# A Verification Framework for Obstruction, Probability, and Time

Anonymous Author(s)
Submission Id: 1252

## ABSTRACT

Verifying strategic behaviour in real-time multi-agent systems under uncertainty is vital for safety- and security-critical domains. Existing obstruction logics treat either adversarial timing (TOL) or probabilistic risk (POTL), but real scenarios require both. We introduce *Probabilistic Timed Obstruction Temporal Logic* (PTOTL), which unifies dense time, probabilities, and cost-bounded obstruction for real-time security games. Interpreted over *Weighted Probabilistic Timed Automaton* (WPTA), PTOTL models attacker–defender interactions where discrete actions and time elapse evolve, and the defender may disable transitions under a per-step budget. We give syntax and semantics, and a symbolic model-checking procedure on a probabilistic zone graph. Despite added strategic and probabilistic features, verification remains PSPACE, not highter than PTCTL or PTATL while offering greater temporal expressiveness. An automotive Moving Target Defense (MTD) case study demonstrates practicality as a specification and verification language.

## KEYWORDS

Strategic reasoning, Cybersecurity, Model checking, Quantitative verification, Security games, Moving Target Defense mechanisms

**ACM Reference Format:**
Anonymous Author(s). 2026. A Verification Framework for Obstruction, Probability, and Time. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 10 pages.

## 1 INTRODUCTION

*Cyber–physical-system* (CPS) requires guarantees under timing, stochasticity, and purposeful interference, particularly when implementing security measures. With autonomous, intelligent, and asynchronous components modeled as *Multi-Agent Systems* (MAS), strategic interactions become explicit while preserving clocks, probabilities, and resource limits. Timed automata with zone abstractions capture dense time [3], and probabilistic model checking quantifies risk on *Markov decision processes* (MDP) and *Probabilistic Timed Automata* (PTA) [12, 49]. In this context, security questions are strategic: *which* agents can enforce *which* objectives *within* a deadline. *Alternating-time Temporal Logic* (ATL) formalizes coalitional ability [4] to work on these strategic conditions; timed TATL adds deadlines [34]; and *Strategic Timed CTL* (STCTL) extends CTL with strategy binders under dense time [8]. Despite differences, these logics reason about time-bounded ability on a fixed arena; in particular, they characterize attacker *choices*, not how a defender can *reshape* the arena under operational constraints. We therefore use *Attack graphs* (AG) as the concrete modeling layer for attacker behavior, while reserving the logical layer for specification and verification of defender objectives. AG support reasoning about security scenarios by encoding multi-stage intrusions as paths over conditions and exploits [39]. While AG handle reachability, cut sets, and time-to-compromise, they are static: they assume a fixed configuration which is a real problem given the evolution of attacks in modern campaigns, where attackers exploit knowledge of these static defenses. To address this, *dynamic maneuver* becomes essential: periodic changes in topology, or exposure invalidate routes learned by an attacker within security boundaries, increase the attacker's per-step cost, and devalue reconnaissance, all under real-time CPS constraints. *Moving Target Defense* (MTD) techniques operate this shift via time-bounded reconfiguration, isolation, route randomization, and service relocation to truncate or delay attack paths and reduce compromise probabilities [26, 47, 56], under constraints. However, without a principled model, the orchestration of these techniques remains heuristic. We must therefore be able to formally specify, from the defender's all this objectives under explicit constraints, e.g., "within $T$ time units the probability of reaching a critical asset remains below $\alpha$ when at most $n$ actions can be inhibited per step". This requires *evolving* AG where suppressing an exploit opportunity corresponds to removing an edge and eliminating its outcome. *Obstruction* and *sabotage* logics formalize this dynamic action view: a player may delete or disable a bounded set of edges and reason about residual paths [9, 20, 46, 55]. Yet timed strategic logics (TATL, STCTL) lack cost-bounded edge disabling with explicit accounting of removed probability mass [8, 34]. Probabilistic real-time logics such as *Probabilistic Timed CTL* (PTCTL) quantify timed behaviors of PTA but keep transitions immutable [38, 49]. Recent obstruction logics treat dense time *Timed Obstruction Logic* (TOL) or probabilities *Probabilistic Obstruction Temporal Logic* (POTL) in isolation, not their combination [21, 44, 45]. We therefore need a logic that *unifies* dense time, quantitative thresholds, and real-time cost-bounded obstruction.

To fill this gap, we propose PTOTL, interpreted over a WPTA that extend PTA with per-transition sabotage costs. PTOTL uses a single top-level sabotage binder ranging over *memoryless* defender strategies that, at each step, may disable a subset of currently enabled edges whose cumulative cost does not exceed a *per-step budget* that resets after each jump. Dense-time windows are expressed with freeze clocks, and path modalities include a probabilistic next and dense-time until/release. Probabilities are evaluated on *substochastic* successor kernels—disabled outcomes drop their mass rather than renormalize reflecting AG/MTD practice. In short, AG *model* the attacker's feasible progress in the plant, while PTOTL *specifies* the defender's guarantees and their constraints. We work in a perfect-information setting with one sabotage envelope to retain decidability while capturing the engineering intent.

In addition, we compare PTOTL with PTCTL and *Probabilistic Timed ATL* (PTATL). PTOTL extends PTCTL [38] by obstruction capability. Relative to PTATL, which quantifies coalitional strategies

*Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus.*

on fixed arenas, PTOTL targets edge-centric disabling under explicit budgets; the formalisms are complementary and incomparable in general, and we provide a sound coalition-to-sabotage abstraction on compiled models [14, 16, 25, 36, 37]. Furthermore, we prove that the model checking is PSPACE-complete for one clock by tight reductions with PTCTL [38, 49]. Finally, we evaluate PTOTL on an automotive case study grounded in documented in-vehicle attacks [23, 48, 51], using PTOTL specifications to assess moving-target defenses under time and probability constraints, yielding quantitative guarantees on bounded-time reachability, and demonstrating practicality to verify the security level and the robustness of these MTD techniques against new threats, to improve safety-critical CPS and support risk–performance trade-offs.

**Related Work.** Timed and probabilistic verification build on timed automata with zone abstractions and on probabilistic model checking for MDP/CTMC/PTA, with mature reductions and tools (e.g., PRISM) [1, 3, 12, 27, 41, 49]. Non-strategic real-time probabilistic logics (e.g., TCTL/PTCTL) use PTA abstractions [49]. Strategic verification originates from ATL/ATL* and their refinements with epistemic operators, recall, and strategy contexts, as well as Strategy Logic [4, 22, 43, 52]. In real time, timed game automata and the logics TATL and STCTL extend strategic reasoning with dense time and map expressiveness/decidability frontiers [8, 19, 29, 34]. Probabilistic strategic variants quantify stochastic coalition success: PATL/PATL* and PTATL integrate probabilities (and, for PTATL, dense time), with model checking under memoryless imperfect information and IMITATOR/PRISM workflows [7, 24, 25, 37, 41]. Related work studies equilibria and controller synthesis in probabilistic real-time and partially observable games [15, 35, 42, 57]. Edge-centric interference has been formalised via Sabotage Modal Logic (SML) and its subset variant (SSML), and by Dynamic Escape Games for weighted reachability [9, 20, 46, 53, 55]. Obstruction Logic and its timed/probabilistic extensions (TOL, POTL) add temporal operators, budgets, and quantitative thresholds [21, 44, 45], but treat dense time or probability in isolation rather than over PTA. As an application backdrop, attack graphs (AG) support multi-stage intrusion analysis, while surveys of Moving Target Defense (MTD) document dynamic reconfiguration (routing/ACL/policy rotation, service relocation) to reshape feasible paths under resource and timing constraints [26, 39, 47, 56]. These lines motivate formalisms that reason about time-bounded, budgeted edge disabling under uncertainty on real-time probabilistic models.

**Structure of the work.** The paper is structured as follows: Section 2 covers background. Section 3 defines PTOTL. Section 4 gives the model-checking algorithm and complexity. Section 5 presents the MTD automotive case study. Section 6 relates PTOTL to PTCTL/PTATL. Section 7 explain future work and concludes.

## 2 BACKGROUND

In this section, we discuss the basic notions used in the technical part, and begin with some general concepts. Let $\mathbb{N}$ be the set of natural numbers containing 0 as $\mathbb{N}_{\geq 0}$, $\mathbb{R}_{\geq 0}$ the set of non-negative reals and $\mathbb{Z}$ the set of integers. Let $X$ and $Y$ be two sets and $|X|$ denotes its cardinality. The set operations of intersection, union, complementation, set difference, and Cartesian product are denoted $X \cap Y, X \cup Y, \overline{X}, X \setminus Y$, and $X \times Y$, respectively. Inclusion and strict inclusion are denoted $X \subseteq Y$ and $X \subset Y$, respectively. The empty set is denoted $\emptyset$. Let $\pi = x_1, \ldots, x_n$ be a finite sequence, $last(\pi)$ denotes the last element $x_n$ of $\pi$.

### 2.1 General Concepts

*Probability Distribution and Space.* Let $Q$ be a finite set and $\mu : Q \longrightarrow [0, 1]$ be a probability distribution function over $Q$ such that $\sum_{q \in Q} \mu(q) = 1$. We denote by $\mathcal{D}(Q)$ the set of all such distributions over $Q$. For a given $\mu \in \mathcal{D}(Q)$, $supp(\mu) = \{q \in Q \mid \mu(q) > 0\}$ is called the support of $\mu$. The standard notation of a probability space is a triple $(\Omega, \mathcal{F}, \mathrm{Pr})$, where $\Omega$ is a sample space that represents all possible outcomes, $\mathcal{F} \subseteq 2^{\Omega}$ is a $\sigma$-algebra over $\Omega$, i.e., it includes the empty subset, and it is closed under countable unions and complement, and $\mathrm{Pr}\colon \mathcal{F} \longrightarrow [0, 1]$ is a probability measure over $(\Omega, \mathcal{F})$. We denote the set of all finite and infinite sequences of elements of $Q$ by $Q^+$ and $Q^*$, respectively.

*Kripke Structure and Markov Chain.* Here, we will formally define Kripke structure (KS) and Markov Chain (MC).

Definition 1 (Kripke Structure). *A Kripke Structure (KS) over a set* Ap *of atomic propositions is a tuple* $\mathcal{K} = \langle Q, q_0, R, \mathcal{L} \rangle$ *where* $Q$ *is a finite, non-empty set of states,* $q_0 \in Q$ *is the initial state,* $R \subseteq Q \times Q$ *is a binary serial relation over* $Q$ *(i.e., for any* $q \in Q$ *there is a* $q' \in Q$ *such that* $\langle q, q' \rangle \in R$*) and* $\mathcal{L} : Q \longrightarrow 2^{\mathrm{Ap}}$ *is a labeling function assigning a set of atomic propositions to any state* $q \in Q$.

Definition 2 (Markov Chain). *A Markov Chain (MC) is a pair* $\mathcal{H} = (Q, \mathbf{P})$ *where* $Q$ *is a (countable) set of states and* $\mathbf{P}\colon Q \times Q \longrightarrow [0, 1]$ *is a transition probability function such that for all state* $q \in Q$, $\Sigma_{q' \in Q} \mathbf{P}(q, q') = 1$. *If* $Q$ *is finite, we consider* $\mathbf{P}$ *as a transition matrix.*

A KS can be extended via MC [40] to define *Probabilistic Kripke Structure (PKS)* as follows.

Definition 3 (PKS). *A PKS over a set* Ap *of atomic propositions is a tuple* $\mathcal{G} = \langle Q, q_0, \mathbf{P}, \mathcal{L} \rangle$ *where* $(Q, \mathbf{P})$ *is a MC,* $q_0$ *is the initial state and* $\mathcal{L} : Q \longrightarrow 2^{\mathrm{Ap}}$ *is a labeling function assigning a set of atomic propositions to any state* $q \in Q$.

*Path.* A (finite or infinite) path over $\mathcal{G}$ is a sequence $\pi = q_0, q_1, q_2, \ldots$ starting at the initial state $q_0$ such that $\mathbf{P}(q_i, q_{i+1}) > 0$ for all $i \in \mathbb{N}$. We write $\pi_i$ for the $i$-th state $q_i$, $\pi_{\leq i}$ for the prefix $q_0, \ldots, q_i$, and $\pi_{\geq i}$ for the suffix $q_i, q_{i+1}, \ldots$. Let $\mathsf{Paths}^{+}_{\mathcal{G},q}$ be the set of non-empty finite paths from $q$, and $\mathsf{Paths}^{\omega}_{\mathcal{G},q}$ the set of infinite paths from $q$. A *history* is any finite prefix of a path; $H$ denotes the set of histories and $last(h)$ the last state of $h$.

*Cylinder.* We measure probabilities of path sets via the standard cylinder construction. For each $q \in Q$, let $(\Omega_q, \mathcal{F}_q, \mathrm{Pr}^q_{\mathcal{G}})$ be the probability space where $\Omega_q$ is the set of infinite paths starting at $q$ and $\mathcal{F}_q$ is the $\sigma$-algebra generated by cylinder sets. For a finite path $\hat{\pi} = q_0, q_1, \ldots, q_n$, its cylinder is $\mathsf{Cyl}(\hat{\pi}) = \{ \pi \in \mathsf{Paths}^{\omega}_{\mathcal{G},q_0} \mid \hat{\pi} \in \mathsf{Prefix}(\pi) \}$. The measure on cylinders is defined b $\mathrm{Pr}^{q_0}_{\mathcal{G}}(\mathsf{Cyl}(\hat{\pi})) = \prod_{i=0}^{n-1} \mathbf{P}(q_i, q_{i+1})$, and it extends uniquely from cylinders to $\mathcal{F}_q$ (we keep the notation $\mathrm{Pr}^q_{\mathcal{G}}$). Not every subset of paths is measurable, but all sets considered here are, see [12] for measure-theoretic details.

*Predecessors and Successors.* Let $\mathcal{G}$ be a PKS and $q \in Q$ one of its states, $\mathrm{pre}(q)$ denotes the set of predecessors of $q$, i.e., $\mathrm{pre}(q) =$

$\{q' \in Q \mid \mathbf{P}(q', q) > 0\}$. Similarly, $\mathrm{post}(q)$ denotes the set of successors of $q$, i.e., $\mathrm{post}(q) = \{q' \in Q \mid \mathbf{P}(q, q') > 0\}$, and $\mathsf{E}(q)$ denotes its outgoing edges $\mathsf{E}(q) = \{e \in Q{\times}Q \mid e = (q, q')$ for some $q' \in Q$ and $\mathbf{P}(q, q') > 0\}$.
Here, we show *Probabilistic Obstruction Temporal Structure* (POTS) an extension of PKS [44], enabling weighted properties in a model.

Definition 4 (POTS). *A POTS (model for short) is given by a tuple $\mathcal{M} = (\mathcal{G}, \mathsf{C})$ where $\mathcal{G} = (Q, q_0, \mathbf{P}, \mathcal{L})$ is a PKS and $\mathsf{C} : Q \times Q \longrightarrow \mathbb{N}$ is a function assigning to any pairs $(q, q')$ a natural number $n \in \mathbb{N}_{\geq 0}$.*

## 2.2 Weighted Timed Automata

We now explore the relation between WTS and Weighted Timed Automata (WTA) [5]. A WTA is an extension of a TA [2] with weight/cost information at both locations and edges, and it can be used to address several interesting questions [5, 19].

Definition 5 (Clock constraints and invariants). *Let $X$ be a finite set of variables over $\mathbb{R}_{\geq 0}$, called clocks. The set $\Phi^{+}(X)$ of clock constraints over the set of clocks $X$ is given by the following grammar:*

$$\phi := true \mid x \sim c \mid x - y \sim c \mid \phi \ \wedge \ \phi$$

*where $x, y \in X$, $c \in \mathbb{N}$, and $\sim \in \ \{<,\ >,\ \leq,\ \geq,\ =\}$.*

The clock constraints of the form $true$, $x \sim c$ are called non-diagonal constraints and those of the form $x - y \sim c$ are called diagonal constraints. The set of non-diagonal constraints over $X$ is denoted by $\Phi(X)$. Clock invariants $\Delta(X)$ are constraints where $\sim \in \{<, \leq\}$.

Definition 6 (Clock valuations). *Given a finite set of clocks X, a clock valuation function, $v : X \longrightarrow \mathbb{R}_{\geq 0}$ assigning to each clock $x \in X$ a non-negative value $v(x)$. We denote $\mathbb{R}^{X}_{\geq 0}$ the set of all valuations. For a clock valuation $v \in \mathbb{R}^{X}_{\geq 0}$ and a time value $d \in \mathbb{R}_{\geq 0}$, $v + d$ is the valuation satisfied by $(v + d)(x) = v(x) + d$ for each $x \in X$. Given a clock subset $Y \subseteq X$, we denote $v[Y \longleftarrow 0]$ the valuation defined as follows: $v[Y \longleftarrow 0](x) = 0$ if $x \in Y$ and $v[Y \longleftarrow 0](x) = v(x)$ else.*

Here, we only consider the weight/cost in the edges (transitions) in our WTA. Formally, a WTA is defined as follows [5].

Definition 7 (WTA). *Let $X$ be a finite set of clocks and* Ap *a finite set of atoms. A WTA is a tuple $\mathcal{A} = (L, l_0, X, \Sigma, T, Inv, W, K)$, where: $(i)$ $L$ is a finite set of locations. $(ii)$ $l_0 \in L$ is an initial location. $(iii)$ $X$ is a finite set of clocks. $(iv)$ $\Sigma$ is a finite set of actions. $(v)$ $T \subseteq L \times \Sigma \times \Phi(X) \times 2^{X} \times L$ is a finite set of transitions. $(vi)$ $Inv : L \longrightarrow \Delta(X)$ is a function assigning to each location a clock invariant. $(vii)$ $W \colon T \longrightarrow \mathbb{N}_{\geq 0}$ is a labeling function on elements of $T$. $(viii)$ $K : L \longrightarrow 2^{\mathrm{Ap}}$ is a labeling function for locations.*

We write $\ell \xrightarrow[w]{a,\phi,Y} \ell'$ as shorthand for $(\ell, a, \phi, Y, \ell')_w \in T$, where $a$ is an action, $\phi \in \Phi(X)$ a guard, $Y \subseteq X$ a reset set, and $w \in \mathbb{N}_{\geq 0}$. Let $\mathsf{W} : T \longrightarrow \mathbb{N}_{\geq 0}$ map an edge $t = (\ell, a, \phi, Y, \ell')_w$ to its weight $\mathsf{W}(t) = w$, interpreted as the (de)activation cost. Costs are *annotations only*: they don't appear in guards/invariants and so don't affect which discrete transitions are enabled. This avoids undecidability issues of HA [33] and preserves the decidability results for WTA [19]. Thus, the semantics of WTA are given by a weighted transition system (WTS), an extension of labelled transition systems (LTS)[50].

Definition 8 (Semantics of WTA). *Let $\mathcal{A} = (L, l_0, X, \Sigma, T, Inv, W, K)$ be a WTA. The semantics of WTA $\mathcal{A}$ is given by a WTS($\mathcal{A}$) = $(S, s_0, \Sigma_\Delta, E, W', K', S_F)$ where: $(i)$ $S \subseteq L \times \mathbb{R}^{X}_{\geq 0}$ is a set of states. $(ii)$ $s_0 = (l_0, v_0)$ with $v_0(x) = 0$ for all $x \in X$ and $v_0 \models Inv(l_0)$. $(iii)$ $\Sigma_\Delta = \Sigma \uplus \mathbb{R}_{\geq 0}$. $(iv)$ $E \subseteq S \times \Sigma_\Delta \times S$ is a transition defined by two rules:*

- ***Discrete transition:*** *$(l, v) \xrightarrow[w]{a} (l', v')$ for $a \in \Sigma$ and $w \in \mathbb{N}_{\geq 0}$ iff $l \xrightarrow[w]{a,\phi,Y} l'$, $v \models \phi$, $v' = v[Y \longleftarrow 0]$ and $v' \models Inv(l')$ and,*
- ***Delay transition:*** *$(l, v) \xrightarrow{d} (l, v + d)$, for some $d \in \mathbb{R}_{\geq 0}$ iff $v + d \models Inv(l)$.*

*$(v)$ $W' = E \longrightarrow \mathbb{N}_{\geq 0}$. $(vi)$ $K'((l, v)) = K(l) \cup \{\phi \in \Phi(X) \mid v \models \phi\}$.*

## 2.3 Predecessor operator and Zone Graph

Since a WTA has infinitely many states, one cannot build a finite-state automaton. Instead, the *zone graph* provides a finite symbolic semantics for TA behaviours [18]. It is both the core implementation technique in TA tools and a basis of decidability results. In a zone graph, clock zones symbolically denote sets of valuations. Over a set of clocks $X$, a zone $Z \subseteq \mathbb{R}^{X}_{\geq 0}$ is the set of evaluations satisfying a constraints conjunction $\phi$, i.e., $Z = \{\ v \in \mathbb{R}^{X}_{\geq 0} \mid v \models \phi\ \}$. A symbolic state zone is a pair $\mathcal{Z} = (l, Z)$ where $l$ is a location and $Z$ a clock zone, it represents all concrete states $(l', v)$ with $l' = l$ and $v \in Z$.

Definition 9 (Discrete and Time Predecessor). *Let $\mathcal{Z}$ be a zone and $e$ an edge of WTS($\mathcal{A}$) (with $w \in \mathbb{N}_{\geq 0}$). Define:*

$$\textit{disc-pred}(e, \mathcal{Z}) \ = \ \{\, z \mid \exists z' \in \mathcal{Z} : \ z \xrightarrow[w]{e} z' \,\},$$

$$\textit{time-pred}(\mathcal{Z}) \ = \ \{\, z \mid \exists z' \in \mathcal{Z},\ \exists d \in \mathbb{R}_{\geq 0} : \ z \xrightarrow[w]{d} z' \,\}.$$

Both $\mathrm{disc\text{-}pred}(e, \mathcal{Z})$ and $\mathrm{time\text{-}pred}(\mathcal{Z})$ are zones (closure under predecessors). Dually, $\mathrm{time\text{-}succ}(l, Z)$ and $\mathrm{post}(e, (l, Z))$ denote, respectively, the sets of time-successors of any state in $(l, Z)$ and the discrete successors of $(l, Z)$ via $e$.

Definition 10 (Predecessor). *Let $\mathcal{Z}$ be a zone and $e$ an edge of WTS($\mathcal{A}$). Define*

$$\textit{pred}(e, \mathcal{Z}) \ = \ \textit{disc-pred}\big(e,\ \textit{time-pred}(\mathcal{Z})\big).$$

In words, $\mathrm{pred}(e, \mathcal{Z})$ collects all states that, after one discrete $e$-step followed by some time delay, reach a state in $\mathcal{Z}$. We write $\mathrm{time\text{-}succ}(l, Z)$ for the set of time-successors of any state in $(l, Z)$, and $\mathrm{post}(e, (l, Z))$ for discrete $e$-successors of $(l, Z)$ with $\mathcal{Z} = (l, Z)$.

## 3 MODEL AND LOGIC

In this section, we define the model WPTA and the PTOTL syntax and semantics, extending previous obstruction logics [21, 44, 45], by unified the quantitative and real-time temporal aspect.

Definition 11 (WPTA). *The WPTA is an extension of WTA [6] and PTA [49]. A WPTA is a tuple $\mathcal{M} = \langle L, \ell_0, \mathcal{X}, \mathrm{Inv}, T, Prob, W, K \rangle$, where: $(i)$ $L$ is a finite set of locations. $(ii)$ $\ell_0 \in L$ is the initial location. $(iii)$ $\mathcal{X}$ is a finite set of clocks. $(iv)$ $\mathrm{Inv} : L \longrightarrow \Phi(\mathcal{X})$ associates an invariant to each location. $(v)$ $T \subseteq L \times \Phi(\mathcal{X}) \times 2^{\mathcal{X}} \times L$ is the finite set of edge transitions. A transition $(\ell, g, r, \ell')$ consists of source location $\ell$, guard $g$, reset $r$, and target location $\ell'$. $(vi)$ $Prob : T \longrightarrow \mathcal{D}(L)$ assigns a probability distribution on targets once a transition is chosen. $(vii)$ $W : T \longrightarrow \mathbb{N}_{\geq 0}$ assigns a non-negative integer weight (cost) to each transition. $(viii)$ $K : L \longrightarrow 2^{\mathrm{Ap}}$ is a labeling function for the location.*

We write $\ell \xrightarrow[w,\,p]{a,\phi,Y} \ell'$ as shorthand for $(\ell, a, \phi, Y, \ell')^{p}_{w} \in T$, where $a$ is an action, $\phi \in \Phi(X)$ a guard, $Y \subseteq X$ a reset set, $w \in \mathbb{N}_{\geq 0}$ and $p \in \mathcal{D}(L)$ is a discrete probability distribution over target locations (finite support). In this notation, the next location $\ell'$ is drawn according to $p$, in particular, $\ell' \in \mathrm{supp}(p)$ iff $p(\ell') > 0$. The semantics of WTA is presented by WTS. To define the semantics of WPTA, we employ Probabilistic Timed Kripke Structure (PTKS), which extend Markov Chain (MC) with real-valued duration [40].

Definition 12 (WPTA semantics). *Let $\mathcal{M} = \langle L, l_0, \mathcal{X}, \mathrm{Inv}, T, Prob, W, K\rangle$ be a WPTA. Semantics of WPTA $\mathcal{M}$ is given by a PTKS($\mathcal{M}$) $= (S, s_0, \hat{P}, \hat{W}, \hat{K})$ where: $(i)$ $S \subseteq L \times \mathbb{R}^{X}_{\geq 0}$ is a set of states. $(ii)$ $s_0 = (l_0, v_0)$ with $v_0(x) = 0$ for all $x \in X$ and $v_0 \models Inv(l_0)$. $(iii)$ $\hat{P} \subseteq S \times \mathcal{D}(S)$ is a probabilistic transition relation. For $s = (l, v) \in S$ :*

- ***Discrete transition:*** *For $t = (\ell, g, r, \ell'_e)^{p}_{w} \in T$ with $v \models g$, let $p := Prob(t) \in \mathcal{D}(L)$ and define $\mu_t \in \mathcal{D}(S)$ by*

$$\mu_t\big(\ell',\, v[r := 0]\big) \;=\; \begin{cases} p(\ell') & \text{if } v[r := 0] \models \mathrm{Inv}(\ell'), \\ 0 & \text{otherwise.} \end{cases}$$

  *Then $(s, \mu_t) \in \hat{P}$ and,*
- ***Delay transition:*** *For any $d \in \mathbb{R}_{\geq 0}$ with $v + t \models \mathrm{Inv}(\ell)$ for all $t \in [0, d]$, let $\mu_d = \delta_{(\ell, v+d)} \in \mathcal{D}(S)$. Then $(s, \mu_d) \in \hat{P}$.*

*$(v)$ $\hat{W} : \{(s, \mu, s') \mid (s, \mu) \in \hat{P},\ \mu(s') > 0\} \longrightarrow \mathbb{N}_{\geq 0}$ is the weight function:*

$$\hat{W}(s, \mu, s') \;=\; \begin{cases} W(t) & \text{if } \mu = \mu_t \text{ for some enabled } t \in T, \\ 0 & \text{if } \mu = \mu_d \text{ for some } d \geq 0. \end{cases}$$

*$(vi)$ $\hat{K} : S \longrightarrow 2^{\mathsf{Ap}}$ is the labelling, e.g. $\hat{K}(\ell, v) = K(\ell)$.*

*Strategy and Outcomes.* A symbolic state is a pair $q = (\ell, Z)$ with $\ell \in L$ and $Z \in \Phi^{+}(X)$. Let $Q \subseteq L \times \Phi^{+}(X)$ be the set of reachable symbolic states of $\mathcal{M}$, and $H$ the set of finite histories over $Q$ (finite sequences). Let $W : T \longrightarrow \mathbb{N}_{\geq 0}$ be the edge-cost function and $n \in \mathbb{N}_{\geq 0}$ a fixed per-step budget. For $q = (\ell, Z)$ define the enabled set $\mathrm{Enabled}(q) = \{ (\ell, g, r, \ell') \in T \mid Z \wedge g \neq \emptyset \}$.

A *zone-based n-strategy* is a function $\mathcal{S} : H \longrightarrow 2^{T}$ such that, for each history $h \in H$ with $q = \mathrm{last}(h)$ and $E_h = \mathcal{S}(h)$,

$$E_h \subseteq \mathrm{Enabled}(q) \qquad \text{and} \qquad \sum_{e \in E_h} W(e) \;\leq\; n.$$

$E_h$ are the transitions *desactivadas* at $q$ under budget $n$. The strategy is *memoryless* if $\mathcal{S}(h) = \mathcal{S}(h')$ whenever $\mathrm{last}(h) = \mathrm{last}(h')$; i.e., it can be viewed as $\mathcal{S} : Q \longrightarrow 2^{T}$ with the same constraints. A (symbolic) path is a sequence $\rho = q_0 \xrightarrow{\sigma_0} q_1 \xrightarrow{\sigma_1} q_2 \cdots$ over the zone graph. It is *compatible* with $\mathcal{S}$ if, for all $i \geq 0$, $(q_i, \sigma_i, q_{i+1}) \notin \mathcal{S}(\rho_{\leq i})$, where $\rho_{\leq i} = q_0, \ldots, q_i$. The set of outcomes from $q$ is $\mathrm{Out}(q, \mathcal{S}) = \{ \rho \text{ infinite path from } q \mid \rho \text{ compatible with } \mathcal{S} \}$. We now introduce the syntax of our logic PTOTL.

Definition 13. *Let* Ap *be a (at most countable) set of atomic propositions, $X$ a finite set of clocks, $\bowtie$ and $\sim\, \in \{<, \leq, =, \geq, >\}$, $n \in \mathbb{N}_{\geq 0}$ a cost budget, and $k \in [0, 1] \cap \mathbb{Q}$ a probability threshold. PTOTL is stratified into* four *layers, state, clock, probabilistic, and timed formulas all evaluated over the event trace (the sequence of discrete jumps) of the underlying model as follows:*

1. ***State formulas*** *$\varphi$:*

$$\varphi \;::=\; \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \beta \mid j.\psi$$

2. ***State–clock formulas*** *$\alpha$:*

$$\alpha \;::=\; \varphi \mid x \sim c \mid \alpha \wedge \alpha \quad (x, y \in X,\ c \in \mathbb{Z})$$

3. ***Timed path formulas*** *$\psi$:*

$$\psi \;::=\; \alpha\, U\, \alpha \mid \alpha\, R\, \alpha$$

   *(Until/Release over the* event index*; each $\alpha$ is checked at the state reached* after *each discrete jump. The freeze $j.\psi$ locally binds the formula clock $j$ to* 0*.)*

4. ***Probabilistic obstruction formulas*** *$\beta$:*

$$\beta \;::=\; \big\langle +^{\bowtie k}_{n} \big\rangle\, \chi \qquad \text{where } \chi \;::=\; \bigcirc_{\mathrm{disc}}\, \varphi \mid \psi$$

   *(States that there exists an n-budget sabotage strategy such that the probability of paths satisfying $\chi$ meets the threshold $\bowtie k$; only a single outermost occurrence is allowed.)*

(i) We use the event-next operator $\bigcirc_{\mathrm{disc}}$ (next *discrete* jump) only inside the probabilistic modality $\langle +^{\bowtie k}_{n} \rangle$ (i.e., in $\chi$), not as a primitive timed operator, thereby avoiding ambiguity with dense time and name clashes with the clock set $X$. (ii) The binder $j.\psi$ is a (TPTL-style) *freeze* operator that binds a fresh time variable $j$ to the current time point, allowing $\psi$ to compare future times against $j$ via clock constraints. As usual, other Boolean/temporal connectives are defined as abbreviations: $\varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$, $\mathbf{F}\alpha \equiv \top\, U\, \alpha$, $\mathbf{G}\alpha \equiv \alpha\, R\, \bot$, etc.

Definition 14 (PTOTL Semantics). *Let $\mathcal{M} = \langle L, \ell_0, \mathcal{X}, \mathrm{Inv}, T, Prob, W, K\rangle$ be a WPTA. A* configuration *is a pair $q = (\ell, v)$ with $\ell \in L$ and valuation $v : \mathcal{X} \cup \mathcal{J} \longrightarrow \mathbb{R}_{\geq 0}$ for system clocks and formula clocks. A timed run $\rho$ alternates time elapse and discrete edges, respecting invariants and guards, and, under the PTKS semantics and a fixed budget-n sabotage strategy $\sigma$, induces a probability measure $\mathrm{Pr}^{\mathcal{M},q}_{\sigma}$ on runs from q. From each run $\rho$ we extract the* event trace *$J(\rho) = s_0 s_1 s_2 \ldots$ where $s_i = (\ell_i, v_i)$ is the state* immediately after *the i-th discrete jump ($s_0$ is the initial configuration). Path modalities are interpreted on this discrete event index.*

***Satisfaction of state formulas $\varphi$.***

- *$\mathcal{M}, q \models \top$ always.*
- *$\mathcal{M}, q \models p$ iff $p \in K(\ell)$.*
- *$\mathcal{M}, q \models \neg\varphi$ iff $\mathcal{M}, q \not\models \varphi$.*
- *$\mathcal{M}, q \models \varphi_1 \wedge \varphi_2$ iff $\mathcal{M}, q \models \varphi_1$ and $\mathcal{M}, q \models \varphi_2$.*
- *$\mathcal{M}, q \models j.\psi$ iff $\mathcal{M}, (\ell, v[j := 0]) \models \psi$.*

***Satisfaction of state–clock formulas $\alpha$.***

- *If $\alpha$ is a state formula, use the rules above.*
- *$\mathcal{M}, q \models (x \sim c)$ iff $v(x) \sim c$, with $x \in \mathcal{X} \cup \mathcal{J}$, $c \in \mathbb{Q}_{\geq 0}$.*
- *$\mathcal{M}, q \models (\alpha_1 \wedge \alpha_2)$ iff $\mathcal{M}, q \models \alpha_1$ and $\mathcal{M}, q \models \alpha_2$.*

***Timed path formulas (event-indexed) $\psi$.*** *We write $(\rho, i) \models_{\mathrm{disc}} \cdot$ when evaluating on the event index i of $J(\rho) = s_0 s_1 s_2 \ldots$, and $s_i \models \cdot$ for state-level satisfaction.*

$$(\rho, i) \models_{\mathrm{disc}} \alpha_1\, U\, \alpha_2 \iff \exists j \geq i :\ s_j \models \alpha_2 \wedge \forall m \in [i, j) : s_m \models \alpha_1,$$

$$(\rho, i) \models_{\mathrm{disc}} \alpha_1\, R\, \alpha_2 \iff \forall j \geq i : \Big( (\forall m \in [i, j) : s_m \not\models \alpha_1) \Rightarrow s_j \models \alpha_2 \Big).$$

*By design, $\psi$ contains no event-next operator.*

***Probabilistic obstruction*** $\beta$. *Let* $\nmid_n$ *be the set of budget-n sabotage strategies (transition-disabling schedulers). For a path subformula* $\chi \in \{\bigcirc_{\text{disc}}\,\varphi,\ \alpha_1 U \alpha_2,\ \alpha_1 R \alpha_2\}$,

$$\mathcal{M}, q \models \langle \nmid_n^{\bowtie k} \rangle \chi \iff \exists\, \sigma \in \nmid_n : \Pr_{\sigma}^{\mathcal{M},q}\big(\{\, \rho \mid (\rho, 0) \models_{\text{disc}} \chi \,\}\big) \bowtie k,$$

*with* $\bowtie \in \{<, \leq, =, \geq, >\}$ *and* $k \in [0,1]$. *Here* $\bigcirc_{\text{disc}} = \bigcirc_{\text{disc}}$ *denotes the* next discrete jump *and is used only inside the probabilistic envelope. (i) For any* $\chi$, *the set* $\{\rho \mid (\rho, 0) \models_{\text{disc}} \chi\}$ *is a countable union of cylinder sets, hence measurable; the probability above is well defined. (ii) Any occurrence of a formula clock* $j$ *must lie within the scope of a binder* $j.\psi$. *(iii) To preserve decidability, PTOTL admits at most one (non-nested) sabotage envelope per formula.*

## 4 MODEL CHECKING

Given a WPTA $\mathcal{M}$ and a closed PTOTL formula $\varphi$, model checking computes the satisfaction set $\mathsf{Sat}(\varphi, \mathcal{M}) \subseteq Q$ over the *probabilistic zone graph* $\mathcal{Z} = (Q, \rightarrow, P)$ and then checks whether the initial symbolic state belongs to it. The distinctive difficulty is the *probabilistic sabotage* operator $\langle \nmid_n^{\bowtie k} \rangle \chi$, which quantifies over per-step budget-$n$ strategies that disable outgoing edges and compares the induced path probability with $k$. Let $\mathfrak{S}_n$ be the set of per-step budget-$n$ sabotage strategies. Under the PTKS semantics, each $\sigma \in \mathfrak{S}_n$ and state $q$ induce a probability measure $\Pr_\sigma^{\mathcal{M},q}$ on runs $\rho$ from $q$. For a path pattern $\chi \in \{\bigcirc_{\text{disc}}\,\varphi,\ \alpha_1 U \alpha_2,\ \alpha_1 R \alpha_2\}$, define

$$\Pr_{\sigma}^{\mathcal{M},q}(\chi) \ := \ \Pr_{\sigma}^{\mathcal{M},q}\big(\{\rho \mid (\rho, 0) \models_{\text{disc}} \chi\}\big)$$

$$\mathsf{Sat}\big(\langle \nmid_n^{\bowtie k} \rangle \chi\big) \ = \ \{\, q \mid \exists \sigma \in \mathfrak{S}_n : \Pr_{\sigma}^{\mathcal{M},q}(\chi) \ \bowtie \ k \,\}$$

The path sets are measurable (countable unions of cylinder sets), hence the probabilities are well-defined. A symbolic state is $q = (\ell, Z)$ with location $\ell$ and canonical convex zone $Z$ (DBM) over system and formula clocks. The enabled set at $q$ is

$$E_q \ = \ \{\, t = (\ell, g, r, \ell') \in T \mid Z \cap [\![g]\!] \neq \emptyset \,\}.$$

A sabotage configuration picks $E' \subseteq E_q$ such that $\sum_{e \in E_q \setminus E'} W(e) \leq n$. Disabled edges contribute *no* probability mass (no renormalisation): for successors $q'$,

$$P^{E'}(q, q') \ := \ \begin{cases} P(q, q') & \text{if some kept edge in } E' \text{ leads from } q \text{ to } q', \\ 0 & \text{otherwise.} \end{cases}$$

The sabotage budget resets *at every discrete step.* $\mathcal{Z}$ is finite because zones are DBMs closed under time elapse and resets, with all constants drawn from $\mathcal{M}$ and $\varphi$; thus only finitely many canonical DBMs arise. For $L \subseteq Q$ define the masked, thresholded predecessor

$$\mathsf{Pre}_{n,\bowtie k}(L) \ := \ \Big\{\, q \;\Big|\; \exists E' \subseteq E_q : \ \textstyle\sum_{e \in E_q \setminus E'} W(e) \leq n \\ \wedge \displaystyle\sum_{q' \in L} P^{E'}(q, q') \ \bowtie \ k \Big\}$$

Let $L_1 = \mathsf{Sat}(\alpha_1)$, $L_2 = \mathsf{Sat}(\alpha_2)$. Then

$$\mathsf{Sat}\big(\langle \nmid_n^{\bowtie k} \rangle \ \bigcirc_{\text{disc}} \ \varphi\big) = \mathsf{Pre}_{n,\bowtie k}\big(\mathsf{Sat}(\varphi)\big),$$

$$\mathsf{Sat}\big(\langle \nmid_n^{\bowtie k} \rangle \ \alpha_1 \, U \, \alpha_2\big) = \mu S. \ L_2 \ \cup \ \big(L_1 \cap \mathsf{Pre}_{n,\bowtie k}(S)\big),$$

$$\mathsf{Sat}\big(\langle \nmid_n^{\bowtie k} \rangle \ \alpha_1 \, R \, \alpha_2\big) = \nu S. \ L_2 \ \cap \ \big(L_1 \ \cup \ \mathsf{Pre}_{n,\bowtie k}(S)\big),$$

with $\mu/\nu$ denoting least/greatest fixpoints over the finite lattice $2^Q$. We now present the concrete algorithms that implement PTOTL model checking over the symbolic probabilistic zone graph induced by the WPTA structure, detailing each semantic case of the grammar and the strategy outcomes.

**Algorithm 1** Model-Checking over a Probabilistic Zone Graph

```
Input: Probabilistic Zone Graph $\mathcal{Z} = (Q, \rightarrow, P)$ of WPTA $\mathcal{M}$,
    closed PTOTL formula $\varphi$
Output: $\mathsf{Sat}(\varphi) \subseteq Q$
for $i = 1$ to $|\varphi|$ do                ▷ bottom-up by syntactic depth
    for all $\psi \in \mathsf{Sub}(\varphi)$ with depth $|\psi| = i$ do
        switch $\psi$ do
            case $\top$
                $\mathsf{Sat}(\psi) \leftarrow Q$
            case $p \in \mathsf{Ap}$
                $\mathsf{Sat}(\psi) \leftarrow \{\, (\ell, Z) \in Q \mid p \in K(\ell) \,\}$
            case $x \sim c$
                $\mathsf{Sat}(\psi) \leftarrow \{\, (\ell, Z) \in Q \mid Z \models x \smile c \,\}$
            case $\psi_1 \wedge \psi_2$
                $\mathsf{Sat}(\psi) \leftarrow \mathsf{Sat}(\psi_1) \ \cap \ \mathsf{Sat}(\psi_2)$
            case $\neg\psi_1$
                $\mathsf{Sat}(\psi) \leftarrow Q \setminus \mathsf{Sat}(\psi_1)$
            case $j.\psi_1$
                $\mathsf{Sat}(\psi) \leftarrow \{\, (\ell,\ Z[j \leftarrow 0]) \mid (\ell, Z) \in \mathsf{Sat}(\psi_1) \,\}$
            case $\langle \nmid_n^{\bowtie k} \rangle \ \bigcirc_{\text{disc}} \ \varphi_1$
                $\mathsf{Sat}(\psi) \leftarrow \mathsf{Pre}_{n,\bowtie k}\big(\mathsf{Sat}(\varphi_1)\big)$
            case $\langle \nmid_n^{\bowtie k} \rangle \ \alpha_1 \ U \ \alpha_2$
                $\mathsf{Sat}(\psi) \leftarrow \mathsf{Until}_{n,\bowtie k}\big(\mathsf{Sat}(\alpha_1),\ \mathsf{Sat}(\alpha_2)\big)$
            case $\langle \nmid_n^{\bowtie k} \rangle \ \alpha_1 \ R \ \alpha_2$
                $\mathsf{Sat}(\psi) \leftarrow \mathsf{Release}_{n,\bowtie k}\big(\mathsf{Sat}(\alpha_1),\ \mathsf{Sat}(\alpha_2)\big)$
return $\mathsf{Sat}(\varphi)$
```

**Algorithm 2** $\mathsf{Pre}_{n,\bowtie k}(L)$

```
Input: budget $n$, target set $L \subseteq Q$, threshold relation $\bowtie$, value
    $k \in [0,1]$
Output: $S = \{q \in Q \mid \exists E' \subseteq \mathit{Enabled}(q) :$
    $\sum_{e \in \mathit{Enabled}(q) \setminus E'} W(e) \leq n \ \wedge \ \sum_{q' \in L} P^{E'}(q, q') \ \bowtie \ k \,\}$
$S \leftarrow \emptyset$
for all $q \in Q$ do
    $E_q \leftarrow \mathit{Enabled}(q)$
    for all $E' \subseteq E_q$ such that $\sum_{e \in E_q \setminus E'} W(e) \leq n$ do
        $p \leftarrow \sum_{q' \in L} P^{E'}(q, q')$                ▷ masked probability, no
    renormalisation
        if $p \ \bowtie \ k$ then
            $S \leftarrow S \cup \{q\}$; break
return $S$
```

The four procedures in Algorithms 1-4 implement the *symbolic* model checking of PTOTL over the probabilistic zone graph of a WPTA. Algorithm 1 traverses the syntax tree of $\varphi$ bottom-up and computes $\mathsf{Sat}(\psi) \subseteq Q$ for each subformula $\psi$. Base cases are handled directly: atomic propositions via the location labelling $K(\ell)$,

**Algorithm 3** $\mathsf{Until}_{n,\bowtie k}(L_1, L_2)$ (least fixpoint)

```
Input: budget $n$, label sets $L_1, L_2 \subseteq Q$, threshold relation $\bowtie$, value $k$
Output: $S = \mu X.\ L_2 \cup (L_1 \cap \mathsf{Pre}_{n,\bowtie k}(X))$
$S \leftarrow L_2$; $changed \leftarrow$ true
while $changed$ do
    $changed \leftarrow$ false
    for all $q \in (L_1 \setminus S)$ do
        $E_q \leftarrow Enabled(q)$; $added \leftarrow$ false
        for all $E' \subseteq E_q$ with $\sum_{e \in E_q \setminus E'} W(e) \le n$ do
            $p \leftarrow \sum_{q' \in S} P^{E'}(q, q')$
            if $p \bowtie k$ then
                $S \leftarrow S \cup \{q\}$; $changed \leftarrow$ true; $added \leftarrow$ true; break
return $S$
```

**Algorithm 4** $\mathsf{Release}_{n,\bowtie k}(L_1, L_2)$ (greatest fixpoint)

```
Input: budget $n$, label sets $L_1, L_2 \subseteq Q$, threshold relation $\bowtie$, value $k$
Output: $S = \nu X.\ L_2 \cap (\, L_1 \cup \mathsf{Pre}_{n,\bowtie k}(X)\,)$
$S \leftarrow Q$; $changed \leftarrow$ true
while $changed$ do
    $changed \leftarrow$ false
    for all $q \in S$ do
        if $q \notin L_2$ then
            $S \leftarrow S \setminus \{q\}$; $changed \leftarrow$ true; continue
        if $q \in L_1$ then
            continue                                  ▷ kept by $L_1$
        $E_q \leftarrow Enabled(q)$; $ok \leftarrow$ false
        for all $E' \subseteq E_q$ with $\sum_{e \in E_q \setminus E'} W(e) \le n$ do
            $p \leftarrow \sum_{q' \in S} P^{E'}(q, q')$
            if $p \bowtie k$ then
                $ok \leftarrow$ true; break
        if not $ok$ then
            $S \leftarrow S \setminus \{q\}$; $changed \leftarrow$ true
return $S$
```

and clock constraints by DBM entailment ($Z \models x \sim c$). Boolean connectives are realised as set operations over symbolic states. The freeze operator $j.\psi$ is interpreted symbolically by resetting the formula clock in the current zone, i.e. $Z[j \leftarrow 0]$, before recursively evaluating $\psi$. Strategic (sabotage) constructs are delegated to the dedicated procedures: *(i)* Algorithm 2 computes $\mathsf{Pre}_{n,\bowtie k}(L)$ and is invoked for $\langle \lightning_n^{\bowtie k} \rangle \bigcirc_{\mathrm{disc}} \varphi_1$. For each zone state $q$, it enumerates sabotage configurations $E' \subseteq Enabled(q)$ with per-step cost $\sum_{e \in Enabled(q) \setminus E'} W(e) \le n$, and checks the *masked* probability $\sum_{q' \in L} P^{E'}(q, q') \bowtie k$, where sabotaged edges contribute zero mass (*no* renormalisation). *(ii)* Algorithm 3 realises the least-fixpoint characterisation of $\langle \lightning_n^{\bowtie k} \rangle\, \alpha_1\, U\, \alpha_2$: it starts from $L_2 = \mathsf{Sat}(\alpha_2)$ and expands backward through $L_1 = \mathsf{Sat}(\alpha_1)$, adding any $q \in L_1$ that admits a budget-feasible sabotage $E'$ with $\sum_{q' \in S} P^{E'}(q, q') \bowtie k$, where $S$ is the current approximation. *(iii)* Algorithm 4 implements the greatest-fixpoint for $\langle \lightning_n^{\bowtie k} \rangle\, \alpha_1\, R\, \alpha_2$: it starts from $S = Q$ and removes states that either violate $L_2$ or cannot sustain the release condition, i.e. states $q \notin L_1$ for which every budget-feasible sabotage configuration $E'$ yields $\sum_{q' \in S} P^{E'}(q, q') \not\bowtie k$. These procedures are faithful to the event-indexed semantics of PTOTL: $\bigcirc_{\mathrm{disc}}$ refers to the next *discrete* jump; the sabotage budget is per-step and resets after each jump; and probability mass on disabled edges is simply dropped (no redistribution). Since the probabilistic zone graph $Q$ is finite (canonical DBMs under bounded constants) and both fixpoint iterations in Algorithms 3 and 4 are monotone over $2^Q$, each loop stabilises in at most $|Q|$ iterations. Hence Algorithm 1 always terminates on finite inputs. Let us now prove the termination and correctness of the Algorithm 1. We first prove that the algorithm always halts on finite inputs.

Lemma 1 (Finiteness of the Probabilistic Zone Graph). *Let $\mathcal{Z}$ be the probabilistic zone graph constructed from a WPTA $\mathcal{M}$ using canonical DBMs and a standard finite extrapolation (e.g. LU-extrapolation) whose bounds are drawn from guards of $\mathcal{M}$ and clock bounds occurring in $\varphi$. Then $\mathcal{Z}$ has finitely many symbolic states.*

Sketch. Symbolic states are pairs $(\ell, Z)$ with $\ell \in L$ and $Z$ a canonical convex zone (DBM) over system/formula clocks. Since all clock constraints use constants from a finite set, the chosen extrapolation guarantees only finitely many canonical DBMs are reachable under time-elapse closure and reset operations. As $L$ is finite, the set $Q = \{(\ell, Z)\}$ is finite. □

Theorem 4.1 (Termination). *For any closed PTOTL formula $\varphi$ and any finite probabilistic zone graph $\mathcal{Z}$, Algorithm 1 terminates.*

Sketch. The algorithm processes subformulas of $\varphi$ by increasing syntactic depth; there are finitely many subformulas. For strategic cases, Algorithms 2–4 operate on the finite state set $Q$. In Pre, each state $q$ admits finitely many sabotage configurations $E' \subseteq Enabled(q)$; enumeration thus terminates. In Until and Release, the sets $S \subseteq Q$ evolve by monotone additions/removals. Since $Q$ is finite, each fixpoint stabilises in at most $|Q|$ iterations. Hence the overall procedure halts. □

Theorem 4.2 (Soundness & Completeness (Correctness)). *For every closed PTOTL formula $\varphi$, the set computed by Algorithm 1 equals the denotational satisfaction set induced by the PTOTL semantics over the PTKS of the underlying WPTA: for all $q \in Q$,*

$$q \in \mathsf{Sat}_{\mathrm{alg}}(\varphi) \iff \mathcal{Z}, q \models \varphi.$$

Proof sketch by structural induction on $\varphi$. *Base cases.* $\top$ is satisfied everywhere; for $p \in \mathrm{Ap}$ the algorithm selects all $(\ell, Z)$ with $p \in K(\ell)$; clock constraints are checked by DBM entailment $Z \models x \sim c$. These coincide with the semantics.

*Booleans.* $\neg$ and $\wedge$ are implemented as complement and intersection over $Q$, matching the Boolean semantics.

*Freeze.* For $j.\psi$, the algorithm symbolically resets the formula clock via the standard DBM update $Z[j \leftarrow 0]$ and then evaluates $\psi$ in $(\ell, Z[j \leftarrow 0])$. This implements the denotation "bind $j$ to the current time and evaluate $\psi$".

*Event-next under sabotage.* For $\langle \lightning_n^{\bowtie k} \rangle \bigcirc_{\mathrm{disc}} \varphi_1$, Algorithm 2 computes the set of $q$ for which *there exists* a budget-feasible $E' \subseteq Enabled(q)$ with masked non-renormalised probability $\sum_{q' \in \mathsf{Sat}(\varphi_1)}$

$P^{E'}(q, q') \bowtie k$. This is exactly the event-indexed probabilistic clause of the semantics.

*Until/Release under sabotage.* For $\langle \dagger_n^{\bowtie k} \rangle\, \alpha_1\, U\, \alpha_2$ and $\langle \dagger_n^{\bowtie k} \rangle\, \alpha_1\, R\, \alpha_2$ the algorithm computes, respectively, the least and greatest fixpoints of the standard backward-characterisations:

$$\mu S.\, L_2 \cup (L_1 \cap \mathsf{Pre}_{n,\bowtie k}(S)) \quad \text{and} \quad \nu S.\, L_2 \cap (L_1 \cup \mathsf{Pre}_{n,\bowtie k}(S)),$$

with $L_i = \mathsf{Sat}(\alpha_i)$. These match the denotational semantics by standard arguments for temporal fixpoints with one-step predecessors. Thus each constructor is computed correctly. □

Theorem 4.3 (Complexity). *Let $|\varphi|$ be the size of the formula, $|Q|$ the number of zone states, and $d = \max_{q \in Q} |Enabled(q)|$. Then the time to compute* $\mathsf{Sat}(\varphi)$ *is* $O\Big(|\varphi| \cdot |Q| \;\; + \;\; |Q| \cdot 2^d\Big)$ *for non-temporal/next-only cases, and* $O\Big(|\varphi| \cdot |Q|^2 \cdot 2^d\Big)$ *with Until/Release, up to polynomial DBM-manipulation costs. In particular:*

- *If $|\mathcal{X}| = 1$ (single system clock), the zone graph size is polynomial, and model checking is in* ***PSPACE****.*
- *For $|\mathcal{X}| \geq 2$, the zone graph can be exponential in the input, yielding an* ***EXPTIME*** *upper bound; hardness follows from classical multi-clock timed model-checking results, so the bound is tight.*

Justification. The bottom-up traversal contributes $O(|\varphi| \cdot |Q|)$. For each state, sabotage enumerates subsets $E' \subseteq Enabled(q)$, i.e. up to $2^d$ configurations; each check requires only polynomial-time DBM tests and summations. Until/Release iterate over $S \subseteq Q$ and stabilise in at most $|Q|$ rounds, yielding the additional $|Q|$ factor. For one clock, canonical extrapolations ensure $|Q|$ is polynomial, so the overall space is polynomial; with $\geq$ 2 clocks, $|Q|$ may be exponential, implying the stated bound. □

## 5 CASE STUDY: AUTOMOTIVE MTD

Modern connected vehicles expose Internet-facing IVI and telematics, an in-vehicle security gateway, and safety-critical controllers over CAN/Ethernet with cellular/Wi-Fi/Bluetooth/V2X ingress. Empirical studies show that multi-stage intrusions commonly enter via IVI or external communications, pivot through the gateway, and attempt to reach safety domains [10, 11, 23, 48, 51, 54, 58]. We instantiate this centralized–domainal vehicle architecture as a WPTA in the Fig. 1 and use PTOTL to certify defender MTD techniques claims under time, probability, and budget. Locations S0–S7 denote the external point of the vehicle (Init), the infotainment (IVI) and communication (COMMS) system, a Secure gateway (SGW) between the external and core functionality, the compute central unit (CCU), the telematic contol unit (TCU), the ADAS and Motion who represent the most critical part of an automotive CPS. All external ingress traverses S3. Each WPTA edge is labelled by an action label $a_x$, a guard for the clock $x$, an optional reset of $x$, a base probability $p$ with sum to 1, and a deactivation cost $w$. The topology consists of multiple ingress routes funnelling through the gateway, by timed gateway to control transfers, and a single actuation jump to *Motion* deactivatable within its guard window for fixed cost $w$ our MTD lever. The model in Fig. 1 shows the validated guards, probabilities, resets, and sabotage costs that drive verification.

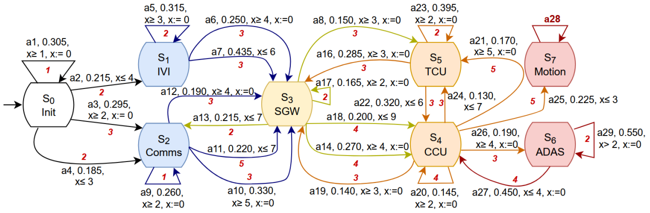

**Figure 1: Automotive WPTA where nodes $S_6$ and $S_7$ represent the attacker's goals, and $S_4$ with $S_5$ the sensitive part and $S_3$ a control point. Larger version in supplementary material.**

We use the semantics of Sections 3–4 and assume: a unique attacker location per discrete step observable by the defender, a *per-step* budget that disables any subset of *currently enabled* outgoing edges at that location, sub-stochastic masking with no renormalisation, and a single dense-time clock $x$.

We evaluate a critical property that bounds the probability for an attacker to reach *Motion* and thus control the vehicle. Let at_S7 label S7 and $j$ be a freeze clock. We denote the property by:

$$g_{\text{motion}} \;:=\; j.\, \big\langle \dagger_B^{\leq \alpha} \big\rangle \,\big(\, \mathsf{true}\, U \,\left(j \leq T \wedge \mathsf{at\_S7}\right) \big).$$

$$g_{\text{motion}} \;:=\; j.\, \big\langle \dagger_4^{\leq 0.76} \big\rangle \,\big(\, \mathsf{true}\, U \,\left(j \leq 9 \wedge \mathsf{at\_S7}\right) \big).$$

i.e., there exists a budget $B$ obstruction strategy for the defender that keeps the probability of reaching Motion within time $T$ at most $\alpha$. Operationally, this bounds the risk of actuator takeover within the mission horizon while respecting a per-step MTD cost limit. It also captures the practical intent of MTD playbooks: to buy time for detection/rollback or to contain lateral movement before Motion is reachable, without assuming perfect patchability [10, 47].

An example of concrete witness strategy deactivates $a25$ whenever its guard holds, to protect the motion. Its per-step cost is 5, so any $B \geq 5$ suffices. Under sub-stochastic masking, we have $P(\mathrm{S4} \longrightarrow \mathrm{S7}) = 0$ whenever $x \leq 3$. Because every feasible route to S7 factors through $a25$, the masked predecessor of S7 is empty, the Until fixpoint never adds S7, and $g_{\text{motion}}$ holds from S0 for any $T$ and any $\alpha \geq 0$. This last-hop certificate is minimal: if $B < 5$, $a25$ cannot be disabled and positive mass toward *Motion* remains. If actuation cannot be masked, upstream obstruction reduces risk. Disabling both gateway families $\{a14, a18\}$ (cost 8) blocks S3 ⟶ S4 throughout their active windows and suffices whenever all $j \leq T$ paths to S7 must pass S4. Disabling only one family (cost 4) removes either the compliant or the exploit traversal, dropping up to 0.270 or 0.200 *base* probability per enabled step; hardening ingress by masking $a11$ (cost 5) suppresses a dominant OTA/back-end relay. These effects are computed exactly by the masked predecessor operator, and feasibility is monotone in $B$: if $g_{\text{motion}}$ holds at budget $B$, it holds for any $B' \geq B$. We can also thinks about another complex real attack scenario were we can assume that if we force the attacker to go from $S_3$ (SGW) to $S_4$ (TCU), this will save enough time to detect the attacker or suspicious behavior the attacker having to force the passage to go to $S_5$ (CCU). Overall, the instance shows how PTOTL expresses and certifies time- and risk-bounded MTD claims on a stochastic real-time substrate, with constructive witnesses and quantitative budgets that align with engineering constraints [11, 26, 39, 47].

# 6 COMPARISON WITH EXISTING LOGIC

We position PTOTL against two reference formalisms for probabilistic real-time reasoning: PTCTL [38] and PTATL [37].

## 6.1 *PTOTL* vs *PTCTL*

We compare PTOTL with PTCTL [38], whose syntax is:

$$\theta ::= a \mid \zeta \mid \neg\theta \mid \theta \vee \theta \mid z.\theta \mid P^{\bowtie\lambda}[\theta_1\, U\, \theta_2],$$

where $a \in AP$ is an atomic proposition, $\zeta$ a DBM zone constraint, $z$ a formula clock, and $\lambda \in [0,1]$ a probability threshold.

Let PTOTL$^0$ denote the 0-fragment of PTOTL, i.e., the set of formulas where the sabotage budget $n$ is fixed to 0. Let $\mathrm{At}(\zeta)$ denote the set of atomic DBM constraints composing $\zeta$ (each of the form $x \sim c$ or $x - y \sim c$) and $\chi$ the *next* operator where $\chi ::= \bigcirc_{\text{disc}}\, \varphi \mid \psi$. We define a mapping $(\cdot)^\bullet$ from PTCTL formulas to PTOTL$^0$ as follows:

$$\begin{array}{rcl}
(a)^\bullet & = & a \\
(\zeta)^\bullet & = & \bigwedge_{\chi \in \mathrm{At}(\zeta)} \chi \\
(\neg\theta)^\bullet & = & \neg(\theta)^\bullet \\
(\theta_1 \vee \theta_2)^\bullet & = & \neg\big(\neg(\theta_1)^\bullet \wedge \neg(\theta_2)^\bullet\big) \\
(z.\theta)^\bullet & = & j.(\theta)^\bullet \\
\big(P^{\bowtie\lambda}[\theta_1 U \theta_2]\big)^\bullet & = & \langle {+}^{\bowtie\lambda}_{0} \rangle\big((\theta_1)^\bullet\ U\ (\theta_2)^\bullet\big).
\end{array}$$

Disjunction is expressed via De Morgan's law, since $\vee$ isn't a primitive operator of PTOTL. We shows that every PTCTL formula translates to PTOTL$^0$. Conversely, PTOTL$^0$ without the *next* operator, matches PTCTL in expressiveness.

Theorem 6.1. *For every WPTA $\mathcal{M}$, symbolic state $s$, and PTCTL formula $\theta$, we have:*

$$\mathcal{M}, s \models_{PTCTL} \theta \quad \textit{iff} \quad \mathcal{M}, s \models_{PTOTL^0 \setminus \chi} (\theta)^\bullet.$$

*Discussion.* This result shows that PTCTL is fully embeddable into the 0-fragment of PTOTL. However, PTOTL is strictly more expressive: (i) positive budgets $n{>}0$ capture cost-bounded obstruction absent from PTCTL; and (ii) even for $n{=}0$, the probabilistic *next* has no counterpart in PTCTL.

## 6.2 *PTOTL* vs *PTATL*

We contrast PTOTL with PTATL [37], which extends ATL with dense-time bounds and probabilistic thresholds for coalitions [14, 16, 25]. These logics differ in their primitives: PTATL quantifies coalition power in concurrent timed games, while PTOTL specifies cost-bounded transition disabling in timing via freeze with only 2 agents. We focus on single-layer PTATL formulas with no nested coalition modalities, and we don't claim completeness. We recall PTATL syntax and give a term-by-term comparison with PTOTL.

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\!\langle A \rangle\!\rangle^{\bowtie z}(\varphi\ U_I\ \varphi) \mid \langle\!\langle A \rangle\!\rangle^{\bowtie z}(\varphi\ R_I\ \varphi),$$

where $p \in AP$ is an atomic proposition, $A \subseteq \mathcal{A}$ is a coalition, $\bowtie \in \{<, \leq, \geq, >\}$, $z \in [0,1]$, and $I \subseteq \mathbb{R}_{\geq 0}$ is a time interval.

To relate these logics (coalition→sabotage), we define $(\cdot)^\dagger$ mapping PTATL to a PTOTL fragment by replacing coalition power with per-step, cost-bounded sabotage. Choose $B : 2^{\mathcal{A}} \to \mathbb{N}$ giving each coalition $A$ a budget $B(A)$, and encode time bounds with a freeze clock $j$:

$$\begin{aligned}
(p)^\dagger &= p \\
(\neg\varphi)^\dagger &= \neg(\varphi)^\dagger \\
(\varphi_1 \wedge \varphi_2)^\dagger &= (\varphi_1)^\dagger \wedge (\varphi_2)^\dagger \\
\big(\langle\!\langle A \rangle\!\rangle^{\bowtie z}\,(\varphi_1\ U_{[a,b]}\ \varphi_2)\big)^\dagger &= j.\ \langle {+}^{\bowtie z}_{B(A)} \rangle\ \Big((\varphi_1)^\dagger\ U\ \big((\varphi_2)^\dagger \wedge a \leq j \leq b\big)\Big) \\
\big(\langle\!\langle A \rangle\!\rangle^{\bowtie z}\,(\varphi_1\ U_{(a,b]}\ \varphi_2)\big)^\dagger &= j.\ \langle {+}^{\bowtie z}_{B(A)} \rangle\ \Big((\varphi_1)^\dagger\ U\ \big((\varphi_2)^\dagger \wedge a < j \leq b\big)\Big) \\
(\langle\!\langle A \rangle\!\rangle^{\bowtie z}\,(\varphi_1\ R_I\ \varphi_2))^\dagger &= j.\ \langle {+}^{\bowtie z}_{B(A)} \rangle\ \Big((\varphi_1)^\dagger\ R\ \big((\varphi_2)^\dagger \wedge j \in I\big)\Big)
\end{aligned}$$

Here, $j \in I$ denotes the guard, and $j.$ aligns the dense-time window for $U/R$. We use a single sabotage envelope per formula; nested coalitions are over-approximated bottom-up. Under assumptions (A1) joint actions compiled to transitions, (A2) coalition power over-approximated by per-step disabling up to $B(A)$, and (A3) time bounds encoded by $j$, one-way soundness holds:

$$\mathcal{M} \models_{\text{PTATL}} \phi \;\Rightarrow\; \mathcal{M} \models_{\text{PTOTL}} \phi^\dagger.$$

Completeness does not hold in general; coalition quantification and imperfect information in PTATL have no primitive in PTOTL.

Theorem 6.2 (Incomparability). *$\mathcal{L}_{PTOTL}$ and $\mathcal{L}_{PTATL}$ are incomparable but complementary: there exist formulas expressible in PTOTL but not in PTATL, and vice versa.*

*Discussion.* For MTD style defenses, where operators *disable transitions* under per-step budgets within timed windows, PTOTL gives concise residual-risk bounds and constructive witnesses.

# 7 CONCLUSION AND FUTURE WORK

We introduced PTOTL, a logic over WPTA that unifies dense time, probabilistic thresholds, and cost-bounded transition disabling under sub-stochastic semantics. We provided syntax/semantics and a symbolic zone-based model checker via masked predecessors and fixpoints. Complexity matches classical bounds *PSPACE-complete*, we show that PTOTL extend PTCTL and is complementary with PTATL. An automotive case study shows concisely captures MTD claims—time-bounded, budgeted reduction of compromise probability with quantitative guarantees and constructive witnesses.

Future work includes considering cumulative global budgets and introducing optimizations; multiple interacting sabotage envelopes to approximate coalitional behaviour; imperfect information and bounded recall, using sound approximations or hybrid methods [13, 17, 28, 30, 31]; quantitative and parametric synthesis; and a reference implementation integrated into VITAMIN [32] for empirical evaluation on realistic security scenarios.

# ACKNOWLEDGMENTS

This work is conducted within the Intelligent Cybersecurity for Mobility Systems (ICMS) research chair at Télécom Paris, founded by Ampere, BCG, IRT SystemX, Renault, Solent, Thales, Valeo, and ZF Group, and supported by the Fondation Mines-Télécom.

# REFERENCES

[1] R. Alur. (1991). *Techniques for Automatic Verification of Real-Time Systems.* Ph.D. Dissertation. Stanford University.

[2] R. Alur and D. Dill. 1994. A theory of timed automata. *Theoretical computer science* 126, 2 (1994), 183–235.

[3] Rajeev Alur and David L. Dill. 1994. A theory of timed automata. *Theoretical Computer Science* 126, 2 (April 1994), 183–235. https://doi.org/10.1016/0304-3975(94)90010-8

[4] R. Alur, T.A. Henzinger, and O. Kupferman. 2002. Alternating-time temporal logic. *J. ACM* 49, 5 (2002), 672–713.

[5] R. Alur, S. La Torre, and G. J. Pappas. 2001. Optimal Paths in Weighted Timed Automata. In *Computation and Control.* 49–62.

[6] Rajeev Alur, Salvatore La Torre, and George J. Pappas. 2001. *Optimal Paths in Weighted Timed Automata.* Springer Berlin Heidelberg, 49–62. https://doi.org/10.1007/3-540-45351-2_8

[7] Étienne André. 2021. *IMITATOR 3: Synthesis of Timing Parameters Beyond Decidability.* Springer International Publishing, 552–565. https://doi.org/10.1007/978-3-030-81685-8_26

[8] J. Arias, W. Jamroga, W. Penczek, L. Petrucci, and T. Sidoruk. (2023). Strategic (Timed) Computation Tree Logic. arXiv:2302.13405 [cs.LO]

[9] G. Aucher, J. Van Benthem, and D. Grossi. 2018. Modal logics of sabotage revisited. *Journal of Logic and Computation* 28, 2 (2018), 269 – 303. https://doi.org/10.1093/logcom/exx034

[10] Maxime Ayrault. 2022. Dynamic Defenses for Improved Resilience of Connected Cars. In *Dynamic Defenses for Improved Resilience of Connected Cars.* HAL. https://theses.hal.science/tel-04498523/file/107694_AYRAULT_2022_archivage.pdf

[11] Maxime Ayrault, Ulrich Kühne, and Étienne Borde. 2022. Finding Optimal Moving Target Defense Strategies: A Resilience Booster for Connected Cars. *Information* 13, 5 (May 2022), 242. https://doi.org/10.3390/info13050242

[12] Christel Baier and Joost-Pieter Katoen. 2008. *Principles of model checking.* MIT Press.

[13] F. Belardinelli, A. Ferrando, and V. Malvone. 2023. An abstraction-refinement framework for verifying strategic properties in multi-agent systems with imperfect information. *Artif. Intell.* 316 (2023), 103847. https://doi.org/10.1016/j.artint.2022.103847

[14] Francesco Belardinelli, Wojciech Jamroga, Damian Kurpiewski, Vadim Malvone, and Aniello Murano. 2019. Strategy Logic with Simple Goals: Tractable Reasoning about Strategies. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-2019).* International Joint Conferences on Artificial Intelligence Organization, 88–94. https://doi.org/10.24963/ijcai.2019/13

[15] Francesco Belardinelli, Wojciech Jamroga, Munyque Mittelmann, and Aniello Murano. 2023. Strategic Abilities of Forgetful Agents in Stochastic Environments. arXiv:2310.17240 [cs.MA] https://arxiv.org/abs/2310.17240

[16] Francesco Belardinelli, Wojtek Jamroga, Munyque Mittelmann, and Aniello Murano. 2024. Verification of Stochastic Multi-Agent Systems with Forgetful Strategies. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems* (Auckland, New Zealand) *(AAMAS '24).* International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 160–169.

[17] F. Belardinelli, A. Lomuscio, V. Malvone, and E. Yu. 2022. Approximating Perfect Recall when Model Checking Strategic Abilities: Theory and Applications. *J. Artif. Intell. Res.* 73 (2022), 897–932. https://doi.org/10.1613/jair.1.12539

[18] J. Bengtsson and W. Yi. 2004. Timed Automata: Semantics, Algorithms and Tools. In *Lectures on Concurrency and Petri Nets.* 87–124. https://doi.org/10.1007/b98282

[19] P. Bouyer, U. Fahrenberg, K. G. Larsen, and N. Markey. 2011. Quantitative analysis of real-time systems using priced timed automata. *Communications of the ACM* (2011). https://doi.org/10.1145/1995376.1995396

[20] D. Catta, J. Leneutre, and V. Malvone. 2023. Attack Graphs & Subset Sabotage Games. *Intelligenza Artificiale* 17, 1 (2023), 77–88. https://doi.org/10.3233/IA-221080

[21] D. Catta, J. Leneutre, and V. Malvone. 2023. Obstruction Logic: A Strategic Temporal Logic to Reason About Dynamic Game Models. In *ECAI 2023 - 26th European Conference on Artificial Intelligence.* https://doi.org/10.3233/FAIA230292

[22] K. Chatterjee, T. Henzinger, and N. Piterman. 2007. Strategy Logic. In *CONCUR07.* 59–73.

[23] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno. 2011. Comprehensive Experimental Analyses of Automotive Attack Surfaces. In *20th USENIX Security Symposium (USENIX Security 11).* USENIX Association, San Francisco, CA. https://www.usenix.org/conference/usenix-security-11/comprehensive-experimental-analyses-automotive-attack-surfaces

[24] Taolue Chen, Vojtěch Forejt, Marta Kwiatkowska, David Parker, and Aistis Simaitis. 2013. Automatic verification of competitive stochastic systems. *Formal Methods in System Design* 43, 1 (Feb. 2013), 61–92. https://doi.org/10.1007/s10703-013-0183-7

[25] Taolue Chen and Jian Lu. 2007. Probabilistic Alternating-time Temporal Logic and Model Checking Algorithm. In *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)*, Vol. 2. 35–39. https://doi.org/10.1109/FSKD.2007.458

[26] J. Cho, D. Sharma, H. Alavizadeh, S. Yoon, Noam B-A., T. Moore, Dan Kim, H. Lim, and F. Nelson. 2020. Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense. *IEEE Communications Surveys & Tutorials* (2020).

[27] E. M. Clarke, O. Grumberg, and D. A. Peled. (1999). *Model Checking.* The MIT Press, Cambridge, Massachusetts.

[28] C. Dima and F. L. Tiplea. 2011. Model-checking ATL under Imperfect Information and Perfect Recall Semantics is Undecidable. *CoRR* (2011). http://arxiv.org/abs/1102.4225

[29] M. Faella, S. La Torre, and A. Murano. 2014. Automata-theoretic decision of timed games. *Theor. Comput. Sci.* 515 (2014), 46–63. https://doi.org/10.1016/J.TCS.2013.08.021

[30] A. Ferrando and V. Malvone. 2022. Towards the Combination of Model Checking and Runtime Verification on Multi-agent Systems. In *20th International Conference, PAAMS 2022.* https://doi.org/10.1007/978-3-031-18192-4_12

[31] A. Ferrando and V. Malvone. 2023. Towards the Verification of Strategic Properties in Multi-Agent Systems with Imperfect Information. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2023.* https://doi.org/10.5555/3545946.3598713

[32] Angelo Ferrando and Vadim Malvone. 2025. VITAMIN: VerIficaTion of A MultI ageNt system. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems* (Detroit, MI, USA) *(AAMAS '25).* International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 3023–3025.

[33] T. A Henzinger, P-H. Ho, and H. Wong-Toi. 1997. HyTech: A model checker for hybrid systems. In *Computer Aided Verification: 9th International Conference, CAV'97.*

[34] T. A. Henzinger and V. S. Prabhu. 2006. Timed Alternating-Time Temporal Logic. In *4th International Conferences on Formal Modelling and Analysis of Timed Systems (FORMATS'06).*

[35] Karel Horák and Branislav Bošanský. 2019. Solving Partially Observable Stochastic Games with Public Observations. *Proceedings of the AAAI Conference on Artificial Intelligence* 33, 01 (July 2019), 2029–2036. https://doi.org/10.1609/aaai.v33i01.33012029

[36] Xiaowei Huang, Kaile Su, and Chenyi Zhang. 2012. Probabilistic Alternating-Time Temporal Logic of Incomplete Information and Synchronous Perfect Recall. *Proceedings of the National Conference on Artificial Intelligence* 1.

[37] Wojciech Jamroga, Marta Kwiatkowska, Wojciech Penczek, Laure Petrucci, and Teofil Sidoruk. 2025. Probabilistic Timed ATL. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems* (Detroit, MI, USA) *(AAMAS '25).* International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1051–1059.

[38] Marcin Jurdzinski, Francois Laroussinie, and Jeremy Sproston. 2008. Model Checking Probabilistic Timed Automata with One or Two Clocks. (2008). https://doi.org/10.48550/ARXIV.0809.0060

[39] K. Kaynar. 2016. A Taxonomy for Attack Graph Generation and Usage in Network Security. *J. Inf. Secur. Appl.* 29, C (2016), 27–56.

[40] Samantha Kleinberg. 2012. *Causality, Probability, and Time.* Cambridge University Press, 241–250.

[41] M. Kwiatkowska, G. Norman, and D. Parker. 2011. PRISM 4.0: Verification of Probabilistic Real-time Systems. In *Proc. 23rd International Conference on Computer Aided Verification (CAV'11) (LNCS, Vol. 6806)*, G. Gopalakrishnan and S. Qadeer (Eds.). Springer, 585–591.

[42] Marta Kwiatkowska, Gethin Norman, David Parker, and Gabriel Santos. 2021. Automatic verification of concurrent stochastic systems. *Formal Methods in System Design* 58, 1–2 (Jan. 2021), 188–250. https://doi.org/10.1007/s10703-020-00356-y

[43] F. Laroussinie, N. Markey, and G. Oreiby. 2006. Model-Checking Timed ATL for Durational Concurrent Game Structures. In *Formal Modeling and Analysis of Timed Systems, 4th International Conference, FORMATS 2006.* https://doi.org/10.1007/11867340_18

[44] Jean Leneutre, Vadim Malvone, and James Ortiz. 2024. Probabilistic Obstruction Temporal Logic: a Probabilistic Logic to Reason about Dynamic Models. https://doi.org/10.48550/ARXIV.2411.00025

[45] Jean Leneutre, Vadim Malvone, and James Ortiz. 2025. Jean Leneutre, Vadim Malvone, and James Ortiz. 2025. Timed Obstruction Logic: A Timed Approach to Dynamic Game Reasoning IFAAMAS.

[46] C. Löding and P. Rohde. 2003. Model Checking and Satisfiability for Sabotage Modal Logic. In *FST TCS 2003: Foundations of Software Technology and Theoretical*

*Computer Science*. https://doi.org/10.1007/978-3-540-24597-1_26

[47] Hector Marco-Gisbert and Ismael Ripoll Ripoll. 2019. Address Space Layout Randomization Next Generation. *Applied Sciences* 9, 14 (2019).

[48] C. Miller and C. Valasek. 2015. Comprehensive Experimental Analyses of Automotive Attack Surfaces. In *Remote Exploitation of an Unaltered Passenger Vehicle. Black Hat USA*. USENIX Association. https://www.usenix.org/legacy/events/sec11/tech/full_papers/Checkoway.pdf

[49] Gethin Norman, David Parker, and Jeremy Sproston. 2012. Model checking for probabilistic timed automata. *Formal Methods in System Design* 43, 2 (Oct. 2012), 164–190. https://doi.org/10.1007/s10703-012-0177-x

[50] G. D. Plotkin. 1981. *A Structural Approach to Operational Semantics*. Technical Report DAIMI FN-19. University of Aarhus. http://citeseer.ist.psu.edu/plotkin81structural.html

[51] Bradley Potteiger, Zhenkai Zhang, Long Cheng, and Xenofon Koutsoukos. [n.d.]. A Tutorial on Moving Target Defense Approaches within Automotive Cyber-Physical Systems. Volume 2 - 2021 ([n. d.]). https://doi.org/10.3389/ffutr.2021.792573

[52] P.Y. Schobbens. 2004. Alternating-Time Logic with Imperfect Recall. *ENTCS* 85, 2 (2004), 82–93.

[53] A. Di Stasio, P. D. Lambiase, V. Malvone, and A. Murano. 2018. Dynamic Escape Game. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018*. http://dl.acm.org/citation.cfm?id=3237984

[54] Bhosale Akshay Tanaji and Sayak Roychowdhury. 2024. A Survey of Cybersecurity Challenges and Mitigation Techniques for Connected and Autonomous Vehicles. *IEEE Transactions on Intelligent Vehicles* (2024), 1–18. https://doi.org/10.1109/TIV.2024.3493938

[55] J. van Benthem. 2005. *An Essay on Sabotage and Obstruction*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-32254-2_16

[56] Bryan C. Ward, Steven R. Gomez, Richard W. Skowyra, David Bigelow, Jason Martin, James Landry, and Hamed Okhravi. 2018. Survey of Cyber Moving Targets Second Edition. https://api.semanticscholar.org/CorpusID:70305693

[57] Rui Yan, Gabriel Santos, Gethin Norman, David Parker, and Marta Kwiatkowska. 2024. Partially Observable Stochastic Games with Neural Perception Mechanisms. In *Formal Methods: 26th International Symposium, FM 2024, Milan, Italy, September 9–13, 2024, Proceedings, Part I* (Milan, Italy). Springer-Verlag, Berlin, Heidelberg, 363–380. https://doi.org/10.1007/978-3-031-71162-6_19

[58] Jianjun Zheng and Akbar Siami Namin. 2019. A Survey on the Moving Target Defense Strategies: An Architectural Perspective. *Journal of Computer Science and Technology* 34, 1 (Jan. 2019), 207–233. https://doi.org/10.1007/s11390-019-1906-z