Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning
Main Track

Time Robustness for Point-Based Semantics of Metric Interval Temporal Logic

Simone Silvetti[1] , **Ivan Compagnucci**[2] , **Francesca Cairoli**[1] , **Catia Trubiani**[2] , **Laura Nenzi**[1]
[1]University of Trieste, Trieste, Italy
[2]Gran Sasso Science Institute, L'Aquila, Italy
simone.silvetti@dia.units.it, ivan.compagnucci@gssi.it,
francesca.cairoli@units.it, catia.trubiani@gssi.it, lnenzi@units.it

Abstract

Time-critical systems must satisfy temporal constraints whose correctness depends not only on event ordering but also on precise timing. Metric Interval Temporal Logic (MITL) provides a formalism to express such requirements. Although robustness has been widely studied under signal-based interpretations, it remains largely unexplored for point-based semantics, where executions are sequences of timestamped facts. In this setting, small timing variations may arbitrarily change Boolean satisfaction, revealing the instability of temporal truth under uncertainty.

We introduce a notion of time robustness for MITL over point-based semantics, interpreting robustness as a margin of validity of temporal interpretations. We define a quantitative semantics and prove soundness with respect to Boolean satisfaction together with a Lipschitz stability property with respect to timestamp perturbations, which induces a metric notion of proximity between interpretations. The semantics admits a polynomial-time evaluation procedure and is illustrated on two case studies (drone surveillance and smart hospital), where robustness empirically correlates with tolerance to temporal noise.

1 Introduction

Many domains rely on knowledge involving temporal constraints, ranging from safety-critical controllers to business processes governed by service-level agreements (Huang et al. 2024; Reimann et al. 2011; Xu and Parnas 2002). For example, after a drug injection, a patient's blood pressure may be required to be measured *not earlier than 10 minutes* and *not later than 20 minutes* (Lanz, Weber, and Reichert 2014). Correctness, therefore, depends not only on the ordering of events but also on their precise *timing relations* (Guan, d'Amorim, and Legunsen 2025; Guan and Legunsen 2025; Colombo, Pace, and Schneider 2008). This raises a semantic question: can validity itself be made stable under temporal uncertainty?

Temporal logics have emerged as a powerful formalism for specifying and reasoning about timing-dependent properties. Logics such as Metric Temporal Logic (MTL) (Koymans 1990) and its fragments, including Metric Interval Temporal Logic (MITL) (Alur, Feder, and Henzinger 1996), extend classical temporal logic with explicit quantitative timing intervals. These extensions allow the expression of requirements such as *every request must be acknowledged within 5 time units* or *a safety condition must hold for the next 10 time units*.

In some applications, knowledge is not limited to the occurrence of events but also includes quantitative observations about the system, such as measured temperature, pressure, or position. In these cases, the represented information is no longer a sequence of timestamped facts but a time-indexed evolution of numerical values. Signal Temporal Logic (STL) (Maler and Nickovic 2004) extends metric temporal reasoning to this setting by allowing predicates over real-valued signals, enabling specifications that combine temporal constraints with quantitative conditions.

This distinction highlights two complementary views of temporal knowledge: event-based representations describe *what happened and when*, while signal-based representations describe *how a quantity evolved over time*.

In event-based representations, timestamps are rarely exact: delays in sensing, logging, or distributed recording introduce temporal uncertainty. As a consequence, traces describing essentially the same course of events may differ only by small timing variations while receiving different Boolean truth values. When behaviors are represented as signals, Boolean satisfaction is often replaced by a graded notion of validity. Quantitative semantics for STL assign a real value measuring how far an observation is from violating a specification, thus capturing a *robustness margin* rather than a binary truth value (Fainekos and Pappas 2009; Donzé and Maler 2010). This notion, often referred to as *spatial robustness*, quantifies tolerance with respect to perturbations of signal values. Such measures have also been complemented by the notion of *temporal robustness*, quantifying how much the timing of relevant signal features may vary while preserving satisfaction (Rodionova et al. 2022). Conceptually, these approaches interpret robustness as tolerance of an observed behavior to uncertainty in measurements and timing. They therefore move from exact truth values to a quantitative assessment of validity.

Despite the extensive study of robustness for signal-based temporal semantics, much less attention has been devoted to event-based interpretations, where models consist of discrete timestamped facts. In this setting, temporal specifications are evaluated over the occurrence times of events rather than over a continuous trajectory.

The difference is not merely representational but seman-

tic. Signal-based semantics describes the evolution of observable quantities, and robustness measures tolerance of a behavior to perturbations. In contrast, point-based semantics represents knowledge as timestamped facts describing when events occurred. Here, uncertainty does not concern the value of a measured quantity but rather the recorded time at which a fact is asserted to hold. Consequently, robustness no longer measures how much a physical trajectory can be perturbed, but rather how much event timestamps can vary before the truth value of a specification changes. In this sense, robustness characterizes the reliability of temporal conclusions drawn from imprecise timestamps.

We introduce a notion of *time robustness* for MITL over finite event traces. Robustness quantifies the maximal timestamp perturbation preserving satisfaction and guarantees that bounded temporal variations cannot change the truth value. This interpretation is natural in domains where knowledge is recorded as timestamped facts (e.g., logs or workflows), where uncertainty concerns when events occur rather than signal values. Accordingly, robustness characterizes the reliability of temporal conclusions under timestamp imprecision. The resulting semantics provides a quantitative notion of proximity between temporal interpretations, illustrated through two case studies.

Contributions. The main contributions of this work can be summarized as follows:

- A quantitative time-robustness semantics for MITL over event-based interpretations (see Section 3, Definition 3).
- Soundness and stability guarantees relating robustness to Boolean satisfaction (see Section 3, Theorems 1 and 2, Corollary 1).
- A polynomial-time evaluation procedure derived from the compositional semantics (see Section 3, Proposition 1).
- Empirical evidence linking robustness to tolerance under temporal uncertainty (see Section 4).

Overall, the proposed semantics equips temporal interpretations with a metric notion of validity, enabling reasoning over temporally uncertain knowledge.

2 Related Work

Signal-based Time Robustness. While extensive research has been devoted to the notion of spatial robustness for the satisfaction of temporal logic specifications under the signal-based interpretation (Fainekos and Pappas 2009; Donzé and Maler 2010; Silvetti et al. 2018), comparatively limited attention has been given to the formalization of time robustness. The earliest contribution in this direction was presented by Donzé and Maler (2010), where time robustness was characterized in terms of the maximum admissible temporal perturbation, that is, the largest uniform temporal shift of the entire trace, either forward or backward, that does not change the truth value of a target formula. Later, Rodionova et al. (2022) extended this concept by permitting asynchronous temporal displacements among the individual signals constituting the trace. Although robustness notions for STL are defined over dense-time signals and may appear related to discrete behaviors, the two settings capture different semantic phenomena. In signal-based semantics, time robustness measures admissible perturbations of continuous trajectories, whereas in the point-based interpretation perturbations affect the timestamps of discrete events while preserving their structural relations. Accordingly, our robustness characterizes the stability of temporal truth under timestamp uncertainty rather than tolerance of signal behavior.

Point-based Spatial Robustness. The most closely related work to ours is (Fainekos and Pappas 2006), which introduced a quantitative robustness semantics for MTL over timed state sequences. In their framework, states take values in a metric space, and robustness quantifies the distance of the observed state values from the boundary of satisfaction. This notion is spatial: it captures how much the state component of a trace can be perturbed before the truth value of a specification changes, while timestamps remain fixed. Our work differs in two respects. First, we consider event traces whose states are discrete labelings (subsets of atomic propositions) rather than elements of a continuous metric space, which is natural in domains where knowledge is recorded as timestamped facts. Second, we define robustness along the temporal dimension as the maximal perturbation to event timestamps that preserves satisfaction. The two notions are therefore complementary: spatial robustness addresses uncertainty in what is observed, while time robustness addresses uncertainty in when it is observed. To the best of our knowledge, no existing approach in the point-based setting defines a time robustness that characterizes the stability of Boolean satisfaction under timestamp perturbations.

MTL Trace Alignment. The trace alignment problem in MTL (Giacomo et al. 2021) and our notion of time robustness represent two perspectives on the same fundamental issue. Trace alignment seeks the minimal modifications needed to make a log trace compliant with a given temporal specification, while our time robustness approximates the maximum perturbation that can be applied to a trace while preserving the (un)satisfiability of a logical requirement.

3 Metric Interval Temporal Logic and Time Robustness

Finite Event Trace. Let $\mathsf{AP} = \{\mathtt{p}_0, \mathtt{p}_1, \ldots, \mathtt{p}_{m-1}\}$, with $m \in \mathbb{N}$, denote a finite set of atomic propositions. We define an *event* as a pair $(t, s) \in \mathbb{R}_{\geq 0} \times 2^{\mathsf{AP}}$, where t is the time when the event occurs, and s denotes the event state, describing the properties that hold at that event. For example, the event state $\{\mathtt{a}, \mathtt{b}\}$ indicates that both predicates $\mathtt{a}$ and $\mathtt{b}$ are true at the event. A *finite event trace* is a finite sequence of events $\sigma = (t_0, s_0)(t_1, s_1) \ldots (t_{n-1}, s_{n-1})$ where $\{t_i\}_{0 \leq i < n}$ is a strictly increasing sequence of non-negative real numbers, with $t_0 = 0$ and $n \in \mathbb{N}$. By Σ_{AP} we denote the set of finite event traces composed of events with states in 2^{AP}, e.g., $\sigma = (0, \{\mathtt{a}\})(10, \{\mathtt{b}, \mathtt{c}\})(20, \{\mathtt{d}\})$ is an event trace with 3 events. We use the notation $\vec{t}(\sigma)$ and $\vec{s}(\sigma)$ to denote, respectively, the time vector and the state vector of σ, $\sigma[i]$ to denote the i-th event in trace σ, and $\| \cdot \|_1$ to

denote the standard ℓ^1 norm.

Logic. MITL (Alur, Feder, and Henzinger 1996) is a fragment of MTL (Koymans 1990) where singular intervals such as $[a, a]$ are not allowed. Its syntax is defined as follows.

Definition 1 (MITL Syntax)**.** *MITL formulae are defined by the following grammar:*

$$\varphi := \top \mid \mathtt{p} \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \mathbf{U}_I \varphi \mid \mathbf{X}_I \varphi$$

where $\top$ is the true constant, $p \in \mathsf{AP}$ is an atomic proposition, $\neg$ and $\wedge$ the Boolean connectives, $\mathbf{U}_I$ the until *operator and $\mathbf{X}_I$ the next operator where $I = [a, b]$ or $I = [a, +\infty)$ is an interval with $0 \leq a < b < +\infty$.*

In addition, we derive the eventually operator $\mathbf{F}_I\varphi := \top \mathbf{U}_I \varphi$, the always operator $\mathbf{G}_I\varphi := \neg\mathbf{F}_I\neg\varphi$, the disjunction operator $\varphi_1 \vee \varphi_2 := \neg(\neg\varphi_1 \wedge \neg\varphi_2)$ and the implication operator $\varphi_1 \rightarrow \varphi_2 := \neg\varphi_1 \vee \varphi_2$. Moreover, we write $\mathbf{G}$ and $\mathbf{F}$ instead of $\mathbf{G}_{[0,+\infty)}$ and $\mathbf{F}_{[0,+\infty)}$.

To motivate our work, consider the following example. In many process-oriented systems, a key property concerns the temporal relationship between a request and its corresponding satisfaction. Such a requirement can be formally expressed as $\varphi = \mathbf{G}(\mathtt{req} \rightarrow \mathbf{F}_I \mathtt{sat})$, where $I \subseteq \mathbb{R}^+$ represents the admissible time interval between the request and its satisfaction. The interpretation of I varies depending on the system under consideration. For example, in a *mortgage release process*, customers typically request a loan quote, and the bank subsequently issues an offer. A key performance indicator (KPI) in this context measures the delay between these two events, where the offer is expected to be provided within a specified upper bound after the request is made. In *client–server systems*, a similar pattern captures response-time requirements, ensuring that the server acknowledges or responds to a client request before a defined deadline. In both cases, the interval bound takes the form $I = [0, T]$, indicating that responses may occur at any time up to the deadline T, with no lower bound on how early they can occur. In contrast, in *chemical processes*, the desired reaction or mixture must reach a stable state within a specific time window $I = [T_0, T_1]$: achieving it too early may induce instability, whereas a delayed response can degrade performance or reduce yield.

In this work, we interpret MITL formulae over finite event traces with the following Boolean *point-based semantics*.

Definition 2 (MITL Boolean semantics)**.** *Let $\varphi \in \mathcal{L}_{MITL}$ be a MITL formula. The satisfaction relation $(\sigma, i) \models \varphi$, indicating that the event trace $\sigma = (t_0, s_0), \ldots, (t_{n-1}, s_{n-1})$ satisfies φ at position $i < n$, is defined inductively as follows:*

$$\begin{aligned}
&(\sigma, i) \models \top \\
&(\sigma, i) \models \mathtt{p} \ \textit{if}\ \mathtt{p} \in s_i \\
&(\sigma, i) \models \neg\varphi \ \textit{if}\ (\sigma, i) \not\models \varphi \\
&(\sigma, i) \models \varphi_1 \wedge \varphi_2 \ \textit{if}\ (\sigma, i) \models \varphi_1 \ \textit{and}\ (\sigma, i) \models \varphi_2 \\
&(\sigma, i) \models \varphi_1 \mathbf{U}_I \varphi_2 \ \textit{if}\ \exists j \geq i, (\sigma, j) \models \varphi_2,\ t_j - t_i \in I, \\
&\qquad\qquad \forall k \in [i, j),\ (\sigma, k) \models \varphi_1 \\
&(\sigma, i) \models \mathbf{X}_I \varphi \ \textit{if}\ i < n-1 \ \textit{and}\ (\sigma, i+1) \models \varphi, \ \textit{and} \\
&\qquad\qquad t_{i+1} - t_i \in I
\end{aligned}$$

Remark 1 (Event witnessing in point-based semantics)**.** *Under point-based semantics, temporal operators are evaluated only at event indices and are witnessed exclusively by existing events. In particular,*

$$(\sigma, i) \models F_I \top$$

holds if and only if there exists an index $j \geq i$ such that $t_j - t_i \in I$. Equivalently, the formula $\top U_I \top$ holds at position i if and only if there exists an event $j \geq i$ whose timestamp lies within the interval $t_i + I$.

This differs fundamentally from signal-based semantics, where $\top$ holds at every real instant and $F_I\top$ is therefore a tautology. In the point-based interpretation, if no event occurs within the time window $t_i + I$, the formula is false.

Example 1. *Consider the formula*

$$\varphi = \mathbf{G}(\mathtt{req} \rightarrow \mathbf{F}_{[10,20]}\mathtt{sat})$$

and the three event traces

$$\begin{aligned}
\sigma_1 &= (0, \{\mathtt{s}\}), (5, \{\mathtt{req}\}), (16, \{\mathtt{sat}\}) \\
\sigma_2 &= (0, \{\mathtt{s}\}), (5, \{\mathtt{req}\}), (22, \{\mathtt{sat}\}) \\
\sigma_3 &= (0, \{\mathtt{s}\}), (5, \{\mathtt{req}\}), (20, \{\mathtt{sat}\})
\end{aligned}$$

illustrated in Figure 1. All the executions satisfy the requirement, since the satisfaction *event (`sat`) occurs between 10 and 20 time units after the* request *event (`req`). However, the three executions differ in their behavior. Suppose we introduce a time perturbation to the satisfaction event (shown in Figure 1 with dotted lines), allowing it to occur slightly earlier or later by the same amount in each trace. In that case, it becomes clear that, under the same perturbation, σ_3 is more likely to satisfy φ than σ_1 or σ_2. Specifically, σ_3 can tolerate a perturbation of up to 5 time units without violating the requirement, while σ_1 and σ_2 can only tolerate perturbations of 1 and 3 time units, respectively.*

In the context of process-aware information systems, such as Business Process Management, this tolerance can be interpreted as the resilience of a trace to satisfy the requirements despite delays or early occurrence of events that may be caused by resource de-allocation, anticipation of precedent activities, or other scheduling fluctuations. This ability of the system to generate event traces that are robust under such perturbations is a desirable quality, which motivated us to introduce the concept of time robustness of event traces concerning the satisfiability of a MITL formula.

Intuitively, the robustness of temporal operators is determined by the event that maximizes the admissible temporal

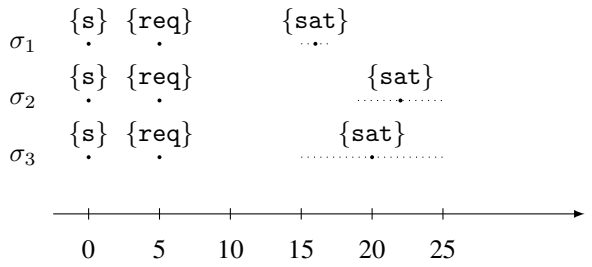


Figure 1: Representation of the three event traces σ_1, σ_2, and σ_3, where each event is indicated by a dot placed at the corresponding time location.

perturbation while preserving satisfaction. Boolean connectives propagate robustness compositionally, while temporal operators combine the admissible perturbation induced by interval constraints with the robustness of their subformulas. For the until operator, robustness aggregates the margin by which the witness event satisfies the interval constraint and the minimal robustness of the prefix. For the next operator, robustness reflects the admissible displacement between consecutive events constrained by the interval I.

Definition 3 (Time robustness of MITL formulae)**.** *We introduce the time robustness as the function* $\rho : \mathcal{L}_{MITL} \times \Sigma_{\mathsf{AP}} \times \mathbb{N} \to \mathbb{R} \cup \{\pm\infty\}$ *defined recursively as follows:*

$$\rho(\top, \sigma, i) = +\infty$$

$$\rho(p, \sigma, i) = \begin{cases} +\infty & \text{if } (\sigma, i) \models p, \\ -\infty & \text{otherwise.} \end{cases}$$

$$\rho(\neg\varphi, \sigma, i) = -\rho(\varphi, \sigma, i)$$

$$\rho(\varphi_1 \wedge \varphi_2, \sigma, i) = \min(\rho(\varphi_1, \sigma, i), \rho(\varphi_2, \sigma, i))$$

$$\rho(\varphi_1 \mathbf{U}_I \varphi_2, \sigma, i) = \max_{j \geq i} \Big(\min\Big(\rho(\varphi_2, \sigma, j),\ d(t_i, t_j, I), \min_{i \leq k < j} \rho(\varphi_1, \sigma, k) \Big) \Big)$$

$$\rho(\mathbf{X}_I \varphi, \sigma, i) = \begin{cases} \min(\rho(\varphi, \sigma, i+1), d(t_i, t_{i+1}, I)) & \text{if } i < n-1, \\ -\infty & \text{otherwise.} \end{cases}$$

where

$$d(t_i, t_j, [a, b]) = \begin{cases} t_i + b - t_j & \text{if } a = 0, \\ \min(t_j - t_i - a, t_i + b - t_j) & \text{otherwise.} \end{cases}$$

As a consequence, we obtain

$$\rho(\mathbf{F}_I \varphi, \sigma, i) = \max_{j \geq i} \Big(\min\big(\rho(\varphi, \sigma, j), d(t_i, t_j, I)\big) \Big),$$

$$\rho(\mathbf{G}_I \varphi, \sigma, i) = \min_{j \geq i} \Big(\max\big(\rho(\varphi, \sigma, j), -d(t_i, t_j, I)\big) \Big).$$

Remark 2. *The function* $d(t_i, t_j, [a, b])$ *quantifies the alignment between* t_j *and the interval* $[t_i + a, t_i + b]$*. When positive, it represents the safety margin keeping* t_j *inside the interval; when negative, it gives the minimal adjustment needed to place* t_j *within the interval. Figure 2 provides visual intuition. Its definition distinguishes two cases. When* $a = 0$*, the interval* $[t_i, t_i + b]$ *is anchored at the current time, so* t_j *cannot precede the interval's left endpoint; the margin is therefore determined solely by the distance to the upper bound* $t_i + b$*. When* $a > 0$*, the interval constrains* t_j *from both sides, so the margin is the minimum of the distances to both bounds.*

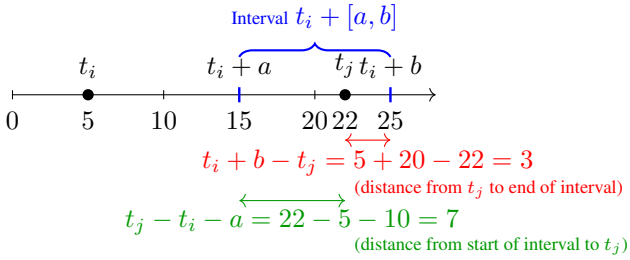


Figure 2: Visualization of the distance function $d(t_i, t_j, [a, b])$ showing the two components: $t_i + b - t_j$ (red) and $t_j - t_i - a$ (green). The function returns the minimum of these two distances, which in this case with $t_i = 5$, $t_j = 22$, and $[a, b] = [10, 20]$ equals $\min(3, 7) = 3$.

We now state an important property of the function $d(t_i, t_j, [a, b])$, expressed in the following lemma.

Lemma 1 (ℓ^1-lipschitzianity of the distance function)**.** *Let us consider two vectors* $\vec{v}$ *and* $\vec{w}$ *of strictly increasing elements both of size* $n < +\infty$*, i.e.,* $\vec{v} = \{(v_0, \ldots, v_{n-1}) \mid v_0 < \cdots < v_{n-1}\}$ *and* $\vec{w} = \{(w_0, \ldots, w_{n-1}) \mid w_0 < \cdots < w_{n-1}\}$*, and any interval* $[a, b]$ *with* $a < b < +\infty$ *then:*

$$\forall\, 0 \leq i < j \leq n,\ |d(v_i, v_j, [a, b]) - d(w_i, w_j, [a, b])| \leq \|\vec{v} - \vec{w}\|_1.$$

Proof. We distinguish two cases.

If $a = 0$, then $|d(v_i, v_j, [0, b]) - d(w_i, w_j, [0, b])| = |v_i + b - v_j - (w_i + b - w_j)| \leq |v_i - w_i| + |v_j - w_j| \leq \|\vec{v} - \vec{w}\|_1$.

If $a \neq 0$, observe that $d(v_i, v_j, [a, b]) = \frac{b-a}{2} - |v_j - v_i - \frac{a+b}{2}|$, which follows from $\min(x, y) = \frac{x+y-|x-y|}{2}$. Hence,

$$\begin{aligned} &|d(v_i, v_j, [a, b]) - d(w_i, w_j, [a, b])| \\ &= \left| \left| w_j - w_i - \frac{a+b}{2} \right| - \left| v_j - v_i - \frac{a+b}{2} \right| \right| \\ &= |w_j - w_i - (v_j - v_i)| \\ &\leq |v_j - w_j| + |v_i - w_i| \\ &\leq \|\vec{v} - \vec{w}\|_1, \end{aligned}$$

where the first inequality uses $||x| - |y|| \leq |x - y|$. □

Remark 3. *In signal-based semantics, time robustness quantifies admissible temporal shifts of signals, either synchronously (the entire signal is shifted) or asynchronously, as in (Rodionova et al. 2022). In contrast, under point-based semantics, we define time robustness with respect to perturbations of individual event timestamps. This captures variations in the occurrence time of facts, a notion that is not naturally represented in signal-based semantics.*

An important property of a quantitative semantics is soundness with the Boolean semantics, which we show in the following theorem.

Theorem 1 (Soundness of time robustness)**.** *The time robustness is sound with respect to Boolean semantics of MITL, meaning that for each formula φ, event trace σ and $i < |\sigma|$ we have:*

$$\rho(\varphi,\sigma,i) > 0 \Rightarrow (\sigma,i) \models \varphi \quad (1)$$
$$\rho(\varphi,\sigma,i) < 0 \Rightarrow (\sigma,i) \not\models \varphi \quad (2)$$
$$(\sigma,i) \models \varphi \Rightarrow \rho(\varphi,\sigma,i) \geq 0 \quad (3)$$
$$(\sigma,i) \not\models \varphi \Rightarrow \rho(\varphi,\sigma,i) \leq 0. \quad (4)$$

Proof. We prove the theorem by using induction on the complexity of the formula.

(1) a) For atomic propositions, the theorem is proved by definition.
b) If $\rho(\neg\varphi,\sigma,i) = -\rho(\varphi,\sigma,i) > 0$ then $\rho(\varphi,\sigma,i) < 0$ which implies, by the induction hypothesis, that $(\sigma,i) \not\models \varphi$ and, by Definition 2 of the Boolean semantics, that $(\sigma,i) \models \neg\varphi$.
c) If $\rho(\varphi_1 \wedge \varphi_2,\sigma,i) = \min(\rho(\varphi_1,\sigma,i),\rho(\varphi_2,\sigma,i)) > 0$ then both $\rho(\varphi_1,\sigma,i) > 0$ and $\rho(\varphi_2,\sigma,i) > 0$ meaning that, by the induction hypothesis, $(\sigma,i) \models \varphi_1$ and $(\sigma,i) \models \varphi_2$, by Definition 2 it implies that $(\sigma,i) \models \varphi_1 \wedge \varphi_2$.
d) If $\rho(\varphi_1 \mathbf{U}_I \varphi_2,\sigma,i) > 0$ then by definition of time robustness there exists $j \geq i$ such that $\rho(\varphi_2,\sigma,j) > 0$, $d(t_i,t_j,I) > 0$ and $\min_{i\leq k<j}(\rho(\varphi_1,\sigma,k)) > 0$ meaning that $\forall i \leq k < j, \rho(\varphi_1,\sigma,k) > 0$. By the induction hypothesis, we have $(\sigma,j) \models \varphi_2$ and $\forall k < j\,(\sigma,k) \models \varphi_1$. Furthermore, by Remark 2 $t_j - t_i \in I$. All the conditions of Definition 2 are satisfied, so that $(\sigma,i) \models \varphi_1 \mathbf{U}_I \varphi_2$.
e) The argument for $\mathbf{X}_I\varphi$ is similar to d).

(2) The proof is similar to (1).

(3) a) For atomic propositions, the theorem is proved by definition.
b) If $(\sigma,i) \models \neg\varphi$ then, by Definition 2, $(\sigma,i) \not\models \varphi$. By using the induction hypothesis, we have $\rho(\varphi,\sigma,i) \leq 0$ and so $\rho(\neg\varphi,\sigma,i) = -\rho(\varphi,\sigma,i) \geq 0$.
c) If $(\sigma,i) \models \varphi_1 \wedge \varphi_2$ then $(\sigma,i) \models \varphi_1$ and $(\sigma,i) \models \varphi_2$ and by induction hypothesis $\rho(\varphi_1,\sigma,i) \geq 0$ and $\rho(\varphi_2,\sigma,i) \geq 0$ which implies that $\rho(\varphi_1 \wedge \varphi_2,\sigma,i) = \min(\rho(\varphi_1,\sigma,i),\rho(\varphi_2,\sigma,i)) \geq 0$.
d) If $(\sigma,i) \models \varphi_1 \mathbf{U}_I \varphi_2$ then by Definition 2 $\exists j \geq i, (\sigma,j) \models \varphi_2$, $t_j - t_i \in I$, $\forall k \in [i,j)$, $(\sigma,k) \models \varphi_1$ and by the induction hypothesis $\rho(\varphi_2,\sigma,j) \geq 0$ and $\min_{i\leq k<j}(\rho(\varphi_1,\sigma,k)) \geq 0$ and, by Remark 2, $d(t_i,t_j,I) \geq 0$. It means that at least one $j \geq i$ we have $\min\left(\rho(\varphi_2,\sigma,j), d(t_i,t_j,I), \min_{i\leq k<j}(\rho(\varphi_1,\sigma,k))\right) \geq 0$ which implies that $\rho(\varphi_1 \mathbf{U}_I \varphi_2,\sigma,i) \geq 0$.
e) The argument for $\mathbf{X}_I\varphi$ is similar to d).

(4) The proof is similar to (3).

□

Theorem 2 (Stability property of time robustness)**.** *Let us consider $\varphi \in \mathcal{L}_{MITL}$ and two trajectories σ and σ' such that $\vec{s}(\sigma) = \vec{s}(\sigma')$, then*

$$\forall i \leq |\sigma|, \quad |\rho(\varphi,\sigma',i) - \rho(\varphi,\sigma,i)| \leq \|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1. \quad (5)$$

When $\rho(\varphi,\sigma',i) \in \{\pm\infty\}$, robustness depends only on the labeling and is therefore invariant under timestamp perturbations.

Proof. The proof is obtained by induction on the structure of the formula, and considers the rewriting of (5) as:

$$\rho(\varphi,\sigma',i) \leq \rho(\varphi,\sigma,i) + \|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1, \quad (6)$$
$$\rho(\varphi,\sigma',i) \geq \rho(\varphi,\sigma,i) - \|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1, \quad (7)$$

for simplicity, let us write $L := \|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1$.

(6) a) If $\rho(p,\sigma,i) = +\infty$ then $(\sigma,i) \models p$; since $\mathbf{s}(\sigma') = \mathbf{s}(\sigma)$ it implies $(\sigma',i) \models p$ and so $\rho(p,\sigma',i) = +\infty$. Moreover, if $\rho(p,\sigma,i) = -\infty$ then $(\sigma,i) \not\models p$; since $\mathbf{s}(\sigma') = \mathbf{s}(\sigma)$ we have $(\sigma',i) \not\models p$ and therefore $\rho(p,\sigma',i) = -\infty$.
b) By applying the induction hypothesis (7) to φ, we obtain $\rho(\neg\varphi,\sigma',i) = -\rho(\varphi,\sigma',i) \leq -\rho(\varphi,\sigma,i) + L = \rho(\neg\varphi,\sigma,i) + L$
c) Applying the induction hypothesis (6) separately to φ_1 and φ_2, we have $\rho(\varphi_1 \wedge \varphi_2,\sigma',i) = \min(\rho(\varphi_1,\sigma',i),\rho(\varphi_2,\sigma',i)) \leq \min(\rho(\varphi_1,\sigma,i) + L, \rho(\varphi_2,\sigma,i) + L) = \min(\rho(\varphi_1,\sigma,i),\rho(\varphi_2,\sigma,i)) + L = \rho(\varphi_1 \wedge \varphi_2,\sigma,i) + L$. The same argumentation can be applied to prove (7) because $\min$ as well as $\max$ are monotonic functions invariant under synchronous translation, i.e., $\min(a+L,b+L) = \min(a,b) + L$.
d) If we look at the definition of $\rho(\varphi_1 \mathbf{U}_I \varphi_2,\sigma',i)$ we notice that it results in nested applications of $\min$ and $\max$ which eventually reach the robustness of φ_1 or φ_2, where the induction hypothesis holds, or a distance function, where the relation $d(t'_i,t'_j,[a,b]) \leq d(t_i,t_j,[a,b]) + L$ is a direct consequence of Lemma 1. The conclusion is straightforward.
e) The proof for $\mathbf{X}_I\varphi$ is similar to d).

(7) The proof follows symmetrically by applying the induction hypothesis to each subformula and using Lemma 1 for the distance function, observing that $\min$ and $\max$ are monotone and preserve the bounds under finite perturbations.

□

The previous theorem establishes a relationship between the magnitude of perturbations that can be applied to a trajectory and the resulting time robustness. In essence, it represents a stability property: if the perturbation remains below a certain threshold, the corresponding difference in robustness will also stay within that threshold. Mathematically:

$$\forall \delta > 0, \quad \|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1 < \delta \Longrightarrow |\rho(\varphi,\sigma',i) - \rho(\varphi,\sigma,i)| < \delta.$$

A direct implication of this theorem is the correctness property of time robustness.

Corollary 1 (Correctness property of time robustness)**.** *Let $\delta > 0$ and $\mathcal{B}(\sigma,\delta) = \{\sigma' \in \Sigma_{\text{AP}} \mid \vec{s}(\sigma') = \vec{s}(\sigma)$ and $\|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1 < \delta\}$. If $\rho(\varphi,\sigma,i) = \delta$ (resp. $\rho(\varphi,\sigma,i) = -\delta$) then $\forall \sigma' \in \mathcal{B}(\sigma,\delta)$ we have $(\sigma',i) \models \varphi$ (resp. $(\sigma',i) \not\models \varphi$).*

Proof. Considering that $\rho(\varphi, \sigma, i) = \delta$ and the stability property of time robustness, we obtain $\|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1 < \rho(\varphi, \sigma, i) \implies |\rho(\varphi, \sigma', i) - \rho(\varphi, \sigma, i)| < \rho(\varphi, \sigma, i)$ which implies $\rho(\varphi, \sigma', i) > 0$ when $\rho(\varphi, \sigma, i) > 0$, and $\rho(\varphi, \sigma', i) < 0$ otherwise.

□

We remark that Theorem 1 does not establish an equivalence when $\rho(\varphi, \sigma, i) = 0$. This is standard in quantitative semantics, where a zero robustness value denotes a boundary case rather than a Boolean truth value, and arbitrarily small timestamp perturbations may affect satisfaction.

The correctness property ensures that if the time robustness value $\rho(\varphi, \sigma, 0)$ is strictly positive (or strictly negative), there exists a neighborhood around the target trajectory σ, denoted in the previous corollary as $\mathcal{B}(\sigma, \rho(\varphi, \sigma, 0))$, that still satisfies (or does not satisfy) the formula φ. This property justifies the term "robustness" for this type of quantitative semantics. It also enables the use of optimization-based methods to address parameter synthesis problems, as is commonly done in the field of continuous signal and relative robustness semantics (Chattopadhyay and Mamouras 2020; Bakhirkin et al. 2017; Fainekos and Pappas 2009).

From a theoretical perspective, time robustness provides a deterministic under-approximation of the set of admissible timestamp perturbations that preserve satisfiability. In practice, this quantity can be interpreted as a resilience margin: if perturbations are applied within this bound, satisfiability is guaranteed. When perturbations exceed this bound, satisfiability is no longer guaranteed but may still occur.

More formally, if $\rho(\varphi, \sigma, 0) = \delta > 0$, Corollary 1 ensures that every perturbed trace σ' satisfying $\|\vec{t}(\sigma') - \vec{t}(\sigma)\|_1 < \delta$ still satisfies φ. Under a stochastic perturbation model Δ on timestamps, this yields the conservative probabilistic bound

$$\mathbb{P}(\sigma + \Delta \models \varphi) \;\geq\; \mathbb{P}\big(\|\Delta\|_1 < \delta\big).$$

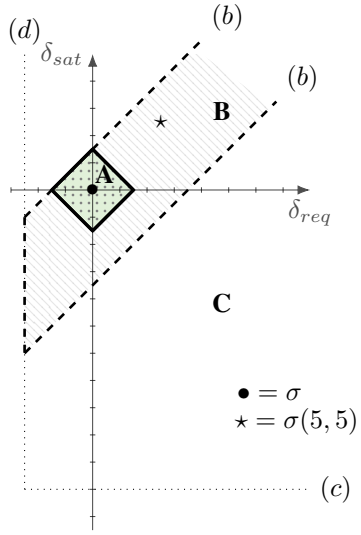
For this reason, in the experimental evaluation we study the empirical relationship between time robustness and the probability of satisfiability under stochastic timestamp perturbations. This allows us to assess whether higher robustness values correlate with increased tolerance to random timing deviations, thereby linking the formal guarantees of the semantics with its practical relevance in monitoring scenarios.

Example 2. *To better illustrate the concept of time robustness, consider again the event trace* $\sigma_2 = (0, \{\mathtt{s}\}), (5, \{\mathtt{req}\}), (22, \{\mathtt{sat}\})$ *and the formula* $\varphi = \mathbf{G}(\mathtt{req} \rightarrow \mathbf{F}_{[10,20]}\mathtt{sat})$ *from Example 1. To identify all possible time displacements of the events* `req` *and* `sat`*, we consider the set*

$$A(\sigma_2, \varphi) = \{\, (\delta_{req}, \delta_{sat}) \in \mathbb{R}^2 \mid \sigma_2(\delta_{req}, \delta_{sat}) \models \varphi \,\},$$

where $\sigma_2(\delta_{req}, \delta_{sat}) = (0, \{\mathtt{s}\}), (5 + \delta_{req}, \{\mathtt{req}\}), (22 + \delta_{sat}, \{\mathtt{sat}\})$*. This leads to the equations shown in Figure 3 (Bottom), whose solution is represented in the chart of Figure 3 (Top). The region enclosed between the dashed lines represents the set of feasible displacements, i.e.,* $A(\sigma_2, \varphi)$*. Region* ***A*** *denotes the subset of feasible displacements within* ℓ^1 *distance smaller than the time robustness value, i.e.,* $B(\sigma_2, \rho(\varphi, \sigma_2, 0))$*. Region* ***B*** *contains additional feasible displacements that still preserve satisfaction of* φ *but lie outside this robustness neighborhood. Region* ***C*** *contains displacements that violate* φ*.*

For instance, the point marked with a star, represented with $\star$*, corresponds to a synchronous delay of both events by 5 time units, which results in a valid event trace. The origin of the chart, represented with* $\bullet$*, corresponds to the original event trace. From this example, it is evident that the robustness region is an under-approximation of the set of time displacements of the target trajectory that preserve the truth value of the formula. Mathematically, this relationship can be expressed as* $B(\sigma_2, \rho(\varphi, \sigma_2, 0)) \subseteq A(\sigma_2, \varphi)$*.*



$$\begin{aligned} D_{sat} &:= \mathbf{t}(\sigma)_{sat} + \delta_{sat} \\ D_{req} &:= \mathbf{t}(\sigma)_{req} + \delta_{req} \\ D_{sat} &> D_{req} && \text{(a)} \\ D_{sat} - D_{req} &\in [10, 20] && \text{(b)} \\ D_{sat} &> 0 && \text{(c)} \\ D_{req} &> 0 && \text{(d)} \end{aligned}$$

Figure 3: Time robustness of the trace $\sigma_2 = (0, \{\mathtt{s}\}), (5, \{\mathtt{req}\}), (22, \{\mathtt{sat}\})$ with respect to $\varphi = \mathbf{G}(\mathtt{req} \rightarrow \mathbf{F}_{[10,20]}\,\mathtt{sat})$. (Top) geometric representation of feasible perturbations in the $(\delta_{req}, \delta_{sat})$ plane. The black dot marks the original trace, while the star denotes a perturbed trace. The region enclosed between dashed lines denotes perturbations preserving satisfaction of φ. Region **A**, shown by the diamond-shaped region, contains perturbations within ℓ^1 distance smaller than $\rho(\varphi, \sigma_2, 0)$. Region **B** contains additional feasible perturbations outside Region **A**, showing that time robustness is an under-approximation of the full feasible set. Region **C** contains perturbations that violate φ. (Bottom) corresponding temporal constraints.

The computed robustness value is an under-approximation of the maximal perturbation preserving satisfaction. This follows from the compositional interpretation of logical connectives via $\min$ and $\max$, which does not capture logical dependencies between subformulas. Consider, for example, the formula $\varphi = \mathbf{F}_I \neg \mathrm{p} \vee \mathbf{F}_J \neg \mathrm{p}$ with $I \cap J = \emptyset$. The trajectory $\sigma = (0, \{\mathtt{s}\})(4, \{\mathtt{p}\})$ satisfies the formula under arbitrary time shifts, yet the robustness value remains finite. We deliberately adopt this conservative approximation to obtain a tractable quantitative semantics; importantly, soundness with respect to Boolean satisfaction and the stability property of Theorem 2 remain unaffected.

Monitoring and Computational Complexity. Definition 3 induces a dynamic-programming monitoring algorithm. For each subformula ψ of φ and each trace position i, the robustness value $\rho(\psi, \sigma, i)$ is computed bottom–up over the syntax tree. Boolean operators ($\neg$, $\wedge$) are evaluated in constant time from previously computed subformula values. For temporal operators, robustness values are computed by scanning the relevant index ranges induced by the interval constraints. In particular, the operators U_I and X_I require linear-time scans over the trace segment satisfying the timing constraints.

Proposition 1 (Computational Complexity)**.** *Let σ be a finite event trace of length n and φ an MITL formula of size $|\varphi|$. All robustness values $\rho(\varphi, \sigma, i)$ for $0 \leq i < n$ can be computed in time $O(n^2 \cdot |\varphi|)$ and space $O(n \cdot |\varphi|)$.*

Proof Sketch. Robustness values are computed via dynamic programming bottom-up over the syntax tree of φ.
Boolean operators: For each index $i < n$, connectives are evaluated in $O(1)$ time using previously stored subformula results.
Temporal operators: For operators such as $\varphi_1 U_I \varphi_2$, the evaluation at each position i requires scanning for a witness $j \geq i$. Since the distance function $d(t_i, t_j, I)$ and the prefix term $\min_{i \leq k < j} \rho(\varphi_1, \sigma, k)$ depend explicitly on both i and j, the search involves $O(n)$ candidate events in the worst case. With $O(|\varphi|)$ subformulas and n positions, the quadratic scan for temporal operators yields an overall complexity of $O(n^2 \cdot |\varphi|)$. Space complexity remains $O(n \cdot |\varphi|)$ to store the DP table.

□

The resulting complexity matches the standard complexity of quantitative monitoring algorithms for signal-based temporal logics, while preserving the additional structure induced by point-based timestamp perturbations.

4 Experimental Evaluation

Our monitoring algorithm directly implements the quantitative semantics introduced in Definition 3. The resulting Python tool, called TIRO, includes detailed replication instructions and is publicly available online (Silvetti et al. 2026). We evaluate it in two case studies: a drone surveillance system and a smart hospital multi-agent system.

4.1 Drone Surveillance System

Context. This is a toy example in which a trajectory records the events of a drone exploring a specific surveillance area. Three disjoint target regions are considered: A, B, and C. When the drone enters a target area, for instance, area A, an event `enterA` is emitted to represent this occurrence.

Trajectories. An example trajectory, depicted in Figure 4, is:

$$\sigma = (0, \{\mathtt{s}\}), (4, \{\mathtt{enterA}\}), (15, \{\mathtt{enterA}\}), (34, \{\mathtt{enterB}\}), (56, \{\mathtt{enterC}\})$$

indicating that the drone starts its path at time 0, then visits region A twice before proceeding through regions B and C.

Properties. An interesting property in this domain is that the surveillance drone must enter the target areas in a specific order while adhering to precise timing constraints, as expressed by the following MITL property.

$$\psi_d = \mathbf{F}_{[10,20]}(\mathtt{enterA} \wedge \mathbf{F}_{[40,50]}(\mathtt{enterB} \wedge \mathbf{F}_{[20,30]}\, \mathtt{enterC}))$$

It is a formula with nested "eventually" operators, specifying that between 10 and 20 seconds, the drone should enter region A; starting from that moment, it should enter region B within the next 40 to 50 time units; and finally, from that point, it should enter region C within the following 20 to 30 seconds.

Experiment and Results. The goal of this use case is to demonstrate how time robustness relates to the robustness of a trajectory under perturbations of its events. To illustrate this correlation, we consider three drone trajectories that satisfy the requirement with different levels of robustness. Afterwards, we apply increasing levels of temporal perturbation to each entry event and analyze how this affects the probability of satisfying the requirement formula. We

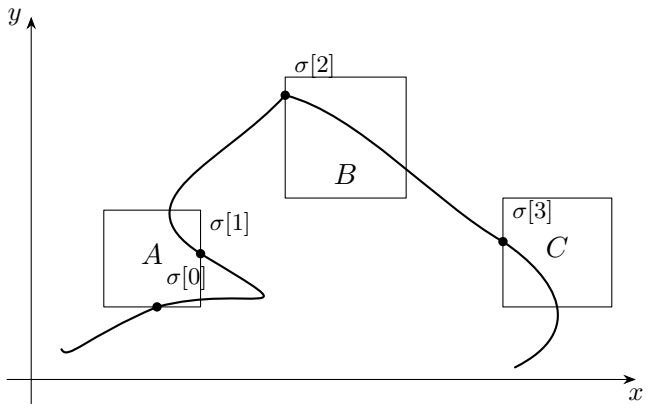


Figure 4: Path σ of the drone visiting the target regions A, B and C. Dots indicate the coordinates corresponding to the entrance in each target region and represent entrance events. The resulting trace is composed of 4 events: $\sigma[0], \sigma[1], \sigma[2], \sigma[3]$.

expect that, under the same degree of perturbation, trajectories with higher robustness will exhibit a higher probability of satisfying the formula. We consider the following three trajectories:

$$\sigma_a = (0, \{\texttt{s}\}), (15, \{\texttt{enterA}\}), (60, \{\texttt{enterB}\}), (85, \{\texttt{enterC}\})$$
$$\sigma_b = (0, \{\texttt{s}\}), (17, \{\texttt{enterA}\}), (61, \{\texttt{enterB}\}), (87, \{\texttt{enterC}\})$$
$$\sigma_c = (0, \{\texttt{s}\}), (13, \{\texttt{enterA}\}), (61, \{\texttt{enterB}\}), (90, \{\texttt{enterC}\})$$

which exhibit decreasing levels of time robustness, as shown in the first column (perturbation 0) of Table 1. For each trajectory, we apply seven perturbation levels, ranging from 0.1 to 6, obtained by adding Gaussian noise with mean 0 and standard deviation equal to the perturbation level to the timestamps of all events in the corresponding trajectory. The results are presented in Table 1, which is organized into three rows, each corresponding to a different trajectory, and reports the average time robustness (with standard deviation) and the probability of satisfaction with respect to ψ_d. These values are computed over 1000 perturbed traces generated for each perturbation intensity. For example, trajectory σ_a perturbed with an intensity of 2.0 yields an average robustness of 1.68 (standard deviation 1.64) and a satisfaction probability of 0.86. As shown in Table 1, when the trajectory is fixed and the perturbation intensity increases, both the average robustness and the probability of satisfaction decrease. Conversely, when the perturbation level is fixed, both quantities are proportional to the robustness of the original (unperturbed) trajectory.

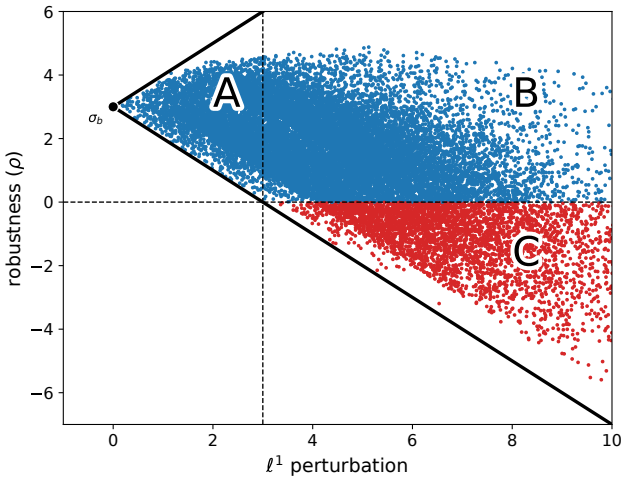


Figure 5: Scatter plot of 20000 perturbed instances of trajectory σ_b at perturbation intensity 2.0. Each point reports the time robustness $\rho(\psi_d, \sigma_b + \delta, 0)$ (y-axis) versus the ℓ^1 norm of the applied perturbation (i.e., $\|\delta\|_1$) (x-axis). The black dot at $(0, 3)$ represents σ_b. All traces with $\|\delta\|_1 < 3$ have positive robustness (region A, satisfaction); violations only occur when $\|\delta\|_1 > 3$ (region C). Points in region B still satisfy ψ_d despite $\|\delta\|_1 > 3$, illustrating that our time robustness is an under-approximation. Blue dots represent traces that satisfy ψ_d; conversely, red dots represent traces that do not satisfy the formula.

As an additional experiment, we considered the specific trajectory σ_b and applied a random perturbation with intensity level equal to 2.0. We generated 20000 perturbed trajectories and report, in Figure 5, the robustness of each trajectory together with the corresponding ℓ^1 norm of the perturbation vector. It is interesting to observe the distribution of the points. First, there are no trajectories with robustness values higher than 5. This is expected, since for the specification ψ_d, the maximum achievable robustness is obtained by σ_a and corresponds to 5. This occurs because the events of σ_a lie exactly at the center of the required temporal intervals: the first event is centered within $[10, 20]$, the second event occurs 45 time units after the first—precisely in the middle of $[40, 50]$—and the final event is 25 units after the second. Second, it is clear that all trajectories with an ℓ^1 perturbation below 3 (recall that $\rho(\psi_d, \sigma_b, 0) = 3$) exhibit positive robustness (region A in the figure), meaning that they satisfy the formula ψ_d. This result aligns with the correctness property of the semantics, i.e., if the applied perturbation is smaller than the absolute value of the robustness, the sign of the robustness remains unchanged. Conversely, all trajectories that violate the formula (region C) have an ℓ^1 perturbation greater than 3. The points in region B correspond to trajectories that still satisfy ψ_d but have ℓ^1 perturbation values greater than 3. This occurs because our notion of time robustness is an under-approximation, as discussed in Example 2.

Finally, all points lie above the line defined by the equation $\rho(\psi_d, \sigma_b+\delta, 0) = 3-\|\delta\|_1$ and below $\rho(\psi_d, \sigma_b+\delta, 0) = 3 + \|\delta\|_1$, where δ denotes the ℓ^1 perturbation of σ_b. This is a clear confirmation of the stability property in the form of Equations (6) and (7), which states that the variation of the robustness is linearly bounded by the ℓ^1 norm of the perturbation, here denoted with δ.

4.2 Smart Hospital Multi-Agent System

Context. We consider a multi-agent simulation of autonomous robots operating in the Ghent University Hospital (Ongenae et al. 2011), modeled as a graph of interconnected rooms. A central coordinator schedules the delivery of medical items from the pharmacy to target rooms. Each item has a priority level (low, medium, high) affecting its handling urgency. Robots can `load`, `move`, and `deliver` items. They may also `fail`, after which another robot can `recover` them. Malicious robots may issue fake repair requests (`fakeFailure`), causing unnecessary interventions and delays. Items are `created`, transported, and eventually `delivered`. If a robot fails during transport, the item may be dropped and later collected by another robot.

Trajectories. As described in the previous paragraph, this use case involves two dynamic entities interacting with each other: the item and the robot. Although these entities interact, we analyze them separately. For example, consider the following two trajectories.

$$\sigma_{item} = (0, \{\texttt{create}\}), (18, \{\texttt{load}\}), (50, \{\texttt{move}\}), (60, \{\texttt{move}\}), (70, \{\texttt{deliver}\})$$
$$\sigma_{robot} = (0, \{\texttt{load}\}), (20, \{\texttt{move}\}), (70, \{\texttt{fail}, \texttt{help}\}), (2100, \{\texttt{recover}\})$$

	Perturbations							
	0	0.1	1.0	2.0	3.0	4.0	5.0	6.0
$\rho(\psi_d, \sigma_a, 0)$	5.0	4.84 (0.08)	3.35 (0.84)	1.68 (1.64)	0.29 (2.47)	-1.46 (3.28)	-3.07 (4.02)	-4.56 (4.77)
$P(\sigma_a \models \psi_d)$	–	1.0	1.0	0.86	0.61	0.39	0.25	0.17
$\rho(\psi_d, \sigma_b, 0)$	3.0	3.00 (0.09)	2.62 (0.87)	2.62 (0.87)	1.30 (1.69)	-0.18 (2.45)	-1.81 (3.38)	-2.96 (4.00)
$P(\sigma_b \models \psi_d)$	–	1.0	1.0	0.79	0.52	0.34	0.26	0.16
$\rho(\psi_d, \sigma_c, 0)$	1.0	1.0 (0.14)	0.46 (1.03)	-0.48 (2.04)	-1.58 (2.92)	-3.14 (3.76)	-4.24 (4.58)	-5.50 (4.99)
$P(\sigma_c \models \psi_d)$	–	1.0	0.71	0.44	0.32	0.23	0.19	0.14

Table 1: Mean±std time-robustness and $P(\sigma \models \psi_d)$, which denotes the empirical probability of satisfaction over perturbed traces for σ_a, σ_b, and σ_c under Gaussian timestamp perturbations.

			φ_a			φ_b			φ_c ($\times 10^3$)		
r	**m**	**m/r**	M	Q_1	Q_3	M	Q_1	Q_3	M	Q_1	Q_3
10	0	0	-558	-55929	-178	-650	-5661	9	-4.9	-6.5	-3.8
10	5	0.5	-574	-3533	-189	-1018	-7247	8	-4.8	-6.1	-3.8
10	10	1	-571	-2033	-201	-1507	-9772	6	-4.9	-6.4	-3.9
20	0	0	-21	-204	20	-316	-4059	10	-4.9	-6.6	-3.8
20	10	0.5	-109	-424	-2	-698	-8275	5	-5.0	-6.4	-3.8
20	20	1	-203	-1093	-42	-1216	-18478	-12	-5.0	-6.6	-3.9
40	0	0	20	20	20	-3	-512	15	-4.1	-5.5	-2.9
40	20	0.5	20	20	20	-14	-606	13	-4.8	-6.2	-3.7
40	40	1	19	-23	20	-237	-1758	4	-4.8	-6.4	-3.8

Table 2: Median (M) and interquartile range (Q_1, Q_3) of time-robustness values for formulas φ_a, φ_b, and φ_c across nine experimental configurations (Use Case 4.2). Column r reports the total number of robots, while m indicates how many of them are malicious ($m \leq r$); m/r denotes their ratio. Values of φ_c are expressed in scientific notation ($\times 10^3$).

σ_{item} represents the lifecycle of an item that is created, loaded, and, after two movements, delivered. Meanwhile, σ_{robot} represents the lifecycle of a robot that loads an item at time 0, starts moving at time 20, and simultaneously fails and issues a help request at time 70. Finally, it is recovered at time 2100.

Properties. To measure the quality (i.e., the level of service) of the autonomous package delivery, we define the following three MITL properties:

$$\varphi_a = \mathbf{G}(\texttt{create} \rightarrow \mathbf{F}_{[0,20]}\, \texttt{load})$$
$$\varphi_b = \mathbf{G}(\texttt{load} \rightarrow \mathbf{F}_{[L,U]}\, \texttt{deliver})$$
$$\varphi_c = \mathbf{G}(\texttt{fail} \rightarrow X_{[0,2500]}\, \texttt{recover})$$

The first two properties, φ_a and φ_b, constrain the system's behavior by imposing timing requirements on item trajectories. Specifically, φ_a expresses that every package must be loaded by a robot within 20 seconds of its creation, while φ_b mandates that, once loaded, a package must be delivered to its destination within a time window $[L, U]$, which depends on the destination, as delivery times naturally increase for more distant locations. To establish reasonable bounds, we estimate them using a reference configuration without malicious robots and zero failures. In this ideal scenario, robots perform at their optimal level. We evaluate the delivery time for each target destination and define L and U as the average time minus and plus two standard deviations, respectively. This way, the specification represents the system's expected performance in the absence of faults, and deviations from this baseline, caused by failures or malicious robots, are directly reflected in the robustness value. The third property (φ_c), instead, is interpreted over robot trajectories and requires that if a robot fails, it must be recovered within 2500 seconds, a bound that incorporates the 1800-second repair time.

Experiment and Results. Each formula is evaluated over all available trajectories, where the number of trajectories corresponds to the number of items delivered during a one-month simulation for properties φ_a and φ_b. For property φ_c, instead, the number of trajectories corresponds to the number of robots; each trajectory tracks the events generated by a single robot over one month. To improve statistical significance, each monthly trajectory is divided into

30 daily sub-trajectories (one per day). Table 2 reports the time-robustness statistics for the three formulas across nine configurations, obtained by varying the number of malicious robots. We use the central communication mechanism and the same simulation parameters as in (Pinciroli and Trubiani 2023). The results are consistent with the findings reported in (Pinciroli and Trubiani 2023). The large negative robustness values observed for φ_b, and, to a lesser extent, for φ_a in low-robot configurations, reflect delays accumulated under failures and malicious behavior. Since the bounds $[L, U]$ are calibrated on an ideal scenario with no failures or malicious robots, dropped items and fake repair requests can substantially increase load–deliver delays. Moreover, because φ_b is under a global operator, a single severely delayed delivery can drive the robustness of the whole trajectory to a large negative value. This explains the asymmetric interquartile ranges in Table 2: Q_3 often stays close to zero, while Q_1 captures the tail of severely delayed deliveries, which decreases as the number of robots increases.

When the total number of robots is fixed and the percentage of malicious robots increases, both properties φ_a and φ_b show a decrease in satisfaction, as reflected by a lower median time-robustness value. This trend is particularly evident for φ_b, which captures the timing of package deliveries. Moreover, the results clearly show, as expected, that the total number of robots is positively correlated with the median time robustness of both formulas. Indeed, when more robots are available, it is more likely that one is ready to load a package (property φ_a) or to complete a delivery in case another robot fails and drops the item (property φ_b). On the other hand, property φ_c exhibits only minor variations with respect to both the number of robots and the fraction of malicious ones.

5 Conclusion

This paper introduced a quantitative semantics for Metric Interval Temporal Logic (MITL) under point-based interpretation, extending the concept of time robustness to event-based systems. The proposed definition quantifies the tolerance of an event trace to temporal perturbations while preserving the satisfaction of a temporal specification. We proved that the semantics is sound with respect to the Boolean interpretation and satisfies key properties, such as correctness and bounded sensitivity to event shifts. A monitoring algorithm implementing the proposed semantics was developed and validated through two representative case studies: a drone surveillance system and a smart hospital multi-agent system. The experiments demonstrated how time robustness can effectively capture the resilience of system behaviors to timing deviations and provide an interpretable measure of quality for event-based systems.

Future research will first investigate the use of the proposed notion of time robustness to define control synthesis problems, following approaches already developed for other types of quantitative semantics. A second promising direction concerns the implementation of an optimal monitoring algorithm and the exploration of its applicability to runtime monitoring scenarios.

AI Declaration

The authors used Claude (Anthropic) to assist with proofreading and language editing of this manuscript. All scientific content, results, and conclusions are solely the responsibility of the authors.

Acknowledgments

This work has been partially funded by the MUR-PRIN project 20228FT78M DREAM (modular software design to reduce uncertainty in ethics-based cyber-physical systems), and MUR Department of Excellence 2023 – 2027 for GSSI.

References

Alur, R.; Feder, T.; and Henzinger, T. A. 1996. The Benefits of Relaxing Punctuality. 43(1):116–146.

Bakhirkin, A.; Ferrère, T.; Maler, O.; and Ulus, D. 2017. On the quantitative semantics of regular expressions over real-valued signals. In *International Conference on Formal Modeling and Analysis of Timed Systems*, 189–206. Springer.

Chattopadhyay, A., and Mamouras, K. 2020. A verified online monitor for metric temporal logic with quantitative semantics. In *International Conference on Runtime Verification*, 383–403. Springer.

Colombo, C.; Pace, G. J.; and Schneider, G. 2008. Dynamic event-based runtime monitoring of real-time and contextual properties. In *International Workshop on Formal Methods for Industrial Critical Systems*, 135–149. Springer.

Donzé, A., and Maler, O. 2010. Robust Satisfaction of Temporal Logic over Real-Valued Signals. In *Formal Modeling and Analysis of Timed Systems*, 92–106.

Fainekos, G. E., and Pappas, G. J. 2006. Robustness of Temporal Logic Specifications. In Havelund, K.; Núñez, M.; Roşu, G.; and Wolff, B., eds., *Formal Approaches to Software Testing and Runtime Verification*, 178–192.

Fainekos, G. E., and Pappas, G. J. 2009. Robustness of temporal logic specifications for continuous-time signals. *Theoretical Computer Science* 410(42):4262–4291.

Giacomo, G. D.; Murano, A.; Patrizi, F.; and Perelli, G. 2021. Timed trace alignment with metric temporal logic over finite traces. In *KR*, 227–236.

Guan, K., and Legunsen, O. 2025. Instrumentation-driven evolution-aware runtime verification. In *International Conference on Software Engineering (ICSE)*, 636–636.

Guan, K.; d'Amorim, M.; and Legunsen, O. 2025. Faster explicit-trace monitoring-oriented programming for runtime verification of software tests. *Proceedings of the ACM on Programming Languages* 9(OOPSLA2):3696–3725.

Huang, Y.; Chen, X.; Jin, Z.; and Zhou, T. 2024. Enabling efficient real-time requirements inconsistency detection for safety-critical systems. In *International Requirements Engineering Conference Workshops (REW)*, 222–229.

Koymans, R. 1990. Specifying Real-Time Properties with Metric Temporal Logic. *Real-Time Syst.* 2(4):255–299.

Lanz, A.; Weber, B.; and Reichert, M. 2014. Time patterns for process-aware information systems. *Requir. Eng.* 19(2):113–141.

Maler, O., and Nickovic, D. 2004. Monitoring temporal properties of continuous signals. In *International symposium on formal techniques in real-time and fault-tolerant systems*, 152–166. Springer.

Ongenae, F.; Myny, D.; Dhaene, T.; Defloor, T.; Van Goubergen, D.; Verhoeve, P.; Decruyenaere, J.; and De Turck, F. 2011. An ontology-based nurse call management system (oNCS) with probabilistic priority assessment. *BMC health services research* 11:26.

Pinciroli, R., and Trubiani, C. 2023. Performance Analysis of Fault-Tolerant Multiagent Coordination Mechanisms. *IEEE Trans. Ind. Informatics* 19(9):9821–9832.

Reimann, F.; Lukasiewycz, M.; Glass, M.; Haubelt, C.; and Teich, J. 2011. Symbolic system synthesis in the presence of stringent real-time constraints. In *Proceedings of the Design Automation Conference*, 393–398.

Rodionova, A.; Lindemann, L.; Morari, M.; and Pappas, G. 2022. Temporal robustness of temporal logic specifications: Analysis and control design. *ACM Transactions on Embedded Computing Systems* 22(1):1–44.

Silvetti, S.; Nenzi, L.; Bartocci, E.; and Bortolussi, L. 2018. Signal convolution logic. In *International Symposium on Automated Technology for Verification and Analysis*, 267–283. Springer.

Silvetti, S.; Compagnucci, I.; Cairoli, F.; Trubiani, C.; and Nenzi, L. 2026. TIRO: Time Robustness for Point-Based Semantics of Metric Interval Temporal Logic. https://github.com/LogArtLab/tiro/.

Xu, J., and Parnas, D. L. 2002. On satisfying timing constraints in hard-real-time systems. *IEEE Transactions on Software Engineering* 19(1):70–84.