Fuzzy modeling with constraint technology*

Przemysław Kobylański
Institute of Industrial
Engineering and Management,
Wrocław University of Technology,
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
przemko@pwr.wroc.pl

Paweł Zieliński
Institute of Mathematics,
Technical University of Opole,
Luboszycka 3,
45-036 Opole, Poland
pziel@po.opole.pl

Abstract

A representation of fuzzy numbers and constraints in the form of descriptors is proposed. This representation in natural way uses mechanism of constraint propagation available in systems based on constraint technology. A computational example, in which the descriptors are implemented in PROLOG with constraint technology, is presented.

Keywords: Fuzzy Modeling, Fuzzy Constraint, Constraint Technology.

1 Introduction

In the paper we propose an approach to representation of fuzzy numbers and fuzzy constraints. The idea of fuzzy constraints is not new. The mathematical framework to define fuzzy constraints, fuzzy constraint satisfaction problem has been first suggested and developed by Dubois et. al [2, 3].

The approach, presented here, has been worked out with the idea of using existing constraint technology systems, and of exploiting built-in classical algorithms for propagation and checking consistency of constraints. It is assumed that such systems enable us to use linear constraints on real or rational numbers and to maximize and minimize linear functions.

To build an application solving a given problem we can use *procedural* or *declarative programming*. In the first case we have to know an algorithm solving the problem and we encode this algorithm in one of imperative programming languages. In the second one we are able to describe only our knowledge about the problem in one of the declarative programming languages, even if we do not know how to solve this problem.

There is a class of declarative programming languages such as logic programming languages LP. One of the most popular logic programming languages is PROLOG [6]. In PROLOG the knowledge about the problem being solved is written as a set of logical formulas (*clauses*). The solution is determined with inference engine such as *linear resolution*.

The presented concept of descriptors was successfully implemented in IF/PROLOG as module `const_fuzzy` which uses the module `const_linear` for linear constraints on rational numbers [4].

2 Models and constraint technology

Mathematical programming models are declarative descriptions of problems. They can be included into logic programs with the constraint technology. The constraint technology extents the LP class of languages to a new CLP (constraint logic programming) class [1, 5]. CLP languages enable constraints to be used from the model as ordinary statements. This feature gives occasion to generate models in many ways, exactly like the execution of the program can be done on many computational paths depending on the satisfaction of conditions in control statements. One of the interesting features of CLP languages is that the constraints from the model can influence on the execution of the program. For instance, if one constraint contradicts with the previously added ones it causes the backtrack and selection of the other execution path.

* This work was supported by grant no. 7T11F02120 from the State Committee for Scientific Research (Komitet Badań Naukowych).

Such full integration of models and declarative programs is one of the main differences between procedural and declarative languages. It allows to build applications solving operations research problems easily.

3 Selected notions and symbols of the fuzzy sets theory

We recall here only those concepts and elements of notation from the fuzzy set theory which we use in the paper.

Definition 1. *Let $\tilde{A}_1,\dots,\tilde{A}_n$ be fuzzy sets in $\mathbb{X}_1,\dots,\mathbb{X}_n$ with the membership functions $\mu_{\tilde{A}_1}(x_1),\dots,\mu_{\tilde{A}_1}(x_n)$, respectively. Then the Cartesian product of the fuzzy sets $\tilde{\Omega}=\tilde{A}_1\times\cdots\times\tilde{A}_n$ is defined as a fuzzy set in $\Omega=\mathbb{X}_1\times\cdots\times\mathbb{X}_n$ whose membership function has the following form:*

$$\begin{aligned}\mu_{\tilde{\Omega}}(\boldsymbol{x}) &= \mu_{\tilde{A}_1\times\cdots\times\tilde{A}_n}(x_1,\dots,x_n)\\ &= \min\{\mu_{\tilde{A}_1}(x_1),\dots,\mu_{\tilde{A}_1}(x_n)\}.\end{aligned}$$

Let us recall the extension principle of Zadeh [8], which provide a general method for extending a non-fuzzy mathematical concept to the fuzzy framework.

Definition 2. *Let f be be mapping from Ω to a set $\mathbb{Y}$ such that $y=f(\boldsymbol{x})$. Then the fuzzy set $\tilde{B}$ in $\mathbb{Y}$ induced by the fuzzy set $\tilde{\Omega}$ in Ω is defined by the following membership function:*

$$\mu_{\tilde{B}}(y)=\begin{cases}\sup\limits_{\substack{\boldsymbol{x}\in\Omega\\ y=f(\boldsymbol{x})}}\mu_{\tilde{\Omega}}(\boldsymbol{x}) & \text{for} \quad f^{-1}(y)\neq\emptyset,\\ 0 & \text{for} \quad f^{-1}(y)=\emptyset.\end{cases} \tag{1}$$

Nguyen [7] proposed an equivalent representation of the extension principle.

Theorem 1. *If there exist $x_1,\dots,x_n$ such that $\mu_{\tilde{B}}(y)=\mu_{\tilde{A}_1\times\cdots\times\tilde{A}_n}(x_1,\dots,x_n)$ for any $y\in\mathbb{Y}$ (the supremum of (1) is attained in $x_1,\dots,x_n$), then the following equality holds:*

$$\tilde{B}^{\lambda}=[f(\tilde{A}_1,\dots,\tilde{A}_n)]^{\lambda}=f(\tilde{A}_1^{\lambda},\dots,\tilde{A}_n^{\lambda}),$$

where $\tilde{A}_i^{\lambda}$ is λ-cut of the fuzzy set $\tilde{A}_i$

Definition 3. *Fuzzy number $\tilde{F}$ is a fuzzy set in $\mathbb{R}$ with a membership function $\mu_{\tilde{F}}(x)$, which is continuous and fulfills the following conditions: normality, i.e. $\sup_{x\in\mathbb{R}}\mu_{\tilde{F}}(x)=1$ and convexity, i.e. for any $x,y\in\mathbb{R}$, $\alpha\in[0,1]$ it holds $\mu_{\tilde{F}}(\alpha x+(1-\alpha)y)\geq\min\{\mu_{\tilde{F}}(x),\mu_{\tilde{F}}(y)\}$.*

Definition 4. *The interval $\tilde{F}^{\lambda}=[\underline{F}^{\lambda},\overline{F}^{\lambda}]=\{x\in\mathbb{R}|\mu_{\tilde{F}}(x)\geq\lambda\},\lambda\in(0,1]$, and $\tilde{F}^{0}=[\underline{F}^{0},\overline{F}^{0}]$ for $\lambda=0$, where $\underline{F}^{0}=\inf\{x|\mu_{\tilde{F}}(x)>0\}$ and $\overline{F}^{0}=\sup\{x|\mu_{\tilde{F}}(x)>0\}$, is called λ-cut of fuzzy number $\tilde{F}$.*

4 Descriptors

In this section we introduce the notion of descriptor of a fuzzy number and a fuzzy constraint. The descriptor is a structure, which uniquely describes a fuzzy number or a fuzzy constraint. It allows to operate on fuzzy numbers, easily, and enables the inference of fuzzy quantities and relationships among them in any systems equipped with the constraint technology.

4.1 Descriptor of fuzzy number

Every fuzzy number $\tilde{F}$ can be decomposed into a family of intervals according to its λ-cuts: $\{\tilde{F}^{\lambda}|0\leq\lambda\leq 1\}$. But it is difficult to manipulate this family in a constraint software. If we replace it by the field $\bigcup_{\lambda\in[0,1]}\tilde{F}^{\lambda}\times\{\lambda\}$, then it turns out that this field can be easy described by means of a finite set of constraints.

Definition 5. *The descriptor $D(\tilde{F})$ of a fuzzy number $\tilde{F}$ is a structure $(x,\lambda)@C(x,\lambda)$, where $C(x,\lambda)$ is the constraint system restricting values of variables $x,\lambda\in\mathbb{R}$: $\{0\leq\lambda\leq 1,\underline{F}^{\lambda}\leq x\leq\overline{F}^{\lambda}\}$.*

Let us give, according to Definition 5, the descriptors of some fuzzy numbers very popular and convenient in applications, i.e. triangular $\tilde{F}=(a,b,c)$ and trapezoidal $\tilde{F}=(a,b,c,d)$ ones. Their descriptors take the following forms:

$$\begin{aligned}\text{triangular:}\quad & D(\tilde{F})=(x,\lambda)@\{0\leq\lambda\leq 1,\\ & a+(b-a)\lambda\leq x\leq c-(c-b)\lambda\},\\ \text{trapezoidal:}\quad & D(F)=(x,\lambda)@\{0\leq\lambda\leq 1,\\ & a+(b-a)\lambda\leq x\leq d-(d-c)\lambda\}.\end{aligned}$$

The special cases of fuzzy numbers, i.e. precise number $\tilde{F}=(a)$ and interval number $\tilde{F}=(a,b)$ have the following descriptors:

$$\begin{aligned}\text{precise:}\quad & D(\tilde{F})=(x,\lambda)@\{0\leq\lambda\leq 1,x=a\},\\ \text{interval:}\quad & D(\tilde{F})=(x,\lambda)@\{0\leq\lambda\leq 1,a\leq x\leq b\}.\end{aligned}$$

4.2 Descriptor of the fuzzy value of an arithmetical expression

In the deterministic case an arithmetical expression is defined by the following formula:

expression::=
number|(expression)|expression $\diamond$ expression,

where $\diamond \in \{+,-,\cdot,/\}$ are the ordinary binary operations. In view of the above formula the value v of any expression may be treated as a function E of numbers $a_1,\dots,a_n$ included in this expression, i.e. $v = E(a_1,\dots,a_n)$, E is a real-valued ordinary function with n variables.

In the fuzzy case a fuzzy arithmetical expression is defined by the following formula:

fuzzy expression::=
fuzzy number |(fuzzy expression)|
fuzzy expression $\tilde{\diamond}$ fuzzy expression,

where $\tilde{\diamond} \in \{\tilde{+},\tilde{-},\tilde{\cdot},\tilde{/}\}$ are extended binary operations, which are a consequence of applying the extension principle to the ordinary operations $\diamond \in \{+,-,\cdot,/\}$ (see Definition 2). Now the situation is different from the deterministic case. Problems arise, when we compute the fuzzy value of an expression with the extended operations by application of the above formula. Let us consider for instance expression $\tilde{A}\tilde{+}\tilde{B}\tilde{-}\tilde{B}$ and fuzzy numbers $\tilde{A} = (0,0)$ and $\tilde{B} = (1,2)$. We get the value of this expression equal to $(-1,1)$. Thus the result is far from the expected $(0,0)$. Similarly, we have $\tilde{B}\tilde{-}\tilde{B} \neq 0$. The reason of these problems is multiple occurrence of variable $\tilde{B}$ in the first and second fuzzy expression.

We define the fuzzy value $\tilde{V}$ in the following way. The fuzzy value $\tilde{V}$ is a function E of fuzzy numbers $\tilde{F}_1,\dots,\tilde{F}_n$, $\tilde{V} = E(\tilde{F}_1,\dots,\tilde{F}_n)$ (for the first fuzzy expression $E(\tilde{A},\tilde{B}) = \tilde{A}\tilde{+}\tilde{B}\tilde{-}\tilde{B}$). By direct application of the extension principle to $E(x_1,\dots,x_n)$ (to $E(a,b) = a+b-b$) we obtain the membership function of $\tilde{V}$:

$$\mu_{\tilde{V}}(x) = \sup_{\substack{x_1,\dots,x_n\in\mathbb{R}\\ x=E(x_1,\dots,x_n)}} \min\{\mu_{\tilde{F}_1}(x_1),\dots,\mu_{\tilde{F}_1}(x_n)\} \quad (2)$$

$$\left(\mu_{\tilde{A}\tilde{+}\tilde{B}\tilde{-}\tilde{B}}(x) = \sup_{\substack{a,b\in\mathbb{R}\\ x=a+b-b}} \min\{\mu_{\tilde{A}}(a),\mu_{\tilde{B}}(b)\}\right).$$

To cope with difficulty due to the multiple occurrence of a variable we suggest computing the descriptor of $\tilde{V}$. Further on in this section we will prove that the computed descriptor is consistent with (2). The descriptor of each fuzzy value includes variable representing possible realization of this value. When a value occurs in a fuzzy expression, then its realization represented by variable is constrained with variables representing values of remaining subexpressions.

We now show how the descriptor of $\tilde{V}$ of any expression may be systematically determined, even for complex expressions. Let $D(\tilde{V}_i) = (x_i,\lambda_i)@C_i(x_i,\lambda_i)$, for $i \in \{1,2\}$, be descriptors of fuzzy values $\tilde{V}_i$ of two expressions. In some cases $D(\tilde{V}_i) = D(\tilde{F}_i)$, since a fuzzy number $\tilde{F}_i$ is also an expression. Then the descriptor of the fuzzy value of expression $\tilde{V}_1\tilde{\diamond}\tilde{V}_2$ has the form:

$$\begin{aligned} D(\tilde{V}_1\tilde{\diamond}\tilde{V}_2) \quad = \quad & (x,\lambda)@C_1(x_1,\lambda_1)\cup C_2(x_2,\lambda_2) \\ & \cup\{\lambda=\lambda_1=\lambda_2, x=x_1\diamond x_2\}. \quad (3) \end{aligned}$$

Thus, application of formula (3) to $\tilde{V} = E(\tilde{F}_1,\dots,\tilde{F}_n)$ consists in imposing constraints. One can also notice that formula (3) implicitly constructs constraint $x = E(x_1,\dots,x_n)$. Hence, the following statement results directly.

Statement 1. *If $D(\tilde{F}_i)$, for $i = 1,\dots,n$, are the descriptors of fuzzy numbers $\tilde{F}_i$ and $D(\tilde{V})$ is the descriptor of $\tilde{V}$, $\tilde{V} = E(\tilde{F}_1,\dots,\tilde{F}_n)$, then the set of constraints $C(x,\lambda)$ of $D(\tilde{V})$ takes the form:*

$$\{0\le\lambda\le 1, \underline{F}_1^{\lambda_1}\le x_1\le\overline{F}_1^{\lambda_1},\dots,\underline{F}_n^{\lambda_n}\le x_n\le\overline{F}_n^{\lambda_n}, \lambda=\lambda_1=\dots=\lambda_n, x=E(x_1,\dots,x_n)\}.$$

To clarify (3) let us consider for instance fuzzy numbers $\tilde{F}_1 = (1,2,3)$ and $\tilde{F}_2 = (2,4,5)$. According to (3) the descriptor of their sum is equal to:

$$\begin{aligned} D(\tilde{F}_1\tilde{+}\tilde{F}_2) \quad &= \quad (x,\lambda)@\left\{\begin{array}{l} 0\le\lambda_1\le 1, \\ 1+\lambda_1\le x_1\le 3-\lambda_1, \\ \\ 0\le\lambda_2\le 1, \\ 2+2\lambda_2\le x_2\le 5-\lambda_2, \\ \\ \lambda=\lambda_1=\lambda_2, \\ x=x_1+x_2 \end{array}\right. \\ &= \quad (x,\lambda)@\left\{\begin{array}{l} 0\le\lambda\le 1, \\ 3+3\lambda\le x\le 8-2\lambda. \end{array}\right. \quad (4) \end{aligned}$$

If we simplify the above system, the constraints $\{0 \leq \lambda \leq 1, 3+3\lambda \leq x \leq 8-2\lambda\}$ describe triangular fuzzy number $\tilde{F} = (3,6,8)$.

Let us determine expression $\tilde{F}_1 \tilde{-} \tilde{F}_1$, where $\tilde{F}_1 = (1,2)$. On account of (3) the descriptor of $\tilde{F}_1 \tilde{-} \tilde{F}_1$ has form:

$$
\begin{aligned}
D(\tilde{F}_1 \tilde{-} \tilde{F}_1) &= (x,\lambda)@\left\{\begin{array}{l} 0 \leq \lambda_1 \leq 1, \\ 1 \leq x_1 \leq 2, \\ \\ 0 \leq \lambda_1 \leq 1, \\ 1 \leq x_1 \leq 2, \\ \\ \lambda = \lambda_1 = \lambda_1, \\ x = x_1 - x_1 \end{array}\right. \\
&= (x,\lambda)@\left\{\begin{array}{l} 0 \leq \lambda \leq 1, \\ x = 0. \end{array}\right.
\end{aligned}
$$

If we simplify and remove redundant constraints, we get the descriptor of fuzzy number $\tilde{F} = (0)$.

An important property of the descriptor of an expression is established by Proposition 1.

Proposition 1. *If $D(\tilde{F}_i)$, for $i = 1,\ldots,n$, are the descriptors of fuzzy numbers $\tilde{F}_i$ and $\tilde{V}$ is the fuzzy value of expression $E(\tilde{F}_1,\ldots,\tilde{F}_n)$, $\tilde{V} = E(\tilde{F}_1,\ldots,\tilde{F}_n)$, whose membership function is the result of the extension principle (see (2)), then a descriptor obtained by applying (3) is the descriptor of $\tilde{V}$.*

Proof. Assume that fuzzy number $\tilde{V}$ ($\tilde{V} = E(\tilde{F}_1,\ldots,\tilde{F}_n)$) has the membership function obtained according to (2). From Definition 5 we get the set of constraints $C(x,\lambda)$ of $D(\tilde{V})$

$$\{0 \leq \lambda \leq 1, \underline{V}^{\lambda} \leq x \leq \overline{V}^{\lambda}\}. \tag{5}$$

Theorem 1 gives the relationship between λ-cuts of $\tilde{V}$ and $\tilde{F}_1,\ldots,\tilde{F}_n$:

$$
\begin{aligned}
\tilde{V}^{\lambda} &= [\underline{V}^{\lambda},\overline{V}^{\lambda}] = E([\underline{F}_1^{\lambda},\overline{F}_1^{\lambda}],\ldots,[\underline{F}_n^{\lambda},\overline{F}_n^{\lambda}]) \\
&= \{x \,|\, \underline{F}_1^{\lambda} \leq x_1 \leq \overline{F}_1^{\lambda},\ldots,\underline{F}_n^{\lambda} \leq x_n \leq \overline{F}_n^{\lambda}, \\
&\qquad x = E(x_1,\ldots,x_n)\},\ \lambda \in [0,1]. \qquad (6)
\end{aligned}
$$

If we put (6) into constraint system (5), we will immediately obtain the set of constraints $C(x,\lambda)$ of the descriptor $D(\tilde{V})$ obtained by means of formula (3) (see Statement 1). This completes the proof. □

4.3 Descriptor of fuzzy constraint

Let us consider crisp constraint $E_1(x_1,\ldots,x_n) \propto E_2(x_1,\ldots,x_n)$, which we will denote by $\mathcal{C}$. Such defined constraint determines the set of solutions in $\mathbb{R}^n$, which satisfy constraint $\mathcal{C}$. Now we generalize crisp constraint $\mathcal{C}$ to the fuzzy case. Then a fuzzy constraint $\tilde{\mathcal{C}}$ $(E_1(\tilde{F}_1,\ldots,\tilde{F}_n) \tilde{\propto} E_2(\tilde{F}_1,\ldots,\tilde{F}_n))$ is a fuzzy set in $\mathbb{R}^n$ whose membership function is expressed by

$$\mu_{\tilde{\mathcal{C}}}(\boldsymbol{x}) = \min\{\mu_{\tilde{F}_1}(x_1),\ldots,\mu_{\tilde{F}_n}(x_n), c(x_1,\ldots,x_n)\}. \tag{7}$$

c is the characteristic function of the set of solutions, which satisfy constraint $\mathcal{C}$, i.e. $c(x_1,\ldots,x_n) = 1$ if $E_1(x_1,\ldots,x_n) \propto E_2(x_1,\ldots,x_n)$, 0 otherwise, $\propto \in \{<\ ,>,\leq,\geq=,\neq\}$ is the ordinary binary relation.

The membership function (7) describes the fuzzy set of solutions satisfying the fuzzy constraint $\tilde{\mathcal{C}}$.

We pass on to definition of the descriptor of a fuzzy constraint.

Definition 6. *The descriptor $D(\tilde{\mathcal{C}})$ of fuzzy constraint $\tilde{\mathcal{C}}$ $(E_1(\tilde{F}_1,\ldots,\tilde{F}_n) \tilde{\propto} E_2(\tilde{F}_1,\ldots,\tilde{F}_n))$ is a structure $(\boldsymbol{x},\lambda)@C(\boldsymbol{x},\lambda)$, where $\boldsymbol{x} = (x_1,\ldots,x_n)$ and $C(\boldsymbol{x},\lambda)$ is the set of constraints:*

$$
\begin{aligned}
&\{0 \leq \lambda \leq 1, \underline{F}_1^{\lambda} \leq x_1 \leq \overline{F}_1^{\lambda},\ldots,\underline{F}_n^{\lambda} \leq x_n \leq \overline{F}_n^{\lambda}, \\
&E_1(x_1,\ldots,x_n) \propto E_2(x_1,\ldots,x_n)\}.
\end{aligned}
$$

The descriptor of a fuzzy constraint can be easy determined in the following way.

Let $D(\tilde{V}_i) = (x_i,\lambda_i)@C_i(x_i,\lambda_i)$, for $i \in \{1,2\}$, be the descriptors of fuzzy values $\tilde{V}_i$ of two expressions. Then the descriptor of fuzzy constraint $\tilde{\mathcal{C}}$ $(\tilde{V}_1 \tilde{\propto} \tilde{V}_2)$ can be determined by means of the following formula:

$$
\begin{aligned}
D(\tilde{\mathcal{C}}) &= D(\tilde{V}_1 \tilde{\propto} \tilde{V}_2) \\
&= (\boldsymbol{x},\lambda)@C_1(x_1,\lambda_1) \cup C_2(x_2,\lambda_2) \qquad (8) \\
&\qquad \cup \{\lambda = \lambda_1 = \lambda_2, x_1 \propto x_2\},\ \boldsymbol{x} = (x_1,x_2).
\end{aligned}
$$

Let us illustrate (8). Descriptor (4) describes the sum of fuzzy numbers $\tilde{F}_1 = (1,2,3)$ and $\tilde{F}_2 = (2,4,5)$. If $\tilde{F}_3 = (5,7,9)$, then the descriptor of fuzzy constraint

$\tilde{F}_3 \tilde{=} \tilde{F}_1 + \tilde{F}_2$ is as follows

$$(\boldsymbol{x},\lambda)@\begin{cases} 0 \le \lambda_1 \le 1, \\ 1+\lambda_1 \le x_1 \le 3-\lambda_1, \\ \\ 0 \le \lambda_2 \le 1, \\ 2+2\lambda_2 \le x_2 \le 5-\lambda_2, \\ \\ 0 \le \lambda_3 \le 1, \\ 5+2\lambda_3 \le x_3 \le 9-2\lambda_3, \\ \\ \lambda = \lambda_1 = \lambda_2 = \lambda_3, \\ x_3 = x_1 + x_2, \end{cases}$$

where $\boldsymbol{x} = (x_1, x_2, x_3)$.

Our example demonstrates that determining the descriptor of a fuzzy constraint consists in imposing constraints, similarly as for the descriptor of a fuzzy value.

We conclude this section with an important property of the descriptor of a fuzzy constraint.

Proposition 2. *If $D(\tilde{V}_i) = (v_i, \lambda_i)@C_i(v_i, \lambda_i)$, for $i \in \{1,2\}$, are the descriptors of fuzzy values $\tilde{V}_i$ of expressions $E_i(\tilde{F}_1, \ldots, \tilde{F}_n)$ and $\tilde{C}$ is a fuzzy constraint, whose membership function is the result of (7), then a descriptor $D(\tilde{V}_1 \tilde{\propto} \tilde{V}_2)$ obtained by applying (8) is the descriptor of $\tilde{C}$.*

Proof. Suppose that $\tilde{C}$ is fuzzy constraint, whose membership function is obtained according to (7). By Definition 6, the descriptor of $\tilde{C}$ has the following system of constraints $C(\boldsymbol{x}, \lambda)$

$$\{0 \le \lambda \le 1, \underline{F}_1^{\lambda} \le x_1 \le \overline{F}_1^{\lambda}, \ldots, \underline{F}_n^{\lambda} \le x_n \le \overline{F}_n^{\lambda}, E_1(x_1, \ldots, x_n) \propto E_2(x_1, \ldots, x_n)\}.$$

Or equivalently:

$$\{0 \le \lambda \le 1, \underline{F}_1^{\lambda_1} \le x_1 \le \overline{F}_1^{\lambda_1}, \ldots, \underline{F}_n^{\lambda_n} \le x_n \le \overline{F}_n^{\lambda_n}, \lambda = \lambda_1 = \cdots = \lambda_n, E_1(x_1, \ldots, x_n) \propto E_2(x_1, \ldots, x_n)\}. \tag{9}$$

By assumption $D(\tilde{V}_i)$, for $i \in \{1,2\}$, are descriptors of fuzzy values $\tilde{V}_i$ of expressions $E_i(\tilde{F}_1, \ldots, \tilde{F}_n)$. From Statement 1 it follows that $C(v_i, \lambda_i)$, for $i \in \{1,2\}$, have the form:

$$\{0 \le \lambda \le 1, \underline{F}_1^{\lambda_1} \le x_1 \le \overline{F}_1^{\lambda_1}, \ldots, \underline{F}_n^{\lambda_n} \le x_n \le \overline{F}_n^{\lambda_n}, \lambda = \lambda_1 = \cdots = \lambda_n, v_i = E_i(x_1, \ldots, x_n)\}.$$

Hence, it can be easily seen that (9) is the set of constraints of descriptor $D(\tilde{C})$ obtained by applying (8) to $D(\tilde{V}_1)$ and $D(\tilde{V}_2)$. □

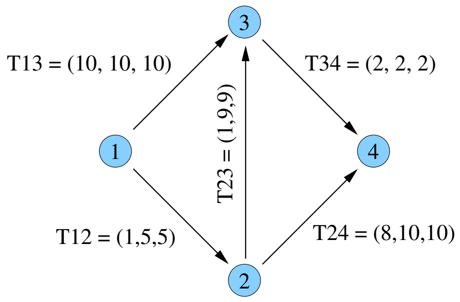

Figure 1: Network with five activities with triangular duration times.

5 Computational example

Let us consider network with five activities (see Figure 1). The duration time of each activity is given by means of triangular or precise number.

The following predicate computes the degree of possibility *POSS* that the project ends within *TIME* units of time:

```
completion(TIME, POSS) :-
 S13 $>= 0, S12 $>= 0, S23 $>= 0,
 S24 $>= 0, S34 $>= 0,
 fuzzy(T13, (10, 10, 10)),
 fuzzy(T12, (1, 5, 5)),
 fuzzy(T23, (1, 9, 9)),
 fuzzy(T34, (2, 2, 2)),
 fuzzy(T24, (8, 10, 10)),
 fuzzy(E1, 0),
 fuzzy(E2, E1+T12+S12),
 fuzzy(E3, E1+T13+S13),
 fuzzy(E3, E2+T23+S23),
 fuzzy(E4, E2+T24+S24),
 fuzzy(E4, E3+T34+S34),
 fuzzy(TOTAL, E4),
 fconstr(FC, [TOTAL], TOTAL =< TIME),
 poss(FC, POSS).
```

The following predicates from module `const_fuzzy` have been used:

`fuzzy(FN, FE)` defines descriptor `FN` of fuzzy number being the value of fuzzy expression `FE`,

`fconstr(FC, FV, FL)` defines descriptor `FC` of fuzzy constraint, given with the logical condition `FL` (`FV` is a list of fuzzy decision variables occurring in the condition `FL`),

`poss(FC, RN)` computes the degree of possibility `RN` of satisfaction of fuzzy constraint `FC`.

Variables T_{ij} and S_{ij} represent the duration time and

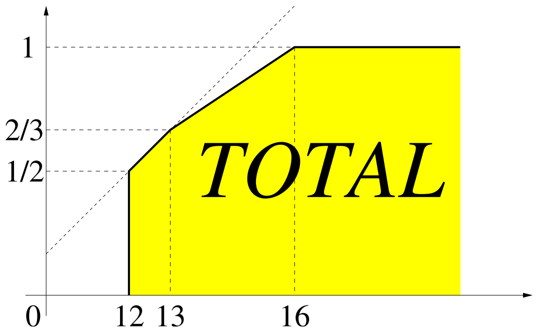

Figure 2: The membership function of $TOTAL$.

the slack of activity (i, j), respectively. Variables E_i express the fuzzy moments of occurrence of event i. Variable $TOTAL$ represents the possibility distribution $\mu_{TOTAL}(x)$ that the project ends within given x units of time.

The following dialog shows computing the degrees of possibility that the project ends within 12, 13 and 16 units of time (prefix `0r` expresses the rational number):

```
[user] ?- completion(12, POSS).
POSS = 0r1/2
[user] ?- completion(13, POSS).
POSS = 0r2/3
[user] ?- completion(16, POSS).
POSS = 1
```

In Figure 2 the membership function of $TOTAL$ is presented. The shaded area, under the graph, is bounded by a piecewise linear curve, which consists of three line segments. The number of line segments depends on the number of possible critical paths, which for certain projects, in the worst case, may be exponential. Owing to the descriptors, it is unnecessary to know the shape of this piecewise linear curve, since full information about it is included in the system of constraints, which describes its descriptor. Therefore, by applying polynomial algorithms for the linear programming problem, one can compute in polynomial time the degree of possibility of completion of project within a given number of units of time.

6 Conclusions

The proposed representation of fuzzy numbers and fuzzy constraints, in the form of descriptors, in natural way uses mechanism of constraint propagation in systems equipped with the constraint technology. The descriptor is a structure, in which the whole of history of all relationships among values or among constraints is recorded. Each fuzzy quantity has, in corresponding descriptor, variable representing its realization. Therefore, if a variable occurs in many expressions and constraints, then everywhere it is realized as the same real number.

The concept of descriptors is based on linear programming. For this reason it is impossible to model directly intractable problems, for instance float calculations in activity networks with fuzzy durations, which is NP-hard. However, taking into account the fact that constraint propagation, which is done during using descriptors, may cause failures and backtracks and thus it is possible to implement programs solving computationally hard problems by partial examination of a search tree.

References

[1] M. Dincbas, P. Van Hentenryck, H. Simonis, A. Aggoun, T. Graf, F. Berthier. The constraint logic programming language CHIP. In *Proceedings of the International Conference on Fifth Generation Computer Systems (FGCS'88)*, 693–702, Tokyo, Japan, December 1988.

[2] D. Dubois, H. Fargier, H. Prade. Fuzzy constraints in job-shop scheduling. *Intelligent Manufacturing*, 6:215–234, 1995.

[3] D. Dubois, H. Fargier, H. Prade. Possibility theory in constraint satisfaction problems: Handling priority preference and uncertainty. *Applied Intelligence*, 6:287–309, 1996.

[4] IF/Prolog V5.0. *Constraints Package*. Siemens Nixdorf Informationssysteme AG.

[5] J. Jaffar, S. Michayov, P.J. Stuckey, R.H.C. Yap. The $CLP(R)$ language and system. *ACM Transaction on Programming Languages and Systems*, 14:339–395, 1992.

[6] U. Nilsson, J. Małuszyński. *Logic, Programming and Prolog*, John Wiley & Sons, 1995 (second edition).

[7] H.T. Nguyen. A note on the extension principle for fuzzy sets. *J. Math. Anal. Appl.*, 68:369–380, 1978.

[8] L.A. Zadeh. *The concept of a linguistic variable and its application to approximate reasoning*, American Elsevier Publishing Co., 1973.