Multi-adjoint logic semantics for a practical fuzzy tool

Hannes Strass, Susana Munoz-Hernandez, and Víctor Pablos Ceruelo

Universidad Politécnica de Madrid
hannes.strass@alumnos.upm.es, {susana,vpablos}@fi.upm.es
http://babel.ls.fi.upm.es/

Appendix

Proposition (\(T_P\) is continuous). Let \(P\) be a well-defined fuzzy logic program and \(I_0 \sqsubseteq I_1 \sqsubseteq \ldots\) a countably infinite increasing sequence of interpretations. Then

\[
T_P \left( \bigcup_{n=0}^{\infty} I_n \right) = \bigcup_{n=0}^{\infty} T_P(I_n).
\]

Proof. For ease of reading, define

\[
I := \bigcup_{n=0}^{\infty} I_n, \quad I' := T_P(I), \quad J'_n := T_P(I_n), \quad J' := \bigcup_{n=0}^{\infty} J'_n.
\]

and let \(A \in B\) and \(I'(A) = x, J'(A) = x'\). We show \(x = x'\).

\(x = zv\) where \(z \in \{\nabla, \blacklozenge\}\): Then there is a clause \(A \leftarrow F.c B_1, \ldots, B_n \in \text{ground}(R)\) with \(I(B_i) = z_i v_i \succ \bot\) for all \(i\) and \(zv \succ (z_1 \circ \cdots \circ z_n) F.c(c, F(v_1, \ldots, v_n))\). By definition of \(I\) and since \(I_0 \sqsubseteq I_1 \sqsubseteq \ldots\) is increasing, there exists an \(m\) such that \(I_m(B_i) = z_i v_i\) for all \(i\) and also \(I_m(B_i) = I_{m'}(B_i)\) for all \(m' > m\). Since \(T_P\) is monotonic, the sequence \(T_P(I_0) \sqsubseteq T_P(I_1) \sqsubseteq \ldots\) is also increasing. By definition of \(T_P\), we hence have \(J'_m(A) = x\) and \(J'_{m'}(A) = x\) for all \(m' > m\). Applying the definition of \(J'\) yields \(x = x'\).

\(x = \nabla v\): Then no program clause that uses ground atoms from \(\text{Dom}(I)\) is applicable to \(A\) and there exists a default value declaration \(\text{default}(p(X_1, \ldots, X_n)) = [\delta_1 \text{ if } \varphi_1, \ldots, \delta_m \text{ if } \varphi_m] \in D\) and a valuation \(\sigma\) such that \(A = \sigma(p(X_1, \ldots, X_n))\), \(\sigma(\varphi_j)\) holds and \(v = \delta_j\). By definition of \(I\), there is also no clause applicable to all of the \(I_k\), the same default rule applies and thus \(J'_k(A) = x\). Again, applying the definition of \(J'\) yields \(x = x'\).

Lemma. Let \(P\) be a well-defined fuzzy logic program. Then

\[
I \models P \text{ iff } T_P(I) \sqsubseteq I.
\]

Proof. “if”: Let \(I' := T_P(I) \sqsubseteq I\).


Let \( k \rightarrow 1 \): A computation tree of height 1 consists of just one node, this is either a

\[ t \in \text{ground}(R) \]  

with \( I(B_i) = z_i v_i \approx \bot \) for all \( i \). Then by definition of \( T_P \) we have \( I'(A) = z' v' \) where \( z' = z_1 \circ \cdots \circ z_n \in \{ \nabla, \diamond \} \) and \( v' = F_c(c, F(v_1, \ldots, v_n)) \). From \( I' \subseteq I \) we get \( z' v' \preceq I(A) \).

\( I \vdash D \): Let \( \text{default}(p(X_1, \ldots, X_n)) = [\delta_1 \text{ if } \varphi_1, \ldots, \delta_m \text{ if } \varphi_m] \in D \), \( \sigma \) be a

valuation, and let \( A := \sigma(p(X_1, \ldots, X_n)) \) be well-typed. Since \( P \) is well-defined, there exists a \( j \) such that \( \sigma(\varphi_j) \) holds and thus \( I'(A) = \triangle v' \). From \( I' \subseteq I \), we again get \( \triangle v' \preceq I(A) \).

“only if”: Let \( I \vdash P \) and \( A \in \overline{B} \) be an arbitrary ground atom. We show \( I'(A) \preceq I(A) \).

\[ I'(A) = z' v' \]  

where \( z' \in \{ \nabla, \diamond \} \): Then there is a clause \( A^{c,F} \vdash B_1, \ldots, B_n \in \text{ground}(R) \) with \( I(B_i) = z_i v_i \approx \bot \) for all \( i \) and \( z' v' = (z_1 \circ \cdots \circ z_n) F_c(c, F(v_1, \ldots, v_n)) \). From \( I \vdash R \) we can now conclude \( z' v' = (z_1 \circ \cdots \circ z_n) F_c(c, F(v_1, \ldots, v_n)) \approx z v \).

\[ I'(A) = \triangle v' \]  

Then there is a default value declaration \( \text{default}(p(X_1, \ldots, X_n)) = [\delta_1 \text{ if } \varphi_1, \ldots, \delta_m \text{ if } \varphi_m] \in D \) and a valuation \( \sigma \) such that \( \sigma(\varphi_j) \) holds and \( v' = \delta_j \). Since \( I \vdash D \), we have \( I'(A) = \triangle \delta_j \preceq I(A) \).

\[ \square \]

**Lemma.** Let \( P \) be a well-defined fuzzy logic program, \( A \in \overline{B} \) be a ground atom.

If there exists a successful computation ending in a computation tree \( t \) of height \( k \) with root node \( A, v \), then \( z v \preceq T_P \uparrow k(\bot)(A) \).

**Proof.** We use induction on \( k \).

\( k = 1 \): A computation tree of height 1 consists of just one node, this is either a

\( C - \) or a \( D - \) node.

\( t = C(A, v) \): Then there exists a fuzzy fact \( A' \leftarrow v' \in R \) such that \( A \) is a

well-typed ground instance of \( A' \) and \( v = v' \). From the definition of \( T_P \) we can conclude \( z v = v \preceq T_P \uparrow 1(\bot)(A) \).

\( t = D(A, v) \): Then there exists a default value declaration \( \text{default}(p(X_1, \ldots, X_n)) = [\delta_1 \text{ if } \varphi_1, \ldots, \delta_m \text{ if } \varphi_m] \in D \) and a substitution \( \mu \) such that \( A = p(X_1, \ldots, X_n) \mu \) is well-typed and \( \varphi_j \mu \) holds. Again, by the definition of \( T_P \), we get \( z v = \triangle \delta_j \preceq T_P \uparrow 1(\bot)(A) \).

\( k \rightarrow k + 1 \): We look at the root node \( C(A, v) \) and assume w.l.o.g. that it has exactly \( n \) children \( B_1, v_1 \), \( 1 \leq i \leq n \). By definition of the transition rules there exists a program clause \( A^{c,F} \vdash B_1', \ldots, B_n' \in R \) and a substitution \( \mu = \text{mgu}(A, B_1, \ldots, B_n, A', B_1', \ldots, B_n') \) and \( v = F_c(c, F(v_1, \ldots, v_n)) \).

Now, all the nodes \( B_i, v_i \) are roots of computation trees \( t_i \) of height \( k_i \leq k \). By the induction hypothesis, \( z_{t_i} v_i \preceq T_P \uparrow k_i(\bot)(B_i) \). Since the sequence \( \bot \subseteq T_P(\bot) \subseteq T_P(\uparrow k_i(\bot)) \subseteq \cdots \) is increasing, we have that \( k_i \leq k \) implies \( T_P \uparrow k_i(\bot) \subseteq T_P \uparrow k(\bot) \) and therefore \( z_{t_i} v_i \preceq T_P \uparrow k(\bot)(B_i) \preceq T_P \uparrow k(\bot)(B_i) \preceq T_P \uparrow k(\bot)(B_i) = z v' \). We can now apply the definition of \( T_P \), that sets \( T_P \uparrow k + 1(\bot)(A) = \nabla \circ z_1' \circ \cdots \circ z_n' F_c(c, F(v_1', \ldots, v_n')) = z' v' \).

**Finally,** \( z_{t_i} v_i \preceq z' v' \) for all \( i \) (induction hypothesis) and monotonicity of \( F_c, F \), and \( \circ \) shows \( z v \preceq z' v' \).

\[ \square \]