QUERY ANSWERING OVER FACT BASES IN ZADEH LOGIC

(This work is partially supported by RFBR grant № 12-01-00705)

Plesnievich G.S., Ph.D. (Tech.Sc.), professor
(National Research University “MPEI”, Krasnokazarmennaya St., 14, Moscow, 111250, Russian Federation, salve777@mail.ru)

Аннотация. Let L be Zadeh logic i.e. the fuzzy propositional logic based on triangular norm min(x, y). A fact in L is an expression of the form \( r \leq \varphi \leq s \) where \( \varphi \in L \) and \( 0 \leq r \leq s \leq 1 \). In a fuzzy interpretation \( I \) of L every fact is true or false, and \( I(\varphi) \in [0,1] \). If and only if the two-side inequality \( r \leq I(\varphi) \leq s \) is satisfied. Thus, the set FL of all facts in L defines a crisp logic with fuzzy interpretations. Logical consequence “\( \models \)” in the logic FL is defined as usual: for any set \( E \) of facts and any fact \( \alpha \), \( E \models \alpha \) if there are no an interpretation \( \lambda \) such that \( I(\alpha) = 1 \) and \( I(\beta) = 1 \) for all \( \beta \in E \). But in the logic FL there is also strong logical consequence \( \models^*: E \models^* \varphi \leq \psi \) if \( E \models \varphi \leq s \) and it is not true that \( E \models r \leq \varphi \leq s \) with \( r > r' \) and not true \( E \models r \leq \varphi \leq s \) with \( s < s' \). A fact base is a finite set \( F \) of facts: \( F = \{ r_1 \leq \theta \leq s_1 \mid 1 \leq i \leq n \} \). One can consider the set \( K = \{ \varphi \mid 1 \leq i \leq n \} \) as a fuzzy knowledge base and \( F \) as an instance of \( K \). A query is an expression of the form \( \varphi \psi \) where \( \psi \in L \). The answer to the query to the fact base \( F \) is the fact \( r \leq \psi \leq s \) such that \( F \models^* r \leq \psi \leq s \).

The problem of query answering over fact bases in FL can be solved by analytical tableau method. The method results in an algorithm with the exponential worst-case estimate (relatively to the size of \( F \cup \{ \kappa \} \) where \( \kappa \) is a query). However, consider the situation when the knowledge base \( K \) and the query \( \kappa \) are fixed but fact bases \( F \) are arbitrary instances of \( K \). Then, it is possible to answer query \( \kappa \) to fact bases \( F \) quickly. But preliminary we should deal with the parametric fact base associated with \( K \).

Under a parametric fact we mean an expression of the form \( a \varphi \leq b \psi \) where \( a \) and \( b \) are not numbers but parameters – variables with values in \([0,1]\). The parametric fact base associated with \( K \) is \( P = \{ a_1 \leq \varphi_1 \leq b_1, \ldots, a_n \leq \varphi_n \leq b_n \mid 1 \leq i \leq n \} \) where \( a_i, b_i \) are different parameters. Thus, if we replace the parameters by specific numbers from \([0,1]\) (with adherence to corresponding inequality) we obtain a specific fact base which is an instance of the parametric fact base \( P \). One can also consider query answering over parametric fact bases. Let \( P \) be a parametric fact base and \( \varphi \psi \) be a query. The answer to the query is the expression \( g \leq \psi \leq h \) such that \( K\lambda \models^* g\lambda \leq \psi \leq h\lambda \) for any substitution \( \lambda \) of numbers from \([0,1]\) for the parameters from \( K \). Here \( g \) and \( h \) are appropriate expressions with the parameters from \( K \).

Using the analytical tableau method, we show how to design an algorithm for finding the expressions \( g \) and \( h \) for a given knowledge base \( K \). So, let a knowledge base \( K = \{ \varphi \mid 1 \leq i \leq n \} \) and a query \( \kappa : \varphi \psi \) be fixed. Suppose we need to answer the query to any fact base \( F = \{ r_1 \leq \theta \leq s_1 \mid 1 \leq i \leq n \} \). Then we (1) apply the algorithm to the parametric fact base \( P = \{ a_1 \leq \varphi_1 \leq b_1, \ldots, a_n \leq \varphi_n \leq b_n \mid 1 \leq i \leq n \} \) and obtain the expressions \( g \) and \( h \); (2) apply the substitution \( \lambda = \{ a_i / r_i, b_i / s_i \mid 1 \leq i \leq n \} \) to \( g \) and \( h \); thus, we obtain the answer \( g\lambda \leq \psi \leq h\lambda \).

Ключевые слова: query, fact base, Zadeh logic, knowledge base, contrary condition, inequalities.

Introduction. Main definitions

Let L be Zadeh logic i.e. the propositional fuzzy logic based on a triangular norm min(x, y). The syntax of L is the same as the syntax of usual propositional logic. The semantics of L is defined by interpretations \( I: L \rightarrow [0,1] \) satisfying the following conditions for any formulas \( \varphi, \psi \in L \):

1. \( I(\neg \varphi) = 1 - I(\varphi) \), \( I(\varphi \land \psi) = \min\{I(\varphi), I(\psi)\} \),
2. \( I(\varphi \lor \psi) = \max\{1 - I(\varphi), 1 - I(\psi)\} \),
3. \( I(\varphi \rightarrow \psi) = \max\{1 - I(\varphi), I(\psi)\} \).

We associate with each formula \( \varphi \in L \) and numbers \( r, s \) the sentence (fact) \( r \leq \varphi \leq s \) which is true or false in \( I \), and \( I(\varphi) \in [0,1] \) is \( \Leftrightarrow_{df} r \leq \varphi \leq s \). (We also write \( \varphi \geq r \) instead of \( r \leq \varphi \leq 1 \), \( \varphi \leq r \) instead of \( 0 \leq \varphi \leq r \).) A fact base \( F \) is a finite set of facts.

Let FL denote the set of all facts for L. Thus, FL is a crisp logic with fuzzy interpretations. As in any logic, there is the logical consequence relation \( \models \) in FL: for any \( E \subseteq FL \) and \( \alpha \in FL \):

\[ E \models \alpha \Leftrightarrow_{df} \] there is no interpretation \( I \) such that \( I(\alpha) = 0 \) and \( I(\beta) = 1 \) for all \( \beta \in E \).

But in FL there is also strong logical consequence \( \models^* \):

\[ E \models^* r \leq \varphi \leq s \Leftrightarrow_{df} E \models r \leq \varphi \leq s \] and it is not true that \( E \models r' \leq \varphi \leq s \) with \( r' > r \) and \( E \models r \leq \varphi \leq s ' \) with \( r' < r \).
A knowledge base is a finite set K of formulas from L. K = {φi | 1 ≤ i ≤ n}. Let us choose numbers 0 ≤ r, s1 ≤ 1 (1 ≤ i ≤ n); then the fact base F = {ri ≤ φi ≤ si | 1 ≤ i ≤ n} is an instance of the knowledge base K. An expression of the form ?ψ is a query to the fact bases – instances of K if ψ is a formula of L in the signature of K. The answer to a query ?ψ to a fact base F is the fact r ≤ s such that F |= r ≤ s ≤ t.

Example 1. Let us consider the knowledge base K = {p ∧ q, q → r} and its instance (the fact base) F = {0.7 ≤ p ∧ q, 0.4 ≤ q → r ≤ 0.6}. Let κ = ?p ∧ ¬r be the query to F. Then 0.4 ≤ p ∧ ¬r ≤ 0.6 is the answer to κ.

Under a parametric fact we mean an expression of the form a ≤ φ ≤ b where a and b are not numbers but parameters – variables with values in [0, 1]. A parametric fact base P for a knowledge base K = {φi | 1 ≤ i ≤ n} is the set of parametric facts with different parameters: P = {ai ≤ φi ≤ bi | 1 ≤ i ≤ n} (ai ≠ bj if i ≠ j).

Let λ be a substitution numbers for parameters: λ = {ri, ai, bi | 1 ≤ i ≤ n} (ri ≤ si). Then, applying λ to P we obtain the fact base PL = {ai ≤ φi ≤ bi | 1 ≤ i ≤ n}. One can put a query ?ψ to the parametric fact base P for a knowledge base K. Then the answer to this query is the expression g ≤ ψ ≤ h such that Kλ |= g ≤ ψ ≤ hλ for any substitution λ where g and h are some expressions containing parameters from K.

Example 2. Let the knowledge base K and the query κ be the same as in Example 1. The parametric fact base for the knowledge base is P = {a ≤ p ∧ q ≤ b, c ≤ q → r ≤ d}. Then g ≤ p ∧ ¬r ≤ h is the answer to κ where
g = min{a, 1 – d}, 
h = case[max{b, 1 - c} if a+c > 1 ∧ b+d > 1, b if a+c ≤ 1 ∧ b+d > 1, 1-c if a+c > 1 ∧ b+d ≤ 1, 0 if a+c ≤ 1 ∧ b+d ≤ 1].

Query answering to fact bases

Let M be the set of all sentences from FL of the forms a ≤ c, a < c, a ≥ c and a > c. Obviously, the problem of logical consequence for logic FL is reduced to the problem of inconsistency for logic M since F |= r ≤ a ≤ s ⇔ F ∪ {a > r} and F ∪ {r < s} are inconsistent sets where F* = {β ≤ r, β ≥ r | r ≤ β ≤ s} ∈ F).

The method of analytical tableau can be applied to solve the problem of inconsistency in M [2]. In Table 1 there are the inference rules for the logic M. This method also can be applied to the problem of finding answers to queries to fact bases. We show by example, how to do it.

Table 1

<table>
<thead>
<tr>
<th>¬φ ≥ c</th>
<th>¬φ ≤ c</th>
<th>¬φ &gt; c</th>
<th>¬φ &lt; c</th>
</tr>
</thead>
<tbody>
<tr>
<td>φ ≤ 1 – c</td>
<td>φ ≥ 1 – c</td>
<td>φ &gt; 1 – c</td>
<td>φ &gt; 1 – c</td>
</tr>
<tr>
<td>φ ∧ ψ ≥ c</td>
<td>φ ∧ ψ ≤ c</td>
<td>φ ∧ ψ &gt; c</td>
<td>φ ∧ ψ &lt; c</td>
</tr>
<tr>
<td>φ ≥ c</td>
<td>φ ≤ c</td>
<td>ψ ≤ c</td>
<td>ψ &gt; c</td>
</tr>
<tr>
<td>ψ ≥ c</td>
<td>ψ &lt; c</td>
<td></td>
<td></td>
</tr>
<tr>
<td>φ ∨ ψ ≥ c</td>
<td>φ ∨ ψ ≤ c</td>
<td>φ ∨ ψ &gt; c</td>
<td>φ ∨ ψ &lt; c</td>
</tr>
<tr>
<td>φ ≥ c</td>
<td>ψ ≥ c</td>
<td>ψ ≤ c</td>
<td>ψ &gt; c</td>
</tr>
<tr>
<td>ψ ≤ c</td>
<td>ψ &lt; c</td>
<td></td>
<td></td>
</tr>
<tr>
<td>φ → ψ ≥ c</td>
<td>φ → ψ ≤ c</td>
<td>φ → ψ &gt; c</td>
<td>φ → ψ &lt; c</td>
</tr>
<tr>
<td>φ ≤ 1 – c</td>
<td>ψ ≥ c</td>
<td>ψ ≥ 1 – c</td>
<td>ψ &lt; c</td>
</tr>
<tr>
<td>ψ ≤ c</td>
<td>ψ &lt; c</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example 3. Let the knowledge base K, the fact base F and the query κ be such as in Example 1. The fact base F can be replaced with the equivalent fact base F* = {p ∧ q ≥ 0.7, q → r ≥ 0.4, q → r ≤ 0.6}. Then F* |= x ≤ p ∧ ¬r ≤ y if and only if the following two sets are inconsistent:
F_1=\{p \land q \geq 0.7, q \rightarrow r \geq 0.4, q \rightarrow r \leq 0.6, p \land \neg r \leq x\},
F_2=\{p \land q \geq 0.7, q \rightarrow r \geq 0.4, q \rightarrow r \leq 0.6, p \land \neg r > y\}.

In Figure 1 the deduction trees A and B for these sets are shown. In A the branch (1) is closed since it contains contrary inequalities \(q \geq 0.7\) and \(q \leq 0.6\). The branch (2) will be closed if we choose \(y\) such that inequalities \(r < 1-y\) and \(r \geq 0.4\) becomes contrary. Also (2) will be closed if \(y=1\) (then \(p > 1\) and that is impossible). Hence, (2) is closed if and only if \(y=1\) or \(0.4 \geq 1-y\), i.e., \(y \geq 0.6\). Therefore, \(h = \min y = 0.6\). In B branches (1) and (2) are closed since they contain contrary inequalities \(q \geq 0.7\) and \(q \leq 0.6\). Clearly, branch (3) is closed if and only if \(1-x = 1\), i.e., \(x = 0\). Therefore, \(g = 0\). Hence, \(p \land \neg r \leq 0.6\) is the answer to \(\kappa\).

**Figure 1. Deduction trees**

A
- [1] \(p \land q \geq 0.7\)
- [5] \(q \rightarrow r \geq 0.4\)
- [2] \(q \rightarrow r \leq 0.6\)
- [3] \(p \land \neg r > y\)
  - \(p \geq 0.7\)
  - \(q \geq 0.7\)
  - \(q \geq 0.4\)
  - \(r \leq 0.6\)
- [4] \(\neg r > y\)
- \(r < 1-y\)
- \(q \leq 0.6\)
- \(r \geq 0.4\)

B
- [1] \(p \land q \geq 0.7\)
- [4] \(q \rightarrow r \geq 0.4\)
- [2] \(q \rightarrow r \leq 0.6\)
- [5] \(p \land \neg r < x\)
  - \(p \geq 0.7\)
  - \(q \geq 0.7\)
  - \(q \geq 0.4\)
- \(p > y\)
- \(r < 1-y\)
- \(p < x\)
- \(r < x\)
- \(r > 1-x\)
- \(r > 1-x\)

**Remark.** The similar method has been offered in [1].

**Query answering for parametric fact bases**

One can find answers to queries to parametric fact bases by applying analytical tableaux method. Here is an example (how to do it).

**Example 4.** Let the knowledge base \(F\), the query \(\kappa\) and the parametric fact base \(P\) are such as in Example 2. In Figure 2 and Figure 3 the deduction trees for the sets \(P_1 = P \cup \{p \land \neg r > y\}\) and \(P_2 = P \cup \{p \land \neg r < x\}\) are shown.

In the first tree, consider two inequalities \(q \geq a\) and \(q \leq 1-c\) which lie on branch (1). Clearly, they are contrary (inconsistent) if and only if \(a > 1-c\), i.e., \(a+c > 1\). We say that \((q \geq a, q \leq 1-c)\) is a **candidate contrary pair** and \(a+c > 1\) is a condition of its contrariness.

In Table 2 there are all the candidate contrary pairs together with the contrariness conditions and with references to the branches closed by the contrary pairs. From the table we see that pairs 1 and 4 (and also pairs 2 and 3) block up all branches of the tree. Therefore, the tree in Figure 2 is closed if and only if the following condition is satisfied:

\((a+c > 1 \land y \geq 1-c) \lor (b+d > 1 \land y \geq b)\) \hspace{1cm} (1)

**Table 2**

<table>
<thead>
<tr>
<th>No.</th>
<th>(A)</th>
<th>(B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>((q \leq 1-c, q \geq a))</td>
<td>(a+c &gt; 1)</td>
</tr>
<tr>
<td>2</td>
<td>((q \leq b, q \geq 1-d))</td>
<td>(b+d &gt; 1)</td>
</tr>
<tr>
<td>3</td>
<td>((p \leq b, p \geq y))</td>
<td>(y \geq b)</td>
</tr>
<tr>
<td>4</td>
<td>((q \leq 1-c, q \geq a))</td>
<td>(y \geq 1-c)</td>
</tr>
</tbody>
</table>

So, \(h = \min \{y\ | \ y \text{ satisfies (1)}\}\). Depending on what conditions \(a+c > 1\) and \(b+d > 1\) are true or false the condition (2.1) is reduced to:

- \(y \geq 1-c \lor y \geq b\) if \(a+c > 1\) and \(b+d > 1\),
- \(y \geq b\) if \(a+c \leq 1\) and \(b+d > 1\),
\[ y \geq 1 - c \quad \text{if} \quad a + c > 1 \text{ and } b + d \leq 1, \]
\[ 0 \quad \text{if} \quad a + c \leq 1 \text{ and } b + d \leq 1. \]

From this we obtain
\[
h = \begin{cases} 
\max \{b, 1 - c\} & \text{if } a + c > 1 \text{ and } b + d > 1, \\
b & \text{if } a + c \leq 1 \text{ and } b + d > 1, \\
1 - c & \text{if } a + c > 1 \text{ and } b + d \leq 1, \\
0 & \text{if } a + c \leq 1 \text{ and } b + d \leq 1.
\end{cases}
\]

In a similar way, considering the tree in Figure 3, we find:
\[ g = \min \{a, 1 - d\}. \]
Example 5. Let the knowledge base $K$, the parametric fact base $P$, the query $\kappa$, and the expressions $g$ and $h$ be such as in Example 4. Let us take the substitution $\lambda = \{ a/0.7, b/1, c/0.4, d/0.6 \}$. Applying $\lambda$ to $P$, $g$ and $h$, we obtain

$$P_\lambda = \{ 0.7 \leq p \land q, 0.4 \leq q \rightarrow r \leq 0.6 \}, g_\lambda = 0.4, h_\lambda = 0.4.$$  

Thus, we have the answer $0.4 \leq p \land \neg r \leq 0.6$ to the query $p \land \neg r$ to the fact base $F$.

Let us consider a general situation when a knowledge base $K$, a query $\kappa$ and a parametric fact base $P$ for $K$ are arbitrary. Let $P$ have parameters $a_i$ ($i = 1, 2, \ldots$). Suppose we construct the deduction trees $T_1$ and $T_2$ for the sets $P \cup \{ \kappa < x \}$ and $P \cup \{ \kappa > y \}$. It is easy to see that in $T_1$ there can be candidate pairs and contrary conditions of the following forms:

$$(p \geq a_r, p \leq a_l), \quad (p \geq 1 - a_r, p \leq a_l), \quad (p \geq a_r, p \leq 1 - a_l), \quad (p \geq 1 - a_r, p \leq 1 - a_l),$$

$$(p \geq a_r, p < x), \quad x \leq a_r, \quad x \leq 1 - a_r, \quad p = a_r, \quad x \leq 0.5.$$  

In the first tree, consider two inequalities $q \geq a$ and $q \leq 1 - c$ which lie on branch (1). Clearly, they are contrary (inconsistent) if and only if $a > 1 - c$ i.e.

Every candidate pair in $T_1$ blocks some branches. Let $b(\pi)$ denote the set of branches which is blocked up by pair $\pi$, and let $c(\pi)$ denote the contrary condition for pair $\pi$. Also, let $b(S) = U \{ b(\pi) | \pi \in E \}$ and $c(S) = \Lambda \{ c(\pi) | \pi \in S \}$ where $S$ is a set of candidate pairs.

A set $S$ of candidate pairs is a covering if $b(S)$ coincides with the set of all branches of $T_1$. Thus, if $S$ is a covering and the condition $S$ is satisfied with a given substitution $\sigma$ then the tree $T_1|\sigma$ is closed. A covering $S$ is (locally) minimal if $S \setminus \{ \pi \}$ is not a covering for each $\pi \in S$.

Let $S_1, S_2, \ldots, S_n$ be all minimal coverings for $T_1$. Take the condition $C = c(S_1) \lor c(S_2) \lor \ldots \lor c(S_n)$. Thus, $C$ is a disjunction of conjunctions made of inequalities of the form: $a_j > a_k, a_j + a_k < 1, a_j + a_k > 1, x \leq a_j, x \leq 1 - a_p, x \leq 2$.

Let $R$ be the set of all conditions which are occurred in $C$ and have no variable $x$. Let 0 be any assignment of truth values 0 or 1 to the conditions from $R$, i.e. $0 : R \rightarrow \{ 0, 1 \}$. One can consider $0$ as a substitution truth values for inequalities. Thus, $C_0$ has the form $C_0^1 \lor C_0^2 \lor \ldots \lor C_0^m$ where $C_0^i = c(S_i)$ and $C_0^i$ has the form $(x \leq e_{i1}) \lor (x \leq e_{i2}) \lor \ldots \lor (x \leq e_{im})$. Let us denote $r(C_0^i) = \{ e_{i1}, e_{i2}, \ldots, e_{im} \}$.

It is clear that:

$$(x \leq e_{i1}) \land (x \leq e_{i2}) \land \ldots \land (x \leq e_{iim})$$

$$(x \geq a_{i1}) \lor (x \leq a_{i2}) \lor \ldots \lor (x \leq a_{iim})$$

$$(x \leq \min r(C_0^i))$$

$$(x \geq \max r(C_0^i)).$$

Let $0^* \theta$ be the conjunction of the inequalities from $R$ or their negations. We include in $0^* \theta$ an inequality if $0 \theta$ assigns 1, and the contrary inequality if $0^* \theta$ assigns 0, i.e.

$$0^* \theta = (\Lambda \{ \sigma | \sigma \in R, \theta(\sigma) = 1 \}) \lor (\Lambda \{ \neg \sigma | \sigma \in K, \theta(\sigma) = 0 \}).$$

We have

$$C = \Lambda \{ 0^* \theta | 0 : R \rightarrow \{ 0, 1 \} \}.$$  

(3)

This can be understood considering the following example.

Example 6. Let $\alpha$ be a formula of propositional variables $p, q$ and $r$: $\alpha = \alpha [p, q, r]$. Then

$$\alpha [p, q, r] = (p \land q \rightarrow \alpha [1, 1, r]) \land (p \land \neg q \rightarrow \alpha [1, 0, r]) \land \neg (p \land q \rightarrow \alpha [0, 1, r]) \land \neg (p \land \neg q \rightarrow \alpha [0, 0, r]).$$

Indeed, for example, if $p = 0, q = 1$ then in the right part of this equality we have $\alpha [0, 1, r]$. Hence, the equality is true for $p = 0, q = 1$.

From (2) and (3) we obtain

$$g = \max \{ x | x \text{ satisfies } C \}$$

$$= \text{case}\{ \min \{ \min r(C_0^i) | 1 \leq j \leq m \} \text{ if } 0^* \theta | 0 : R \rightarrow \{ 0, 1 \} \}.$$  

In the similar way we obtain the expression for $h$ (see Table 3).

Table 3

$$g = \text{case}\{ \max \{ \min r(C_0^i) | 1 \leq j \leq m \} \text{ if } 0^* \theta | 0 : R \rightarrow \{ 0, 1 \} \}.$$  

5
\( h = \text{case}\{\max\{r(C^j)|1 \leq j \leq m\} \text{ if } 0^R \rightarrow [0, 1]\} \)

**Example 7.** In the tree \( T_2 \) (Figure 3) there are the candidate pairs which are written in Table 3. From here \( R = \{a+c>1, b+d>1\} \). There are exactly four substitutions

- \( \theta_1 = \{1/(a+c>1), 1/(b+d>1)\} \)
- \( \theta_2 = \{1/(a+c>1), 0/(b+d>1)\} \)
- \( \theta_3 = \{0/(a+c>1), 1/(b+d>1)\} \)
- \( \theta_4 = \{0/(a+c>1), 0/(b+d>1)\} \)

Then we have

- \( \theta_1^* = (a+c>1) \land (b+d\leq1) \)
- \( \theta_2^* = (a+c>1) \land (b+d\leq1) \)
- \( \theta_3^* = (a+c\leq1) \land (b+d>1) \)
- \( \theta_4^* = (a+c\leq1) \land (b+d\leq1) \)

For the condition

\( C = (a+c>1 \land y \geq 1-c) \lor (b+d>1 \land y \geq b) \)

we have

- \( C_{\theta_1} = ((y \geq 1-c) \lor (y \geq b)) \)
- \( C_{\theta_2} = (y \geq 1-c) \)
- \( C_{\theta_3} = (y \geq b) \)
- \( C_{\theta_4} = 0 \)

Hence,

\( r(C_{\theta_1}) = \{1-c, b\}, r(C_{\theta_2}) = \{1-c\}, r(C_{\theta_3}) = \{b\}, r(C_{\theta_4}) = \{\} \)

Using the formula presented in the second row of Table 3 we obtain the expression \( g \) coinciding with what is presented in Example 2.

**References**
