CONSTRAINT LOGIC PROGRAMMING AND INTUITIONISTIC FUZZY LOGICS

Krassimir T. Atanassov
Math. Research Lab. - IPACT, P.O.Box 12, Sofia-1113, BULGARIA

The idea that a connection exists between the concepts of Constraint Logic Programming (ICLP) [1] on one hand and that of Intuitionistic Fuzzy Sets (IFSs) [2, 3] and Intuitionistic Fuzzy Logics (IFLs) [4-7] on the other hand, is discussed.

On the basis of these concepts, the concept of "Intuitionistic Fuzzy Constraint Logic Programming" (IFCLP) is introduced.

Initially, we shall give some remarks on IFLs.

To each proposition (in the classical sense) one can assign its truth value: truth - denoted by 1, or falsity - 0. In the case of fuzzy logics this truth value is a real number in the interval [0, 1] and can be called "truth degree" of a particular proposition. In [4] we add one more value - "falsity degree" - which will be in the interval [0, 1] as well. Thus one assigns to the proposition p two real numbers $\mu(p)$ and $\nu(p)$ with the following constraint to hold:

$$\mu(p) + \nu(p) \leq 1.$$

Let this assignment be provided by an evaluation function V defined over a set of propositions S in such a way that:

$$V(p) = \langle \mu(p), \nu(p) \rangle.$$

Hence the function $V: S \dashrightarrow [0, 1] \times [0, 1]$ gives the truth and falsity degrees of all propositions in S.

We assume that the evaluation function V assigns to the logical truth T: $V(T) = \langle 1, 0 \rangle$, and to F: $V(F) = \langle 0, 1 \rangle$.

The evaluation of the negation $\neg p$ of the proposition p will be defined through:

$$V(\neg p) = \langle \nu(p), \mu(p) \rangle.$$

Obviously, when $\nu(p) = 1 - \mu(p)$, i.e.

$$V(p) = \langle \mu(p), 1 - \mu(p) \rangle,$$

for $\neg p$ we get:

$$V(\neg p) = \langle 1 - \mu(p), \mu(p) \rangle,$$

which coincides with the result for ordinary fuzzy logic (see e. g., [8]).

When the values V(p) and V(q) of the propositions p and q are known, the evaluation function V can be extended also for operations "&", "$\vee$" through the definition :

$$V(p \,\&\, q) = \langle \min(\mu(p), \mu(q)), \max(\nu(p), \nu(q)) \rangle,$$

$$V(p \vee q) = \langle \max(\mu(p), \mu(q)), \min(\nu(p), \nu(q)) \rangle,$$

Depending on the way of definition of the operation "$\supset$" different variants of IFPC can be obtained. Here we shall use the variant called in [4] "max-min":

$$V(p \supset q) = \langle \max(\gamma(p), \mu(q)), \min(\mu(p), \gamma(q))\rangle$$

By analogy with the operations over IFS it will be convenient to define for the propositions p, q $\in$ S:

$$\neg V(p) = V(\neg p),$$
$$V(p) \wedge V(q) = V(p \,\&\, q),$$
$$V(p) \vee V(q) = V(p \veebar q),$$
$$V(p) \rightarrow V(q) = V(p \supset q).$$

For the needs of the discussion below we shall define the notion of intuitionistic fuzzy tautology (IFT) through:

"A is an IFT" iff "if $V(A) = \langle a, b\rangle$, then $a \geq b$".

In [5] are defined the following IF-interpretations of the modal operators

$$V(\Box p) = \langle a, 1-a\rangle,$$
$$V(\Diamond p) = \langle 1-b, b\rangle$$

and

$$\Box\, V(p) = V(\Box p),$$
$$\Diamond\, V(p) = V(\Diamond p),$$

where $V(p) = \langle a, b\rangle$. Here we shall define analogously of the operators defined over IFS with the forms ($\alpha, \beta \in [0, 1]$):

$$V(D_{\alpha}(p)) = \langle a + \alpha.(1 - a - b), b + (1 - \alpha).(1 - a - b))$$

$$V(F_{\alpha,\beta}(p)) = \langle a + \alpha.(1 - a - b), b + \beta.(1 - a - b)) \text{ and } \alpha + \beta \leq 1$$

$$V(G_{\alpha,\beta}(p)) = \langle \alpha.a, \beta.b)$$

$$V(H_{\alpha,\beta}(p)) = \langle \alpha.a, b + \beta.(1 - a - b)\rangle$$

$$V(J_{\alpha,\beta}(p)) = \langle a + \alpha.(1 - a - b), \beta.b\rangle$$

$$V(\bar{H}_{\alpha,\beta}(p)) = \langle \alpha.a, b + \beta.(1 - \alpha.a - b)\rangle$$

$$V(\bar{J}_{\alpha,\beta}(p)) = \langle a + \alpha.(1 - a - \beta.b), \beta.b\rangle$$

The definition for the quantifiers is as follows (see [6]):

$$V(\forall xA) = \langle \min_{x\in E} \mu(A), \max_{x\in E} \gamma(A)\rangle$$

and

$$V(\exists xA) = \langle \max_{x\in E} \mu(A), \min_{x\in E} \gamma(A)\rangle.$$

We must note, that these operators are not studied in the frames of the modal logic, yet.

Let p be a proposition and let V be a truth-value function, which juxtaposes to the proposition p and to the time-moment $t \in T$ (T is a fixed set which we shall call "time-scale" and it is strictly oriented by the relation "<") the ordered pair:

$$V(p, t) = \langle \mu(p, t), \gamma(p, t)\rangle.$$

Let

$$T' = \{t'/t' \in T \,\&\, t' < t\}$$

$$T'' = \{t''/t'' \in T \,\&\, t'' > t\}$$

We shall define for given p and t the operators $P(p,t)$, $F(p,t)$, $H(p,t)$, $G(p,t)$: $[0, 1] \times [0, 1] \times T \rightarrow [0, 1] \times [0, 1] \times T$, for which:

$$X(p,t) = X(<\mu(p,t), \gamma(p,t)>)$$

for $X \in \{P, F, H, G\}$ and:

- $V(P(p,t)) = <\mu(p,t'), \gamma(p,t')>$,
  where $t' \in T'$ satisfies the conditions:
  (a) $\mu(p,t') - \gamma(p,t') = \max_{t* \in T'} (\mu(p,t*) - \gamma(p,t*))$,
  (b) if there exist more than one such element of T', then t' is the maximal.
- $V(F(p,t)) = <\mu(p,t''), \gamma(p,t'')>$,
  where $t'' \in T''$ satisfies the conditions:
  (a) $\mu(p,t'') - \gamma(p,t'') = \max_{t* \in T''} (\mu(p,t*) - \gamma(p,t*))$,
  (b) if there exist more than one such element of T'', then t'' is the minimal.
- $V(H(p,t)) = <\mu(p,t'), \gamma(p,t')>$,
  where $t' \in T'$ satisfies the conditions:
  (a) $\mu(p,t') - \gamma(p,t') = \min_{t* \in T'} (\mu(p,t*) - \gamma(p,t*))$,
  (b) if there exist more than one such element of T', then t' is the maximal.
- $V(G(p,t)) = <\mu(p,t''), \gamma(p,t'')>$,
  where $t'' \in T''$ satisfies the conditions:
  (a) $\mu(p,t'') - \gamma(p,t'') = \min_{t* \in T''} (\mu(p,t*) - \gamma(p,t*))$,
  (b) if there exist more than one such element of T'', then t'' is the minimal.

* * *

Below, we shall follow the exposition and terminology of the paper [1] and we shall introduce the elements of the IFCLP. Thus the comparison between both these texts will be saved labour.

Let $SORT = U\{SORT_i\}$ denote the finite set of sorts in question. A signature of an n-ary function (predicate, variable) symbol f is a sequence of n+1 (resp. n, 1) elements of SORT. By the term sort of f, we mean the last element in the signature of the function symbol f. The symbols Σ and Π denote denumerable collections of function symbols (and their signatures) and predicate symbols (and their signatures) respectively. Let W denote a collection of variables, $\sigma(\Sigma)$ and $\sigma(\Sigma \cup W)$ denote, respectively, ground terms and the terms possibly containing variables. An atom is of the form $p(t_1, t_2, \ldots, t_m)$, where p is an m-ary symbol in Π and $t_i \in \sigma(\Sigma \cup W)$ and $V(t_i) \in [0, 1] \times [0, 1]$, for every i ($1 \le i \le m$) and $V(p(t_1, t_2, \ldots, t_m)) \in [0, 1] \times [0, 1]$.

A structure is defined over alphabets Π and Σ of predicate and function symbols, where Π contains the equality symbol "=", which needs no signature. Such a structure $R(\Sigma, \Pi)$ consists of:

(a) a collection DR of non-empty sets DR_s, where s ranges over the sorts in SORT;

(b) an assignment, to each n-ary $f \in \Sigma$, of a function

$$DR_{s_1} \times DR_{s_2} \times \ldots \times DR_{s_n} \to DR_s,$$

where $(s_1, s_2, \ldots, s_n, s)$ is the signature of f;

(c) an assignment, to each n-ary $p \in \Pi$, excepting the symbol "=", of a function $DR_{s_1} \times DR_{s_2} \times \ldots \times DR_{s_n} \in [0, 1] \times [0, 1]$, where $(s_1, s_2, \ldots, s_n)$ is the signature of p.

An atomic (Π, Σ)-constraint is simply an atom over the alphabets Σ and Π; a (Π, Σ)-constraint is a possibly empty finite set of those atoms. Intuitively, a constraint can be a logical expression of atomic constraints connected by logical operations conjunction, disjunction, negation, implication, by modal operators "necessity" and "possibility" and their IF-extensions (the operators D_α, $F_{\alpha,\beta}$, $G_{\alpha,\beta}$, etc.) and by temporal operators (in contrast to [1], where it can be only a conjunction of atomic constraints).

A $R(\Pi, \Sigma)$-valuation on an expression over Π and Σ is a mapping from each distinct variable in the expression into DR_s, where s is the sort of the variable in question. Where Θ is a $R(\Pi, \Sigma)$-valuation on the term t, we write $t\Theta$ to denote the appropriate element in DR_s, where s is the sort associated with t. Similarly, where Θ is a $R(\Pi, \Sigma)$-valuation on the atomic (Π, Σ)-constraint c, $c\Theta$ denotes the proposition such that either

(a) $R(\Pi, \Sigma) \vdash c\Theta$ (i.e. $c\Theta$ is an IFT)

or

(b) $R(\Pi, \Sigma) \vdash \neg c\Theta$ (i.e., $c\Theta$ is an IF false, i.e., if $V(\neg c\Theta) = \langle a, b \rangle$, then $a < b$).

If C is a possibly infinite set of atomic (Π, Σ)-constraints, we write

(a) $R(\Pi, \Sigma) \vdash C\theta$, if for all $c \in C$ $R(\Pi, \Sigma) \vdash c\theta$

(b) $R(\Pi, \Sigma) \vdash \neg C\theta$, otherwise.

Whenever (a) holds, we say that C is $R(\Pi, \Sigma)$-solvable and that θ is an $R(\Pi, \Sigma)$-solution of C.

Where $A = p(t_1, t_2, \ldots, t_n)$ is a (Π_p, Σ)-atom and θ a (Π, Σ)-valuation of $t_1, t_2, \ldots, t_n$, let $A\theta$ denote $p(t_1\theta, t_2\theta, \ldots, t_n\theta)$. The $R(\Pi, \Sigma)$-base of a program P is then given by:

$\{p(x_1, x_2, \ldots, x_n)\theta$:
p is an n-ary symbol in Π and
θ is a (Π, Σ)- valuation of the variables
$x_1, x_2, \ldots, x_n\}$.

If S is a subset of the $R(\Pi, \Sigma)$-base of a program P, we write $(S)_p$ to denote the subset of S containing elements associated with the predicate symbol $p \in \Pi_p$. An R-model of a program P is given by a subset I of the R-base such that for every rule in P

$$A \leftarrow (c \mid B_1, B_2, \ldots, B_n)$$

where $n \geq 0$, and for every valuation θ on A, c, $B_1, B_2, \ldots, B_n$, such that θ is an R-solution of c, $\{B_1\theta, B_2\theta, \ldots, B_n\theta\} \subset I$ implies $A\theta \in I$.

Here we shall add to the text of [1] the following possible constructions base on the constructions in InF-PROLOG [8].

Initially, we must note, that the right-hand part of the rule has some truth value which is obtained by the function V and which is an element of the set $[0, 1] \times [0, 1]]$. We can somehow calculate the truth value of the left-hand part of the rule. In general, the rules of P can have one of the following forms:

(1) $A \leftarrow (c \mid B_1, B_2, \ldots, B_n;\ \mu, \gamma)$,

(2) $A \leftarrow (c \mid B_1, B_2, \ldots, B_n;\ M, N)$,

where $M, N \subset [0, 1]$ and $\sup M + \sup N \leq 1$,

(3) $A;\ M_1, N_1 \leftarrow (c \mid B_1, B_2, \ldots, B_n;\ M_2, N_2)$,

where $M_1, N_1 \subset [0, 1]$ and $\sup M_1 + \sup N_1 \leq 1$ and

$M_2, N_2 \subset [0, 1]$ and $\sup M_2 + \sup N_2 \leq 1$,

(4) $A;\ X_{\alpha,\beta} \leftarrow (c \mid B_1, B_2, \ldots, B_n)$, where $\alpha, \beta \in [0, 1]$,

where $X \in \{D_\alpha, F_{\alpha,\beta}$ (in this case $\alpha + \beta \leq 1$), $G_{\alpha,\beta}, H_{\alpha,\beta}, \bar{H}_{\alpha,\beta}, J_{\alpha,\beta}, \bar{J}_{\alpha,\beta}\}$ or a combination of them.

The sense of these forms is as follows:

Case 1: if <a, b> is the truth-value of the right hand and $a \geq \mu$ and $b \leq \gamma$, then A also receive the value <a, b>; else the rule is not activated.

Case 2: if <a, b> is the truth-value of the right hand and $a \in M$ and $b \in N$, then A alsoreceive the value <a, b>; else the rule is not activated.

Case 3: if <a, b> is the truth-value of the right hand and $a \in M_2$ and $b \in N_2$, then A receive the value

$$\langle \inf M_1 + (a - \inf M_2).(\sup M_1 - \sup M_1)/(\sup M_2 - \sup M_2),$$
$$\inf N_1 + (a - \inf N_2).(\sup N_1 - \sup N_1)/(\sup N_2 - \sup N_2)\rangle;$$

otherwiss the rule is not activated.

Case 4: if <a, b> is the truth-value of the right-hand part, then A receive the value $X_{\alpha,\beta}\langle a, b\rangle$.

Let $d = p(d_1, \ldots, d_n)$ be an element in the R-base of a structure $R(\Pi, \Sigma)$. We say that d is (finitely) definable if there is a (finite) possibly infinite set of (Π, Σ)-constraints $c(x_1, \ldots, x_n)$ containing distinguished variables $x_1, \ldots, x_n$ such that one $R(\Pi, \Sigma)$-solution of c maps $x_1, \ldots, x_n$ into $d_1, \ldots, d_n$ respectively. The element d is definable uniquely if all $R(\Pi, \Sigma)$-solutions of a maps $x_1, \ldots, x_n$ into $d_1, \ldots, d_n$ respectively. An element is a limit element if it is definable uniquely but not by means of a (finite) (Π, Σ)-constraint.

A (many-sorted) first-order theory $I(\Pi, \Sigma)$ is a set of well-formed formulas over the alphabets Π and Σ where formulas are defined in the usual way. A (ground) $d(\Pi, \Sigma)$-substitution is a finite set of the form $\{x_1/t_1, x_2/t_2, \ldots, x_n/t_n\}$ where

(a) the t_i $(1 \leq i \leq n)$ are (ground) terms over Σ which do not contain any occurrences of x_j $(1 \leq j \leq n)$

(b) x_i and t_i have the same sort $(1 \leq i \leq n)$. For these substitutions, we define the notions of applications, composition and relative generality in the usual manner, e.g., we say that a substitution Θ is more general than Γ, denoted by $\Gamma \leq_I \Theta$, if there exists a (Π, Σ)-substitution ∂ such that $I \vdash (\Gamma = \Theta\partial)$.

We shall sometimes associate a substitution

$$\Theta = \{x_1/t_1, x_2/t_2, \ldots, x_n/t_n\}$$

with a system of equations, denoted by $\hat{\Theta}$, as follows $\hat{\Theta} = \{x_1 = t_1, x_2 = t_2, \ldots, x_n = t_n\}$.

A (Π, Σ)-constraint c such that $I(\Pi, \Sigma) \vdash \bar{\exists}c$, where $\bar{\exists}c$ is the existential closure of c, is said to be $I(\Sigma, \Pi)$-satisfable. A I-satisfier Θ of a (Σ, Π)-constraint c is an $R(\Sigma, \Pi)$-substitution such that $I(\Pi, \Sigma) \vdash c\Theta\Gamma$ for all (Σ, Π)-substitutions Γ of $c\Theta$. Clearly $I(\Pi, \Sigma) \vdash c\Theta\Gamma$ implies that c is $I(\Pi, \Sigma)$-satisfiable. A most general I-satisfier Θ of a (Σ, Π)-constraint c is aI-satisfier of c such that $\Gamma \le_I \Theta$ for all I-satisfiers Γ of c.

A constraint logic program is defined over (sorted) alphabets Π_p and Σ where $\Pi_p \cap \Pi = \emptyset$. Such a program consists of a finite set of constraint rules, this last being of the form

$A \leftarrow (c \mid)$ or

$A \leftarrow (c \mid B_1, B_2, \ldots, B_n)$

where c is a possibly empty (Σ, Π)-constraint and A and B_i ($1 \le i \le n$) are atoms over Π_p and Σ. A goal is also defined over (sorted) alphabets Π_p and Σ, and it is of the form

$\leftarrow (c \mid)$ or

$\leftarrow (c \mid B_1, B_2, \ldots, B_n)$

where c is a possibly empty or infinite set of (Π, Σ)-constraints and B_i ($1 \le i \le n$) are atoms over Π_p and Σ; a finite goal contains only a (finite) (Π, Σ)-constraints; a unit goal contains only one (Π_p, Σ)-atom.

These constructions (from [1]) can be generalized to each of the forms (1) - (4), described above. Let the last form of the rules be noted by (0).

The function $T_{(P,R)}$ maps from and into the R-base:

$T_{(P,R,i)}(S) = \{d \in$ R-base:

there exists a rule in P from form (i)
and a R-valuation Θ such that
(a) $R \vdash (A\Theta = d)$
(b) $R \vdash c\Theta$
(c) $\{B_1\Theta, B_2\Theta, \ldots, B_n\Theta\} \subset S\}$

Using $T_{(P,R,i)}(S)$ we can build in a natural way, the following kinds of sets, where α is a (not necessaryly finite) ordinal,

$T_{(P,R,i)} \mid 0 = \emptyset$

$T_{(P,R,i)} \mid (\alpha + 1)$ = if α is a successor ordinal then

$T_{(P,R,i)}(T_{(P,R,i)} \wedge \alpha)$

else

$\cup\{T_{(P,R,i)} \mid \beta\}$, over all $\beta < \alpha$

and

$T_{(P,R,i)} \mid 0$ = R-base

$T_{(P,R,i)} \mid (\alpha + 1)$ = if α is a successor ordinal then

$T_{(P,R,i)}(T_{(P,R,i)} \vee \alpha)$

else

$\cap\{T_{(P,R,i)} \mid \beta\}$, over all $\beta < \alpha$.

We now define the counter-part of this function $T_{(P,R,i)}$, this time with respect to a corresponding theory I.

After this we shall extend the truth value function V to a function related to a temporal parameter z. Each of the forms (0) - (4) can be extended to the form for which the truth value of the right-hand part is $\langle a(z), b(z)\rangle$ and the calculation of the left-hand part is as above. Now we shall obtain essentially new extension of the objects, described in [1]. The function $T_{(P,R,i)}$ defined in [1] maps from and into unit goals. Now it has the form (we shall use the symbol "-" to denote finite sequences of objects such as terms, atoms, etc. and let for the object $\bar{x}$, $arg(\bar{x})$ is the list of all different arguments of such a sequence):

$T_{(P,R,i,z)}(S) = \{\leftarrow(c \mid p(\bar{x})):$

there exists a rule in P of the form (i) and z is an element of some (fixed) temporal scale and $z \in arg(\bar{x})$

$p(\bar{t}) \leftarrow(c_0 \mid p_1(\bar{t}_1), \ldots, p_n(\bar{t}_n))$

(for case (0); for the other cases, the rule has the respective form) where

(a) for each $1 \le i \le n$

there exists $\leftarrow(c_i \mid p_i(\bar{x}_i) \in S$

which share no common variables, such that

$c' = c_0 \cup \{\bar{x}_1 = \bar{t}_1, \ldots, \bar{x}_n = \bar{t}_n\} \cup \{c_1, \ldots, c_n\}$

is I-satisfiable;

(b) $c = c' \cup \{\bar{x} = \bar{t}\}$ }.

For brevity, we shall write the rules only in form (0) but they can be in the other forms, too.

Let $[\leftarrow(c \mid p(\bar{t}))]$ denote all those elements in the R-base associated with the symbol p obtained by R-instantiating variables in c so as to obtain a true value, i.e.

$[\leftarrow(c \mid p(\bar{t}))] = \{p(\bar{x})\Theta\colon \Theta$ is a R-solution of $c \cup \{\bar{x} = \bar{t}\}\}$.

Similarly, $[\leftarrow(c \mid p_1(\bar{t}_1), \ldots, p_k(\bar{t}_k))]$ denotes

$\cup\{p_i(\bar{x}_i)\Theta\colon \Theta$ is a R-solution of $c \cup \{\bar{x}_1 = \bar{t}_1\} \cup \ldots \cup \{\bar{x}_n = \bar{t}_n\}$.

Where P is a constraint logic program, a (P, R)-derivation step from a (not necessarily finite) goal

$\leftarrow (c \mid A_1, A_2, \ldots, A_n)$

results in a goal of the form

$\leftarrow (\bar{c} \mid \bar{B}_1, \bar{B}_2, \ldots, \bar{B}_n)$

if there exist n variants of constraint rules in P with no variables in common with G and with each other

$A'_1 \leftarrow (c_1 \mid \bar{B}_1)$

$A'_2 \leftarrow (c_2 \mid \bar{B}_2)$

. . .

$A'_n \leftarrow (c_n \mid \bar{B}_n)$

such that $\bar{c}$ is $c_1 \cup c_2 \cup \ldots \cup c_n \cup \{A_1 = A'_1, \ldots, A_n = A'_n\}$ and $\bar{c}$ is R-solvable.

This paper is based on [9].

The described constructions are a basis of the following problems, which will be objects of disscussion in other papers of ours:

1. To show that every ordinary program in the frames of the constraint logic programming can be realized by means of InF-PROLOG.
2. To construct an extension of InF-PROLOG containing the rules of the forms (0) - (4) in the temporal case (the rules on InF-PROLOG are in non-temporal variants of the forms (0) - (4)).
3. To construct constraint and temporal expert systems as extensions to the ordinary expert systems.

REFERENCES:

[1] Jaffar J., Lassez J.-L., Constraint logic programming, Monash Univ. Techn. Report No. 74, 1986.

[2] Atanassov K., Intuitionistic fuzzy sets, Fuzzy sets and Systems, Vol. 20 (1986), No. 1, 87-96.

[3] Atanassov K., More on intuitionistic fuzzy sets. Fuzzy sets and systems, 33, 1989, No. 1, 37-46.

[4] Atanassov K., Two variants of intuitonistic fuzzy propositional calculus. Preprint IM-MFAIS-5-88, Sofia, 1988.

[5] Atanassov K., Two variants of intuitionistic fuzzy modal logic. Preprint IM-MFAIS-3-89, Sofia, 1989.

[6] Atanassov K., Gargov G., Intuitionistic fuzzy logic. Compt. rend. Acad. bulg. Sci., Tome 43, N. 3, 1990, 9-12.

[7] Atanassov K., Remark on a temporal intuitionistic fuzzy logic Second Sci. Session of the "Mathematical Foundation of Artificial Intelligence" Seminar, Sofia, March 30, 1990, Prepr. IM-MFAIS-1-90, 1-5.

[8] Mendelson E., Introduction to mathematical logic, Princeton, NJ: D. Van Nostrand, 1964.

[9] Atanassov K., Intuitionistic fuzzy constraint logic programming, Preprint MRL-1-92, Sofia, 1992.