FUZZY TEMPORAL PROLOG

TU VAN LE
Faculty of Information Sciences & Engineering, University of Canberra
PO Box 1, Belconnen, ACT 2616, Australia
E-mail: vanle@ise.canberra.edu.au

ABSTRACT

Fuzzy and temporal logics are combined to produce a form of Horn-clause logic programming system which is capable of processing the rules that contain fuzzy times and that have their certainty varying with time. The system’s operational and model-theoretic semantics are described and the system’s soundness is established.

1. Introduction

During the past ten years, there have been several attempts to incorporate fuzzy logic into Prolog to make the system capable of handling imprecise knowledge and queries (Atanassov and Georgiev⁴, Le⁷,⁸, Liu and Li⁹–¹¹, Martin et al.¹²). On the other hand, temporal logic has been also used as the framework for some logic programming systems that accept temporal information in the form of tensed expressions (Gabbay³, Galton¹, Hale⁵). Other methods of representing and matching fuzzy times are found in (Dubois and Prade², Hirschman and Story⁶, Nokel¹³, Qian¹⁴), which have applications mainly in fuzzy rules-based systems. So far, however, there have been no attempts to combine fuzzy logic and temporal logic into some form of logic programming system that can handle time-dependent knowledge such as the following:

“If the rotational speed of the compressor is high, then it is likely now that the reactor’s temperature will start to increase after around 3 minutes.”

This rule contains two time-dependent imprecise expressions. The first one is the term “likely now”, which means the truth degree of the rule varies with time (i.e. it is a fuzzy subset of the time domain), and the second one is “after around 3 minutes”, which represents a fuzzy time gap.

In this paper, I present an extended Prolog system that incorporates both fuzzy logic and temporal logic, and allows the above mentioned sentence to be represented by a rule of the form $A \rightarrow^\alpha F\lambda B$, where $\alpha$ symbolizes a fuzzy subset of the time domain, $\lambda$ is a fuzzy number (i.e., a convex normalised single modal-valued fuzzy subset of the set of numbers), and $F$ is the classical temporal connective expressing “it will be true”. This rule is read as “If $A$ holds, then it is $\alpha$-likely that $B$ will hold after $\lambda$ time units from now”. The idea behind the use of time-dependent fuzzy rules is that in reality, something may be more true
now than at the next moment.

The paper is organised as follows. The next two sections respectively describe the syntax of FT-Prolog programs and their model-theoretic semantics. Section 4 discusses the computational aspect of FT-Prolog, and Section 5 establishes the soundness of the system.

2. Syntax of an FT-Prolog Program

Throughout this paper, I assume the time domain \( T = (Z, <, +) \), where \( Z \) is the set of all integers, and \(<, +\) are the usual symbols for “less than” relation and “addition” function.

**Definition 2.1**

1. An FTP-program is a finite set of FTP-clauses.
2. An FTP-clause has the form \( A \not\subseteq B \), where \( A \) is a head, \( B \) is a body, and \( \alpha \) is a symbol representing a fuzzy set in the time domain \( T \).
3. A head is a literal, and a body is either empty or a conjunction of literals.
4. A literal has one of the forms \( A, FA, PA, F\lambda A, P\lambda A \), where \( A \) is an atomic formula, \( F \) and \( P \) are the classical temporal connectives for “future” and “past” respectively, and \( \lambda \) is a symbol representing a fuzzy integer.

Note immediately that the classical always clause \( \Box (A \leftarrow B) \) and current clause \( A \leftarrow B \) in temporal logic are particular cases of FTP-clauses. In fact, if \( \chi_S \) denotes the characteristic function of the set \( S \), we have:

\[
\Box (A \leftarrow B) \equiv A \not\subseteq B
\]

\[
A \leftarrow B \equiv A \chi_{\{t_0\}} B
\]

where \( t_0 \) is some particular time which is chosen as the current time. For simplicity of notation, we write 1 in place of \( \chi_T \).

**Definition 2.2** An FTP-query has the form \( ?Q \) and is read as “When and how likely is \( Q \) true?”, where \( Q \) is a conjunction of literals.

3. Model-theoretic Semantics of an FT-Prolog Program

Let \( \mathcal{P} \) be an FTP-program (i.e., a finite set of FTP-clauses). The Herbrand base of \( \mathcal{P} \) is denoted by \( \mathcal{H}(\mathcal{P}) \) and is defined to be the set of all ground atomic formulas (i.e., formulas containing no variables) that can be formed by the predicate, function, and constant symbols in \( \mathcal{P} \).
Definition 3.1 A Herbrand interpretation of $\mathcal{P}$ is a mapping $I : \mathcal{H}(\mathcal{P}) \to [0, 1]^T$, which assigns to each ground atomic formula a fuzzy subset of the time domain $T$.

Intuitively, an interpretation $I$ of $\mathcal{P}$ assigns a truth degree (from 0 to 1) to each ground atomic formula $A \in \mathcal{H}(\mathcal{P})$ at each time instance $t \in T$. Thus, $I(A)$ is a fuzzy subset of the time domain $T$, and we denote its membership function by $\mu_A$. That is, $I(A)(t) = \mu_A(t)$, for every $t \in T$.

Definition 3.2 Let $\mathcal{M}$ be a Herbrand interpretation of $\mathcal{P}$. Then, by definition:

1. $\mathcal{M}$ is a (Herbrand) model of $\mathcal{P}$ (notation $\mathcal{M} \models \mathcal{P}$) if $\mathcal{M}$ is a model of every clause $C$ in $\mathcal{P}$ (notation $\mathcal{M} \models C$).

2. $\mathcal{M} \models C$ iff $\mathcal{M} \models C^\theta$, for every ground instance $C^\theta$ of $C$.

3. $\mathcal{M} \models C^\theta$ iff $\mathcal{M}_t \models C^\theta$, for every $t \in T$.

Here, the notation $\mathcal{M}_t \models C^\theta$ is read as "$C^\theta$ holds in $\mathcal{M}$ at time $t". The meaning of this is given in the following definition.

Definition 3.3 Let $A$ be a ground atomic formula in $\mathcal{H}(\mathcal{P})$, $B_1$ be any ground literal, $\alpha$ and $\beta$ represent two fuzzy subsets of the time domain $T$, $\lambda$ be a fuzzy integer, and $H \preceq B$ be a ground instance of a clause in $\mathcal{P}$. Also let $\mathcal{M}$ be a Herbrand interpretation of $\mathcal{P}$ which is associated with a membership function $\mu$, and let $t \in T$. Then, by definition (for simplicity, we write $\exists s > t$ to mean $\exists s \in T \ s > t$):

1. $\mathcal{M}_t \models A \preceq$ iff $\mu_A(t) \geq \alpha(t)$.

2. $\mathcal{M}_t \models FA \preceq$ iff $\exists s > t \ \mu_A(s) \geq \alpha(t)$.

3. $\mathcal{M}_t \models PA \preceq$ iff $\exists s < t \ \mu_A(s) \geq \alpha(t)$.

4. $\mathcal{M}_t \models F\lambda A \preceq$ iff $\exists s > t \ \lambda(s - t) \geq \alpha(t) \land \mu_A(s) \geq \alpha(t)$.

5. $\mathcal{M}_t \models P\lambda A \preceq$ iff $\exists s < t \ \lambda(t - s) \geq \alpha(t) \land \mu_A(s) \geq \alpha(t)$.

6. $\mathcal{M}_t \models B_1 \land B \preceq$ iff $\mathcal{M}_t \models B_1 \preceq \land \mathcal{M}_t \models B \preceq$.

7. $\mathcal{M}_t \models H \preceq B$ iff $\mathcal{M}_t \models B \preceq \Rightarrow \mathcal{M}_t \models H \preceq \beta$.

Recall that $(\alpha \land \beta)(t) = \min(\alpha(t), \beta(t))$, for every $t \in T$. In the next section, we shall also need the following notation:

$$(\lambda \oplus \rho)(t) = \max_{r + s = t} \min(\lambda(r), \rho(s)).$$
Definition 3.4 Let $\mathcal{P}$ be an FTP-program, and $A$ be any formula. We say that $A$ is a consequence of $\mathcal{P}$ (notation $\mathcal{P} \models A$) if any (Herbrand) model of $\mathcal{P}$ is also a model of $A$.

4. The Computation Process of FT-Prolog

Consider an FTP-program $\mathcal{P}$ and a query $\mathcal{Q}$, which means “When and how likely is $\mathcal{Q}$ true?” The system’s computation process leading to an answer to the query is described in this section. Here, the notation $\mathcal{P} \vdash_{\alpha} \mathcal{Q}\theta$ is to be read as “From the program $\mathcal{P}$, the system infers that $\mathcal{Q}\theta$ has a truth degree of $\alpha(t)$ at each time point $t$”. Thus, if $t_0$ is such that $\alpha(t_0) = \max_t \alpha(t)$, then $\mathcal{Q}\theta$ has highest truth degree at the time $t_0$.

Case 1: $Q$ is an atomic formula.

(1.1) If there is a clause $A \leftarrow \in \mathcal{P}$ and a substitution $\theta$ such that $\mathcal{Q}\theta = A\theta$, then $\mathcal{P} \vdash_{\alpha} \mathcal{Q}\theta$.

(1.2) If there is a clause $A \leftarrow B \in \mathcal{P}$ and a substitution $\theta$ such that $\mathcal{Q}\theta = A\theta$, and there is a substitution $\theta'$ such that $\mathcal{P} \vdash_{\beta} B\theta\theta'$, then $\mathcal{P} \vdash_{\alpha \land \beta} \mathcal{Q}\theta\theta'$.

Case 2: $Q$ has the form $FA$, where $A$ is an atomic formula.

(2.1) If there is a clause $FA_1 \leftarrow$ or a clause $F\lambda A_1 \leftarrow$ in $\mathcal{P}$ and a substitution $\theta$ such that $A\theta = A_1\theta$, then $\mathcal{P} \vdash_{\alpha} \mathcal{Q}\theta$.

(2.2) If there is a clause $FA_1 \leftarrow B$ or a clause $F\lambda A_1 \leftarrow B$ in $\mathcal{P}$ and a substitution $\theta$ such that $A\theta = A_1\theta$, and there is a substitution $\theta'$ such that $\mathcal{P} \vdash_{\beta} B\theta\theta'$, then $\mathcal{P} \vdash_{\alpha \land \beta} \mathcal{Q}\theta\theta'$.

(2.3) If there is a clause $FA_1 \leftarrow B$ or a clause $F\lambda A_1 \leftarrow B$ in $\mathcal{P}$ and a substitution $\theta$ such that $A\theta = A_1\theta$, and there is a substitution $\theta'$ such that $\mathcal{P} \vdash_{\beta} F B\theta\theta'$, then $\mathcal{P} \vdash_{\beta} \mathcal{Q}\theta\theta'$.

Case 3: $Q$ has the form $F\nu A$, where $A$ is an atomic formula and $\nu$ is a fuzzy integer.

(3.1) If there is a clause $F\lambda A_1 \leftarrow$ in $\mathcal{P}$ and a substitution $\theta$ such that $A\theta = A_1\theta$, and if $\gamma(t) = (\alpha(t) - \max_s |\lambda(s - t) - \nu(s - t)|) \lor 0$, then $\mathcal{P} \vdash_{\gamma} \mathcal{Q}\theta$.

(3.2) If there is a clause $F\lambda A_1 \leftarrow B$ in $\mathcal{P}$ and a substitution $\theta$ such that $A\theta = A_1\theta$, and there is a substitution $\theta'$ such that $\mathcal{P} \vdash_{\beta} B\theta\theta'$, and if

$$
\gamma(t) = (((\alpha \land \beta)(t) - \max_s |\lambda(s - t) - \nu(s - t)|) \lor 0),
$$

then $\mathcal{P} \vdash_{\gamma} \mathcal{Q}\theta\theta'$. 

(3.3) If there is a clause $F \lambda A_1 \vdash B$ in $\mathcal{P}$ and a substitution $\theta$ such that $A\theta = A_1\theta$, and there is a substitution $\theta'$ such that $\mathcal{P} \vdash_{\beta} F\rho B\theta\theta'$, and if

$$\gamma(i) = (\beta(t) - \max_s |(\lambda \oplus \rho)(s - t) - \nu(s - t)|) \lor 0,$$

then $\mathcal{P} \vdash_{\gamma} Q\theta\theta'$.

Case 4 and 5: $Q$ has the form $PA$ or $P\nu A$.

The computations are similar to those in Case 2 and Case 3 respectively.

Case 6: $Q$ has the form $A \land B$.

If there are substitutions $\theta$ and $\theta'$ such that $\mathcal{P} \vdash_{\alpha} A\theta$ and $\mathcal{P} \vdash_{\beta} B\theta\theta'$, then $\mathcal{P} \vdash_{\alpha \land \beta} Q\theta\theta'$.

Example 4.1 Consider the following FTP-program, in which $\alpha$ and $\beta$ are symbols representing fuzzy sets in the time domain $T$, $\lambda$ represents a fuzzy integer, and $a$ is a constant symbol.

1. $Fp(X) \not\leftarrow q(X)$
2. $F\lambda r(X) \not\leftarrow p(X)$
3. $q(a) \not\leftarrow$

Given the query $?Fr(X)$, the system's computation is summarised in the resolution tree shown below. This tree shows that the result of the system's computation is $\mathcal{P} \vdash_{\alpha \land \beta} Fr(a)$, which means that $r(a)$ will have a truth degree $(\alpha \land \beta)(t)$, which depends on the time $t$ taken as the current time.
Observe that Example 4.1 contains clauses that represent the typical situations of many dynamic systems in chemical reaction, biological evolution, earthquake prediction, etc.

5. Soundness of FT-Prolog

Consider an FTP-program $\mathcal{P}$ and a query $Q$. It can be easily verified that if there is a substitution $\theta$ such that $\mathcal{P} \vdash_\gamma Q\theta$, then for every ground instance $Q\theta\theta'$ of $Q\theta$, we have $\mathcal{P} \vdash_\gamma Q\theta\theta'$. Therefore, it suffices to prove the soundness of the system's computation process for ground queries. This is established by the following theorem.

**Theorem 5.1** Let $\mathcal{P}$ be an FTP-program, $Q$ a ground query, and $\gamma$ a fuzzy subset of the time domain. Then

$$\mathcal{P} \vdash_\gamma Q \Rightarrow \mathcal{P} \models Q \models_\gamma.$$

**Proof.** The proof is by induction on the number of derivations. We shall consider each case of the system's computation process, and in each case, let $M$ be a (Herbrand) model of $\mathcal{P}$, and $t \in T$. Thus, $M$ is a model of every clause in $\mathcal{P}$.

**Case 1:** $Q$ is an atomic formula. This case is straightforward.

**Case 2:** $Q$ has the form $FA$, where $A$ is an atomic formula.

1. Suppose that there is a clause $FA_1 \models \mathcal{P}$ and a substitution $\theta$ such that $A = A_1\theta$. Then $M, t \models FA_1\theta \models_\gamma$, which is the same as $M, t \models Q \models_\gamma$. In the case there is a clause $F\lambda A_1 \models \mathcal{P}$ and a substitution $\theta$ such that $A = A_1\theta$, we have $M, t \models F\lambda A_1\theta \models_\gamma$, which implies $M, t \models FA_1\theta \models_\gamma$, and so $M, t \models Q \models_\gamma$.

2. Suppose that there is a clause $FA_1 \models \mathcal{P}$ and a substitution $\theta$ such that $A = A_1\theta$, and there is a ground substitution $\theta'$ such that $\mathcal{P} \vdash_\beta B\theta\theta'$; also let $\gamma = \alpha \land \beta$. Then, by the definition of model of $\mathcal{P}$, we have $M, t \models (FA_1 \models \mathcal{P} B)\theta\theta'$, and by the induction assumption, $M, t \models B\theta\theta' \models_\gamma$. It follows that $M, t \models FA_1\theta \models \mathcal{P} \alpha \land \beta$. Hence, $M, t \models Q \models_\gamma$. In the case the clause is of the form $F\lambda A_1 \models \mathcal{P} B$ rather than $FA_1 \models \mathcal{P} B$, the proof is almost the same, as $M, t \models F\lambda A_1\theta \models \mathcal{P} \alpha \land \beta$ implies $M, t \models FA_1\theta \models \mathcal{P} \alpha \land \beta$.

3. Suppose that there is a clause $FA_1 \models \mathcal{P}$ and a substitution $\theta$ such that $A = A_1\theta$, and there is a ground substitution $\theta'$ such that $\mathcal{P} \vdash_\gamma FB\theta\theta'$. Then, for any $s \in T$, we have $M, s \models (FA_1 \models \mathcal{P} B)\theta\theta'$. Also, by the induction
assumption, \( M, t \models FB\theta \theta' \rightleftharpoons \). That is, \( \exists s > t \ M, s \models B\theta \theta' \rightleftharpoons \), which implies \( M, s \models FA_1 \theta \rightleftharpoons \). Hence, \( \exists u > s > t \ M, u \models A_1 \theta \rightleftharpoons \). That is, \( M, t \models FA_1 \theta \rightleftharpoons \), which is the same as \( M, t \models Q \rightleftharpoons \). In the case the clause is of the form \( F \lambda A_1 \rightleftharpoons B \) rather than \( FA_1 \rightleftharpoons B \), the proof is almost the same using the fact that \( M, s \models F \lambda A_1 \theta \rightleftharpoons \) implies \( M, s \models FA_1 \theta \rightleftharpoons \).

**Case 3:** \( Q \) has the form \( F \nu A \), where \( A \) is a ground atomic formula and \( \nu \) is a fuzzy integer.

(3.1) Suppose that there is a clause \( F \lambda A_1 \rightleftharpoons \) in \( \mathcal{P} \) and a substitution \( \theta \) such that \( A = A_1 \theta \), and \( \gamma(t) = (\alpha(t) - \max_s |\lambda(s - t) - \nu(s - t)|) \lor 0 \). Then we have \( M, t \models F \lambda A_1 \theta \rightleftharpoons \), that is, \( \exists s > t \ \lambda(s - t) \geq \alpha(t) \land \mu_{A_1 \theta}(s) \geq \alpha(t) \). It follows that

\[
\nu(s - t) \geq \lambda(s - t) - \max_s |\lambda(u - t) - \nu(u - t)| \\
\geq \alpha(t) - \max_s |\lambda(u - t) - \nu(u - t)|.
\]

Hence \( \nu(s - t) \geq \gamma(t) \), and also \( \mu_{A_1 \theta}(s) \geq \alpha(t) \geq \gamma(t) \). Thus, \( M, t \models F \nu A_1 \theta \rightleftharpoons \), that is, \( M, t \models Q \rightleftharpoons \).

(3.2) Suppose that there is a clause \( F \lambda A_1 \rightleftharpoons B \) in \( \mathcal{P} \) and a substitution \( \theta \) such that \( A = A_1 \theta \), and there is a substitution \( \theta' \) such that \( \mathcal{P} \vdash_{\beta} B\theta \theta' \), and \( \gamma(t) = ((\alpha \land \beta)(t) - \max_s |\lambda(s - t) - \nu(s - t)|) \lor 0 \). Then \( M, t \models (F \lambda A_1 \rightleftharpoons B)\theta \theta' \) and \( M, t \models B\theta \theta' \rightleftharpoons \rightleftharpoons \) imply \( M, t \models FA_1 \theta \rightleftharpoons \). Hence, by the same reasoning as in (3.1), we have \( M, t \models F \nu A_1 \theta \rightleftharpoons \), that is, \( M, t \models Q \rightleftharpoons \).

(3.3) Suppose that there is a clause \( F \lambda A_1 \rightleftharpoons B \) in \( \mathcal{P} \) and a substitution \( \theta \) such that \( A = A_1 \theta \), also there is a substitution \( \theta' \) such that \( \mathcal{P} \vdash_{\beta} F \rho B\theta \theta' \), and \( \gamma(t) = (\beta(t) - \max_s |(\lambda \oplus \rho)(s - t) - \nu(s - t)|) \lor 0 \). Then, from \( M, t \models F \rho B\theta \theta' \rightleftharpoons \), it follows that

\[
\exists s > t \ \rho(s - t) \geq \beta(t) \land M, s \models B\theta \theta' \rightleftharpoons \ (1)
\]

Also, since \( M, s \models (F \lambda A_1 \rightleftharpoons B)\theta \theta' \), it follows that \( M, s \models FA_1 \theta \rightleftharpoons \). That is,

\[
\exists u > s \ \lambda(u - s) \geq \beta(t) \land \mu_{A_1 \theta}(u) \geq \beta(t) \ (2)
\]

By combining (1) and (2), we have \( \exists u > t \ (\lambda \oplus \rho)(u - t) \geq \beta(t) \land \mu_{A_1 \theta}(u) \geq \beta(t) \).

This implies (by the reasoning similar to that in (3.1) and (3.2)) that \( \exists u > t \ \nu(u - t) \geq \gamma(t) \land \mu_{A_1 \theta}(u) \geq \gamma(t) \). That is, \( M, t \models F \nu A_1 \theta \rightleftharpoons \), which is the same as \( M, t \models Q \rightleftharpoons \).
The proofs of Cases 4 and 5 are similar to those of Cases 2 and 3, and the proof of Case 6 is straightforward.

As for the completeness, it is well known that Prolog itself is incomplete, due to its top-down and left-to-right selection rule. Since FT-Prolog is an extension of Prolog, it inherits this weakness, unless a different selection rule is employed, which is outside the scope of this paper.

6. References

14 D. Qian, 'Representation and use of imprecise temporal knowledge in dynamic systems', In Fuzzy Sets and Systems, 50 (1992), 59-77.