

Fuzzy Time in Linear Temporal Logic

ACHILLE FRIGERI, Politecnico di Milano

LILIANA PASQUALE, Lero—The Irish Software Engineering Research Centre

PAOLA SPOLETINI, Università dell'Insubria

In the past years, the adoption of adaptive systems has increased in many fields of computer science, such as databases and software engineering. These systems are able to automatically react to events by collecting information from the external environment and generating new events. However, the collection of data is often hampered by uncertainty and vagueness. The decision-making mechanism used to produce a reaction is also imprecise and cannot be evaluated in a crisp way, as it depends on vague temporal constraints expressed by humans. Logic has been extensively used as an abstraction to express vagueness in the satisfaction of system properties, as well as to enrich existing modeling formalisms. However, existing attempts to fuzzify the temporal modalities still have some limitations. Existing fuzzy temporal languages are generally obtained from classical temporal logic by replacing classical connectives or propositions with their fuzzy counterparts. Hence, these languages do not allow us to represent temporal properties, such as “almost always” and “soon,” in which the notion of time is inherently fuzzy. To overcome these limitations, we propose a temporal framework, fuzzy-time temporal logic (FTL), to express vagueness on time. This framework formally defines a set of fuzzy temporal modalities that can be customized by choosing a specific semantics for the connectives. The semantics of the language is sound, and the introduced modalities respect a set of mutual relations. We also prove that under the assumption that all events are crisp, FTL reduces to linear temporal logic (LTL). Moreover, for some of the possible fuzzy interpretations of the connectives, we identify adequate sets of temporal operators, from which it is possible to derive all of the other ones.

1. INTRODUCTION

In the past years, the adoption of adaptive systems has increased in many fields of computer science. Adaptive systems must automatically react to achieve or maintain their requirements, depending on their internal state and the changes that can take

Author's addresses: Achille Frigeri, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milan, Italy; email: achille.frigeri@polimi.it; Liliana Pasquale, Lero - the Irish Software Engineering Research Centre, University of Limerick, Limerick, Ireland; email: liliana.pasquale@lero.ie; Paola Spoletini, Dipartimento di Scienze Teoriche e Applicate Università dell'Insubria, Via Mazzini 5, 21100 Varese, Italy; email: paola.spoletini@uninubria.it

place in the surrounding environment. Examples of such systems are active databases [Paton and Díaz 1999], active sensor networks [Levis et al. 2005], and smart grids [SmartGrids 2013].

Event-driven architectures [Mühl et al. 2006] are a common architectural paradigm to design adaptive systems. This paradigm is based on the idea that the actions performed by the system are generated as a reaction to the events occurred inside and outside it. In many cases, providing such active functionality requires detection of the occurrence of other relevant events, according to a set of inference rules. These rules are generally defined by domain experts and are formalized by designers. Domain experts must provide the set of basic events to be collected, which serve as input to the rules, their interrelationships, and parameters, for identifying the occurrence of new events.

However, the collection of data is often hampered by uncertainty and vagueness that can arise from the imprecision of the monitoring infrastructure, as well as unreliable data sources and networks. The inference rules that are used to produce a reaction are also imprecise. They often depend on the evaluation of untimed or temporal properties that are vague, since they are expressed by humans, or they must provide a certain degree of flexibility. For all of these reasons, such rules cannot be formalized in a crisp way, using traditional crisp (temporal) logic. For example, in the customer's domain, a smart grid might have to satisfy the following property: "all appliances must be available almost always." This rule is vague since the concept of availability cannot be assessed precisely, as it may depend on the customers' perception. The temporal period ("almost always"), during which the availability property must be satisfied, is vague as well. For these reasons, it becomes fundamental to identify a suitable formalism to represent vague properties as suitable untimed or temporal formulae.

Fuzzy logic has been adopted as a mathematical abstraction to express vagueness in the satisfaction of formulae and can be used to represent the properties mentioned previously. However, although propositional fuzzy logic has been deeply investigated and has even been used to extend and enrich typical modeling formalisms, such as Petri nets [Looney 1988; Cardoso and Camargo 1998] and UML [Ma et al. 2012], the fuzzy version of the temporal modalities has often been neglected and requires further investigation. Few attempts [Thiele and Kalenka 1993; Lamine and Kabanza 2000; Dutta 1988; Dubois and Prade 1989; Moon et al. 2004; De Alfaro et al. 2009; Mukherjee and Dasgupta 2013] to manage time have been made, but all of these approaches mainly focus on the uncertainty of the information and do not take into account the truth degree of temporal expressions. The semantics of existing fuzzy temporal operators is based on the idea of replacing classical connectives or propositions with their fuzzy counterparts. Indeed, existing fuzzy languages do not allow us to represent additional temporal properties, such as "almost always" or "soon." This kind of modality may be useful when we need to specify situations in which a formula is slightly satisfied, since an event happens a little bit later than expected, a property is always satisfied except for a small set of time instants, or a property is maintained for a time interval that is slightly smaller than the one requested.

To overcome the limitations of existing formalisms, we propose fuzzy-time temporal logic (FTL), a temporal framework to express vagueness on time. This framework formally defines a set of fuzzy temporal modalities that can be customized by choosing a specific semantics for the connectives. Indeed, the semantics of connectives is not defined univocally, as with the term *fuzzy logic*, we do not refer to a specific formal system, but to some form of many-valued logic. In this article, we focus on the three basic continuous t-norms: Łukasiewicz, Gödel-Dummett, and Product [Hájek 1998b]. This choice is motivated by the fact that several metamathematical properties (completeness theorems, deduction theorems, complexity) were proved for these t-norms (e.g., see Baaz et al. [2001] and Aguzzoli et al. [2005]), and more important, any continuous t-norm is an ordinal sum construction of them. For historical and practical reasons, we also

consider the fuzzy logic introduced by Zadeh [1965], which although is not a t-norm-based logic, has been broadly used in artificial intelligence and in the analysis of fuzzy control systems.

The semantics of the language is sound, and the introduced modalities respect a set of mutual relations. We also prove that under the assumption that all events are crisp, FTL reduces to linear temporal logic (LTL). Finally, for some of the fuzzy interpretations of the connectives, we identify an adequate set of temporal operators, from which it is possible to derive all of the others. Moreover, our framework is well suited for representing adaptive rules and properties, as demonstrated through the proposed example. The main contribution of this article is to provide a formalism for modeling temporal properties of systems characterized by vagueness and for which LTL may not be appropriate. Therefore, to encourage the use of FTL by nonlogicians, the syntax of the language was intentionally kept as intuitive as possible. Furthermore, we provide evidence that the semantics of FTL is appropriate, as it preserves LTL properties and operators—from which FTL operators often derive naturally.

The article is organized as follows. Section 2 discusses related work. Section 3 provides some background knowledge about fuzzy logic and points out its differences with regard to probability theory. Section 4 presents the FTL framework by illustrating some interesting properties of its operators. Section 5 shows some relationships between FTL and LTL, and provides some adequate sets of connectives for the considered interpretations. Section 6 provides some examples of possible FTL specifications in the smart grids domain, and Section 7 concludes the article.

2. RELATED WORK

In computer science, fuzzy logic has mainly been used to represent the uncertainty due to the unpredictability of the environment or the imprecision of the measurements. Many attempts have been made to use fuzzy logic to monitor the satisfaction of temporal properties of the system and/or the environment. For example, for each classic temporal operator (always, eventually, until, etc.), Lamine and Kabanza [2000] add a corresponding fuzzy temporal one. These operators keep the same semantics of their crisp counterparts, with the only difference that the Boolean connectives (not, and, or) are replaced with the corresponding operations in Zadeh's interpretation (see operations respectively associated with negation, t-norm, and t-conorm later in Table II). The authors evaluate a fuzzy proposition over a history (i.e., a sequence of states) and associate a weight with the evaluation made at each state. The weights and the extent to which the history is needed to evaluate a proposition are defined empirically, depending on the application and the properties expressed by the proposition itself. Similarly, Thiele and Kalenka [1993] define a fuzzy "interpretation" of the classical temporal operators. They also introduce new fuzzy temporal operators to represent the short or long time distance in which a specific property must be satisfied, both in the future or in the past. Despite that the aforementioned approaches represent a first step toward the fuzzification of time, they do not associate a specific fuzzy semantics with the temporal modalities. Instead, temporal modalities have a fuzzy semantics that only depends on the interpretation given to their (sub-)argument, which is an untimed fuzzy formula. De Alfaro et al. [2009] provide some metric extensions of the classical linear and branching relations such as trace inclusion, trace equivalence, simulation, and bisimulation. They consider a quantitative setting in which propositions are not interpreted as Boolean values, but as elements of arbitrary metric spaces. They show a full logical characterization of the induced distances in terms of quantitative versions of LTL and μ -calculus, together with an algorithm for computing the distances over finite systems.

Other work ([Dutta 1988; Dubois and Prade 1989; Moon et al. 2004]) uses fuzzy temporal logic to express uncertainty on the time instant at which some specific events

may occur and on the temporal relationships among events and states. Dutta [1988] defines the occurrence of an event as the possibility of its occurrence in any time interval. In this way, the author can evaluate a set of temporal relations between a pair of events: if an event precedes/follows another one, the degree an event overlaps another one, or whether an event immediately follows another one. Similarly, Dubois and Prade [1989] represent dates as a possibility distribution. Hence, it is possible to express different situations: whether a date is precisely known or not (i.e., it is within an interval), whether a date is fuzzily known (i.e., the interval boundaries that contain the date are not clearly known), or whether a date is attached to an event that may not occur. Indeed, the authors use fuzzy sets to represent time points that are possibly/necessarily after or before a date, and use fuzzy comparators to express relations between time instants. Moon et al. [2004] do not consider uncertainty on the time instants, but fuzzify temporal events and states and define an order relation among events and states, represented as a directed graph. A recent contribution related with the notion of vagueness on time is Mukherjee and Dasgupta [2013]. The authors introduce a notion of fuzzy time able to extend a real-time logic based on interval relevant in software engineering (MITL) to “fuzzy” intervals in a spirit similar to Thiele and Kalenka [1993]. In this case, the use of a dense real-time temporal logic is proved to be important in determining the quality of satisfaction of a formula.

In requirements engineering, fuzzy logic has been adopted to perform trade-off analysis [Liu and Yen 1996] among conflicting functional requirements. In particular, aggregation functions are used to combine correlated requirements into high-level ones. Fuzzy logic has also been exploited to express uncertain requirements [Liu et al. 2007; Whittle et al. 2009; Baresi et al. 2010]. Liu et al. [2007] introduce a methodology to elicit nonfunctional requirements through fuzzy membership functions that allow representing the uncertainty about the human perception. RELAX [Whittle et al. 2009] is a notation to express uncertain requirements, whose assessment is affected by the imprecision of the measurement. Finally, FLAGS [Baresi et al. 2010] extends traditional LTL by adding new operators to represent transient/small violations in the temporal domain. Its main purpose is to provide a notion of satisfaction level of requirements in the temporal domain. In particular, the authors use this approach to tolerate small deviations of the satisfaction of the requirements during or within a temporal interval. Despite that the purpose of FLAGS is similar to our approach, the syntax and the semantics of the language are not formally described, and the relations among temporal operators are not provided.

3. BACKGROUND

This section provides a general definition of fuzzy logic and points out the differences between a fuzzy and a probabilistic approach for the evaluation of temporal properties. Finally, the section introduces Fuzzy Linear-Time Temporal Logic (FLTL) [Lamine and Kabanza 2000], which is the only attempt of introducing fuzziness in LTL, and discusses its limitations.

3.1. General Formalization of Fuzzy Logic

The term *fuzzy* was explicitly used for the first time in the seminal work of Zadeh [1965] about fuzzy sets, where he presented the theory of classes with unsharp boundaries. In this work, the logical formalism of fuzzy sets shares the same syntax of propositional logic (PL), but its formulae have a truth value comprised between 0 and 1. Conjunction and disjunction are interpreted as min and max operations, respectively.

As pointed out by Zadeh [1994], two main directions in fuzzy logic have to be distinguished. In a broad sense, fuzzy logic has been used to support fuzzy control and express the vagueness of natural languages, without demonstrating its formal properties. In

Table I. General Properties of the Interpretations of Connectives

| | Boundary Value | Commutativity | Associativity | Monotonicity |
|-------------|--|---------------|---------------|---|
| Negation | $\ominus 0 = 1$ $\ominus 1 = 0$ | - | - | $\alpha \leq \beta \Rightarrow \ominus \alpha \geq \ominus \beta$ |
| t-Norm | $\alpha \otimes 0 = 0$ $\alpha \otimes 1 = \alpha$ | yes | yes | $\beta \geq \gamma \Rightarrow \alpha \otimes \beta \geq \alpha \otimes \gamma$ $\alpha \otimes \beta \leq \alpha$ |
| t-Conorm | $\alpha \oplus 0 = \alpha$ $\alpha \oplus 1 = 1$ | yes | yes | $\beta \geq \gamma \Rightarrow \alpha \oplus \beta \geq \alpha \oplus \gamma$ $\alpha \oplus \beta \geq \alpha$ |
| Implication | $1 \oslash \beta = \beta$ $0 \oslash \beta = \alpha \oslash 1 = 1$ $\alpha \oslash 0 = \ominus \alpha$ | no | no | $\alpha \leq \beta \Rightarrow \alpha \oslash \gamma \geq \beta \oslash \gamma$ $\beta \leq \gamma \Rightarrow \alpha \oslash \beta \leq \alpha \oslash \gamma$ $\alpha \oslash \beta \geq \max\{\ominus \alpha, \beta\}$ |

Table II. Examples of Interpretation for Connectives

| | Zadeh [Zadeh 1965] | Gödel-Dummett [Gödel 1933; Dummett 1959] | Łukasiewicz [Łukasiewicz 1920, 1970] | Product [Hájek et al. 1996] |
|------------------------|-----------------------------|---|---|--|
| $\ominus \alpha$ | $1 - \alpha$ | $\begin{cases} 1, & \alpha = 0 \\ 0, & \alpha > 0 \end{cases}$ | $1 - \alpha$ | $\begin{cases} 1, & \alpha = 0 \\ 0, & \alpha > 0 \end{cases}$ |
| $\alpha \otimes \beta$ | $\min\{\alpha, \beta\}$ | $\min\{\alpha, \beta\}$ | $\max\{\alpha + \beta - 1, 0\}$ | $\alpha \cdot \beta$ |
| $\alpha \oplus \beta$ | $\max\{\alpha, \beta\}$ | $\max\{\alpha, \beta\}$ | $\min\{\alpha + \beta, 1\}$ | $\alpha + \beta - \alpha \cdot \beta$ |
| $\alpha \oslash \beta$ | $\max\{1 - \alpha, \beta\}$ | $\begin{cases} 1, & \alpha \leq \beta \\ \beta, & \alpha > \beta \end{cases}$ | $\min\{1 - \alpha + \beta, 1\}$ | $\begin{cases} 1, & \alpha \leq \beta \\ \beta/\alpha, & \alpha > \beta \end{cases}$ |

a narrow sense, “fuzzy logic is a logical system which is an extension of multivalued logic and is intended to serve as a logic of approximate reasoning.” In this article, we use the term *fuzzy logic* to refer both to Zadeh logic [Zadeh 1965], which in computer science is often called *fuzzy logic*, and to continuous t-norm fuzzy logics [Hájek 1998b].

Although Zadeh logic has been applied extensively in soft computing, in the past few years, new families of fuzzy logics with strong logical characterization arose. Among them, the t-norm-based fuzzy logics attract our interest, as they are often axiomatizable and, indeed, have a strong algebraic characterization and can support the demonstration of the results about completeness and complexity [Hájek 1998b; Baaz et al. 2001].

In this article, we conceive fuzzy logic as a many-valued logic [Łukasiewicz 1970], whose formulae have a truth value comprised between 0 and 1 and the semantics of the connectives satisfies some monotonicity laws. The semantics of a fuzzy logic must also be *coherent* with PL, which means that fuzzy logic and PL must share the same syntax, fuzzy logic must reduce to PL when all predicates assume value 0 or 1, and conjunction and disjunction must be commutative and associative connectives. The semantics of the conjunction (\wedge), disjunction (\vee), negation (\neg), and implication (\Rightarrow) is inferred by considering a continuous t-norm (\otimes) [Klement et al. 2000], its associated t-conorm (\oplus), a negation function (\ominus), and an implication function (\oslash), respectively. In case of a t-norm fuzzy logic, the negation is the pseudocomplement (i.e., $\ominus \alpha = \max\{\beta \in [0, 1] \mid \alpha \otimes \beta = 0\}$), whereas the implication function becomes the residuum of the t-norm (i.e., $\alpha \oslash \beta = \max\{\gamma \in [0, 1] \mid \alpha \otimes \gamma \leq \beta\}$). In the rest of the article, we will refer to these functions as the interpretation of connectives. Note also that the family of (continuous) t-norm fuzzy logics is infinite, as demonstrated by the infinite class of Dubois-Prade [Dubois and Prade 1982] and Yager [1980] t-norms and t-conorms.

Table I summarizes some useful properties of the connectives of a fuzzy logic, and Table II provides the interpretation of these connectives for the Zadeh logic and three other well-known t-norm fuzzy logics.

Once an interpretation of the connectives is identified, the evaluation of a (fuzzy) formula can be represented as a function v_i from the set of well-formed formulae to $[0, 1]$, which extends the interpretation $i : AP \rightarrow [0, 1]$ used to evaluate an atomic proposition in AP .

The following proposition describes some well-known properties of t-norms and t-conorms.

PROPOSITION 3.1. *Let \otimes be a t-norm and \oplus be a t-conorm, $\alpha, \beta \in [0, 1]$, and $d^+, d^\times : [0, 1]^2 \rightarrow \{0, 1\}$ be the drastic sum and the drastic product defined respectively by*

$$\begin{aligned} d^+(\alpha, \beta) &= 1 \Leftrightarrow \alpha + \beta > 0, \\ d^\times(\alpha, \beta) &= 1 \Leftrightarrow \alpha \cdot \beta = 1. \end{aligned}$$

Then,

$$\begin{aligned} \max\{\alpha, \beta\} &\leq \alpha \oplus \beta \leq d^+(\alpha, \beta), \\ d^\times(\alpha, \beta) &\leq \alpha \otimes \beta \leq \min\{\alpha, \beta\}. \end{aligned}$$

For a continuous t-norm, it is possible to define two connectives referred to as lattice (or weak) conjunction ($\bar{\wedge}$) and lattice disjunction ($\bar{\vee}$). The semantics of these connectives is given by

$$p \bar{\wedge} q \equiv p \wedge (p \Rightarrow q), \quad (1)$$

$$p \bar{\vee} q \equiv ((p \Rightarrow q) \Rightarrow q) \bar{\wedge} ((q \Rightarrow p) \Rightarrow p). \quad (2)$$

Nevertheless, they reduce respectively to the max and min operations, as stated in the following well-known proposition.

PROPOSITION 3.2. *Let $p_\alpha, p_\beta \in AP$ such that $i(p_\alpha) = \alpha$, and $i(p_\beta) = \beta$, then, for each continuous t-norm,*

$$\begin{aligned} v_i(p_\alpha \bar{\wedge} p_\beta) &= \min\{\alpha, \beta\}, \\ v^i(p_\alpha \bar{\vee} p_\beta) &= \max\{\alpha, \beta\}. \end{aligned}$$

3.2. Fuzzy Logic and Probability

In general, software engineering solutions [Calinescu and Kwiatkowska 2009] use probability-based approaches to deal with uncertainty and vagueness. They focus on observable events, whose occurrence is uncertain, and neglect those events that are vague and cannot be clearly assessed.

For example, the statement “tomorrow there will be a power outage” is uncertain, since it is not possible to know the truth value of the formula. However, by applying probability theory (e.g., by analyzing the frequency of power outages during the last month), it is possible to state that, for example, the probability that the aforementioned statement will be true is 3.8%. When a direct observation can be performed (i.e., tomorrow), it is still possible to assess whether an outage took place or not, and, indeed, the probability value can collapse either to 0 or 1.

Instead, the statement “tomorrow the number of power outages will be low” is not tractable from a probabilistic point of view if the event is not clearly measurable—that is, if the concept of “low” is not defined in an observable way. In this case, we are not facing the problem of uncertainty of an event but the vagueness of its definition. Indeed, assigning the truth degree of 0.038 to the aforementioned statement means that tomorrow it will be necessary to face a “high number of outages.” Even a direct observation of the number of outages will not necessarily cause this value to collapse to 0 or 1.

3.3. Fuzzy Linear-Time Temporal Logic

This section briefly describes FLTL [Lamine and Kabanza 2000], which is an extension of Zadeh logic with temporal operators. FLTL has the same syntax of LTL. In particular, let Φ be the set of well-formed formulae and AP the set of propositional letters, then $\varphi \in \Phi$ if and only if

$$\varphi := p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\varphi \mid \mathbf{G}\varphi \mid \varphi\mathbf{U}\varphi,$$

where $p \in AP$. The semantics of a formula $\phi \in \Phi$ is defined with regard to a linear time structure $\pi_\sigma = (S, w_0, w, L)$, where S is a set of states, w_0 is the initial state, $w \in w_0S^\omega$ is an infinite path, and $L : S \rightarrow [0, 1]^{AP}$ is a fuzzy labeling function. The evaluation $v(\varphi, w^i)$ of a formula $\varphi \in \Phi$ along the path w from the i -th instant is a real number in $[0, 1]$ recursively defined by

$$\begin{aligned} v(p, w^i) &= L(w_i)(p), \\ v(\neg\varphi, w^i) &= 1 - v(\varphi, w^i), \\ v(\varphi \wedge \psi, w^i) &= \min\{v(\varphi, w^i), v(\psi, w^i)\}, \\ v(\mathbf{X}\varphi, w^i) &= v(\varphi, w^{i+1}), \\ v(\mathbf{G}\varphi, w^i) &= \min\{v(\varphi, w^i), v(\mathbf{G}\varphi, w^{i+1})\}, \\ v(\varphi\mathbf{U}\psi, w^i) &= \max\{v(\psi, w^i), \min\{v(\varphi, w^i), v(\varphi\mathbf{U}\psi, w^{i+1})\}\}. \end{aligned}$$

It is easy to see that FLTL extends LTL in the sense that if for all $s \in S$ and $p \in AP$ is $L(s)(p) \in \{0, 1\}$, then $v(\varphi, w^i) = 1$ if and only if $w^i \models \varphi$.

However, FLTL cannot represent the vagueness in the temporal dimension. Fuzzification just addresses Boolean connectives and keeps a crisp semantics for the temporal modalities. For example, when we evaluate the ‘‘globally’’ (always) operator, it may not be suitable to consider the minimum truth value encountered. This semantics does not allow us to tolerate transient violations that take place for a small number of times compared to a long time interval. For example, if we assess the truth of the statement ‘‘this week no power outage happened,’’ by using FLTL we cannot tolerate any power outage, and even one power outage is enough to drop to zero the truth value of this formula. Furthermore, even if FLTL allows us to express some statements about the future, such as ‘‘tomorrow power outages will take place,’’ we cannot express other statements such as ‘‘soon a power outage will happen.’’ For these reasons, the language that we propose in this article, although partially inspired by FLTL, introduces a completely new approach to fuzzify the temporal domain.

4. FUZZY-TIME TEMPORAL LOGIC

This section describes the syntax and semantics of FTL. In a narrow sense, FTL is not a logic, but it is a general formalism whose properties depend on an underlying propositional (fuzzy) logic, such as the ones introduced in Section 3.1. Since the choice of the underlying PL can have consequences both on the theoretical and practical levels, a designer should be fully aware of these impacts when he or she selects a proper PL for FTL.

4.1. Syntax

FTL extends LTL to include fuzziness in time. Let AP be a numerable set of atomic propositions, $\neg, \wedge, \vee, \Rightarrow$ be the (fuzzy) connectives, and O and T be the sets of unary and binary (fuzzy) temporal modalities, respectively. Then, φ belongs to the set Φ of well-formed FTL formulae (from now on, formulae) if it is defined as follows:

$$\varphi := p \mid \neg\varphi \mid \varphi \sim \varphi \mid \mathcal{O}\varphi \mid \varphi\mathcal{T}\varphi,$$

where $p \in AP$, \sim is a binary connective, $\mathcal{O} \in O$, and $\mathcal{T} \in T$. As unary operators, we consider \mathcal{X} (next), \mathcal{S} (soon), \mathcal{F} (eventually), \mathcal{F}_t (eventually in the next t instants), \mathcal{G}

(always), \mathcal{G}_t (always in the next t instants), \mathcal{AG} (almost always), \mathcal{AG}_t (almost always in the next t instants), \mathcal{L}_t (lasts t instants), \mathcal{W}_t (within t instants), where $t \in \mathbb{N}$. Binary operators are \mathcal{U} (until), \mathcal{U}_t (bounded until), \mathcal{AU} (almost until), and \mathcal{AU}_t (bounded almost until). We admit the use of $\mathcal{X}^j(\cdot)$ as a shorthand for j applications of \mathcal{X} . For example, $\mathcal{X}^2(\cdot) \equiv \mathcal{X}(\mathcal{X}(\cdot))$. We also set $\mathcal{X}^0\varphi \equiv \varphi$. From now on, operators *Soon*, \mathcal{AG} , \mathcal{AG}_t , \mathcal{L}_t , \mathcal{W}_t , \mathcal{AU} , and \mathcal{AU}_t will be indicated as “almost” operators.

4.2. Semantics

The semantics of a formula φ is defined with regard to a linear time structure (S, s_0, π, L) , where S is the set of states, s_0 is the initial state, π is an infinite path $\pi = s_0s_1 \dots \in S^\omega$, and $L : S \rightarrow [0, 1]^{AP}$ is the (fuzzy) labeling function that assigns to each state an evaluation for each atomic proposition in AP . π^i indicates the suffix of π , by starting from the i -th position and s^i is the first state of π^i . Besides, we adopt an *avoiding function* $\eta : \mathbb{Z} \rightarrow [0, 1]$. We assume that $\eta(i) = 1$ for all $i \leq 0$, and there exists $n_\eta \in \mathbb{N}$ such that η is strictly decreasing in $\{0, \dots, n_\eta\}$ and $\eta(n') = 0$ for all $n' \geq n_\eta$. Function η expresses the penalization assigned to the number of events that we want to ignore in evaluating the truth degree of a formula that contains an “almost” operator. For example, we interpret the formula “almost always p ” as “always p except for a small number of cases,” and we evaluate the formula according to the number of avoided events. Hence, the evaluation of a formula that contains the operator \mathcal{AG} realizes a trade-off between the number of avoided events and the penalization assigned to this number.

Since we are dealing with a multivalued logic, it makes no sense to define a crisp satisfiability relation. Instead, to define the semantics of a formula φ along a path, we express a *fuzzy satisfiability function* $v : S^\omega \times \Phi \rightarrow [0, 1]$, where $v_\pi(\varphi) = v \in [0, 1]$ means that the truth degree of φ along π is v . We say that two formulae φ and ψ in Φ are logically equivalent, in symbols $\varphi \equiv \psi$, if and only if $v_\pi(\varphi) = v_\pi(\psi)$ for each linear time structure, for each path π , and for each avoiding function.

The truth degree of a formula is defined, as usual, recursively on its structure. Let $p \in AP$ and π^i be a path, then,

$$\begin{aligned} v_{\pi^i}(p) &= L(s^i)(p), \\ v_{\pi^i}(\neg\varphi) &= \ominus v_{\pi^i}(\varphi), \\ v_{\pi^i}(\varphi \wedge \psi) &= v_{\pi^i}(\varphi) \otimes v_{\pi^i}(\psi), \\ v_{\pi^i}(\varphi \vee \psi) &= v_{\pi^i}(\varphi) \oplus v_{\pi^i}(\psi), \\ v_{\pi^i}(\varphi \Rightarrow \psi) &= v_{\pi^i}(\varphi) \oslash v_{\pi^i}(\psi), \end{aligned}$$

where $p \in AP$, $i \in \mathbb{N}$, and $\ominus, \otimes, \oplus, \oslash$ are the operations, between real numbers, defining the chosen semantics of the connectives ($\neg, \wedge, \vee, \Rightarrow$). We are now able to introduce the semantics of the temporal operators provided by FTL.

Next. Operator “next” (\mathcal{X}) has the same semantics of its corresponding LTL operator **X**:

$$v_{\pi^i}(\mathcal{X}\varphi) = v_{\pi^{i+1}}(\varphi).$$

Soon. Operator “soon” (*Soon*) extends the semantics of the “next” operator by tolerating at most n_η time instants of delay. In other words, the greater the number of tolerated instants, the greater the penalization will be:

$$v_{\pi^i}(\text{Soon } \varphi) = \bigoplus_{j=i+1}^{i+n_\eta} v_{\pi^j}(\varphi) \cdot \eta(j - i - 1).$$

PROPOSITION 4.1. *From the monotonicity of the t-conorm \oplus (see Table I), it naturally follows that*

$$v_{\pi^i}(\mathcal{X}\varphi) \leq v_{\pi^i}(\text{Soon } \varphi).$$

Example 1. This operator can be used to express requirements on the response time of a software service. For example, the formula $rp \Rightarrow \text{Soon } rr$ could express the fact that if a request is performed (rp), a response should be received (rr) as soon as possible. The sooner a response is received, the higher the requirement satisfaction will be.

Eventually. Operator “eventually” (\mathcal{F}) and its bounded version (\mathcal{F}_t) also maintain the same semantics of their corresponding LTL operator **F**. Namely,

$$\begin{aligned} v_{\pi^i}(\mathcal{F}_t\varphi) &= \bigoplus_{j=i}^{i+t} v_{\pi^j}(\varphi), \\ v_{\pi^i}(\mathcal{F}\varphi) &= \bigoplus_{j \geq i} v_{\pi^j}(\varphi) = \lim_{t \rightarrow +\infty} v_{\pi^i}(\mathcal{F}_t\varphi). \end{aligned}$$

Observe that for \mathcal{F}_t the equivalences $\mathcal{F}_0\varphi \equiv \varphi$ and $\mathcal{F}_t\varphi \equiv \varphi \vee \mathcal{X}\mathcal{F}_{t-1}\varphi$ hold, for $t \geq 0$. The semantics of \mathcal{F} requires a passage to the limit, whose existence is ensured by the fact that the sequence $v_{\pi^i}(\mathcal{F}_t\varphi)_{t \in \mathbb{N}}$ is increasing, as the t-conorm \oplus is monotonic. These properties are summarized in the following proposition.

PROPOSITION 4.2. *For all $\varphi \in \Phi$ and $t \leq t'$,*

$$v_{\pi^i}(\varphi) \leq v_{\pi^i}(\mathcal{F}_t\varphi) \leq v_{\pi^i}(\mathcal{F}_{t'}\varphi) \leq v_{\pi^i}(\mathcal{F}\varphi).$$

Within. Operator “within” (\mathcal{W}_t) is inherently bounded, and its semantics is defined by

$$v_{\pi^i}(\mathcal{W}_t\varphi) = \bigoplus_{j=i}^{i+t+n_\eta-1} v_{\pi^j}(\varphi) \cdot \eta(j-t-i).$$

Formula $\mathcal{W}_t p$ states that subformula p is supposed to hold in at least one of the next t instants or, possibly, in the next $t+n_\eta$ instants. In the last case, we apply a penalization for each instant after the t -th.

PROPOSITION 4.3. *The semantics of operator \mathcal{W}_t can be expressed by only using operators \mathcal{X} and Soon . More formally, for all $\varphi \in \Phi$ and $t > 0$,*

$$\mathcal{W}_t\varphi \equiv \mathcal{F}_t\varphi \vee \mathcal{X}^{t+1}\text{Soon } \varphi,$$

and

$$\mathcal{W}_0\varphi \equiv \text{Soon } \varphi.$$

COROLLARY 4.4. *For all $\varphi \in \Phi$ and $t \in \mathbb{N}$,*

$$\begin{aligned} v_{\pi^i}(\mathcal{W}_t\varphi) &\geq v_{\pi^i}(\mathcal{F}_t\varphi), \\ \lim_{t \rightarrow +\infty} v_{\pi^i}(\mathcal{W}_t\varphi) &= v_{\pi^i}(\mathcal{F}\varphi). \end{aligned}$$

PROOF. The first property follows immediately from the previous proposition. For the second property, observe that $v_{\pi^i}(\mathcal{F}_{n_\eta}\varphi) \geq v_{\pi^i}(\text{Soon } \varphi)$, and then actually

$$v_{\pi^i}(\mathcal{F}_{t+n_\eta}\varphi) \geq v_{\pi^i}(\mathcal{W}_t\varphi) \geq v_{\pi^i}(\mathcal{F}_t\varphi),$$

and applying the squeeze theorem, we demonstrate the thesis.

Example 2. This operator can be used to specify requirements on the maximum response time required for a specific request and allows tolerating small delays. Consider a meeting scheduler application that requires each participant i to provide his or her availabilities (ap_i) within a certain time (e.g, within 5 days) after the request for availability is sent (ar_i). This requirement can be expressed by the formula $\bigotimes_{i \in P}(ar_i \Rightarrow \mathcal{W}_5 ap_i)$, where P is the set of participants and the time unit is a day.

Always. Operator “always” (\mathcal{G}) and its bounded version (\mathcal{G}_t) extend the semantics of their corresponding LTL operator \mathbf{G} . Namely,

$$\begin{aligned} v_{\pi^i}(\mathcal{G}_t \varphi) &= \bigotimes_{j=i}^{i+t} v_{\pi^j}(\varphi), \\ v_{\pi^i}(\mathcal{G} \varphi) &= \bigotimes_{j \geq i} v_{\pi^j}(\varphi) = \lim_{t \rightarrow +\infty} v_{\pi^i}(\mathcal{G}_t \varphi). \end{aligned}$$

As for \mathcal{F}_t , observe that for \mathcal{G}_t the equivalences $\mathcal{G}_0 \varphi \equiv \varphi$ and $\mathcal{G}_t \varphi \equiv \varphi \wedge \mathcal{X} \mathcal{G}_{t-1} \varphi$ hold, for $t \geq 0$. Similarly to \mathcal{F} , the semantics of \mathcal{G} also requires a passage to the limit, whose existence is ensured by the fact that the sequence $v_{\pi^i}(\mathcal{G}_t \varphi)_{t \in \mathbb{N}}$ is decreasing, as the t-norm \otimes is monotonic (see Table I). These properties are summarized in the following proposition.

PROPOSITION 4.5. *For all $\varphi \in \Phi$ and $t' \leq t$,*

$$\begin{aligned} v_{\pi^i}(\mathcal{G} \varphi) &\leq v_{\pi^i}(\mathcal{G}_t \varphi) \leq v_{\pi^i}(\mathcal{G}_{t'} \varphi) \\ &\leq v_{\pi^i}(\mathcal{G}_1 \varphi) = v_{\pi^i}(\varphi \wedge \mathcal{X} \varphi) \leq v_{\pi^i}(\mathcal{G}_0 \varphi) = v_{\pi^i}(\varphi). \end{aligned}$$

From Propositions 4.1, 4.3, and 4.5, we obtain the following corollary.

COROLLARY 4.6. *For all $\varphi \in \Phi$ and $t, t' \in \mathbb{N}$,*

$$\begin{aligned} v_{\pi^i}(\mathcal{G} \varphi) &\leq v_{\pi^i}(\mathcal{F} \varphi), \\ v_{\pi^i}(\mathcal{G}_t \varphi) &\leq v_{\pi^i}(\mathcal{F}_{t'} \varphi), \\ v_{\pi^i}(\mathcal{G}_t \varphi) &\leq v_{\pi^i}(\mathcal{W}_{t'} \varphi). \end{aligned}$$

Almost Always. Operator “almost always” (\mathcal{AG}) and its bounded version (\mathcal{AG}_t) allow us to evaluate a property over the path π^i , by avoiding at most n_η evaluations of this property, and, at the same time, introducing a penalization for each avoided case. If more cases are avoided, penalization will be more severe. Hence, a trade-off should be identified between the number of avoided cases and the assigned penalization. Formally, for $t > 0$, let I_t be the initial segment of \mathbb{N} of length $t + 1$ (i.e., $I_t = \{0, 1, \dots, t\}$) and $\mathcal{P}^k(I_t)$ the set of subsets of I_t of cardinality k , then

$$\begin{aligned} v_{\pi^i}(\mathcal{AG}_0 \varphi) &= v_{\pi^i}(\varphi), \\ v_{\pi^i}(\mathcal{AG}_t \varphi) &= \max_{j \in I_t} \max_{H \in \mathcal{P}^{t-j}(I_t)} \bigotimes_{h \in H} v_{\pi^{i+h}}(\varphi) \cdot \eta(j), \\ v_{\pi^i}(\mathcal{AG} \varphi) &= \lim_{t \rightarrow +\infty} v_{\pi^i}(\mathcal{AG}_t \varphi). \end{aligned}$$

As we will see later, the sequence $v_{\pi^i}(\mathcal{AG}_t \varphi)_{t \in \mathbb{N}}$ is not monotonic. Nevertheless, we can still prove that the semantics of \mathcal{AG} is well defined.

PROPOSITION 4.7. *The semantics of \mathcal{AG} is well defined.*

PROOF. We prove that given $\varphi \in \Phi$, it is possible to recursively define n propositional letters $p_0, \dots, p_{n_\eta-1}$ such that

$$v_{\pi^i}(\mathcal{AG} \varphi) = \max_{j \leq n_\eta-1} \{\mathcal{G} p_j \cdot \eta(j)\}.$$

Let us define p_0 as

$$\forall i \in \mathbb{N}, v_{\pi^i}(p_0) = v_{\pi^i}(\varphi).$$

Then, for all $0 < m \leq n_\eta$, we recursively obtain p_m from p_{m-1} in the following way. Let h_m be the minimum in $\mathbb{N} \cup \{\infty\}$ such that for all $k \in \mathbb{N}$, $v_{\pi^h}(p_m) \leq v_{\pi^k}(p_{m-1})$. Then, we set

$$\begin{cases} v_{\pi^j}(p_m) = v_{\pi^j}(p_{m-1}), & j < h; \\ v_{\pi^j}(p_m) = v_{\pi^{j+1}}(p_{m-1}), & j \geq h. \end{cases}$$

Hence, for all $t \geq j$,

$$v_{\pi^t}(\mathcal{G}_{t-j} p_j) \leq \max_{H \in \mathcal{P}^{t-j}(I_1)} \bigotimes_{h \in H} v_{\pi^{i+h}}(\varphi).$$

The first term corresponds to choose $H = I_t \setminus \{h_1, \dots, h_j\}$. The converse inequality also holds, since it derives from the monotonicity of the operation \otimes . Then, the passage to the limit

$$\lim_{t \rightarrow +\infty} v_{\pi^t}(\mathcal{AG}_t \varphi) = \lim_{t \rightarrow +\infty} \max_{j \in \mathbb{N}} \{\mathcal{G}_{t-j} p_j \cdot \eta(j)\} = \max_{j \in \mathbb{N}} \{\mathcal{G} p_j \cdot \eta(j)\}$$

implies the thesis.

Note that the maximum in preceding definition can be expressed in any of the fuzzy logic that we are considering. Indeed, in the Zadeh logic the maximum is simply the (standard) \vee , and in a t-norm fuzzy logic it is the lattice disjunction \vee . We decide to use the maximum to find the best matching between the number of avoided cases and the penalization due to η . Indeed, if we define the semantics of \mathcal{AG} via the (strong) disjunction as

$$v_{\pi^i}(\mathcal{AG}_t \varphi) = \bigoplus_{j=0}^t \bigoplus_{H \in \mathcal{P}^{t-j}(I_1)} \bigotimes_{h \in H} v_{\pi^{i+h}}(\varphi) \cdot \eta(j)$$

and consider the Łukasiewicz's interpretation for the connective \vee , then a formula $\mathcal{AG} p$ will often evaluate to 1 due to the high number of considered cases, and (almost) independently from the evaluations of p at each time instant.

In the following proposition, we show how to reduce the complexity of the evaluation of operator \mathcal{AG} by exploiting the monotonicity of the t-conorm.

PROPOSITION 4.8. *It is possible to evaluate the truth degree of formula $\mathcal{AG}_t p$ by performing $O(n_\eta(\log(t) + 1))$ comparisons, $O(t)$ applications of the norm \otimes , and $O(n_\eta)$ multiplications.*

PROOF. We consider the same technique applied in the proof of Proposition 4.7.

Let $(a_k)_{k \leq n}$ be a finite sequence of indices such that $\forall k \leq n, a_k \leq t$, and $\forall h \leq k \leq n$,

Table III. Example of Definition of a Predicate p and an Avoiding Function η

| i | 0 | 1 | 2 | 3 |
|----------------|-----|-----|-----|-----|
| $v_{\pi^i}(p)$ | 0.1 | 0.2 | 1 | 0.1 |
| $\eta(i)$ | 1 | 0.5 | 0.3 | 0 |

$v_{\pi^{a_h}}(p) \leq v_{\pi^{a_k}}(p)$, then

$$\begin{aligned}
 v_{\pi^i}(\mathcal{AG}_t p) &= \max_{j \leq n_\eta} \left\{ \bigotimes_{\substack{h \notin \{a_1, \dots, a_j\} \\ h \leq n_\eta}} v_{\pi^{i+h}}(p) \cdot \eta(j) \right\} \\
 &= \max_{1 \leq j \leq n_\eta} \left\{ v_{\pi^i}(\mathcal{G}_t p), \bigotimes_{\substack{h \notin \{a_1, \dots, a_j\} \\ h \leq n_\eta}} v_{\pi^{i+h}}(p) \cdot \eta(j) \right\}.
 \end{aligned} \tag{3}$$

Finding the indices a_i requires at most $O(n_\eta \log(t))$ comparisons (e.g., by applying the heapsort algorithm), and extra $O(n_\eta)$ comparisons are used to evaluate the maximum. $O(t)$ applications of \otimes are needed, observing that the operation is associative, and indeed, the value obtained at one step can be used for calculating the value for the following step.

From equality (3), we also infer the following corollary.

COROLLARY 4.9. *For all $\varphi \in \Phi$ and $t \in \mathbb{N}$,*

$$\begin{aligned}
 v_{\pi^i}(\mathcal{AG}_t \varphi) &\geq v_{\pi^i}(\mathcal{G}_t \varphi), \\
 v_{\pi^i}(\mathcal{AG} \varphi) &\geq v_{\pi^i}(\mathcal{G} \varphi).
 \end{aligned}$$

Observe that in general it is not possible to establish a priori that inequality holds between $v_{\pi^i}(\mathcal{AG}_t \varphi)$ and $v_{\pi^i}(\mathcal{AG}_{t'} \varphi)$, with $t \neq t'$, as this also depends on function η . For example, let us consider a predicate p together with an avoiding function η , whose behaviors are described in Table III. If we consider Zadeh's interpretation of connectives, then $v_{\pi^0}(\mathcal{AG}_1 p) = 0.1$, $v_{\pi^0}(\mathcal{AG}_2 p) = 0.3$, and $v_{\pi^0}(\mathcal{AG}_3 p) = 0.06$, and the sequence $v_{\pi^i}(\mathcal{AG}_t p)_{t \in \mathbb{N}}$ is not monotonic.

Example 3. This operator can be used to specify requirements related to the availability of a software service. For example, the formula $\mathcal{AG}(rp \Rightarrow \mathcal{X}rr)$ could express that almost always a software service should be responsive. In particular, the more times the service is responsive—that is, a response is received (rr) right after a request is performed (rp)—the more the requirement expressed by the formula will be satisfied.

Lasts. Operator “lasts” (\mathcal{L}_t) is bounded and expresses a property that lasts for t consecutive time instants from the current one, possibly avoiding some evaluations of the events at the end of the considered time interval. Then, the semantics of this operator is defined as follows:

$$v_{\pi^i}(\mathcal{L}_t \varphi) = \max_{0 \leq j \leq \min\{t, n_\eta - 1\}} \{v_{\pi^i}(\mathcal{G}_{t-j} \varphi) \cdot \eta(j)\}.$$

PROPOSITION 4.10. *Let $\varphi \in \Phi$ and $t \in \mathbb{N}$, then the sequence $v_{\pi^i}(\mathcal{L}_t \varphi)_{t \in \mathbb{N}}$ is decreasing, and its limit is $v_{\pi^i}(\mathcal{G} \varphi)$. Moreover, the following inequalities hold:*

$$v_{\pi^i}(\mathcal{G}_t \varphi) \leq v_{\pi^i}(\mathcal{L}_t \varphi) \leq v_{\pi^i}(\mathcal{AG}_t \varphi).$$

PROOF. The fact that the sequence $v_{\pi^i}(\mathcal{L}_t\varphi)_{t \in \mathbb{N}}$ is decreasing follows immediately from the definition and from Proposition 4.5. Again, from definition

$$v_{\pi^i}(\mathcal{G}_t\varphi) \leq v_{\pi^i}(\mathcal{L}_t\varphi) \leq v_{\pi^i}(\mathcal{G}_{t-n_\eta}\varphi),$$

and then passing to the limit, the sequence $v_{\pi^i}(\mathcal{L}_t\varphi)_{t \in \mathbb{N}}$ converges to $v_{\pi^i}(\mathcal{G}\varphi)$. The inequality $v_{\pi^i}(\mathcal{L}_t\varphi) \leq v_{\pi^i}(\mathcal{AG}_t\varphi)$ is a straightforward consequence of Proposition 4.8. \square

Example 4. This operator can be used to express requirements related to the session duration of a software service, which can have an impact on other properties, such as system usability. For example, the formula $\mathcal{L}_2(-disconnect)$ could express that the closer to at least two time units (e.g., hours) is the duration of a session, the higher the requirements formalized by the formula will be satisfied.

Until. The semantics of operator “until” (\mathcal{U}) and its bounded version (\mathcal{U}_t) naturally extends the one assigned to the corresponding LTL operator \mathbf{U} , for $t > 0$:

$$\begin{aligned} v_{\pi^i}(\varphi \mathcal{U}_0 \psi) &= v_{\pi^i}(\psi), \\ v_{\pi^i}(\varphi \mathcal{U}_t \psi) &= \max_{i \leq j \leq i+t} (v_{\pi^j}(\psi) \otimes v_{\pi^i}(\mathcal{G}_{j-1}\varphi)), \\ v_{\pi^i}(\varphi \mathcal{U} \psi) &= \lim_{t \rightarrow +\infty} v_{\pi^i}(\varphi \mathcal{U}_t \psi), \end{aligned}$$

Analogously to \mathcal{AG} , the maximum is used to find the best matching between the evaluation of ψ and φ .

PROPOSITION 4.11. *The semantics of operator \mathcal{U} is well defined. Moreover, $v_{\pi^i}(\varphi \mathcal{U} \psi) \leq v_{\pi^i}(\mathcal{F}\psi)$.*

PROOF. For the first part, it is sufficient to prove that the sequence $v_{\pi^i}(\varphi \mathcal{U}_t \psi)_{t \in \mathbb{N}}$ is increasing. This is obvious, as for all $t > 0$,

$$v_{\pi^i}(\varphi \mathcal{U}_t \psi) = \max\{v_{\pi^i}(\varphi \mathcal{U}_{t-1} \psi), v_{\pi^i}(\mathcal{G}_{t-1}\varphi \wedge \mathcal{X}^t \psi)\}.$$

For the second part, let $p \in AP$ be such that $\forall j \geq i, v_{\pi^j}(p) = 1$. Then, $v_{\pi^i}(\varphi \mathcal{U} \psi) \leq v_{\pi^i}(p \mathcal{U} \psi)$, and from Proposition 3.1, we have

$$v_{\pi^i}(\varphi \mathcal{U} \psi) \leq v_{\pi^i}(p \mathcal{U} \psi) = \max_{j \geq i} (v_{\pi^j} \models \psi) \leq v_{\pi^i}(\mathcal{F}\psi).$$

In particular, for all $t \in \mathbb{N}$, we can state

$$v_{\pi^i}(\psi) = v_{\pi^i}(\varphi \mathcal{U}_0 \psi) \leq v_{\pi^i}(\varphi \mathcal{U}_t \psi) \leq v_{\pi^i}(\varphi \mathcal{U} \psi) \leq v_{\pi^i}(\mathcal{F}\psi). \quad (4)$$

Almost Until. Operator “almost until” (\mathcal{AU}) and its bounded version (\mathcal{AU}_t) are obtained by replacing operator \mathcal{G}_t with its relaxed version \mathcal{AG}_t in the definition of \mathcal{U} (\mathcal{U}_t):

$$\begin{aligned} v_{\pi^i}(\varphi \mathcal{AU}_0 \psi) &= v_{\pi^i}(\psi), \\ v_{\pi^i}(\varphi \mathcal{AU}_t \psi) &= \max_{0 \leq j \leq t} (v_{\pi^{i+j}}(\psi) \otimes v_{\pi^i}(\mathcal{AG}_j \varphi)), \\ v_{\pi^i}(\varphi \mathcal{AU} \psi) &= \lim_{t \rightarrow +\infty} v_{\pi^i}(\varphi \mathcal{AU}_t \psi), \end{aligned}$$

for $t > 0$. Similarly to \mathcal{U} , we can state the following.

PROPOSITION 4.12. *The semantics of operator \mathcal{AU} is well defined. Moreover, for all $t \in \mathbb{N}$,*

$$v_{\pi^i}(\psi) = v_{\pi^i}(\varphi \mathcal{AU}_0 \psi) \leq v_{\pi^i}(\varphi \mathcal{U}_t \psi) \leq v_{\pi^i}(\varphi \mathcal{AU}_t \psi) \leq v_{\pi^i}(\varphi \mathcal{AU} \psi). \quad (5)$$

PROOF. As for \mathcal{U} , we can observe that for all $t > 0$,

$$v_{\pi^i}(\varphi \mathcal{A}\mathcal{U}_t \psi) = \max \{v_{\pi^i}(\varphi \mathcal{A}\mathcal{U}_{t-1} \psi), v_{\pi^i}(\mathcal{A}\mathcal{G}_{t-1} \varphi \wedge \mathcal{X}^t \psi)\}.$$

The sequence $v_{\pi^i}(\varphi \mathcal{A}\mathcal{U}_t \psi)_{t \in \mathbb{N}}$ is increasing, and the semantics of $\mathcal{A}\mathcal{U}$ is well defined. The latter part follows from Corollary 4.9. \square

Mutual relations between FTL operators can be subsumed from the propositions stated previously. In particular, for all $\mathcal{O} \in \mathcal{O}$, $v_{\pi^i}(\mathcal{G}\varphi) \leq v_{\pi^i}(\mathcal{O}\varphi) \leq v_{\pi^i}(\mathcal{F}\varphi)$. Notice that all the inequalities are always not strict, as in general the values of all operators may coincide—that is, it is possible to define a formula φ such that $v_{\pi^i}(\mathcal{G}\varphi) = v_{\pi^i}(\mathcal{F}\varphi)$ for some path π . However, this can happen only in the following special case.

PROPOSITION 4.13. *Let $\varphi \in \Phi$ and $i \in \mathbb{N}$, then $v_{\pi^i}(\mathcal{G}\varphi) = v_{\pi^i}(\mathcal{F}\varphi)$ if and only if $(\pi^j \models \varphi)$ is constant for all $j \geq i$.*

PROOF. For the first implication, observe that if $v_{\pi^j}(\varphi)$ is constant for all $j \geq i$, then for all $j, j' \geq i$, $v_{\pi^j}(\mathcal{F}\varphi) = v_{\pi^{j'}}(\mathcal{G}\varphi) = v_{\pi^i}(\varphi)$. Conversely, suppose $h, k \geq i$ exist such that $v_{\pi^h}(\varphi) = a < v_{\pi^k}(\varphi) = b$. Then, from Proposition 3.1, it follows that

$$v_{\pi^i}(\mathcal{G}\varphi) \leq \min_{j \geq i} \{v_{\pi^j}(\varphi)\} \leq a < b \leq \min_{j \geq i} \{v_{\pi^j}(\varphi)\} \leq v_{\pi^i}(\mathcal{F}\varphi).$$

Example 5. This operator can be used to express conditional requirements on a service response time. For example, the formula $(rp \Rightarrow \mathcal{W}_{200} rr) \mathcal{A}\mathcal{U}(\text{num.requests} \geq 100)$ could express that a service must almost always exhibit a response time less than equal 200 time units (e.g., milliseconds) while the load is lower than 100 requests per second.

4.3. COMMENTS

Note that we can consider a different avoiding function for each occurrence of operators. For example, in the formula of Example 5, we may prefer to tolerate a long delay in evaluating \mathcal{W}_{200} , but we accept to avoid only a few number of events in evaluating $\mathcal{A}\mathcal{U}$. In this case, we can define two functions, $\eta_{\mathcal{W}}$ and $\eta_{\mathcal{A}\mathcal{U}}$, such that for all $i \in \mathbb{N}$, $\eta_{\mathcal{W}}(i) \geq \eta_{\mathcal{A}\mathcal{U}}(i)$.

Moreover, notice that the semantics we have chosen for our operators is arbitrary, and many other variants can be proposed. However, the preceding properties show that our choice is reasonable. For example, the “almost” operators are more lax, since their evaluation has a greater value compared to that of their traditional counterparts, exactly as one would expect.

Finally, as we have already stated at the beginning of this section, the underlying propositional fuzzy logic has a deep impact on the evaluation of FTL formulae. Then, in describing a complex system, the designer should carefully take into account its properties to choose the most suitable interpretation. This can be done considering that the three t-norm-based fuzzy logics that we studied admit a complete set of axioms.

A set of formulae (actually, schemata) is a *complete* set of axioms for a t-norm fuzzy logic, if a formula is a *t-tautology* (i.e., its evaluation is 1 for any interpretation of the atomic propositions) if and only if it can be proved from the axioms using some deduction rules.

Gödel-Dummett, Łukasiewicz, and Product logic extend the basic fuzzy propositional logic (BL), which is the logic of continuous t-norms [Hájek 1998b]. Hájek’s standard completeness theorem states that A1–A7 (Table IV) is a complete set of axioms for BL by using modus ponens (i.e., from φ and $\varphi \Rightarrow \psi$ infer ψ) as the only deduction rule (Cignoli et al. [2000], first proved in Hájek [1998a] with a greater set of axioms).

Table IV. Axiom Schemata for BL, Gödel-Dummett, Łukasiewicz, and Product Logic

| | | |
|-----|---|--|
| A1 | $(\varphi \Rightarrow \psi) \Rightarrow ((\psi \Rightarrow \chi) \Rightarrow (\varphi \Rightarrow \chi))$ | sufficing integrality commutativity of \wedge conjunction |
| A2 | $(\varphi \wedge \psi) \Rightarrow \varphi$ | |
| A3 | $(\varphi \wedge \psi) \Rightarrow (\psi \wedge \varphi)$ | |
| A4 | $(\varphi \wedge (\varphi \Rightarrow \psi)) \Rightarrow (\psi \wedge (\psi \Rightarrow \varphi))$ | |
| A5a | $(\varphi \Rightarrow (\psi \Rightarrow \chi)) \Rightarrow ((\varphi \wedge \psi) \Rightarrow \chi)$ | |
| A5b | $((\varphi \wedge \psi) \Rightarrow \chi) \Rightarrow (\varphi \Rightarrow (\psi \Rightarrow \chi))$ | |
| A6 | $(\varphi \Rightarrow (\psi \Rightarrow \chi)) \Rightarrow ((\psi \wedge \varphi) \Rightarrow \chi) \Rightarrow \chi$ | |
| A7 | $\perp \Rightarrow \varphi$ | <i>ex falso quodlibet</i> |
| AG | $\varphi \Rightarrow (\varphi \wedge \varphi)$ | idempotence of \wedge |
| AŁ | $\neg\neg\varphi \Rightarrow \varphi$ | double negation |
| AΠ | $\neg\neg\varphi \Rightarrow ((\varphi \Rightarrow (\varphi \wedge \psi)) \Rightarrow (\psi \wedge \neg\neg\psi))$ | |

Gödel-Dummett, Łukasiewicz, and Product logic are (completely) axiomatized respectively by (A1–A7, AG) [Dummett 1959; Horn 1969a, 1969b], (A1–A7, AŁ) [Cignoli and Torrens 2003], and (A1–A7, AΠ) [Cintula 2001].

If in the real model each property is equivalent to its double negation, the designer should choose the Łukasiewicz interpretation. Moreover, in the Łukasiewicz case, idempotence of entailment fails, in particular $\varphi \not\equiv \varphi \wedge \varphi$ (which, on the other hand, holds in Gödel-Dummett logic). This can be useful if we are interested in putting emphasis on resource boundedness or on simultaneous occurrence of resources. For example, consider a vending machine and let the proposition 20¢ mean “a coin of twenty cents is inserted.” Then, formula $20\text{¢} \wedge 20\text{¢}$ should not be equivalent to the former, as it actually means that 40 cents were inserted. When linearity is requested, Gödel-Dummett logic is the best choice, as it is actually equivalent to intuitionistic logic [Glivenko 1929] with the prelinearity schema $(\varphi \Rightarrow \psi) \vee (\psi \Rightarrow \varphi)$. Despite Product logic being more difficult to understand, it is characterized by some interesting properties: both formulae $\neg\neg\varphi \Rightarrow \varphi$ (double negation) and $\varphi \Rightarrow (\varphi \wedge \varphi)$ (idempotency) fail to be t-tautologies, formulae $\neg\varphi \Rightarrow (\neg\varphi \wedge \neg\varphi)$ (weak idempotency), $(\varphi \Rightarrow \psi) \vee (\psi \Rightarrow \varphi)$ (prelinearity), and $\neg\neg\chi \Rightarrow (((\varphi \wedge \chi) \Rightarrow (\psi \wedge \chi)) \Rightarrow (\varphi \Rightarrow \psi))$ (cancellation by a nonzero element) are theorems.

5. REDUCTIONS AND EQUIVALENCES

This section proves that under the assumption that all events are crisp, FTL reduces to LTL and provides some adequate sets of connectives, from which it is possible to infer all others.

Reduction to LTL. We can prove that the semantics of FTL extends, in some sense, that of LTL, as stated in the following two propositions.

PROPOSITION 5.1. *Let $p, q \in AP$ such that for all $j \geq i$, $(\pi^j \models p), (\pi^j \models q) \in \{0, 1\}$, then*

$$\begin{aligned} v_{\pi^i}(\mathcal{F}p) &= 1 \Leftrightarrow \pi^i \models \mathbf{F}p, \\ v_{\pi^i}(\mathcal{G}p) &= 1 \Leftrightarrow \pi^i \models \mathbf{G}p, \\ v_{\pi^i}(p \mathcal{U} q) &= 1 \Leftrightarrow \pi^i \models p \mathbf{U} q. \end{aligned}$$

PROOF. It follows, through straightforward calculation, by applying the boundary value in Table I.

PROPOSITION 5.2. *Given any $p \in AP$ and $i \in \mathbb{N}$, let $v_{\pi^i}(p) \in \{0, 1\}$, and $\eta(1) = 0$, then FTL reduces to LTL.*

PROOF. Notice that by definition, *Soon* reduces to \mathcal{X} , \mathcal{W}_t to \mathcal{F}_t , \mathcal{AG}_t and \mathcal{L}_t to \mathcal{G}_t , and \mathcal{AU}_t to \mathcal{U}_t . Then, the thesis follows by applying an argument similar to the one used in the previous proposition.

Adequate Sets. An adequate (or functionally complete) set of connectives for a given logic is a set of connectives that is sufficient to equivalently express any formula of the logic. For example, it is well known that \mathbf{X} and \mathbf{U} , together with \wedge and \neg , form an adequate set of connectives for LTL. An adequate set is not necessarily a subset of the original set of operator—for example, $\{\text{NAND}\}$ and $\{\top, \neq\}$ are adequate sets for PL, and none of the included connectives is primitive. Clearly, adequate sets also depend on the interpretation of the connectives. We denote by $\text{FTL}(\mathbf{Z})$, $\text{FTL}(\mathbf{G})$, $\text{FTL}(\mathbf{L})$, and $\text{FTL}(\Pi)$ the logics whose semantics are based on Zadeh, Gödel-Dummett, Łukasiewicz, and Product interpretation, respectively.

To find small adequate sets of connectives for $\text{FTL}(\mathbf{Z})$, $\text{FTL}(\mathbf{G})$, $\text{FTL}(\mathbf{L})$, and $\text{FTL}(\Pi)$, we introduce

- n_η special atomic propositions $\top, \tilde{p}_1, \dots, \tilde{p}_{n_\eta-1}, \perp$ such that $v_{\pi^i}(\top) = 1$, $v_{\pi^i}(\perp) = 0$, and $v_{\pi^i}(\tilde{p}_j) = \eta(j)$, for all $i \in \mathbb{N}$, and $1 \leq j < n_\eta$,
- $n_{\eta-1}$ extra operators \odot^j , for $1 \leq j < n_\eta$, whose semantics is

$$v_{\pi^i}(\odot^j \varphi) = v_{\pi^i}(\varphi) \cdot \eta(j).$$

We are now able to find a finite adequate set of connectives for each considered interpretation in Table II.

THEOREM 5.3. *Let $\perp \in AP$, then the set $\{\wedge, \neg, \mathcal{X}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}, \odot^1, \dots, \odot^{n_\eta-1}\}$ is an adequate set for $\text{FTL}(\mathbf{Z})$.*

PROOF. Observe that in $\text{FTL}(\mathbf{Z})$, $\varphi \vee \psi \equiv \neg((\neg\varphi) \wedge (\neg\psi))$ and $\varphi \Rightarrow \psi \equiv (\neg\varphi) \vee \psi$. Moreover, $\mathcal{F}\varphi \equiv \perp \mathcal{U}\varphi$ and $\mathcal{G}\varphi \equiv \varphi \mathcal{U}\perp$. For the other operators, the reductions follow directly from the definition, using the new operators \odot^j . For example, *Soon* φ is equivalent to $\bigvee_{j=1}^{n_\eta} \odot^j(\mathcal{X}^j \varphi)$. \square

THEOREM 5.4. *Let $\perp \in AP$, then the set $\{\wedge, \Rightarrow, \mathcal{X}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}, \odot^1, \dots, \odot^{n_\eta-1}\}$ is an adequate set for $\text{FTL}(\mathbf{G})$.*

PROOF. On the one hand, in $\text{FTL}(\mathbf{G})$, connective \vee is interpreted as the maximum and coincides with the lattice disjunction \vee (see Proposition 3.2). On the other hand, \vee is defined in terms of only \wedge and \Rightarrow . Moreover, it is easy to see that $\neg\varphi \equiv \varphi \Rightarrow \perp$, and then both \vee and \neg can be expressed with connectives \wedge and \Rightarrow . Operators \mathcal{F} and \mathcal{G} are not dual in $\text{FTL}(\mathbf{G})$, nevertheless, as for $\text{FTL}(\mathbf{Z})$, the equivalences $\mathcal{F}\varphi \equiv \perp \mathcal{U}\varphi$ and $\mathcal{G}\varphi \equiv \varphi \mathcal{U}\perp$ hold. The other operators follow the same reduction rule used for $\text{FTL}(\mathbf{Z})$. For example,

$$\mathcal{AG}_t \varphi \equiv \bigvee_{j \in I_t} \bigvee_{H \in \mathcal{P}^{t-j}(I_t)} \bigwedge_{h \in H} \odot^j(\mathcal{X}^h \varphi),$$

and

$$\varphi \mathcal{U}_t \psi \equiv \bigvee_{0 \leq j \leq t} \left(\psi \wedge \left(\bigwedge_{0 \leq h < j} \mathcal{X}^h \varphi \right) \right).$$

THEOREM 5.5. *Let $\perp \in AP$, then the set $\{\wedge, \Rightarrow, \mathcal{X}, \mathcal{F}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}, \odot^1, \dots, \odot^{n_\eta-1}\}$ is an adequate set for $\text{FTL}(\mathbf{L})$.*

PROOF. Differently from Gödel-Dummett logic, in Łukasiewicz logic De Morgan's laws hold—that is, $\varphi \vee \psi \equiv \neg((\neg\varphi) \wedge (\neg\psi))$ and $\varphi \wedge \psi \equiv \neg(\neg\varphi \vee \neg\psi)$, and this clearly

extends to $\text{FTL}(\mathbb{L})$, and, moreover, $\neg\varphi \equiv \varphi \Rightarrow \perp$. Then, it follows that \vee and \neg can be expressed in terms of only \wedge and \Rightarrow , and operators \mathcal{F} and \mathcal{G} are dual in $\text{FTL}(\mathbb{L})$ —that is, $\mathcal{G}\varphi \equiv \neg\mathcal{F}\neg\varphi$ and $\mathcal{F}\varphi \equiv \neg\mathcal{G}\neg\varphi$, so we can always rewrite \mathcal{G} in terms of \mathcal{F} . This last result does not hold in the Gödel-Dummett case: even if the connectives \vee and \neg can be expressed in terms of \wedge and \Rightarrow , this is possible only considering lattice connectives and Proposition 3.2; as for this logic, De Morgan’s laws do not hold, due to the different definition of negation.

For the other operators, we can proceed similarly to the previous case, observing that (again from Proposition 3.2) the min/max operators are expressible as lattice conjunction/disjunction, which are defined in terms of \wedge and \Rightarrow . For example, for the formula $\mathcal{AG}_t\varphi$, we proceed as in Theorem 5.4 but considering the lattice disjunction $\underline{\vee}$ instead of the standard disjunction \vee to correctly evaluate the maximum of the considered subformulae—that is,

$$\mathcal{AG}_t\varphi \equiv \bigvee_{j \in I_t} \bigvee_{H \in \mathcal{P}^{t-j}(I_t)} \bigwedge_{h \in H} \odot^j(\mathcal{X}^h\varphi).$$

Analogously, we have

$$\mathcal{L}_t\varphi \equiv \bigvee_{0 \leq j \leq \min\{t, n_\eta - 1\}} \odot^j \left(\bigwedge_{0 \leq h \leq t-j} \mathcal{X}^h\varphi \right),$$

but

$$\mathcal{F}_t\varphi \equiv \bigvee_{0 \leq j \leq t} \mathcal{X}^j\varphi,$$

as in this last case, the operation \oplus is correctly replaced by the connective \vee in Łukasiewicz logic. \square

THEOREM 5.6. *Let $\{\top, \tilde{p}_1, \dots, \tilde{p}_{n_\eta-1}, \perp\} \subseteq AP$, then the set $\{\wedge, \vee, \Rightarrow, \mathcal{X}, \mathcal{F}, \mathcal{G}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}\}$ is an adequate set for $\text{FTL}(\Pi)$.*

PROOF. As for Łukasiewicz logic, in Product logic the equivalence $\neg\varphi \equiv \varphi \Rightarrow \perp$ holds. Nevertheless, when considering disjunction, we must observe that, differently from Gödel-Dummett and Łukasiewicz logics, \vee cannot be expressed in terms of \wedge (see Hájek [1998b]). Then, it is easy to see that \mathcal{F} and \mathcal{G} are not dual and cannot be expressed by using only operator \mathcal{U} . Indeed, no unbounded operator (except *Soon*) can be expressed in terms of another one. For bounded operators, we can proceed as in the previous cases using lattice conjunction/disjunction for min/max operations, but avoiding the use of the operator \odot^i , as $\odot^i(\varphi) \equiv \varphi \wedge \tilde{p}_i$. Then, for example, we have

$$\mathcal{W}_t\varphi \equiv \bigvee_{0 \leq j \leq t+n_\eta-1} (\mathcal{X}^j\varphi \wedge \tilde{p}_{j-t}),$$

$$\mathcal{L}_t\varphi \equiv \bigvee_{0 \leq j \leq \min\{t, n_\eta - 1\}} \left(\left(\bigwedge_{0 \leq h \leq t-j} \mathcal{X}^h\varphi \right) \wedge \tilde{p}_j \right),$$

and

$$\mathcal{AG}_t\varphi \equiv \bigvee_{j \in I_t} \bigvee_{H \in \mathcal{P}^{t-j}(I_t)} \bigwedge_{h \in H} (\mathcal{X}^h\varphi \wedge \tilde{p}_j),$$

and again observe the correct use of lattice/standard disjunction.

Table V. Adequate Sets for FTL(Z), FTL(G), FTL(L), and FTL(Π)

| Logic | Adequate Set |
|--------------|--|
| FTL(Z) | $\wedge, \neg, \mathcal{X}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}, \odot^1, \dots, \odot^{n_\eta-1}$ |
| FTL(G) | $\wedge, \Rightarrow, \mathcal{X}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}, \odot^1, \dots, \odot^{n_\eta-1}$ |
| FTL(L) | $\wedge, \Rightarrow, \mathcal{X}, \mathcal{F}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}, \odot^1, \dots, \odot^{n_\eta-1}$ |
| FTL(Π) | $\wedge, \Rightarrow, \vee, \mathcal{X}, \mathcal{F}, \mathcal{G}, \mathcal{U}, \mathcal{AG}, \mathcal{AU}$ |

The results of the four previous theorems are summarized in Table V. Finally, note that the adoption of the adequate sets in Table V can possibly cause a superexponential blow-up of the length of the formulae. For example, expanding the lattice disjunctions as showed in equivalences (1) and (2), it is easy to see that the simple formula $\mathcal{AG}_t p$, with $p \in AP$, is equivalent in FTL(Π) to a formula of length $O(3^{2^{t+1}} \cdot t)$ that only contains connectives \wedge, \Rightarrow , and \mathcal{X} .

6. EXAMPLES OF PROPERTIES AND SPECIFICATIONS

This section illustrates how FTL can be applied in practice on a smart grid. In particular, we use FTL to formalize a set of properties that a smart grid should satisfy. Smart grids exploit information about the behavior of suppliers and consumers in an electrical grid to improve the efficiency, reliability, and sustainability of the use of electricity. They can operate on different domains, from the provider level to the customer level. In general, in the customer domain, smart grids are designed to maximize the availability of appliances to ensure that all the required tasks are performed and optimize the consumption of energy to achieve economical and ecological benefits.

Typically, smart grids rely on the Energy Management System (EMS) to periodically compute metering data regarding energy consumption, and they use these data to balance the work load of the appliances. In particular, the EMS sends proper operational control data to the appliances to schedule their tasks and tune their functioning to avoid outages. Such control data are computed by analyzing the metering data. Hence, it becomes important to express requirements about the amount of energy consumed and the availability of appliances. However, to ensure a proper behavior of the overall system, it might be preferable to tolerate a few number of outages or the transient unavailability of some appliances instead of stating strict constraints that would be unrealistic.

The following example defines a set of formulae, under the assumption that the smart grid controls a single appliance (A_1). However, the provided formulae can be easily modified to cover the cases when more than one appliance must be controlled.

The first requirement (R1), which the smart grid should satisfy, is the following: “ A_1 must be available almost always during the day.” To state this property, we need to introduce the (fuzzy) proposition a , which measures whether the availability of A_1 is high. More precisely, if π is the path of the daily minutes, $v_{\pi^i}(a)$ expresses the truth degree of proposition “at the i -th minute of the day, the availability of A_1 is high.” Availability is, in general, measured as the time difference between the instant when a request is issued and the instant when the appliance is active. This time difference can be estimated in seconds, and this makes reasonable the choice of minutes as time granularity. Using this definition of availability, the predicate a can be evaluated as follows. If A_i is the actual time delay at the i -th minute, M_i the mean time delay of the i -th minute of the day computed daily over the last month, and σ_i^2 the variance, let $\Delta_i = A_i - M_i$, then

$$v_{\pi^i}(a) = \begin{cases} \min \left\{ 1, \frac{1}{\sigma_i^2} (\Delta_i + \frac{3}{2}\sigma_i^2) \right\}, & \Delta_i \geq -\frac{3}{2}\sigma_i^2; \\ 0, & \text{otherwise.} \end{cases}$$

An appropriate avoiding function could be

$$\eta(n) = \begin{cases} e^{-(n/20)^2}, & n \leq 20 \\ 0, & \text{otherwise.} \end{cases}$$

Hence, once a and η are defined, the first requirement can be expressed as

$$\mathcal{AG}_{1440} a. \tag{R1}$$

This formula evaluates the availability of A_1 during the day, tolerating exceptions for no more than 20 minutes during the day. Indeed, if the availability is below the average for no more than 4 minutes, then the evaluation of $\mathcal{AG}_{1440} a$ is, *at least*, $e^{-(16/20)^2} \sim 0.53$, *independently* of the worst value of the availability. Observe that if we consider the mean delay calculated all over the day, we may obtain less expressive results, since in case of one big delay, the evaluation of the daily availability will dramatically decrease. Notice also that the proposed formula is more suitable for this context than $\mathcal{G}_{1440} a$, which gives a value corresponding to *at most* the worst availability, when it is evaluated along π .

The second requirement (R2) is that “*as soon as new metering data are available, a new operational control data must be sent by the EMS to A_1 .*” To formalize R2, we consider the crisp propositions d and c . The former is satisfied if new metering data are available, whereas the latter is satisfied if an operational control signal is sent by the EMS to A_1 . Using these propositions, R2 can be expressed as

$$d \Rightarrow \mathcal{W}_1 c. \tag{R2}$$

This formula allows us to tolerate small delays in the transmission of operational control data, differentiating the evaluation of small delays from big ones, which can be considered equivalent to the absence of transmission. Instead, this delay cannot be tolerated by using LTL, in which the same proposition would be expressed as $d \Rightarrow c$ or $d \Rightarrow \mathbf{X}c$. Notice that if we do not evaluate the formula from the first minute, R2 can also be written as $d \Rightarrow \text{Soon } c$. Observe that R2 is implicitly universally quantified over the time ($\mathcal{G}(d \Rightarrow \mathcal{W}_1 c)$). This means that always, as soon as new metering data are available, a new operational control data must be sent.

The last requirement (R3) specifies the following property: “*there is no outage in the day until the energy consumption is low or moderate.*” Let s be a crisp proposition, whose evaluation is 0 if the appliance is disconnected, and let p be the (fuzzy) proposition “the energy consumption is low or moderate.” This fuzzy proposition can be evaluated to 1 if the consumed energy is below a given threshold C . Its value can decrease linearly to 0, when the consumed energy is between C and $C + x$, where $C + x$ is the maximal energy that can be tolerated before evaluating p as 0. Clearly, both C and x have to be assigned considering the parameters of the system. Hence, the third requirement can be formalized as

$$s \mathcal{U}_{1440} \neg p. \tag{R3}$$

If this formula is evaluated at the beginning of each day and holds, it guarantees that there is no outage in case the energy consumption is not high. Notice that if the requirement is modified in “*there is no outage until the energy consumption is low or moderate,*” R3 can be expressed as

$$\mathcal{G}(s \mathcal{U} \neg p). \tag{R3_{mod}}$$

Moreover, in case this requirement is relaxed to “*the outages of the day are negligible until the energy consumption is low or moderate,*” then it can be formalized by using

the fuzzy temporal operator \mathcal{AU} as follows:

$$s \mathcal{AU}_{1440} \neg p. \quad (\mathbf{R3}_{rel})$$

The choice of operator \mathcal{AU}_{1440} is suitable because it tolerates (i.e., its satisfaction does not drop to 0) a few numbers of outages of the appliance that occur during the day.

Finally, as we have already noticed in Section 4.3, the choice of a specific interpretation for the connectives is highly important to get more precise results, although all of the inequalities that we proved are still valid independently of the interpretation. If we consider formula $\mathcal{AG}_{1440} s$ (“the daily number of outages is negligible”), then for the evaluation of formula

$$\mathcal{AG}_{1440} s \vee \chi^{1440} \mathcal{AG}_{1440} s$$

it is quite natural to choose the Zadeh or Gödel-Dummett interpretation instead of the Łukasiewicz interpretation, namely, the truncated sum of their evaluations. This choice avoids the predicates of this formula to “saturate,” since a long sequence of days with many outages cannot be equivalent to a day with no outages.

Notice that this is just a simple example to show the characteristics of FTL and the benefits derived by using it to model the requirements of adaptive systems. However, the framework is suitable to represent much more complex (adaptive) systems, especially if they should satisfy relaxed temporal constraints.

7. CONCLUSIONS

This article introduces FTL, which is used to express vagueness on time. The semantics of the temporal operators provided by FTL is highly flexible, as it allows us to select a particular interpretation for the connectives, which best suits the kind of property to be formalized. We prove that FTL extends LTL, since under the assumption that all events are crisp, FTL reduces to LTL. We show that the temporal operators introduced by our logic respect a set of interesting relations, and we also identify some adequate sets of connectives. In future work, we are investigating verification techniques for checking the truth degree of the FTL formulae of the system under analysis both via an automata-based model [Fiorentini et al. 2012; Frigeri et al. 2012] and via the reduction to a problem tractable by an SMT solver, as done in Bersani et al. [2011]. This technique will modify the traditional reachability analysis, according to the peculiarities of the FTL language. Finally, considering that FTL is particularly suitable for describing requirements of adaptive system, in which vagueness is often embedded with uncertainty, we are planning to investigate the relationship between FTL and probabilistic languages.

REFERENCES

- Stefano Aguzzoli, Brunella Gerla, and Zuzana Haniková. 2005. Complexity issues in basic logic. *Soft Computing* 9, 12, 919–934.
- Matthias Baaz, Petr Hájek, Franco Montagna, and Helmut Veith. 2001. Complexity of t-tautologies. *Annals of Pure and Applied Logic* 113, 1–3, 3–11.
- Luciano Baresi, Liliana Pasquale, and Paola Spoletini. 2010. Fuzzy goals for requirements-driven adaptation. In *Proceedings of the 18th International Requirements Engineering Conference*. IEEE, Los Alamitos, CA, 125–134.
- Marcello M. Bersani, Achille Frigeri, Matteo Rossi, and Pierluigi San Pietro. 2011. Completeness of the bounded satisfiability problem for constraint LTL. In *Reachability Problems*, Lecture Notes in Computer Science, Vol. Springer, 58–71.
- Radu Calinescu and Marta Z. Kwiatkowska. 2009. Using quantitative analysis to implement autonomic IT systems. In *Proceedings of the 31st International Conference on Software Engineering (ICSE'09)*. 100–110.
- Janette Cardoso and Heloisa Camargo. 1998. *Fuzziness in Petri Nets*. Studies in Fuzziness and Soft Computing, Vol. 22. Physica-Verlag, Heidelberg.

- Roberto Cignoli, Francesc Esteva, Lluís Godo, and Antoni Torrens. 2000. Basic fuzzy logic is the logic of continuous t-norms and their residua. *Soft Computing* 4, 2, 106–112.
- Roberto Cignoli and Antoni Torrens. 2003. Hájek basic fuzzy logic and Łukasiewicz infinite-valued logic. *Archive for Mathematical Logic* 42, 4, 361–370.
- Petr Cintula. 2001. About axiomatic systems of product fuzzy logic. *Soft Computing* 5, 3, 243–244.
- Luca De Alfaro, Marco Faella, and Mariëlle Stoelinga. 2009. Linear and branching system metrics. *IEEE Transactions on Software Engineering* 35, 2, 258–273.
- Didier Dubois and Henri Prade. 1982. A class of fuzzy measures based on triangular norms. *International Journal of General Systems* 8, 1, 43–61.
- Didier Dubois and Henri Prade. 1989. Processing fuzzy temporal knowledge. *IEEE Transactions on Systems, Man, and Cybernetics* 19, 4, 729–744.
- Michael Dummett. 1959. A propositional calculus with denumerable matrix. *Journal of Symbolic Logic* 25, 97–106.
- Soumitra Dutta. 1988. An event based fuzzy temporal logic. In *Proceedings of the 18th International Symposium on Multiple-Valued Logic*. IEEE, Los Alamitos, CA, 64–71.
- Nicholas Fiorentini, Achille Frigeri, Liliana Pasquale, and Paola Spoletini. 2012. Time modalities over many-valued logics. In *Proceedings of the 13th Italian Conference on Theoretical Computer Science (ICTCS'12)*.
- Achille Frigeri, Liliana Pasquale, and Paola Spoletini. 2012. An Approach for Evaluating Fuzzy-Time Temporal Properties. Retrieved July 6, 2014, from <https://sites.google.com/site/ftlmodelchecking/>.
- Valery Glivenko. 1929. Sur quelques points de la logique de M. Brouwer. *Bulletin de la Classe des Sciences de l'Académie Royale de Belgique* 15, 183–188.
- Kurt Gödel. 1933. Zum intuitionistischen Aussagenkalkül. *Ainzeiger Akademie der Wissenschaften Wien, Mathematische-naturwissensch* 69, 65–66.
- Petr Hájek. 1998a. Basic fuzzy logic and BL-algebras. *Soft Computing* 2, 124–128.
- Petr Hájek. 1998b. *Metamathematics of Fuzzy Logic*. Kluwer, Dordrecht.
- Petr Hájek, Lluís Godo, and Francesc Esteva. 1996. A complete many-valued logic with product conjunction. *Archive of Mathematical Logic* 35, 3, 191–208.
- Alfred Horn. 1969a. Free L-algebras. *Journal of Symbolic Logic* 34, 475–480.
- Alfred Horn. 1969b. Logic with truth values in a linearly ordered Heyting algebra. *Journal of Symbolic Logic* 34, 395–408.
- Erich P. Klement, Radko Mesiar, and Endre Pap. 2000. *Triangular Norms*. Kluwer, Dordrecht.
- Khaled Ben Lamine and Froduald Kabanza. 2000. Using fuzzy temporal logic for monitoring behaviour-based mobile robots. In *Proceedings of the IASTED International Conference on Robotics and Applications*. 116–121.
- Philip Levis, David Gay, and David Culler. 2005. Active sensor networks. In *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation, Volume 2*. 343–356.
- Xiaoqing F. Liu, Manooch Azmoodeh, and Nektarios Georgalas. 2007. Specification of non-functional requirements for contract specification in the NGOSS framework for quality management and product evaluation. In *Proceedings of the 5th International Workshop on Software Quality*. 36–41.
- Xiaoqing F. Liu and John Yen. 1996. An analytic framework for specifying and analyzing imprecise requirements. In *Proceedings of the 18th International Conference on Software Engineering*. 60–69.
- Carl G. Looney. 1988. Fuzzy Petri nets for rule-based decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics* 18, 1, 178–183.
- Jan Łukasiewicz. 1920. O logice trójwartościowej. *Ruch Filozoficzny* 6, 170–171.
- Jan Łukasiewicz. 1970. Three-valued logic. In *Selected Works by Jan Łukasiewicz*. North-Holland, 87–88.
- Zhiming Ma, Li Yan, and Fu Zhang. 2012. Modeling fuzzy information in UML class diagrams and object-oriented database models. *Fuzzy Sets and Systems* 186, 1, 26–46.
- Seong-ick Moon, Kwang Hyung Lee, and Doheon Lee. 2004. Fuzzy branching temporal logic. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 34, 2, 1045–1055.
- Gero Mühl, Ludger Fiege, and Peter Pietzuch. 2006. *Distributed Event-Based Systems*. Springer-Verlag.
- Subhankar Mukherjee and Pallab Dasgupta. 2013. A fuzzy real-time temporal logic. *International Journal of Approximate Reasoning* 54, 9, 1452–1470.
- Norman W. Paton and Oscar Díaz. 1999. Active database systems. *ACM Computing Surveys* 31, 1, 63–103.
- SmartGrids. 2013. Smart Grids European Technology Platform. Retrieved July 6, 2014, from <http://www.smartgrids.eu/>.

- Helmut Thiele and Susanne Kalenka. 1993. On fuzzy temporal logic. In *Proceedings of the 2nd International Conference on Fuzzy Systems*. IEEE, Los Alamitos, CA, 1027–1032.
- Jon Whittle, Pete Sawyer, Nelly Bencomo, and Betty H. C. Cheng. 2009. RELAX: Incorporating uncertainty into the specification of self-adaptive systems. In *Proceedings of the 17th International Requirements Engineering Conference*. IEEE, Los Alamitos, CA, 79–88.
- Ronald R. Yager. 1980. On a general class of fuzzy connectives. *Fuzzy Sets and Systems*, 235–242.
- Lofti A. Zadeh. 1965. Fuzzy sets. *Information and Control* 8, 3, 338–353.
- Lofti A. Zadeh. 1994. Fuzzy logic (abstract): Issues, contentions and perspectives. In *Proceedings of the 22nd Annual Computer Science Conference on Scaling Up: Meeting the Challenge of Complexity in Real-World Computing Applications*. ACM, New York, NY, 407.