB reaches a time limit of 10 min-



(a) Solution quality varying *n*. (b) Solution quality varying *t*. (c) Elicited preferences varying *d*.

(d) Elicited preferences varying *i*. (e) Execution time varying *n*. (f) Elicited preferences varying *n*.

Figure 1: Step limit 100000, tabu steps 1000,random walk probability 0.2, average on 100 problems, default values: *d*=50% (for Figures (a), (b), and (d)), *d*=35% (for Figures (e) and (f)), *i*=30%, *t*=10%, *n*=10, $m = 5$.

utes with just 25 variables, while the WORST algorithm needs the same time to solve instances of size 100.

5 CONCLUSIONS

We developed and tested a local search algorithm to solve incomplete fuzzy CSPs. We tested different elicitation strategies and our best strategies have shown good results compared with the branch and bound solver described in (Gelain et al., 2010a). More precisely, our local search approach shows a very good solution quality when compared with complete algorithms. In addition it shows better scaling properties than such a complete method.

REFERENCES

Aglanda, A., Codognet, P., and Zimmer, L. (2004). An adaptive search for the NSCSPs. In *Proc. CSCLP'04*.

Bistarelli, S., Montanari, U., and Rossi, F. (1997). Semiring-based constraint solving and optimization. *Journal of the ACM*, 44(2):201–236.

Codognet, P. and Diaz, D. (2001). Yet another local search method for constraint solving. In *Proc. SAGA'01*.

Fargier, H., Lang, J., and Schiex, T. (1996). Mixed constraint satisfaction: a framework for decision problems under incomplete knowledge. In *Proceedings of AAAI'96*, volume 1, pages 175–180. AAAI Press.

Gelain, M., Pini, M. S., Rossi, F., and Venable, K. B. (2007). Dealing with incomplete preferences in soft constraint problems. In *Proc. CP'07*, LNCS 4741, pages 286–300. Springer.

Gelain, M., Pini, M. S., Rossi, F., Venable, K. B., and Walsh, T. (2008). Dealing with incomplete preferences in soft constraint problems. In *Proc. CP'08*, LNCS 5202, pages 402–417. Springer.

Gelain, M., Pini, M. S., Rossi, F., Venable, K. B., and Walsh, T. (2010a). Elicitation strategies for soft constraint problems with missing preferences: Properties, algorithms and experimental studies. *AI Journal*, 174(3-4):270–294.

Gelain, M., Pini, M. S., Rossi, F., Venable, K. B., and Walsh., T. (2010b). A local search approach to solve incomplete fuzzy and weighted csps. In *Proc. CP'10 Workshop on Preferences and Soft Constraints*.

Glover, F. and Laguna, M. (1997). *Tabu Search*. Kluwer, Norwell, MA.

Hoos, H. H. and Stutzle, T. (2004). *Stochastic Local Search: Foundations and Applications*. Elsevier - Morgan Kaufmann.

Pini, M. S., Rossi, F., and Venable, K. B. (2010). Soft constraint problems with uncontrollable variables. *Journal of Experimental & Theoretical Artificial Intelligence*.

Rossi, F., Beek, P. V., and Walsh, T. (2006). *Handbook of Constraint Programming*. Elsevier.