Fuzzy Modeling of Priority and Preference in Constraint Satisfaction Problems

D. Dubois[1], H. Fargier[1] and H. Prade[1]

In classical Constraint Satisfaction Problems (CSPs), hard constraints restrict the possible values of a set of variables. However, real world constraint problems are often flexible, and classical CSPs are idealizations that do not account for the preference among feasible solutions. Moreover, some constraints may have priority over others. This paper advocates the use of fuzzy set and possibility theory as a realistic approach for the representation of these two aspects: preference relations among possible instantiations and priorities among constraints. In a Fuzzy Constraint Satisfaction Problem (FCSP), a constraint is satisfied to a degree (rather than satisfied or not satisfied) and the acceptability of a potential solution becomes a gradual notion. Even if the FCSP is partially inconsistent, best instantiations are provided owing to the relaxation of some constraints.

INTRODUCTION

Classical Constraint Satisfaction Problems (CSPs) only consider a set of hard constraints that every solution must satisfy. This rigid representation framework has several drawbacks. First, some problems are over-constrained and have no solutions. A relaxation of the less rigid or important constraints must be performed in order to obtain a solution. Discovering that a problem has no solution may be time-consuming and devising an efficient constraint relaxation method is far from easy. Alternatively, other problems lead to a large set of equally possible solutions, although there often exist preferences among them which remain unexpressed. However, a standard CSP procedure will pick a solution at random. As a matter of fact, in practice, constraints are not always strict and it is desirable to extend the CSP framework in order to accommodate flexible constraints. Devising a framework for representing the flexibility of constraints will avoid artificially unfeasible problems (constraints being self-relaxable) and will avoid the random choice of solutions to loosely constrained problems. By flexible constraints, it is meant either soft constraints, which directly express preferences among solutions (i.e. this is a ranking of instantiations which are more or less acceptable for the satisfaction of a soft constraint), or prioritized constraints that can be violated if they conflict with more prioritary constraints.

In soft constraints, the flexibility accounts for the possibility of going away from instantiations that satisfy the constraints ideally. Notice that the interest in soft constraints can be traced back to the early CSP literature. In 1975, Waltz [1] heuristically distinguished

1. Institut de Recherche en Informatique de Toulouse (I.R.I.T.)- CNRS, Université Paul Sabatier, France.

Scientia Iranica, Vol. 2, No. 4, © Sharif University of Technology, January 1996.

between "likely" instantiations of a constraint and "unlikely" ones which are only considered if necessary. Also in computer vision, in 1976 Rosenfeld et al. [2] modelled preference in the detection of convex objects for scene labeling problems and proposed the use of a fuzzy degree of constraint satisfaction. The idea of representing relative preferences by means of weights is also at work in the relaxation labeling process described in [3]. More recent works either propose the use of fuzzy sets in modeling such constraints [4-8], or progressively relax the constraints when preferences are in conflict [9].

In prioritized constraints the flexibility lies in the ability to discard constraints involved in inconsistencies, provided that they are not too important. Generally, a weight is associated with each constraint and the request is to minimize the greatest priority levels of the violated constraints [10,11]. More generally, Brewka et al. [12] and Borning et al. [13] identify different forms of constraint relaxation, viewing each constraint as a strict partial order on value assignment and weighing the importance of constraints. In particular, Brewka et al. provide a formal semantics in relation to nonmonotonic reasoning by means of maximal-consistent subsets of constraints.

Freuder [14], Freuder and Wallace [15] as well as Satoh [16] have devised theoretical foundations for the treatment of flexibility in CSPs. Satoh tries to apply results in nonmonotonic reasoning based on circumscription to the handling of prioritized constraints so as to induce preference relations on the solution set. A similar point of view is adopted by Lang [17] where prioritized constraints are expressed in possibilistic logic (i.e. a logic with weighted formulas which has a nonmonotonic behavior in case of partial inconsistency). Taking a dual point of view, Freuder [14] regards a flexible problem as a collection of classical CSPs. A metric can then be defined that evaluates the distance between them. Subsequently, the question is how to "find the solutions to the closest solvable problems".

In order to take into account both types of flexibility, a generalization of the CSP framework has been proposed, the Fuzzy Constraint Satisfaction Problem framework (FCSP) [18], based on Zadeh possibility theory [19](see [20] for an introduction). The main point is that both types of flexible constraints are regarded as local criteria that rank-order (partial) instantiations and can be represented by means of fuzzy relations. In a FCSP, constraint satisfaction or violation are no longer an all-or-nothing notion. An instantiation is compatible with a flexible constraint to a degree (belonging to some totally ordered scale). The notion of consistency of a FCSP also becomes a matter of degree. The question is then to combine the satisfaction degrees of the fuzzy constraints in order to determine the total ordering induced over the potential solutions and to choose the best ones. Moving a step further, the use of this framework is proposed also for handling more complex constraints, e.g. constraints with safeguard.

From an algorithmic point of view, the possibility of extending Waltz' algorithm to fuzzy constraints has been pointed out by Dubois and Prade [21] and Yager [22]. As shown in [23,24], all the classical CSP algorithms (e.g. tree search, AC3, PC2) can be easily adapted to FCSPs. More generally, our framework reveals itself powerful enough to accommodate the definitions of local consistency of a problem (arc-consistency, 3-consistency, k-consistency). Interestingly enough, investigations by the second author [23] indicate that the theoretical results relating levels of local consistency of a CSP to its global consistency [25,26] remain valid in FCSPs.

The next section deals with representation issues concerning flexible constraints. Fuzzy subsets on Cartesian products of domains, i.e. fuzzy relations, are used to model soft and/or prioritized constraints. An illustrative example is provided. The agreement of this representation with the preferential semantics of possibility theory is emphasized. Then, the extension (resp. projection) of fuzzy constraints to larger (resp. smaller) Cartesian products of domains is recalled as well as the conjunctive or disjunctive combinations of fuzzy relations for representing compound constraints. Finally, this section discusses the modeling of more sophisticated constraints, namely, prioritized constraints with safeguard (in order to guarantee the satisfaction of a weaker constraint in case of violation of the prioritized one). The third section formally defines the FCSP framework and compares it to other approaches related to flexibility in CSP. This section then presents the essentials of a Branch and Bound algorithm performing the search for the best solutions. Nonmonotonic aspects of FCSPs are also outlined.

REPRESENTING FLEXIBLE CONSTRAINTS

A hard constraint C relating a set of decision variables $\{x_1, \dots, x_n\}$ ranging in respective domains $D_1, \dots, D_n$ is classically described by an associated relation R. R is the crisp subset of $D_1 \times \cdots \times D_n$ that specifies the tuples $d = (d_1, \dots, d_n)$ of values which are compatible with C. The set $\{x_1, \dots, x_n\}$ of variables related by R will be denoted by $V(R)$.

Fuzzy Model of a Soft Constraint

A soft constraint C will be described by means of an associated fuzzy relation R [27], i.e. the fuzzy subset of $D_1 \times \cdots \times D_n$ of values that more or less satisfy C. R is defined by a membership function μ_R which associates a level of satisfaction $\mu_R(d_1, \dots, d_n)$ in a totally ordered set L (with top denoted 1 and bottom denoted 0) to each tuple $(d_1, \dots, d_n) \in D = D_1 \times \cdots \times D_n$. This membership grade indicates to what extent $d = (d_1, \dots, d_n)$ is compatible with (or satisfies) C. Thus, the notion of constraint satisfaction becomes a matter of degree:

$\mu_R(d_1, \dots, d_n) = 1$
means $(d_1, \dots, d_n)$ totally satisfies C,

$\mu_R(d_1, \dots, d_n) = 0$
means $(d_1, \dots, d_n)$ totally violates C,

$0 < \mu_R(d_1, \dots, d_n) < 1$
means $(d_1, \dots, d_n)$ partially satisfies C .

Hard constraints are particular cases of soft constraints, since they only involve levels 0 and 1. A soft constraint involving preferences between values is regarded as a local criterion, ordering the instantiations of C, preference levels being represented in the scale L. $\mu_R(d_1, \dots, d_n) > \mu_R(d'_1, \dots, d'_n)$ means that the first instantiation is preferred to the second one. Interpreting the preference degrees as membership degrees leads to representation of a soft constraint by a fuzzy relation.

The assumption of a totally ordered satisfaction scale underlying the above setting may be questioned. The very use of a satisfaction scale instead of just an ordering relation is crucial when it comes to the aggregation of local satisfaction levels. Indeed due to the famous Arrow theorem (e.g. [28]), it is very difficult to merge several ordering relations that are not commensurate. The satisfaction scale need not be totally ordered, strictly speaking, since a complete lattice will do as well. In the following, it is assumed that L is a totally ordered set, i.e. a chain. However, the scale of membership need not be numerical, as pointed out years ago [29]. A qualitative scale makes sense on finite domains. However, on continuous domains, as in the case of temporal constraints with continuous time, it is much more natural and simpler to assume that the satisfaction scale is the unit interval; then, levels of satisfaction reflect distances to ideal values in the domain.

Fuzzy Model of a Prioritized Constraint

Fuzzy relations also offer a suitable formalism for the expression of prioritized constraints. When it is possible to, a priori, exhibit a total preorder over the respective priorities of the constraints, these priorities will be represented by levels in another scale V. A priority degree $Pr(C)$ is attached to each constraint C and indicates to what extent it is imperative that C be satisfied. First, consider the case of hard constraints. $Pr(C) = 1$ means that C is an absolutely imperative constraint, while $Pr(C) = 0$ indicates that it is completely possible to violate C (C has no incidence in the problem). Given two constraints C and C',

$Pr(C) > Pr(C')$ means that the satisfaction of C is more necessary than the satisfaction of C'. If C and C' cannot be satisfied simultaneously, solutions compatible with C will be preferred to solutions compatible with C'.

In fact, the scale V can be interpreted as a "violation scale": the greater $Pr(C)$, the less it is possible to violate C. This remark leads us to relate the satisfaction scale L to the violation scale V, considering that there exists an order-reversing bijection from V to L such that $L = c(V) = \{c(v), v \in V\}$. $c(0)$ and $c(1)$ are respectively the top element and the bottom element of L, and $v \geq v'$ in V implies $c(v) \leq c(v')$ in L. This is one of the basic modeling assumptions in this paper. The c-complement of the level of priority of a constraint is interpreted as the extent to which the constraint can be violated, using the reversed scale $L = c(V)$ as a satisfaction scale ; L is nothing but V put upside down. Since $Pr(C)$ represents to what extent it is necessary to satisfy C, $c(Pr(C))$ indicates to what extent it is possible to violate C, i.e. to satisfy its negation. In other words, the constraint C is considered as satisfied at least to degree $c(Pr(C))$ whatever the considered solution, whether it satisfies C or not. More precisely, the prioritized constraint $(C, Pr(C))$ is considered as totally satisfied by a tuple if C is satisfied, and satisfied to degree $c(Pr(C))$ if the tuple violates C. Hence, $c(V)$ can be identified to a satisfaction scale as in the previous section and a prioritized constraint C may be represented by the fuzzy relation (see Figure 1):

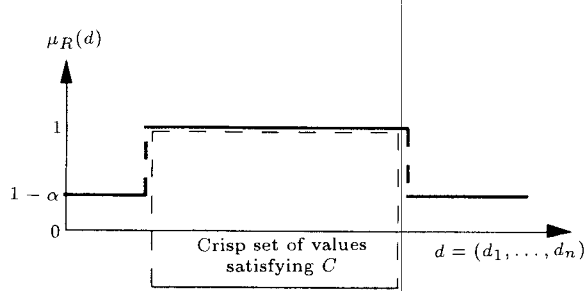

Figure 1. A hard (or crisp) constraint C with priority $Pr(C) = \alpha$ when $c(x) = 1 - x$.

$$\mu_R(d_1, \ldots, d_n) = c(0) = 1 \quad \text{if } (d_1, \ldots, d_n) \text{ satisfies } C \ ;$$
$$\mu_R(d_1, \ldots, d_n) = c(Pr(C)) \quad \text{if } (d_1, \ldots, d_n) \text{ violates } C \ .$$

Note that when $Pr(C) = 1$, the characteristic function of C is recovered, while when $Pr(C) = 0$ the constraint C degenerates into the whole domain D.

Conversely, a soft constraint C, where preferences are described in terms of a finite number of satisfaction degrees $0 = \alpha_0 < \alpha_1 < \cdots < \alpha_p < 1$ in a scale L, can be represented by a finite set of prioritized constraints $\{C_j, 0 \leq j < p\}$ using the scale L put upside down as a priority scale, via an order-reversing map c:

$$Pr(C_j) = c(\alpha_j) \text{ defining } R_j = \{(d_1, \ldots, d_n), \mu_R(d_1, \ldots, d_n) \geq \alpha_{j+1}\}, j = 0, p-1 \ .$$

If, moreover, it is assumed that c is involutive, that is $c(c(\alpha)) = \alpha$ (this hypothesis is made throughout the whole paper), then it is straightforward to reconstruct the soft constraint C by means of the set of prioritized constraints $\{(C_j, Pr(C_j)), 0 \leq j < p\}$, as shown in Figure 2, where:

$$\mu_R(d) = \min_j \max(c(Pr(C_j)), \mu_{R_j}(d)) \quad \text{for every tuple } d = (d_1, \ldots, d_n) \ . \qquad (1)$$

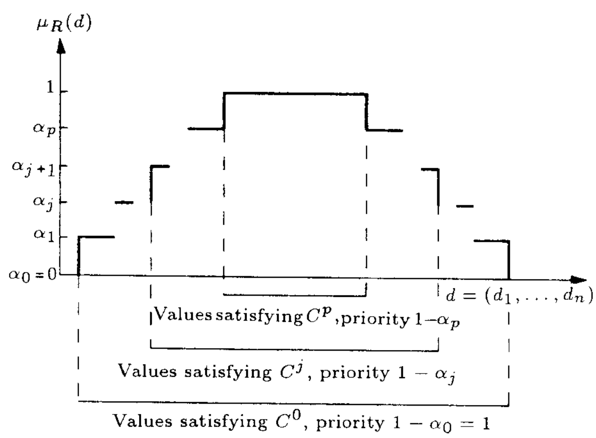

Figure 2. Decomposition of a soft constraint into a family of prioritized constraints when $c(x) = 1 - x$.

Finally, a prioritized soft constraint C corresponds to the following fuzzy relation:

$$\mu_{R'}(d) = \max(c(Pr(C)), \mu_R(d)) ,$$

where R is the fuzzy relation describing the preferences of C only. Viewing the soft constraint expressed by R as a family of nested prioritized constraints, the global priority $Pr(C)$ attached to the soft constraint C means that the priorities higher than $Pr(C)$ are neglected in the expression of R since:

$$\max(c(Pr(C)), \mu_R(d)) =$$

$$\max(c(Pr(C)), \min_j \max(c(Pr(C_j)), \mu_{R_j}(d))) =$$

$$\min_j \max(c(\min(Pr(C), Pr(Cj))), \mu_{R_j}(d)) .$$

To conclude with representation issues, prioritized and soft constraints can be cast in a unique setting that is called "flexible constraints", modeled by fuzzy sets, where flexibility means the capability of self-relaxation. This capability is locally imbedded in the description of the constraint, thus avoiding the necessity of a specific constraint relaxation procedure to be triggered when a set of constraints is found inconsistent. This unification presupposes a strong link between levels of constraint satisfaction and levels of constraint priority, using a single ordered scale L for both priority and satisfaction and an order-reversing map c that changes one notion into the other. For simplicity, $L = [0,1]$ and $c(x) = 1 - x$ are sometimes used in the following. However, all results to be presented remain valid on a qualitative scale.

Example

A course must involve 7 sessions, namely x lectures, y exercise sessions and z training sessions (C_1). There must be about 2 training sessions (C_2), i.e. ideally 2, possibly 1 or 3. Dr. B, who gives the exercise part of the course, wants to manage 3 or 4 sessions (C_3). Prof. A, who gives the lectures, wants to give about 4 lectures (C_4), i.e. ideally 4 lectures, possibly 3 or 5. The request of Dr. B is less important than the one of Prof. A and is itself less important than the imperative constraints C_1 and C_2. In this example, flexibility is modeled using a five level scale $L = (\alpha_0 = 0 < \alpha_1 = c(\alpha_3) < \alpha_2 = c(\alpha_2) < \alpha_3 = c(\alpha_1) < \alpha_4 = 1)$, where c is the order-reversing operation. The priorities of C_3 and C_4 are respectively α_2 and $\alpha_3 (\alpha_2 < \alpha_3)$. The domain of variables x, y and z is the set $\{0, 1, 2, 3, 4, 5, 6, 7\}$. The following model can be used:

C_1 : classical hard constraint:

$$\mu_{R_1}(x, y, z) = 1 \text{ if } x + y + z = 7 ;$$

$$\mu_{R_1}(x, y, z) = 0 \text{ otherwise.}$$

C_2 : soft constraint (see Figure 3a):

$$\mu_{R_2}(z) = 1 \text{ if } z = 2 ;$$

$$\mu_{R_2}(x) = \alpha_3 \text{ if } z = 1 \text{ or } z = 3 ;$$

$$\mu_{R_2}(z) = 0 \text{ otherwise .}$$

C_3 : prioritized constraint $Pr(C_3) = \alpha_2$ (see Figure 3b):

$$\mu_{R_3}(y) = 1 \text{ if } y = 3 \text{ or } y = 4 ;$$

$$\mu_{R_3}(y) = c(\alpha_2) = \alpha_2 \text{ otherwise .}$$

C_4 : soft and prioritized constraint $Pr(C_4) = \alpha_3$ (see Figure 4):

$$\mu_{R_4}(x) = 1 \text{ if } x = 4 ;$$

$$\mu_{R_4}(x) = \max(\alpha_3, c(\alpha_3)) = \alpha_3$$
$$\text{if } x = 3 \text{ or } x = 5 ;$$

$$\mu_{R_4}(x) = c(\alpha_3) = \alpha_1 \text{ otherwise .}$$

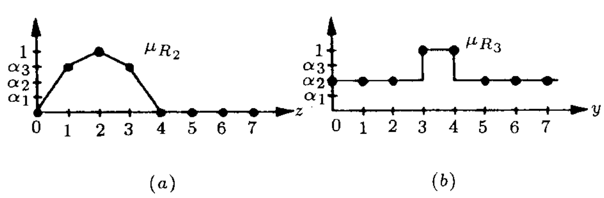

Figure 3. Modeling of C_2(a) and C_3(b) by means of fuzzy unary restriction.

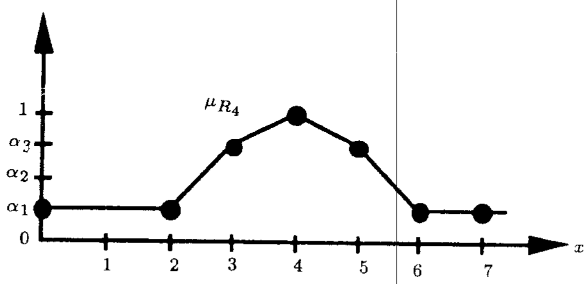

Figure 4. Modeling of C_4 by means of a fuzzy unary restriction.

Operations on Fuzzy Relations

Flexible constraints are modeled by qualitative fuzzy relations. The usual operations on crisp relations can be easily generalized to fuzzy relations [26]. To do so, the fact that, being totally ordered, the satisfaction scale L is a complete distributive lattice, where the minimum and the maximum of two elements make sense is exploited. The following definitions extend classical set-theoretic notions used in constraint-directed problem-solving:

- A fuzzy relation R' is said to be included into R if and only if (see Figure 5):

$$\forall(d_1,\ldots,d_n) \in D_1 \times \cdots \times D_n,$$
$$\mu_{R'}(d_1,\ldots,d_n) \le \mu_R(d_1,\ldots,d_n).$$

  This definition is a generalization of the classical set inclusion. In terms of constraints, C' is tighter than C and C is a relaxation (or a weakening) of C'.

- The projection of a fuzzy relation R on $\{x_{k_1},\ldots,x_{k_{nk}}\} \subseteq V(R)$ is a fuzzy relation $R^{\downarrow\{x_{k_1},\ldots,x_{k_{nk}}\}}$ on $\{x_{k_1},\ldots,x_{k_{nk}}\}$ such that:

$$\mu_{R^{\downarrow\{x_{k_1},\ldots,x_{k_{nk}}\}}}(d_{k_1},\ldots,d_{k_{nk}}) = \sup_{\{d/d^{\downarrow\{x_{k_1},\ldots,x_{k_{nk}}\}}=(d_{k_1},\ldots,d_{k_{nk}})\}} \mu_R(d)\ ,$$

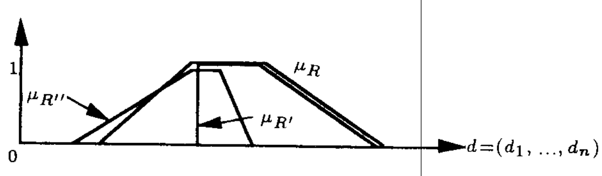

Figure 5. $R' \subseteq R$ and $R'' \not\subseteq R$.

  where $d^{\downarrow\{x_{k_1},\ldots,x_{k_{nk}}\}}$ denotes the classical restriction of $d = (d_1,\ldots,d_n)$ to $\{x_{k_1},\ldots,x_{k_{nk}}\}$. This definition is a generalization of the projection of ordinary relations. $\mu_{R^{\downarrow\{x_{k_1},\ldots,x_{k_{nk}}\}}}(d_{k_1},\ldots,d_{k_{nk}})$ estimates to what level of satisfaction the instantiation $(d_{k_1},\ldots,d_{k_{nk}})$ can be extended to an instantiation that satisfies C.

- The cylindrical extension of a fuzzy relation R to $\{x_{k_1},\ldots,x_{k_{nk}}\} \supseteq V(R)$ is a fuzzy relation $R^{\uparrow\{x_{k_1},\ldots,x_{k_{nk}}\}}$ on $\{x_{k_1},\ldots,x_{k_{nk}}\}$ such that:

$$\mu_{R^{\uparrow\{x_{k_1},\ldots,x_{k_{nk}}\}}}(d_{k_1},\ldots,d_{k_{nk}}) = \mu_R((d_{k_1},\ldots,d_{k_{nk}})^{\downarrow V(R)})\ .$$

  This definition is a generalization of the cylindrical extension of ordinary relations. $\mu_{R^{\uparrow\{x_{k_1},\ldots,x_{k_{nk}}\}}}(d_{k_1},\ldots,d_{k_{nk}})$ estimates to what extent the instantiation $(d_{k_1},\ldots,d_{k_{nk}})$ satisfies C.

- The conjunctive combination (or join) of two fuzzy relations R_i and R_j is a fuzzy relation $R_i \otimes R_j$ over $V(R_i) \cup V(R_j) = \{x_1,\ldots,x_k\}$ such that (see Figure 6):

$$\mu_{R_i\otimes R_j}(d_1,\ldots,d_k) = \min(\mu_{R_i}((d_1,\ldots,d_k)^{\downarrow V(R_i)}), \mu_{R_j}((d_1,\ldots,d_k)^{\downarrow V(R_j)}))\ .$$

  $\mu_{R_i\otimes R_j}(d_1,\ldots,d_k)$ estimates to what extent $(d_1,\ldots,d_k)$ satisfies both C_i and C_j. When $V(R_i) = V(R_j), \otimes$ is a generalization of classical set intersection. All properties of the standard intersection (associativity, commutativity, etc.) hold as long as negation is not involved; in particular, there holds $(R_i \otimes R_j)^{\downarrow V(R_i)} \subseteq R_i$ and $(R_i \otimes R_i) = R_i$.

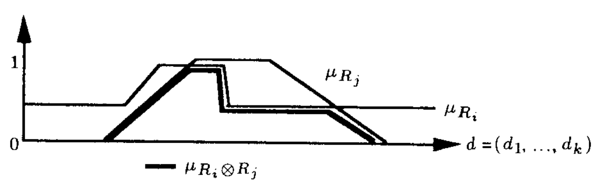

Figure 6. Conjunctive combination of two fuzzy relations R_i and R_j .

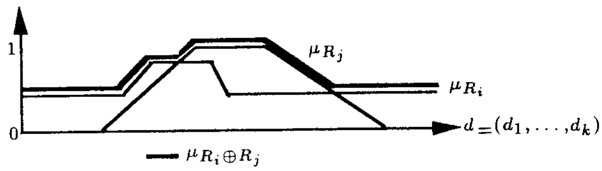

Figure 7. Disjunctive combination of two fuzzy relations R_i and R_j .

Note that the use of the combination rule, allowed by the presence of a unique satisfaction scale, underlies an assumption of commensurability between satisfaction levels pertaining to different constraints. The user who specifies the constraints must describe them by means of this unique scale L, or by means of the dual scale L^T. For instance, in the example given previously, the satisfaction level α_3 of C_4 for $x \in \{3,5\}$ is assumed to be equal to the satisfaction level for $z \in \{1,3\}$ and $\alpha_1 < c(Pr(C_3)) < c(Pr(C_4))$. Although natural and often implicit, this assumption must be emphasized.

- The disjunctive combination of two fuzzy relations R_i and R_j is a fuzzy relation $R_i \oplus R_j$ over $V(R_i) \cup V(R_j) = \{x_1, \dots, x_k\}$ such that (see Figure 7):

$$\mu_{R_i \otimes R_j}(d_1, \dots, d_k) = \max(\mu_{R_i}((d_1, \dots, d_k)^{\downarrow V(R_i)}), \mu_{R_j}((d_1, \dots, d_k)^{\downarrow V(R_j)})) .$$

$\mu_{R_i \oplus R_j}(d_1, \dots, d_k)$ estimates to what extent $(d_1, \dots, d_k)$ satisfies either C_i or C_j. When $V(R_i) = V(R_j)$, $\oplus$ is a generalization of classical set union. All properties of set union (associativity, commutativity, distributivity over intersection, etc.) hold, if negation is not involved.

Prioritized Constraints with Safeguard

The framework of fuzzy constraints offers a convenient tool for representing more sophisticated constraints than the previously encountered ones, for instance prioritized constraints with safeguard as well as nested conditional constraints [24]. Indeed, one may like to express that a constraint C, even with a rather low priority $Pr(C) = \alpha$, can never be completely violated, in the sense that if C is violated, at least a more permissive, minimal, constraint C' is still satisfied. Let R and R' be the fuzzy relations associated with C and C', respectively, with $R \subseteq R'$ (C' is more permissive than C, i.e. C' is a relaxation of C). The whole constraint C^* corresponding to the pair (C, C') can be viewed as the conjunction of a prioritized constraint (C) and a weaker but imperative, possibly soft, constraint (C'). This conjunction is represented by the fuzzy relation R^*, pictured in Figure 8, and expressed by:

$$\forall d \in D_1 \times \cdots \times D_n,$$
$$\mu_{R^*}(d) = \min(\max(\mu_R(d), c(\alpha)), \mu_{R'}(d)) .$$

This is a particular case of the decomposition of a soft constraint into prioritized ones when C and C' are hard. Indeed, such constraints express both a requirement with priority α less than 1 and a weaker requirement with priority 1 and R^* is of the form of Statement 1:

$$\mu_{R^*}(d) = \min(\max(\mu_R(d), c(\alpha)), \max(\mu_{R'}(d), c(Pr(C')))) \text{with} Pr(C') = 1 .$$

See [30] for the use of such constraints in fuzzy database querying systems. Interestingly enough, R^* can be decomposed either as a disjunction or as a conjunction of two fuzzy relations, depending on which fuzzy relation, R or R', the priority weight is combined with. Indeed:

$$\begin{aligned}\mu_{R^*}(d) &= \min(\max(\mu_R(d), c(\alpha)), \mu_{R'}(d)) \\ &= \min(\max(\mu_R(d), c(\alpha)), \max(\mu_R(d), \mu_{R'}(d))) \text{ since } R \subseteq R' \\ &= \max(\mu_R(d), \min(c(\alpha), \mu_{R'}(d))) .\end{aligned}$$

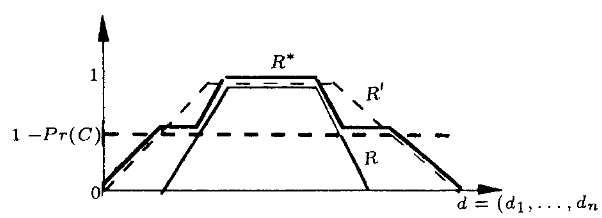

Figure 8. Representation of a prioritized fuzzy constraint with safeguard.

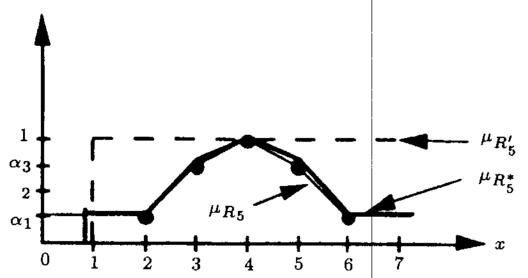

Figure 9. Modeling of C_5 by fuzzy constraint with safeguard constraint.

It expresses that satisfying a constraint with safeguard corresponds to either satisfying its stronger form C or its weaker form C', the satisfaction degree being upper-bounded in this second case by $c(\alpha)$.

For instance, a flexible C_5 constraint prescribing: "Prof. A wants to give about four lectures; anyway, he will never accept giving no lecture" is represented by the fuzzy relation R_5^* pictured in Figure 9.

STATING AND SOLVING FUZZY CONSTRAINT SATISFACTION PROBLEMS

Definition

A Fuzzy Constraint Satisfaction Problem (FCSP) P involves a set of n decision variables $X = \{x_1, \dots, x_n\}$ each ranging on its respective domain $D_1, \dots, D_n$ and a set of m fuzzy relations $R = \{R_1, \dots, R_m\}$ representing a set $C = \{C_1, \dots, C_m\}$ of hard, soft or prioritized constraints (domains are assumed to be discrete in the following). A unary relation R_j of R is supposed to be associated to each variable x_j. It represents the values which are, a priori, more or less feasible (i.e. here, preferred) for x_j (by default, $R_j = D_j$). If all the constraints are unary or binary, the FCSP is called a fuzzy constraint network.

Classically, an instantiation of $\{x_{k_1}, \dots, x_{k_{nk}}\} \subseteq X$ is locally consistent if it satisfies all the constraints in the subnetwork restricted to $\{x_{k_1}, \dots, x_{k_{nk}}\}$. Like constraint satisfaction, the notion of consistency is now a matter of degree. The definition of the conjunctive combination states that $\mu_{[R_{i_1} \otimes \cdots \otimes R_{i_p}]}(d_{k_1}, \dots, d_{k_{nk}})$ estimates to what extent $(d_{k_1}, \dots, d_{k_{nk}})$ satisfies all the constraints $C_{i_1}, \dots, C_{i_p}$. Hence, the degree of local consistency of $(d_{k_1}, \dots, d_{k_{nk}})$ is defined by:

$$\begin{aligned}\mathrm{Cons}(d_{k_1}, \dots, d_{k_{nk}}) &= \mu_{[\otimes_{\{R_i \in R / V(R_i) \subseteq \{x_{k_1}, \dots, x_{k_{nk}}\}\}} R_i]}(d_{k_1}, \dots, d_{k_{nk}}) \\ &= \min_{\{R_i \in R / V(R_i) \subseteq \{x_{k_1}, \dots, x_{k_{nk}}\}\}} \\ &\quad (\mu_{R_i}((d_{k_1}, \dots, d_{k_{nk}})^{\downarrow V(R_i)})) \,.\end{aligned}$$

It should be noticed that: $\forall Y \subseteq \{x_{k_1}, \dots, x_{k_{nk}}\}$,

$$\mathrm{Cons}((d_{k_1}, \dots, d_{k_{nk}})^{\downarrow Y}) \geq \mathrm{Cons}(d_{k_1}, \dots, d_{k_{nk}}).$$

Considering a complete instantiation of X, $\mu_{[R_1 \otimes \cdots \otimes R_m]}(d_1, \dots, d_n)$ is the satisfaction degree of all the constraints by $(d_1, \dots, d_n)$, i.e. the satisfaction degree of problem P by $(d_1, \dots, d_n)$. It is the membership degree of $(d_1, \dots, d_n)$ to the fuzzy set $\rho = R_1 \otimes \cdots \otimes R_m$ which is nothing but the (fuzzy) set of solutions of P. As for classical CSPs, solutions are consistent instantiations of X. $\mathrm{Cons}(d_1, \dots, d_n) = \mu_\rho(d_1, \dots, d_n) > 0$, i.e. solutions that are not totally unfeasible.

These degrees discriminate among the potential solutions since they induce a total preorder over the instantiations; this preorder does not depend on whether L is a numerical scale or not. In other terms, the FCSP approach to flexiblity is more qualitative than quantitative. Actually, solving a classical CSP means separating the set of all instantiations into two classes: the instantiations which are solutions to the problem and those which are not. Introducing flexibility just refines this order.

It should be noticed that the best instantiations of X may get a satisfaction degree lower than 1 if some constraints are conflicting. The FCSP approach can handle partially inconsistent problems. The consistency degree of the FCSP is the satisfaction degree of the best instantiations:

$$\text{Cons}(P) = \sup_{\{(d_1,\ldots,d_n)\in D_1\times\cdots\times D_n\}} \mu_\rho(d_1,\cdots,d_n) = \sup_{\{(d_1,\ldots,d_n)\in D\}}[\min_{\{R_i\in R\}} \mu_{R_i}((d_1,\cdots,d_n)^{\downarrow V(R_i)})],$$

where $D = D_1 \times \cdots \times D_n$. The best solutions of P are those which satisfy the global problem to the maximal degree $\mu_{R_1\otimes\cdots\otimes R_m}(d_1,\cdots,d_n) = \mu_\rho(d_1,\cdots,d_n)(= \text{Cons}(P))$, i.e. those which maximize the satisfaction level of the least satisfied constraint. If there are some instantiations which perfectly satisfy all the constraints ($\text{Cons}(P) = 1$), they are the best solutions. Otherwise, an implicit relaxation of flexible constraints is performed, achieving a trade-off between antagonistic constraints [10]. A solution will be found as long as the problem is not totally inconsistent.

The example which was given before is partially inconsistent. The best solution and the consistency degree are respectively ($x = 3, y = 3, z = 1$) and $\alpha_3 < 1$. The constraint over the number of training sessions and Prof. A's constraint are slightly relaxed according to their flexibility. The other potential solutions (e.g. ($x = 4, y = 1, z = 2$) or ($x = 2, y = 3, z = 2$)) are less consistent (their respective satisfaction degrees are α_2 and α_1).

Discussion

The FCSP approach is in accordance with Freuder's view of constraint relaxation by partial satisfaction [14]. Indeed, an FCSP involving p different satisfaction levels is equivalent to p CSPs. For each level $\alpha_j > 0, \alpha_j \in L$, a CSP P^{α_j} is formed by the set of hard constraints $C_i^{\alpha_j}$ containing the tuples that satisfy C_i to a degree greater than or equal to α_j. Considering that a weight is associated to each possible relaxation of each constraint C_i, the metric associated to this space is defined by the maximum among the weights of the relaxations performed. The set of best solutions to the flexible problem is the set of solutions of the consistent P^{α_j} of highest α_j (the closest solvable problem using Freuder's terminology).

The FCSP approach is different from probabilistic or cost-based approaches, where the best solutions are those satisfying the maximal number of constraints [15], or those for which the sum of satisfaction degrees is maximal [31]. These additive approaches allow for the violation of a constraint to be counterbalanced by the satisfaction of other constraints. The word "constraint" is then hardly justified. In an FCSP, when an instantiation violates a hard constraint, it becomes totally inconsistent: $\mu_{R_1\otimes\cdots\otimes R_m}(d_1,\cdots,d_n) = 0$. Thus, being in accordance with the principle of constraint satisfaction, no constraint can be violated—except according to its relaxation capacities, which are expressed by the FCSP formalism. Additive satisfaction pooling methods also presuppose that constraints are independent or, at least, not redundant. This ideal is difficult to achieve and appears contradictory with the purpose of constraint propagation, which is to produce redundant constraints. Note that the two methods of aggregation of satisfaction levels correspond to the two basic approaches to the definition of social welfare in utility theory(e.g. [28]): utilitarianism, which maximizes the sum of the individual utilities, and egalitarianism which maximizes the minimal individual utility. Only the latter is compatible with the usual treatment of constraints.

Although in accordance with the approach described here, Satoh's approach [16] differs in the way priorities between constraints are expressed. Indeed, Satoh uses second-order logic to describe priorities. Moreover, the ordering of solutions depends on how many constraints are satisfied. In our approach, solutions which satisfy an FCSP to the same degree are not discriminated, even if some of them satisfy more constraints. In other terms, the best solutions in the sense of Satoh are among the best according to the FCSP definition. However, a so-called lexicographic ordering may be used in FCSP, if needed, to discriminate solutions sharing the same global satisfaction degree, as for instance in [10,7]. This mode of aggregation is also known in the social welfare literature under the name "leximin aggregation" (see [28]). The definition of the leximin ordering of two vectors $v_1 = (\mu_1,\ldots,\mu_n)$ and $v_2 = (\lambda_1,\ldots,\lambda_n)$ in L^n is as follows:

1. Rearrange the vectors in increasing order, say $\mu_{i_1} \leq \mu_{i_2} \leq \cdots \leq \mu_{i_n}$ and $\lambda_{j_1} \leq \lambda_{j_2} \leq \cdots \leq \lambda_{j_n}$.
2. Perform a lexicographical comparison starting from the first component, i.e.:

$$v_2 > v_1 \Leftrightarrow \exists k \leq n \text{ such that } \forall\, m < k$$

$$\lambda_{j_m} = \mu_{i_m} \text{ and } \lambda_{j_k} > \mu_{i_k}.$$

In the example, the instantiation ($z = 2, y = 1, x = 4$) which satisfies C_1, C_2, C_3 and C_4 to degrees $(1, 1, \alpha_2, 1)$ is considered in an FCSP as equally good as ($z = 1, y = 1, x = 5$) which satisfies the constraints to degrees $(1, \alpha_3, \alpha_2, \alpha_3)$. The lexicographic ordering, which is a refinement of the min-induced ordering, will prefer the first instantiation to the second one. Note that if $L = \{0, 1\}$, i.e. if the FCSP is a classical CSP, the solutions which are the best according to the lexicographic ordering are those satisfying the maximal number of constraints, as in Freuder's view of partial constraint satisfaction [15]. In other terms, the lexicographic ordering in an FCSP, which is more precisely studied in [32,33], is a generalization of Freuder's ordering in a classical CSP.

As a general model based on possibility theory, the FCSP approach generalizes the frameworks that model softness by means of fuzzy sets [4,5,7,8] as well as those dealing with constraint priorities by searching to minimize the priority of the violated constraints [10,7,13,11]. More precisely, some of them use an inclusion-based refinement of the min-induced ordering [13], or a lexicographic refinement [10,7], which is itself a refinement of the inclusion-based ordering. See [32,33] for a discussion on the selection of preferred solutions in FCSP by means of these three criteria.

A Generic Solving Method for FCSPs

Finding a solution to a classical CSP is an NP-complete task. Hence, finding the best solution of FCSP is at least NP-hard. In fact, it reduces to a sup/min optimization formulation:

$$\sup_{\{(d_1,\ldots,d_n)\in D_1\times\cdots\times D_n\}} \left[\min_{\{R_i\in\{R_1,\ldots,R_m\}\}} \times \left(\mu_{R_i}((d_1,\ldots,d_n)^{\downarrow V(R_i)})\right)\right].$$

This kind of problem can be solved using classical Branch and Bound algorithms [14, 17, 11], such as Depth-First Branch and Bound. It is a natural extension of backtracking, the standard approach to CSPs. Using such a classical tree search algorithm, variables are instantiated in a predetermined sequence, say $(x_1, \ldots, x_n)$. The root of the tree is the empty assignment. Intermediate nodes $(d_1, \ldots, d_k)$ denote partial instantiations and leaves are complete instantiations of $(x_1, \ldots, x_n)$. For each leaf $(d_1, \ldots, d_n)$ in the tree, $\mu_\rho(d_1, \ldots, d_n)$ may be computed. The leaves that maximize μ_ρ are searched for via a depth-first exploration of the tree.

The use of fuzzy constraints makes it possible to prune each branch that necessarily leads to suboptimal leaves that can be proved worse than the best of the already evaluated solutions. In other terms, it is useless to extend intermediary nodes $(d_1, \ldots, d_k)$ such that $\mu_{[\rho^{\downarrow\{x_1,\ldots,x_k\}}]}(d_1, \ldots, d_k) \leq b_{inf}$, b_{inf} being a lower bound of Cons (P). The calculation of $\mu_{[\rho^{\downarrow\{x_1,\ldots,x_k\}}]}(d_1, \ldots, d_k)$ requires the extension of $(d_1, \ldots, d_k)$ into a complete instantiation but the definition of local consistency provides an upper bound for it. Indeed:

$$\begin{aligned}\mathrm{Cons}(d_1,\ldots,d_k) = \\ &\mu_{[\otimes_{\{R_i\in R/V(R_i)\subseteq\{x_1,\ldots,x_k\}\}} R_i]}(d_1,\ldots,d_k) \\ &= \min_{\{R_i\in R/V(R_i)\subseteq\{x_1,\ldots,x_k\}\}} \\ &\quad \times \left(\mu_{R_i}\left((d_1,\ldots,d_k)^{\downarrow V(R_i)}\right)\right) \\ &\geq \min_{\{R_i\in R\}} \left(\mu_{R_i}\left((d_1,\ldots,d_k)^{\downarrow V(R_i)}\right)\right) \\ &= \mu_{[\rho^{\downarrow\{x_1,\ldots,x_k\}}]}(d_1,\ldots,d_k)\end{aligned}$$

Hence:

$$\mathrm{Cons}(d_1,\ldots,d_k) \geq \mu_{[\rho^{\downarrow\{x_1,\ldots,x_k\}}]}(d_1,\ldots,d_k).$$

This bound decreases when extending the nodes of the search tree and becomes exact for the leaves. Moreover, it may be incrementally computed as the tree is explored downward:

$$\mathrm{Cons}(d_1,\ldots,d_{k+1}) = \min\Big(\mathrm{Cons}(d_1,\ldots,d_k), \min_{\{R_i\in R, x_{k+1}\in V(R_i) \text{ and } V(R_i)\subseteq\{x_1,\ldots,x_{k+1}\}\}} \times \mu_{R_i}\left((d_1,\ldots,d_{k+1})^{\downarrow V(R_i)}\right)\Big).$$

Like the incremental computation of consistency in classical CSPs, the incremental computation of $\mathrm{Cons}(d_1,\ldots,d_{k+1})$ considers each constraint only once.

Hence, the search starts with a lower bound b_{inf} (for pruning) and an upper bound b_{sup} of $\mathrm{Cons}(P)$; b_{inf} and b_{sup} may respectively be initialized to 0 and 1, or to better lower and upper bounds of $\mathrm{Cons}(P)$ if available. The consistency of the root is taken as b_{sup}. At each step, the current partial instantiation $(d_1,\ldots,d_k)$ is tentatively extended to variable x_{k+1}. If there is a value d_{k+1} such that $\mathrm{Cons}(d_1,\ldots,d_{k+1}) > b_{\mathrm{inf}}, d_{k+1}$ is assigned to x_{k+1}, and if no value consistent enough can be found for x_{k+1}, the algorithm backtracks to the most recent variable assignment. When a solution $(d_1,\ldots,d_n)$ is reached whose consistency is greater than b_{inf}, it is thus the best current solution; b_{inf} is updated to $\mathrm{Cons}(d_1,\ldots,d_n)$ since it is a better lower bound of $\mathrm{Cons}(P)$. If $\mathrm{Cons}(d_1,\ldots,d_n) < b_{\mathrm{sup}}$, the algorithm backtracks in order to find a solution better than the current one. It should be noticed that partial instantiations $(d_1,\ldots,d_k)$, which have been extended to a solution $(d_1,\ldots,d_n)$ whose consistency is equal to $\mathrm{Cons}(d_1,\ldots,d_k)$, do not have better extensions; hence, these extensions do not have to be explored.

Figure 10 shows a search tree corresponding to the example given previously.

Circumstances may impose resource bounds. In particular, real time processing may require immediate answers that can be refined later, if time allows. The Depth-First Branch and Bound process is well suited to provide resource-bounded solutions. The best instantiation available can simply be reported when, for example, a time limit is exceeded. In our example, the discovery of the best solution requests 10 nodes and 37 checks of consistency (i.e. computations of the satisfaction degree of a constraint). The first solution (consistency: α_1) is reached after 3 node extensions and 10 consistency checks and the best solution (consistency: α_3) is encountered after 7 node extensions and 23 consistency checks, the remaining computational effort being used to prove that there is no better solution (see Figure 10).

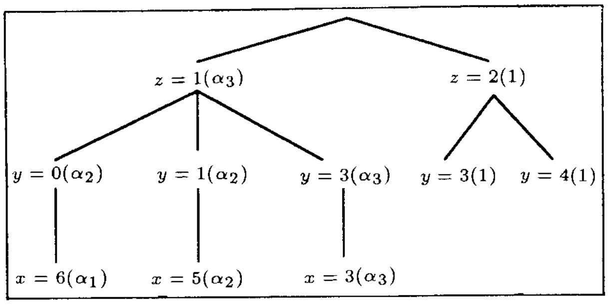

Figure 10. A search tree for the example (the tree is explored from the root to leaves and from left to right).

This kind of algorithm has clearly a worst-case behavior not worse than classical backtracking. Both algorithms, in the worst case, will end up trying all possible combinations of values and testing all the constraints among them. It may actually save effort as stressed in [14]. In the example using a pure backtrack search, 44 nodes and 155 consistency checks were needed; the best solution being reached after 9 node extensions and 28 consistency checks.

The improvement of the search depends on the bounds b_{inf} and b_{sup}. The higher b_{inf}, the more efficient the pruning of useless branches and the lower b_{sup}, the sooner the search will stop. For instance, there are:

- 6 nodes and 24 consistency checks for $b_{\mathrm{inf}} = \alpha_3$ and $b_{\mathrm{sup}} = 1$.
- 7 nodes and 23 consistency checks for $b_{\mathrm{inf}} = 0$ and $b_{\mathrm{sup}} = \alpha_3$.
- 3 nodes and 10 consistency checks for $b_{\mathrm{inf}} = b_{\mathrm{sup}} = \alpha_3$.

It is possible to develop a large class of tree search algorithms (e.g. beam search as in [34]) based on the same principles and integrating different enhancements or variants (see [11]). Heuristics for choosing the instantiation ordering of the variables like those proposed by Dechter and Meiri [35] may be used, since they only consider structural characteristics. Dynamic search rearrangement may also be applied. When extending the current instantiation, the variable having the least number of values whose degree of satisfaction is greater than b_{inf} should be chosen first. A variant for assessing the priority of a variable may be to consider the set of values which are consistent with the current instantiation with a degree greater than b_{inf}:

$$\text{Priority}(x_j/d_1,\ldots,d_k) = \text{Cardinality}\{d_j, \text{Cons}(d_1,\ldots,d_k,d_j) > b_{\text{inf}}\}.$$

For the selection of the value of a variable, the value(s) having the highest degree of satisfaction may be chosen first.

Nonmonotonicity in FCSPs

Using the classical CSP approach, the set of solutions shrinks when new constraints are added and eventually becomes empty in case of conflicting constraints. In the FCSP framework, adding a new constraint to a problem P may rule out all the previously best solutions if they satisfy the new constraint to a degree lower than Cons(P). However, as long as the new problem (say P') is not totally inconsistent, a new set of best solutions appears that satisfies the new problem to a degree Cons(P') $\leq$ Cons(P). Indeed, it holds that:

$$R_1 \otimes \cdots \otimes R_m \otimes R_{m+1} \subseteq R_1 \otimes \cdots \otimes R_m,$$

where $\subseteq$ stands for the fuzzy set inclusion, but generally:

$$\{(d_1,\ldots,d_n)/\mu_{R_1\otimes\cdots\otimes R_m\otimes R_{m+1}}(d_1,\ldots,d_n) = \text{Cons}(P')\} \not\subseteq \{(d_1,\ldots,d_n)/\mu_{R_1\otimes\cdots\otimes R_m}(d_1,\ldots,d_n) = \text{Cons}(P)\}.$$

Hence, the set of best solutions does not decrease monotonically when new constraints are added. The nonmonotonic behavior of soft constraints has been noticed by Satoh [16]. The type of nonmonotonicity at work here is the same as the one captured by possibilistic logic [36] and appears only in the presence of inconsistency. It has been precisely characterized by Benfehrat et al. [37] as the class of preferential inference relations satisfying the rational monotonicity property of Lehmann [38]. In fact, adding a new constraint may lead to four situations:

- The new constraint is redundant: $R_1 \otimes \cdots \otimes R_m \subseteq R_{m+1}$; the set of best solutions remains unchanged.
- The new constraint is totally compatible with P : Cons(P) = Cons(P'); the set of best solutions is included in the previous one but may remain unchanged.
- The new constraint is partially inconsistent with P : Cons(P') < Cons(P); constraints are implicitly relaxed according to their flexiblity and the set of best solutions is not necessarily included in the previous one.
- The new constraint is totally incompatible with P : Cons(P') = 0; the set of best solutions is empty.

  In the example which was given before, the consistency of the problem is α_3 and the set of best solutions consists of a single one, namely $\{(x=3, y=3, z=1)\}$:

- Adding the redundant hard constraint $z \leq 3$ does not change the consistency of the problem nor the set of best solutions.
- Adding the compatible hard constraint $y + z = 4$ neither changes the consistency of the problem nor the set of best solutions; however, the satisfaction degrees of other instantiations decrease (e.g. the satisfaction degrees of $(x=4, y=1, z=2)$ become 0 instead of α_2).
- Adding the hard constraint $y + z = 3$, the consistency of the problem becomes α_2 and the new set of best solutions is $\{(x=4, y=$

$1, z = 2), (x = 4, y = 2, z = 1), (x = 4, y = 0, z = 3)\}$.

- Adding the hard constraint $x + y = 3$, the problem becomes totally inconsistent.

As a consequence of this nonmonotonic behavior, the problem of solution maintenance in FCSPs appears to be more complex than in classical CSPs. Pruned branches in a previous search through the tree have to be developed contrary to the method proposed by Van Hentenryck [39]. The question of relaxing or deleting a constraint is not separately considered in the FCSP model, since the relaxation capacities of the constraints are supposed to be explicitly represented by means of preference among values and priority degrees. In other terms, in the FCSP model, the allowed weakening and deletion of constraints are already captured by the flexibility of the constraints (as far as preferences remain unchanged). On the contrary, when constraints have to dynamically be added or strengthened, e.g. the priority of a constraint (resp. the satisfaction degree of a value) increases (resp. decreases), the nonmonotonicity phenomena described above takes place.

CONCLUSION

The rich expressive power of possibility theory provides a general and unified framework for the representation and the management of flexible constraints involving preferences on values as well as prioritized constraints. The FCSP formalism, which is a generalization of classical CSPs, nevertheless offers a large variety of efficient problem solving tools. Most classical CSP algorithms easily extend, as well as most of the CSP theoretical results and their applications (e.g. tree clustering). This is due to the fact that FCSP's are not additive, but solely based on commensurate orderings, so that all useful properties of the Boolean structure underlying classical CSP's remain valid. The FCSP framework is currently applied to constraint-based approaches in jobshop scheduling [40] where flexible constraints and uncertain parameters are usual features.

As it turns out, explicitly taking the flexibility of the problem into account does not drastically increase the worst-case computational cost of the search procedure; the complexity of filtering procedures may be increased by a factor reflecting the number of different levels used to describe flexibility in the application under concern. Moreover, the problem of finding a feasible solution is changed into an optimization problem of the bottleneck kind, to which Branch and Bound procedures may apply. Of course, in practice finding an optimal solution is generally more computationally expensive than finding a feasible solution. However, experiments carried out in the area of scheduling indicate that the first feasible solution found in the FCSP framework is often obtained more quickly than when preferences are neglected [23]. Moreover, the FCSP approach bypasses empirical relaxation techniques which are needed when a set of constraint is globally unfeasible. Constraint relaxation often happens to be more expensive, difficult to formulate and suboptimal. On the contrary, the FCSP approach can handle partially inconsistent problems. A solution (the instantiation with the maximal satisfaction degree) will be provided as long as the problem is not totally inconsistent. Hence, fuzzy constraints are also useful to guide the search procedure towards "interesting" solutions. Theoretical extensions of the framework are planned with a view to developing computational tools for handling refinements of the global minimum-based satisfaction ordering used here that may be judged as not sufficiently discriminant [23,33]. Finally, this formalism suggests a nonmonotonic framework for dynamic CSPs, when for instance in computer aided-design, default constraints which are used in a first step analysis, are then dynamically modified by the designer.

Moreover, the framework offered by possiblility theory enables us to represent ill-known parameters, whose precise value is neither accessible nor under our control, under the form of so-called possibility distributions (where the possible values are rank-ordered according to their level of plausibility). Ill-known parameters contrast with decision variables on which a decision-maker has control. It is shown in [24] that constraints whose satisfaction depends on these ill-known parameters can be represented in the setting of possibility theory as well. In the presence of ill-known parameters, robust solutions should be searched for, such that the constraints be satisfied whatever the values of these ill-known parameters. Possibility theory implements this idea in a flexible way.

REFERENCES

1. Waltz, D." Understanding line drawings of scenes with shadows", in *The Psychology of Computer Vision*, P.H. Winston, Ed., McGraw-Hill, New York, USA, pp 19–92 (1975).

2. Rosenfeld, A., Hummel, R.A. and Zucker, S.W. "Scene labeling by relaxation operations", *IEEE Trans. on Systems, Man and Cybernetics*, **6**, pp 420–433 (1976).

3. Hummel, R. and Zucker, S. "On the foundations of relaxation labeling processes", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **5**(3), pp 267–287 (1983).

4. Bowen, J., Lai, R. and Bahler, D. "Fuzzy semantics and fuzzy constraint networks", *Proc. of the 1st IEEE Conf. on Fuzzy Systems (FUZZ-IEEE'92)*, San Fransisco, USA, pp 1009–1016 (1992).

5. Bowen, J., Lai, R. and Bahler, D. "Lexical imprecision in fuzzy constraint networks", *Proc. of the National Conf. on Artificial Intelligence (AAAI'92)*, pp 616–620 (1992).

6. Guan, Q. and Friedrich, G. "Extending constraint satisfaction problem solving in structural design", *Proc. of the 5th Inter. Conf. IEA/AIE*, Paderborn, LNAI 604, Springer Verlag, Berlin, Germany, pp 341–350 (1992).

7. Freuder, E.C. and Snow, P. "Improved relaxation and search methods for approximate constraint satisfaction with a maximin criterion", *Proc. of the 8th Biennial Conf. of the Canadian Society for Computational Studies of Intelligence*, Ontario, Canada, May 22–25, pp 227–230 (1990).

8. Martin-Clouaire, R. "Dealing with soft constraints in a constraint satisfaction problem", *Proc. of the Inter. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'92)*, Mallorca, Spain, July 6–10, pp 37–40 (1992).

9. Faltings, B., Haroud, D. and Smith, I. "Dynamic constraint propagation with continuous variables", *Proc. of the Europ. Conf. on Artificial Intelligence (ECAI'92)*, pp 754–758 (1992).

10. Descottes, Y. and Latombe, J.C. "Making compromises among antagonist constraints", *Artificial Intelligence*, **27**, pp 149–164 (1985).

11. Schiex, T. "Possibilistic constraint satisfaction problems or how to handle soft constraints", *Proc. of the 8th Conf. on Uncertainty in Artificial Intelligence*, D. Dubois, M.P. Wellman, B. D'Ambrosio and P. Smets, Eds., Stanford, CA, July 17–19, Morgan & Kaufmann, San Mateo, CA USA, pp 268–275 (1992).

12. Brewka, G., Guesgen, H. and Hertzberg, J. "Constraint relaxation and nonmonotonic reasoning", German National Research Center (GMD), Report TR-92-02 (1992).

13. Borning, A., Maher, M., Marindale, A. and Wilson, M. "Constraint hierarchies and logic programming", *Proc. of the Inter. Conf. on Logic Programming,* Lisbon, Portugal, pp 149–164 (June 1989).

14. Freuder, E.C. "Partial constraint satisfaction", *Proc. of the Inter. Joint Conf.*

on Artificial Intelligence (IJCAI' 89), Detroit, MI, USA, pp 278–283 (1989).

15. Freuder, E.C. and Wallace, R. "Partial constraint satisfaction", *Artificial Intelligence*, **58**, pp 21–71 (1992).

16. Satoh, K. "Formalising soft constraint by interpretation ordering", *Proc. of the Europ. Conf. on Artificial Intelligence (ECAI'90)*, Stockholm, Sweden, pp 585–590 (1990).

17. Lang, J. "Possibilistic logic as a logical framework for min-max discrete optimization problems and prioritized constraints", *Proc. of the Inter. Workshop on Fundamentals of Artificial Intelligence Research (FAIR'91)*, Smolenice, Czechoslovakia, Sept. 8–12, Lecture Notes in Computer Science, **535**, Springer Verlag, Berlin, Germany, pp 112–126 (1991).

18. Dubois, D., Fargier, H. and Prade, H. "The calculus of fuzzy restriction as a basis for flexible constraint satisfaction", *Proc. of the 2nd IEEE Inter. Conf. on Fuzzy Systems (FUZZ-IEEE' 93)*, San Fransisco, CA, USA, March 28-April 1st, pp 1131–1136 (1990).

19. Zadeh, L.A. "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems*, **1**, pp 3–28 (1978).

20. Dubois, D. and Prade, H. (with the collaboration of Farreny, H., Martin-Clouaire, R. and Testemale, C.) *Possibility Theory—An Approach to Computerized Processing of Uncertainty*, Plenum Press, New York, USA (1988).

21. Dubois, D. and Prade, H. "Processing fuzzy temporal knowledge", *IEEE Trans. on Systems, Man and Cybernetics*, **19**(4), pp 729–744 (1989).

22. Yager, R.R. "Some extensions of constraint propagation of label sets", *Int. J. of Approximate Reasoning*, **3**, pp 417–435 (1989).

23. Fargier, H. "Problèmes de satisfaction de contraintes flexibles–Application à l'ordonnancement de production", Thèse de l'Université P. Sabatier, Toulouse, France (1994).

24. Dubois, D., Fargier, H. and Prade, H. "Possibility theory in constraint satisfaction problems: Handling priority, preference and uncertainty", *Applied Intelligence*, to appear (1995).

25. Freuder, E.C. "A sufficient condition for backtrack-free search", *J. of the ACM*, **32** (4), pp 755–761 (1982).

26. Dechter, R. "From local to global consistency", *Artificial Intelligence*, **55**, pp 87–107 (1992).

27. Zadeh, L.A. "Calculus of fuzzy restrictions", in *Fuzzy Sets and Their Applications to Cognitive and Decision Processes*, L.A. Zadeh et al., Eds., Academic Press, New York, USA, pp 1–39 (1975).

28. Moulin, H. *Axioms of Cooperative Decision Making*, Cambridge University Press, Cambridge, MA, USA (1988).

29. Goguen, J.A. "L-fuzzy sets", *J. of Mathematical Analysis and Application*, **18**, pp 145–174 (1967).

30. Dubois, D. and Prade, H. "Tolerant fuzzy pattern matching: An introduction", *J. of Fuzzy Logic and Intelligent Systems*, Seoul, Korea, **3**(2), pp 3–17 (1993).

31. Sadeh, N. "Look-ahead techniques for micro-opportunistic job shop scheduling", Report CS-91-102, Carnegie Mellon University (1991).

32. Fargier, H., Lang, J. and Schiex, T. "Selecting preferred solutions in fuzzy constraint satisfaction problems", *Proc. of the 1st Europ. Conf. on Fuzzy Information Technologies (EUFIT'93)*, Sept. 7–10, ELITE-Foundation, Aachen, Germany, pp 1128–1134 (1993).

33. Dubois, D., Fargier, H. and Prade, H. "Refinements to the maximin approach to decision-making in fuzzy environment", *Fuzzy Sets and Systems*, to appear (1996).

34. Fox, M., Allen, B. and Strohm, G.A. "Job-shop scheduling: An investigation in constraint-directed reasoning", *Proc. of the National Conf. on Artificial Intelligence (AAAI'82)*, Pittsburgh, USA, pp 155–158 (1982).

35. Dechter, R. and Meiri, I. "Experimental evaluation of preprocessing, techniques in Constraint Satisfaction Problems", *Proc. of the Inter. Joint Conf. on Artificial Intelligence (IJCAI' 89)*, Detroit, MI, USA, pp 271–277 (1989).

36. Dubois, D. and Prade, H. "Possibilistic logic, preferential models, non-monotonicity and related issues", *Proc. of the Inter. Joint Conf. on Artificial Intelligence (IJCAI'91)*, Sydney, Australia, Aug. 24–30, pp 419–424 (1991).

37. Benfehrat, S., Dubois, D. and Prade, H."Representing default rules in possibilistic logic" *Proc. of the 3rd Inter Conf. on Principles of Knowledge Representation and Reasoning (KR'92)*, B. Nebel, C. Rich and W. Swartout, Eds., Cambridge, MA, Morgan & Kaufmann, San Mateo, CA USA, pp 673–684 (1992).

38. Lehmann, D. "What does a conditional knowledge base entail?", *Proc. of the 1st Inter. Conf. on Principles of Knowledge Representation and Reasoning (KR'89)*, Toronto, Canada, pp 212–221 (1989).

39. Van Hentenryck, P. "Incremental constraint satisfaction in logic programming", *Proc. ICLP 90*, pp 189–202 (1990).

40. Dubois, D., Fargier, H. and Prade, H. "Fuzzy constraints in job-shop scheduling", *J. of Intelligent Manufacturing*, **6**, pp 215–235 (1995).