



HAL
open science

Execution-time opacity control for timed automata

Étienne André, Marie Duflot, Laetitia Laversa, Engel Lefauchaux

► **To cite this version:**

Étienne André, Marie Duflot, Laetitia Laversa, Engel Lefauchaux. Execution-time opacity control for timed automata. 22nd International Conference on Software Engineering and Formal Methods (SEFM 2024), Alexandre Madeira; Alexander Knapp, Nov 2024, Aveiro (Portugal), Portugal. 10.1007/978-3-031-77382-2_20 . hal-04732493v2

HAL Id: hal-04732493

<https://inria.hal.science/hal-04732493v2>





Submitted on 9 Dec 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

Execution-time opacity control for timed automata^{*}

Étienne André^{1,2}, Marie Dufлот³, Laetitia Laversa^{1,4}, and
Engel Lefauchaux³

¹ Université Sorbonne Paris Nord, CNRS, Laboratoire d'Informatique de Paris Nord,
LIPN, F-93430 Villetaneuse, France

² Institut universitaire de France (IUF)

³ Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

⁴ Université Paris Cité, IRIF, CNRS UMR 8243, Paris, France

Abstract. Timing leaks in timed automata (TA) can occur whenever an attacker is able to deduce a secret by observing some timed behavior. In execution-time opacity, the attacker aims at deducing whether a private location was visited, by observing only the execution time. It can be decided whether a TA is opaque in this setting. In this work, we tackle control, and show that we are able to decide whether a TA can be controlled at runtime to ensure opacity. Our method is constructive, in the sense that we can exhibit such a controller. We also address the case when the attacker cannot have an infinite precision in its observations.

Keywords: timed automata, opacity, side-channel attacks, timed control

1 Introduction

In order to infer sensitive information, side-channels attacks [27] exploit various observable aspects of a system rather than directly exploiting its computational processes, such as power consumption, electromagnetic emissions, or time. In particular, by observing subtle differences in timing, attackers can infer valuable information about the internal state of the system. For example, in [21], a timing attack vulnerability is identified in the Chinese public key cryptography standard; the authors show how the most significant zero-bit leakage obtained from the execution time allows to extract the secret key.

Timed automata (TAs) [1] are a powerful formalism to reason about real-time systems mixing timing constraints and concurrency. Timing leaks can occur whenever an attacker is able to deduce a secret by observing some behavior of a TA.

^{*} This is the author (and extended) version of the manuscript of the same name published in the proceedings of the 22nd International Conference on Software Engineering and Formal Methods (SEFM 2024). The final version is available at [10.1007/978-3-031-77382-2_20](https://doi.org/10.1007/978-3-031-77382-2_20). This work is partially supported by ANR-NRF ProMiS (ANR-19-CE25-0015 / 2019 ANR NRF 0092) and by ANR BisoUS (ANR-22-CE48-0012).

Related works Franck Cassez proposed in [20] a first definition of *timed* opacity for TAs: the system is opaque when an attacker can never deduce whether some secret sequence of actions (possibly with timestamps) was performed, by only observing a given set of observable actions together with their timestamp. It is then proved in [20] that it is undecidable whether a TA is opaque. This notably relates to the undecidability of timed language inclusion for TAs [1]. The undecidability holds even for the restricted class of *event-recording automata* [2] (a subclass of TAs), for which language inclusion is known to be decidable. The aforementioned negative result leaves hope only if the definition or the setting is changed, which was done in four main lines of works.

First, in [29,30], the input model is simplified to *real-time automata* [23], a restricted formalism compared to TAs considering a single clock, reset at each transition. [26] works on constant-time labeled automata, a subclass of real-time automata where events occur at constant values. In this setting, initial-state and current-state opacity become decidable. In [31], Zhang studies labeled real-timed automata (a subclass of labeled TAs); in this setting, state-based (at the initial time, the current time, etc.) opacity is proved to be decidable by extending the observer (that is, the classical powerset construction) from finite automata to labeled real-timed automata.

Second, in [5,7], the opacity was studied in the setting of Cassez' definition, but with restrictions in the model: one-clock automata, one-action automata, or over discrete time. Similarly, in [25], discrete-timed automata with several clocks are considered and transformed into tick automata in order to verify the current-state opacity.

Third, in [4], the authors consider a time-bounded notion of the opacity of [20], where the attacker has to disclose the secret before an upper bound, using a partial observability. Deciding opacity in this setting is shown to be decidable for TAs.

Fourth, in [9], an alternative definition to Cassez' opacity is proposed, by studying execution-time opacity (ET-opacity): the attacker has only access to the *execution time* of the system, as opposed to Cassez' partial observations where some events (with their timestamps) are observable. In that case, most problems for TAs become decidable.

Several works address non-interference for TAs. Different notions of equivalence (e.g., bisimulation) can be considered for this property. Several papers [13,14,8] present some decidability results, while control is considered in [15]. General security problems for TAs are surveyed in [11].

Note that a preliminary version of control is considered in [6] for ET-opacity, but only untimed, i.e., the actions could only be enabled or disabled once and for all, thus severely diminishing the possibilities to render the system ET-opaque. Finally note that [9] considers also *parametric* versions of the opacity problems, in which timing parameters [3] can be used in order to make the system ET-opaque. Our notion of control is orthogonal to parameter synthesis, as another way to ensure the system becomes ET-opaque.

Controller synthesis can be described and solved thanks to game theory; finding a strategy for a controller can be equivalent to computing a winning strategy in a corresponding game. Several game models have been considered, as timed games that can be used to solve synthesis problem on timed automata. In this context, [12] aims to restrict the transition relation in order to satisfy certain properties, while [24] completes this result, minimizing the execution time, and [17] studies the reachability with robust strategies only.

Contributions In this work, we aim at tuning a system to make it ET-opaque, by *controlling* it at runtime.

Our attacker model is as follows: the attacker has a knowledge of the system model, but can only observe the execution time. This can correspond to an attacker observing the energy consumption of a device, clearly denoting the execution time of a program or process; or to an attacker observing communications over a shared network, with an observable message acknowledging the end of execution.

As usual, we consider that the system actions are partitioned between controllable and uncontrollable. Our controller relies on the following notion of strategy: at each timestamp, the strategy enables only a subset of the controllable actions. Our technique relies on a novel *ad-hoc* construction inspired by the region automaton for TAs. We show that the existence of a controller to make the system ET-opaque is decidable, and our approach is constructive. We also address the case when the attacker cannot have an infinite precision in its observations.

Outline Section 2 recalls the necessary material. Section 3 defines the control problem for ET-opacity. Sections 4 and 5 define the core of our approach. Section 6 extends our method when the attacker cannot have an infinite precision in observing the execution time. Section 7 highlights future works.

2 Preliminaries

Clocks are real-valued variables that all evolve over time at the same rate. Throughout this paper, we assume a set $\mathbb{X} = \{x_1, \dots, x_H\}$ of *clocks*. A *clock valuation* is a function $\mu : \mathbb{X} \rightarrow \mathbb{R}_{\geq 0}^H$. Given $R \subseteq \mathbb{X}$, we define the reset of a valuation μ with respect to R , denoted by $[\mu]_R$, as follows: $[\mu]_R(x) = 0$ if $x \in R$, and $[\mu]_R(x) = \mu(x)$ otherwise. We write $\mathbf{0}$ for the clock valuation assigning 0 to all clocks. Given a constant $d \in \mathbb{R}_{\geq 0}$, $\mu + d$ denotes the valuation s.t. $(\mu + d)(x) = \mu(x) + d$, for all $x \in \mathbb{X}$. We assume $\bowtie \in \{<, \leq, =, \geq, >\}$. A *constraint* C is a conjunction of inequalities over \mathbb{X} of the form $x \bowtie d$, with $d \in \mathbb{Z}$. A table of notations is available in Appendix A.

We define timed automata as in [1], with an extra private location, which encodes the secret that shall not be leaked.

Definition 1 (Timed automaton). A timed automaton (TA) \mathcal{A} is a tuple $\mathcal{A} = (\Sigma, L, \ell_0, \ell_{priv}, F, \mathbb{X}, I, E)$ where: 1) Σ is a finite set of actions, 2) L is a

finite set of locations, 3) $\ell_0 \in L$ is the initial location, 4) $\ell_{priv} \in L$ is the private location, 5) $F \subseteq L \setminus \{\ell_{priv}\}$ is the set of final locations, 6) $\mathbb{X} = \{x_1, \dots, x_H\}$ is a finite set of clocks, 7) I is the invariant, assigning to every $\ell \in L$ a constraint $I(\ell)$, 8) E is a finite set of edges $e = (\ell, g, a, R, \ell')$ where $\ell, \ell' \in L$ are the source and target locations, $a \in \Sigma \cup \{\varepsilon\}$, where ε denotes the silent action, $R \subseteq \mathbb{X}$ is a set of clocks to be reset, and g is a constraint over \mathbb{X} (called guard).

Example 1. Fig. 1a depicts a TA \mathcal{A}_1 with a single clock x , where $\Sigma = \{a, b, u\}$. The edge e_1 between ℓ_0 and the private location ℓ_{priv} is available only when the valuation of x is equal to 0. The edge e_6 between ℓ_0 and ℓ_2 resets x .

Since we are only interested in the (first) arrival time in a final location, the following assumption does not restrict our framework, but simplifies the subsequent definitions and results.

Assumption 1. *We consider every final location as urgent (where time cannot elapse): formally, there exists $x \in \mathbb{X}$ such that, for all $(\ell, g, a, R, \ell') \in E, \ell' \in F$, we have $x \in R$ and “ $x = 0$ ” $\in I(\ell')$. Moreover, final locations cannot have outgoing transitions: formally, there is no $(\ell, g, a, R, \ell') \in E$ s.t. $\ell \in F$.*

Definition 2 (Semantics of a TA). *Let $\mathcal{A} = (\Sigma, L, \ell_0, \ell_{priv}, F, \mathbb{X}, I, E)$ be a TA, the semantics of \mathcal{A} is given by the timed transition system $TTS_{\mathcal{A}} = (S, s_0, \Sigma \cup \mathbb{R}_{\geq 0}, \delta)$, with 1) $S = \{(\ell, \mu) \in L \times \mathbb{R}_{\geq 0}^H \mid \mu \models I(\ell)\}$, 2) $s_0 = (\ell_0, \mathbf{0})$, 3) δ consists of the discrete and (continuous) delay transition relations: i) discrete transitions: $(\ell, \mu) \xrightarrow{e} (\ell', \mu')$, if $(\ell, \mu), (\ell', \mu') \in S$, and there exists $e = (\ell, g, a, R, \ell') \in E$, such that $\mu' = [\mu]_R$, and $\mu \models g$. ii) delay transitions: $(\ell, \mu) \xrightarrow{d} (\ell, \mu + d)$, with $d \in \mathbb{R}_{\geq 0}$, if $\forall d' \in [0, d], (\ell, \mu + d') \in S$.*

We write $(\ell, \mu) \xrightarrow{d, e} (\ell', \mu')$ for a combination of a delay and a discrete transitions when $\exists \mu'' : (\ell, \mu) \xrightarrow{d} (\ell, \mu'') \xrightarrow{e} (\ell', \mu')$.

Given a TA \mathcal{A} with semantics $(S, s_0, \Sigma \cup \mathbb{R}_{\geq 0}, \delta)$, a *run* of \mathcal{A} is a finite alternating sequence of states of $TTS_{\mathcal{A}}$ and pairs of delays and edges starting from s_0 of the form $s_0, (d_0, e_0), s_1, \dots, s_n$ where for all $i < n, e_i \in E, d_i \in \mathbb{R}_{\geq 0}$ and $s_i \xrightarrow{d_i, e_i} s_{i+1}$. The duration of a run $\rho = s_0, (d_0, e_0), s_1, \dots, (d_{n-1}, e_{n-1}), s_n$ is $dur(\rho) = \sum_{0 \leq i \leq n-1} d_i$. We define $last(\rho) = s_n$.

Extra clock We will need an extra clock z that will help us later to keep track of the elapsed absolute time. This clock is reset exactly every 1 time unit, and therefore each reset corresponds to a “tick” of the absolute time. (Note that its actual value remains in $[0, 1]$ and therefore always matches the fractional part of the absolute time.) In all subsequent region constructions, we assume the existence of $z \in \mathbb{X}$. For each location ℓ , we add the constraint “ $z \leq 1$ ” to $I(\ell)$, and we add a self-loop edge $(\ell, z = 1, \varepsilon, \{z\}, \ell)$.

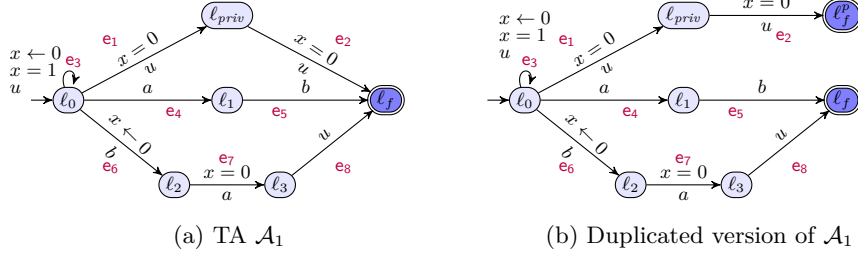


Fig. 1: A TA and its duplicated version (introduced in Section 4)

2.1 Regions

Given a TA \mathcal{A} , for a clock x_i , we denote by c_i the largest constant to which x_i is compared within the guards and invariants of \mathcal{A} : formally, $c_i = \max_j \{d_j \mid x_i \bowtie d_j \text{ appears in a guard or invariant of } \mathcal{A}\}$. Given a clock valuation μ and a clock x_i , $\lfloor \mu(x_i) \rfloor$ (resp. $\text{fr}(\mu(x_i))$) denotes the integral (resp. fractional) part of $\mu(x_i)$.

Definition 3 (Equivalence relation [1]). Two clocks valuations μ, μ' are equivalent, denoted by $\mu \approx \mu'$, when the following three conditions hold for any clocks $x_i, x_j \in \mathbb{X}$:

1. $\lfloor \mu(x_i) \rfloor = \lfloor \mu'(x_i) \rfloor$ or $\mu(x_i) > c_i$ and $\mu'(x_i) > c_i$;
2. if $\mu(x_i) \leq c_i$ and $\mu(x_j) \leq c_j$: $\text{fr}(\mu(x_i)) \leq \text{fr}(\mu(x_j))$ iff $\text{fr}(\mu'(x_i)) \leq \text{fr}(\mu'(x_j))$;
3. if $\mu(x_i) \leq c_i$: $\text{fr}(\mu(x_i)) = 0$ iff $\text{fr}(\mu'(x_i)) = 0$.

The equivalence relation \approx is extended to the states of $TTS_{\mathcal{A}}$: given two states $s = (\ell, \mu), s' = (\ell', \mu')$ of $TTS_{\mathcal{A}}$, we write $s \approx s'$ iff $\ell = \ell'$ and $\mu \approx \mu'$. We denote by $[s]$ and call *region* the equivalence class of a state s for \approx . The set of all regions of \mathcal{A} is denoted $\mathcal{R}_{\mathcal{A}}$. A region $r = [(\ell, \mu)]$ is *final* whenever $\ell \in F$. The set of final regions is denoted by $\mathcal{R}_{\mathcal{A}}^F$. A region r is *reachable* when there exists a run ρ such that $\text{last}(\rho) \in r$.

Region automaton We now define a region automaton based on [19, Proposition 5.3] with two-component labels on the transitions: the first component indicates how the fractional part of z evolved, with symbol “0” corresponding to an absence of change, symbol “0⁺” corresponding to a change remaining in $(0, 1)$ and symbol “1” corresponding to a change where the fractional part either starts or ends at 0. The second component is either ε if the transition represents time elapsing in the TA, or provides the action that labels the corresponding discrete transition in the TA.

Given a state $s = (\ell, \mu)$, and $d \in \mathbb{R}_{\geq 0}$, we write $s + d$ to denote $(\ell, \mu + d)$. We write $r \cup r'$ for $\{s \mid s \in r \text{ or } s \in r'\}$.

Definition 4 (Labeled Region Automaton). For a given TA \mathcal{A} , the region automaton $\mathcal{R}_{\mathcal{A}}$ is given by the tuple $(\mathcal{R}_{\mathcal{A}}, \Sigma^R, \delta^R)$ where:

1. $\mathcal{R}_{\mathcal{A}}$ is the set of states,
2. $\Sigma^{\mathcal{R}} = \{0, 0^+, 1\} \times (\Sigma \cup \{\varepsilon\})$,
3. given two regions $r, r' \in \mathcal{R}_{\mathcal{A}}$ and $\zeta \in \Sigma^{\mathcal{R}}$, we have $(r, \zeta, r') \in \delta^{\mathcal{R}}$ if there exist $s = (\ell, \mu) \in r, s' = (\ell', \mu') \in r'$ such that one of the following holds:
 - (a) $\zeta = (0, a)$ and $(\ell, \mu) \xrightarrow{e} (\ell', \mu') \in \delta$ in $TTS_{\mathcal{A}}$ with $e = (\ell, g, a, R, \ell')$ for some g and R ;
 - (b) $\zeta = (0^+, \varepsilon)$ and $\exists d \in \mathbb{R}_{>0}$ such that 1) $s \xrightarrow{d} s'$, 2) $\forall 0 < d' < d, s + d' \in r \cup r'$ and 3) $\text{fr}(\mu(z)) \neq 0$ and $\text{fr}(\mu'(z)) \neq 0$;⁵
 - (c) $\zeta = (1, \varepsilon)$ and $\exists d \in \mathbb{R}_{>0}$ such that 1) $s \xrightarrow{d} s'$, 2) $\forall 0 < d' < d, s + d' \in r \cup r'$, and 3) $\text{fr}(\mu'(z)) = 0$ iff $\text{fr}(\mu(z)) \neq 0$.

We write $r \xrightarrow{\zeta}_{\mathcal{R}} r'$ for $(r, \zeta, r') \in \delta^{\mathcal{R}}$.

2.2 Execution-time opacity of a TA

Let us now recall from [9] the notions of private and public runs. Given a TA \mathcal{A} and a run ρ , we say that ℓ_{priv} is *visited on the way to a final location* in ρ when ρ is of the form $(\ell_0, \mu_0), (d_0, \mathbf{e}_0), (\ell_1, \mu_1), \dots, (\ell_m, \mu_m), (d_m, \mathbf{e}_m), \dots, (\ell_n, \mu_n)$ for some $m, n \in \mathbb{N}$ such that $\ell_m = \ell_{priv}$ and $\ell_n \in F$. We denote by $Visit^{priv}(\mathcal{A})$ the set of those runs, and refer to them as *private runs*. We denote by $DVisit^{priv}(\mathcal{A})$ the set of all the durations of these runs.

Conversely, we say that ℓ_{priv} is *avoided on the way to a final location* in ρ when ρ is of the form $(\ell_0, \mu_0), (d_0, \mathbf{e}_0), (\ell_1, \mu_1), \dots, (\ell_n, \mu_n)$ with $\ell_n \in F$ and $\forall 0 \leq i < n, \ell_i \notin \{\ell_{priv}\}$. We denote the set of those runs by $Visit^{pub}(\mathcal{A})$, referring to them as *public runs*, and by $DVisit^{pub}(\mathcal{A})$ the set of all the durations of these public runs.

These concepts can be seen as the set of execution times from the initial location ℓ_0 to a final location while visiting (resp. not visiting) ℓ_{priv} .

Example 2. Consider the following two runs of the TA \mathcal{A}_1 in Fig. 1a. Note that we use (ℓ_0, v) as a shortcut for (ℓ_0, μ) such that $\mu(x) = v$.

$$\begin{aligned} \rho_1 &= (\ell_0, 0), (1, \mathbf{e}_3), (\ell_0, 0), (0, \mathbf{e}_1), (\ell_{priv}, 0), (0, \mathbf{e}_2), (\ell_f, 0) \\ \rho_2 &= (\ell_0, 0), (0.1, \mathbf{e}_6), (\ell_2, 0), (0, \mathbf{e}_7), (\ell_3, 0), (0.8, \mathbf{e}_8), (\ell_f, 0.8) \end{aligned}$$

Run $\rho_1 \in \overline{Visit^{priv}}(\mathcal{A}_1)$ is a private run, and $dur(\rho_1) = 1 \in D\overline{Visit^{priv}}(\mathcal{A}_1)$. Run $\rho_2 \in \overline{Visit^{pub}}(\mathcal{A}_1)$ is a public run with $dur(\rho_2) = 0.8 \in D\overline{Visit^{pub}}(\mathcal{A}_1)$.

Definition 5 (Full ET-opacity). A TA \mathcal{A} is fully ET-opaque when $D\overline{Visit^{priv}}(\mathcal{A}) = D\overline{Visit^{pub}}(\mathcal{A})$.

⁵ Condition 2) ensures that we only move from one region to the “next” one (no intermediate region), and condition 3) adds that we stay in the same region for z (changing the region for z is handled in item (c)).

That is, if for any run of duration d reaching a final location after visiting ℓ_{priv} , there exists another run of the same duration reaching a final location but not visiting ℓ_{priv} , and vice versa, then the TA is fully ET-opaque.

Example 3. Consider again \mathcal{A}_1 in Fig. 1a. Each time x is equal to 1, we can reset it via e_3 , and take edges e_1 and e_2 instantaneously. It results that $DVisit^{priv}(\mathcal{A}_1) = \mathbb{N}$. We have seen that $0.8 \in DVisit^{priv}(\mathcal{A}_1)$ in Example 2. So $DVisit^{priv}(\mathcal{A}_1) \neq \overline{DVisit^{priv}(\mathcal{A}_1)}$ and \mathcal{A}_1 is not fully ET-opaque.

3 Problem: Controlling TA to achieve ET-opacity

We assume $\Sigma = \Sigma_c \uplus \Sigma_u$ where Σ_c (resp. Σ_u) denotes controllable (resp. uncontrollable) actions. The uncontrollable actions are always available, whereas the controllable actions can be enabled and disabled at runtime.

The controller has a *strategy*, i.e., a function $\sigma : \mathbb{R}_{\geq 0} \rightarrow 2^{\Sigma_c}$ which associates to each time a subset of Σ_c , these actions are enabled, while others are disabled.

Definition 6 (Semantics of a controlled TA). *Given a TA $\mathcal{A} = (\Sigma, L, \ell_0, \ell_{priv}, F, \mathbb{X}, I, E)$ and a strategy $\sigma : \mathbb{R}_{\geq 0} \rightarrow 2^{\Sigma_c}$, the semantics of the controlled TA \mathcal{A}^σ is given by $TTS_{\mathcal{A}^\sigma} = (S, s_0, \Sigma \cup \mathbb{R}_{\geq 0}, \delta^\sigma)$ with*

1. $S = \{(\ell, \mu, t) \in L \times \mathbb{R}_{\geq 0}^H \times \mathbb{R}_{\geq 0} \mid \mu \models I(\ell)\}$, $s_0 = (\ell_0, \mathbf{0}, 0)$,
2. δ^σ consists of the discrete and (continuous) delay transition relation:
 - (a) discrete transitions: $(\ell, \mu, t) \xrightarrow{e}_\sigma (\ell', \mu', t)$, if $(\ell, \mu, t), (\ell', \mu', t) \in S$ and there exists $e = (\ell, g, a, R, \ell') \in E$ such that $\mu' = [\mu]_R$, $\mu \models g$, and $a \in \sigma(t) \cup \Sigma_u$ (a is enabled by the strategy at time t , or uncontrollable);
 - (b) delay transitions: $(\ell, \mu, t) \xrightarrow{d}_\sigma (\ell, \mu + d, t + d)$, with $d \in \mathbb{R}_{\geq 0}$, if $\forall d' \in \mathbb{R}_{> 0}$ such that $d' < d$, $(\ell, \mu + d', t + d') \in S$.

We write $(\ell, \mu, t) \xrightarrow{d, e}_\sigma (\ell', \mu', t')$ for a combination of a delay and discrete transitions when $\exists \mu''$ such that $(\ell, \mu, t) \xrightarrow{d}_\sigma (\ell, \mu'', t) \xrightarrow{e}_\sigma (\ell', \mu', t')$.

A run $\rho = (\ell_0, \mu_0), (d_0, \mathbf{e}_0), \dots, (\ell_n, \mu_n)$ is σ -compatible when, $\forall 0 \leq i < n$, it holds that $(\ell_i, \mu_i, \sum_{j < i} d_j) \xrightarrow{d_i, \mathbf{e}_i}_\sigma (\ell_{i+1}, \mu_{i+1}, \sum_{j \leq i} d_j)$. We let

- $Visit_\sigma^{priv}(\mathcal{A}) = \{\rho \mid \rho \text{ is private and } \sigma\text{-compatible}\}$,
- $Visit_\sigma^{pub}(\mathcal{A}) = \{\rho \mid \rho \text{ is public and } \sigma\text{-compatible}\}$,
- $DVisit_\sigma^{priv}(\mathcal{A}) = \{dur(\rho) \mid \rho \text{ is private and } \sigma\text{-compatible}\}$,
- $DVisit_\sigma^{pub}(\mathcal{A}) = \{dur(\rho) \mid \rho \text{ is public and } \sigma\text{-compatible}\}$.

Definition 7 (Full ET-opacity with strategy). *For a strategy σ , a TA \mathcal{A} is fully ET-opaque with σ when $DVisit_\sigma^{priv}(\mathcal{A}) = \overline{DVisit_\sigma^{pub}(\mathcal{A})}$.*

Finitely-varying strategies To be able to manage strategies in the following constructions, we expect them to behave in a “reasonable” way. We only consider *finitely-varying strategies* where the number of changes are finite for any closed time interval. Indeed, we can assume that a controller cannot change infinitely frequently its strategy in a finite time: it is unrealistic to consider, in a bounded interval, neither a system that can perform an infinite number of actions, nor a controller that can make an infinite number of choices. Finitely-varying strategies are reminiscent of non-Zeno behaviors, in the sense that finitely-varying strategies have a finite number of strategy changes in every bounded interval.

More formally, a strategy σ is *finitely-varying* when, for any closed time interval I , there is a finite partition p_1, \dots, p_n of I such that each p_i is an interval within which σ makes the same choice (i.e., for all $\tau_1, \tau_2 \in p_i$, $1 \leq i \leq n$, $\sigma(\tau_1) = \sigma(\tau_2)$). See Example 12 in Appendix G for an example of non-finitely-varying strategy.

In this paper, we are interested in several ET-opacity control problems i.e., related to a strategy σ making the TA \mathcal{A}^σ ET-opaque. Those problems differ on whether we want to prove the existence of a finitely-varying strategy enforcing the ET-opacity or synthesize this strategy.

Full ET-opacity finitely-varying controller emptiness problem:

INPUT: A TA \mathcal{A}

PROBLEM: Decide whether the set of finitely-varying strategies σ such that \mathcal{A}^σ is fully ET-opaque is empty.

Full ET-opacity finitely-varying controller synthesis problem:

INPUT: A TA \mathcal{A}

PROBLEM: Synthesize a finitely-varying strategy σ such that \mathcal{A}^σ is fully ET-opaque.

Example 4 (Non-ET-opaque TA). There is no strategy such that TA \mathcal{A}_1 in Fig. 1a is fully ET-opaque. Indeed, as $\Sigma_u = \{u\}$ and $\Sigma_c = \{a, b\}$, the transitions e_1 and e_2 are uncontrollable, so we can reach ℓ_f at any integer time along a run visiting ℓ_{priv} . However, even if we can allow transitions a and b at integer times too, the last transition e_3 is uncontrollable and can be taken at any time. Then, location ℓ_f can be reached at any time along a run not visiting ℓ_{priv} .

4 The belief automaton

In this section, we build an automaton called the “belief automaton”, that will allow us to determine in which regions the system can be after a given execution time. This automaton considers a duplicated TA instead of the original TA in order to distinguish the final state reached by a private or a public run.⁶

⁶ This could equally have been encoded using a Boolean variable remembering whether ℓ_{priv} was visited, as in [10].

4.1 Separating private and public runs

We define a duplicated version of a TA \mathcal{A} , denoted by \mathcal{A}^{dup} , making it possible to decide if a given run avoids ℓ_{priv} or not just looking at the final reached location. Note that any run of \mathcal{A} has an equivalent one in \mathcal{A}^{dup} where each location is replaced by its duplicated version if a previous visited location is ℓ_{priv} . In particular, $DVisit^{priv}(\mathcal{A}) = DVisit^{priv}(\mathcal{A}^{dup})$ and $DVisit^{priv}(\mathcal{A}) = DVisit^{priv}(\mathcal{A}^{dup})$.

Definition 8 (Duplicated TA). Let $\mathcal{A} = (\Sigma, L, \ell_0, \ell_{priv}, F, \mathbb{X}, I, E)$ be a TA. The associated duplicated TA is $\mathcal{A}^{dup} = (\Sigma', L', \ell'_0, \ell'_{priv}, F', \mathbb{X}', I', E')$ where: 1) $\Sigma' = \Sigma$, 2) $L' = L_{pub} \uplus L_{priv}$ with $L_{pub} = L \setminus \ell_{priv}$ and $L_{priv} = \{\ell^p \mid \ell \in L\} \cup \{\ell_{priv}\}$, 3) $\ell'_0 = \ell_0$, 4) $\ell'_{priv} = \ell_{priv}$, 5) $F' = \{\ell_f^p \mid \ell_f \in F\} \cup F$, 6) $\mathbb{X}' = \mathbb{X}$, 7) I' is the invariant such that $\forall \ell \in L, I'(\ell) = I'(\ell^p) = I(\ell)$, 8) $E' = \{(\ell_1, g, a, R, \ell_2) \mid (\ell_1, g, a, R, \ell_2) \in E \text{ and } \ell_1 \neq \ell_{priv}\} \cup \{(\ell_1^p, g, a, R, \ell_2^p) \mid (\ell_1, g, a, R, \ell_2) \in E\} \cup \{(\ell_{priv}, g, a, R, \ell^p) \mid (\ell_{priv}, g, a, R, \ell) \in E\}$

Example 5. Fig. 1b depicts \mathcal{A}_1^{dup} , the duplicated version of \mathcal{A}_1 in Fig. 1a. We do not depict unreachable locations. Observe that each run avoiding ℓ_{priv} ends in ℓ_f , and that the only outgoing transition of ℓ_{priv} is modified to go to ℓ_f^p .

4.2 Beliefs

A *belief*⁷, denoted by \mathbf{b} , represents the set of regions in which the attacker believes to be, according to their observations: elapsed time and enabled actions.

We write $\mathcal{C} \subseteq \Sigma_c$ for a set of enabled actions, and define as follows the belief automaton, constructing the belief the attacker has, according to elapsed time and enabled actions.

Definition 9 (Belief automaton). For \mathcal{A} a TA with $\Sigma = \Sigma_c \uplus \Sigma_u$, we define the belief automaton as the tuple $\mathcal{B}_{\mathcal{A}} = (\mathfrak{S}^{\mathcal{B}_{\mathcal{A}}}, \mathfrak{Q}^{\mathcal{B}_{\mathcal{A}}}, \perp, \mathfrak{D}^{\mathcal{B}_{\mathcal{A}}})$ where:

1. $\mathfrak{S}^{\mathcal{B}_{\mathcal{A}}} = 2^{\mathcal{R}_{\mathcal{A}^{dup}}} \cup \{\perp\}$ is the set of states,
2. $\mathfrak{Q}^{\mathcal{B}_{\mathcal{A}}} = \{0, 0^+, 1\} \times 2^{\Sigma_c}$,
3. \perp is the initial state,
4. $\mathfrak{D}^{\mathcal{B}_{\mathcal{A}}} \subseteq (\mathfrak{S}^{\mathcal{B}_{\mathcal{A}}} \times \mathfrak{Q}^{\mathcal{B}_{\mathcal{A}}} \times \mathfrak{S}^{\mathcal{B}_{\mathcal{A}}})$ and
 - (a) $(\perp, (0, \mathcal{C}), \mathbf{b}) \in \mathfrak{D}^{\mathcal{B}_{\mathcal{A}}}$ iff \mathbf{b} is the largest set such that $\forall r \in \mathbf{b}, \exists n \geq 0$, $[s_0] \xrightarrow{(0, a_1)}_{\mathcal{R}} \dots \xrightarrow{(0, a_n)}_{\mathcal{R}} r$ in $\mathcal{R}_{\mathcal{A}^{dup}}$ with $\forall 1 \leq i \leq n, a_i \in (\mathcal{C} \cup \Sigma_u)$,
 - (b) $(\mathbf{b}, (\dagger_1, \mathcal{C}), \mathbf{b}') \in \mathfrak{D}^{\mathcal{B}_{\mathcal{A}}}$ iff $\mathbf{b} \neq \perp$, \mathbf{b}' is the largest set such that $\forall r' \in \mathbf{b}'$, $\exists r \in \mathbf{b}, \exists n \geq 1, r \xrightarrow{(\dagger_1, \varepsilon)}_{\mathcal{R}} \dots \xrightarrow{(\dagger_n, a_n)}_{\mathcal{R}} r'$ in $\mathcal{R}_{\mathcal{A}^{dup}}$ with $\forall 1 < i \leq n, a_i \in (\mathcal{C} \cup \Sigma_u \cup \{\varepsilon\})$ and $\dagger_1 \in \{0^+, 1\}$ and $\forall 1 < i \leq n, \dagger_i \in \{0, 0^+\}$.

We first consider transitions from the initial belief \perp : time cannot elapse here; one can do a sequence of actions allowed at time 0 (condition 4a). Then, from the other beliefs, a transition is made of a sequence of transitions from the region automaton, the first one of which lets time elapse, and all the following actions need to be allowed at that time (condition 4b).

⁷ We follow the vocabulary from, e.g., [16]. This is also close to the concept of *estimator* (e.g., [25]).

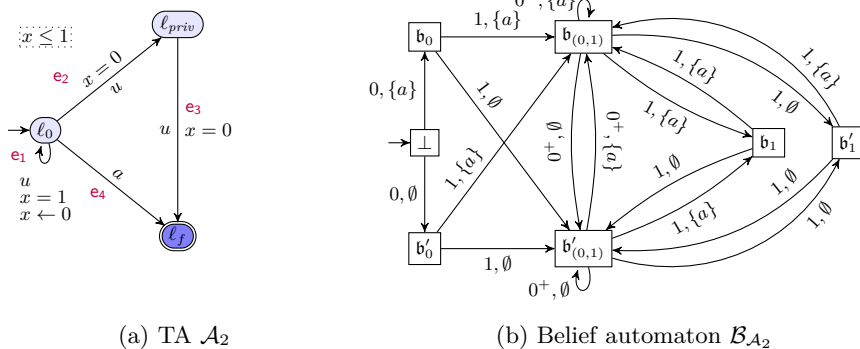


Fig. 2: \mathcal{A}_2 and the corresponding belief automaton

Example 6. In this example and the following ones, we associate each belief either to an open interval or to an integer, and, by abuse of notation, since the automaton has only one clock, we write $(\ell, (\tau, \tau'))$ for the region containing the state $(\ell, \mu(x_1))$ with $\mu(x_1) \in (\tau, \tau'), \tau \in \mathbb{N}, \tau' = +\infty$ if $\tau = c_1, \tau' = \tau + 1$ otherwise. Similarly, we write (ℓ, τ) for the region containing the state $(\ell, \mu(x_1)), \mu(x_1) = \tau \in \mathbb{N}$.

Let \mathcal{A}_2 be the TA in Fig. 2a. With the global invariant $x \leq 1$, we have the following beliefs. Here, the value of clock z is not given as, in this example, it is equivalent to the value of x . (See Example 11 in Appendix G for an example with z .) The corresponding belief automaton is depicted in Fig. 2b.

$$\begin{aligned}
 \mathbf{b}'_0 &= \{(\ell_0, 0), (\ell_{priv}, 0), (\ell_f^p, 0)\} & \mathbf{b}_0 &= \mathbf{b}'_0 \cup \{(\ell_f, 0)\} \\
 \mathbf{b}'_{(0,1)} &= \{(\ell_0, (0, 1)), (\ell_{priv}, (0, 1))\} & \mathbf{b}_{(0,1)} &= \mathbf{b}'_{(0,1)} \cup \{(\ell_f, (0, 1))\} \\
 \mathbf{b}'_1 &= \{(\ell_0, 1), (\ell_0, 0), (\ell_{priv}, 0), (\ell_f^p, 0)\} & \mathbf{b}_1 &= \mathbf{b}'_1 \cup \{(\ell_f, 0)\}
 \end{aligned}$$

Bad beliefs Intuitively, a bad belief allows to discriminate private and public runs. For a given TA \mathcal{A} , we denote $\text{Secret}_{\mathcal{A}} = \{[(\ell, \mu)] \mid \ell \in L_{priv}, \mu \in \mathbb{R}_{\geq 0}^H\}$ the set of regions reachable on a run visiting ℓ_{priv} in \mathcal{A}^{dup} , and $\overline{\text{Secret}}_{\mathcal{A}} = \{[(\ell, \mu)] \mid \ell \in L_{pub}, \mu \in \mathbb{R}_{\geq 0}^H\}$ the set of regions reachable on a run not visiting ℓ_{priv} in \mathcal{A}^{dup} .

For a given TA \mathcal{A} , a belief \mathbf{b} is said to be *bad for full ET-opacity* when exactly one of the following two conditions is satisfied: 1) $(\mathbf{b} \cap \mathbf{R}_{\mathcal{A}}^F \cap \text{Secret}_{\mathcal{A}} \neq \emptyset)$, or 2) $(\mathbf{b} \cap \mathbf{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}} \neq \emptyset)$. This means that finishing in this belief leaks an information to the attacker: only one final state is possible (private or public, but not both).

Beliefs and strategy For a TA \mathcal{A} and a strategy σ , we denote by \mathbf{b}_t^σ the set of regions in which we can be after a time t while following a strategy σ in \mathcal{A} , i.e., $r \in \mathbf{b}_t^\sigma$ when there exists ρ in \mathcal{A}^{dup} such that ρ is σ -compatible, $\text{last}(\rho) \in r, r \in \mathbf{R}_{\mathcal{A}^{dup}}$ and $\text{dur}(\rho) = t$.

We define \mathbb{B}_A^σ as the set of beliefs reachable by a strategy σ . Formally, for a given strategy σ : $\mathbb{B}_A^\sigma = \{\mathbf{b}_t^\sigma \mid t \in \mathbb{R}_{\geq 0}\}$.

We can first characterize the ET-opacity of a controlled TA with beliefs. The proof of this lemma can be found in [Appendix B.1](#).

Lemma 1 (Beliefs characterization for full ET-opacity). *A TA \mathcal{A} is fully ET-opaque with a strategy σ iff, for all $\mathbf{b} \in \mathbb{B}_A^\sigma$, \mathbf{b} is not bad for full ET-opacity.*

A strategy for a belief automaton, called a **\mathbf{b} -strategy**, is a function which associates to a sequence of actions the next available action. Formally, a **\mathbf{b} -strategy** is a function $\alpha : (\mathfrak{A}^{\mathcal{B}_A})^* \rightarrow \mathfrak{A}^{\mathcal{B}_A}$. A *state* in a belief automaton controlled by a **\mathbf{b} -strategy** is made of the sequence of actions performed until that state is reached, and the current belief.

Definition 10 (Controlled belief automaton). *For a belief automaton \mathcal{B}_A and a \mathbf{b} -strategy α , we define $\mathcal{B}_A^\alpha = (\mathfrak{S}^{\mathcal{B}_A^\alpha}, \mathfrak{A}^{\mathcal{B}_A^\alpha}, (\varepsilon, \perp), \mathfrak{D}^{\mathcal{B}_A^\alpha})$ the belief automaton controlled by α as follows:*

1. $\mathfrak{S}^{\mathcal{B}_A^\alpha} = (\mathfrak{A}^{\mathcal{B}_A})^* \times \mathfrak{S}^{\mathcal{B}_A}$ is the set of states,
2. $\mathfrak{A}^{\mathcal{B}_A^\alpha} = \mathfrak{A}^{\mathcal{B}_A} = \{0, 0^+, 1\} \times 2^{\Sigma_c}$ is the alphabet,
3. (ε, \perp) is the initial state,
4. $\mathfrak{D}^{\mathcal{B}_A^\alpha} \subseteq (\mathfrak{S}^{\mathcal{B}_A^\alpha} \times \mathfrak{A}^{\mathcal{B}_A^\alpha} \times \mathfrak{S}^{\mathcal{B}_A^\alpha})$ and $((v, \mathbf{b}), (\dagger, \mathfrak{C}), (v \cdot (\dagger, \mathfrak{C}), \mathbf{b}')) \in \mathfrak{D}^{\mathcal{B}_A^\alpha}$ if $(\mathbf{b}, (\dagger, \mathfrak{C}), \mathbf{b}') \in \mathfrak{D}^{\mathcal{B}_A}$, and $\alpha(v) = (\dagger, \mathfrak{C})$ with $\dagger \in \{0, 0^+, 1\}$.

We define a *finitely-varying \mathbf{b} -strategy* α in such a way that it corresponds to a finitely-varying strategy for a TA: for all $k \in \mathbb{N}$ such that $\alpha^{(k)}(\varepsilon) = (0^+, \mathfrak{C})$ for some \mathfrak{C} , there exists a $k' > k \in \mathbb{N}$ such that $\alpha^{(k')}(\varepsilon) = (1, \mathfrak{C}')$ for some \mathfrak{C}' , with $\alpha^{(i)}(\varepsilon) = \alpha(\dots(\alpha(\varepsilon)))$ where α is applied i times.

In order to define the correspondence between a strategy for a TA and a **\mathbf{b} -strategy** for the corresponding belief automaton, we need to associate an interval of time $t(v)$ to a sequence of transitions $v = (\dagger_1, \mathfrak{C}_1) \cdots (\dagger_n, \mathfrak{C}_n)$, knowing that two transitions of the form $(1, \mathfrak{C})$ are necessary to go from an integer time to another one. Let $\#\dagger(v)$ be the number of indices $i, 0 < i \leq n$, such that $\dagger_i = 1$; then

$$t(v) = \begin{cases} \{\#\dagger(v)/2\} & \text{if } \#\dagger(v) \equiv 0 \pmod{2} \\ (\lfloor \#\dagger(v)/2 \rfloor, \lfloor \#\dagger(v)/2 \rfloor + 1) & \text{otherwise.} \end{cases}$$

Example 7. Let $v_1 = (0, \mathfrak{C}_0) \cdot (1, \mathfrak{C}_1) \cdot (0^+, \mathfrak{C}_2) \cdot (1, \mathfrak{C}_3) \cdot (1, \mathfrak{C}_4)$ and $v_2 = v_1 \cdot (1, \mathfrak{C}_5)$. By definition, $t(v_1) = (1, 2)$ and $t(v_2) = \{2\}$. This means that for all runs ρ_1 and ρ_2 corresponding respectively to v_1 and v_2 , we have $\text{dur}(\rho_1) \in (1, 2)$ and $\text{dur}(\rho_2) = 2$.

Definition 11 (Strategy correspondence). *Let \mathcal{A} be a TA. We say that σ , a finitely-varying strategy of \mathcal{A} , corresponds to α , a finitely-varying \mathbf{b} -strategy of \mathcal{B}_A , denoted by $\sigma \vdash \alpha$, when:*

1. $\alpha(\varepsilon) = (0, \sigma(0))$,
2. for $v \in (\mathfrak{A}^{\mathcal{B}_A})^+$,

- if $v = v' \cdot (1, \mathfrak{C}_0) \cdot (0^+, \mathfrak{C}_1) \cdot \dots \cdot (0^+, \mathfrak{C}_n)$, $n \geq 0$, and there exist $\tau_0 < \dots < \tau_{n+1}$ such that for all $0 \leq i \leq n+1$, $\tau_i \in t(v)$, $\sigma(\tau_i) = \mathfrak{C}_i$, for all $0 \leq i \leq n$, $\mathfrak{C}_i \neq \mathfrak{C}_{i+1}$ and there is no τ'_i , $\tau_i < \tau'_i < \tau_{i+1}$, such that $\sigma(\tau_i) \neq \sigma(\tau'_i) \neq \sigma(\tau_{i+1})$, then $\alpha(v) = (0^+, \mathfrak{C}_{n+1})$,
- otherwise, $\alpha(v) = (1, \mathfrak{C})$ for $\tau \in t(v \cdot (1, \mathfrak{C}))$, $\sigma(\tau) = \mathfrak{C}$ and there is no $\tau' < \tau$ such that $\tau' \in t(v \cdot (1, \mathfrak{C}))$ and $\sigma(\tau') \neq \sigma(\tau)$.

In other words, within a time interval of the form $(t, t+1)$ with $t \in \mathbb{N}$, the \mathfrak{b} -strategy α does not keep track of the precise timestamps of strategy changes in σ but makes the same changes in the same order. A belief \mathfrak{b} is *reachable in* $\mathcal{B}_{\mathcal{A}}^\alpha$ when there exists a word $v \in (\mathfrak{A}^{\mathcal{B}_{\mathcal{A}}})^*$ such that the state (v, \mathfrak{b}) is reachable.

We prove in the following that a strategy in the TA matches a \mathfrak{b} -strategy in the belief automaton as do the runs and sequences of actions in belief automaton.

Lemma 2 (Strategy correspondence). *Let \mathcal{A} be a TA. Given a finitely-varying strategy σ of \mathcal{A} , there is a finitely-varying \mathfrak{b} -strategy α of $\mathcal{B}_{\mathcal{A}}$ such that $\sigma \Vdash \alpha$. Reciprocally, given a finitely-varying \mathfrak{b} -strategy α of $\mathcal{B}_{\mathcal{A}}$, there is a finitely-varying strategy σ of \mathcal{A} such that $\sigma \Vdash \alpha$.*

Proof. Let σ be a finitely-varying strategy for \mathcal{A} . For any interval $[i, i+1)$, $i \in \mathbb{N}$, let $p_{i_0}, p_{i_1}, \dots, p_{i_{n_i}}$ be the smallest finite partition into intervals such that $p_{i_0} = \{i\}$ and σ makes the same choice within each p_{i_j} . We build the \mathfrak{b} -strategy α :

- $\alpha(\varepsilon) = (0, \sigma(0))$,
- for all $v \in (\mathfrak{A}^{\mathcal{B}_{\mathcal{A}}})^+$, with $|v|$ denoting the length of the word v ,
 - if $t(v) = \{\iota\}$, $\iota \in \mathbb{N}$, is a singleton then by assumption, $p_{\iota_0} = t(v)$ and we set $\alpha(v) = (1, \sigma(\tau))$ for some $\tau \in p_{\iota_1}$,
 - otherwise, $v = v' \cdot (1, \mathfrak{C}_1) \cdot v''$ such that $v'' = (0^+, \mathfrak{C}_2) \cdot \dots \cdot (0^+, \mathfrak{C}_k)$ and as σ is finitely-varying, there exist ι, n_ι such that $\bigcup_{j=1}^{n_\iota} p_{\iota_j} = t(v)$ and, as we took the smallest partition, we know that for all $1 \leq j < n$, for any $\tau \in p_{\iota_j}$, $\tau' \in p_{\iota_{j+1}}$, $\sigma(\tau) \neq \sigma(\tau')$. Then,

$$\alpha(v) = \begin{cases} (0^+, \sigma(\tau)), & \text{for some } \tau \in p_{\iota_{2+|v''|}} & \text{if } |v''| < n_\iota - 1 \\ (1, \sigma(\tau)), & \text{for some } \tau \in p_{\iota'_0}, \iota'_0 = \iota + 1 & \text{otherwise.} \end{cases}$$

By construction, α is finitely-varying and $\sigma \Vdash \alpha$.

Conversely, let α be a finitely-varying \mathfrak{b} -strategy, let us build σ such that $\sigma \Vdash \alpha$. Let $\tau \in \mathbb{R}_{\geq 0}$ be a time instant and $v = (\dagger_0, \mathfrak{C}_0) \cdot \dots \cdot (\dagger_k, \mathfrak{C}_k) = (\alpha)^{(1)}(\varepsilon) \cdot \dots \cdot (\alpha)^{(k)}(\varepsilon)$ be the smallest sequence such that $t(v) = \{\iota\}$ is a singleton and $\iota \geq \tau$. As the strategy is finitely-varying this sequence is finite, and we have $\dagger_k = 1$, then:

- either τ is an integer, then we fix $\sigma(\tau) = \mathfrak{C}_k$;
- otherwise, let $j < k$ be the greatest integer such that $\dagger_j = 1$. Consider then the sequence $(\alpha)^{(j)}(\varepsilon) \cdot \dots \cdot (\alpha)^{(k)}(\varepsilon) = (1, \mathfrak{C}_j) \cdot (0^+, \mathfrak{C}_{j+1}) \cdot \dots \cdot (0^+, \mathfrak{C}_{k-1}) \cdot (1, \mathfrak{C}_k)$ and denote $m = k - j$, we define $\sigma(\tau) = \mathfrak{C}_i$ where $i = j + \lfloor \text{fr}(\tau) * m \rfloor$.

By construction, σ is finitely-varying and $\sigma \Vdash \alpha$. □

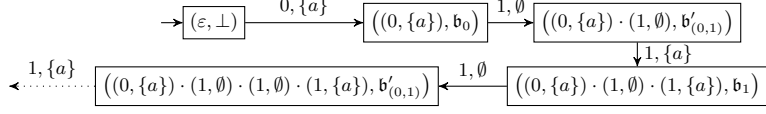


Fig. 3: $\mathcal{B}_{\mathcal{A}_2}^\alpha$: First states of the controlled belief automaton for TA \mathcal{A}_2 and \mathfrak{b} -strategy α

We define a run that follows a \mathfrak{b} -strategy as follows.

Definition 12 (Feasible run). Let \mathcal{A} be a TA, ρ be a run of \mathcal{A}^{dup} and $v \in (\mathfrak{A}^{\mathcal{B}\mathcal{A}})^*$. We say that ρ admits v , denoted by $\rho \models v$, when one of the following holds:

1. $\rho = (\ell_0, \mathbf{0})$ and $v = (0, \mathfrak{C}_0)$ with $\mathfrak{C}_0 \subseteq \Sigma_c$,
2. $\rho = \rho', (d_{n-1}, \mathfrak{e}_{n-1}), (\ell_n, \mu_n)$ with $\mathfrak{e}_{n-1} = (\ell_{n-1}, g, a, R, \ell_n)$ and one of the following holds:
 - (a) $d_{n-1} = 0$, $\rho' \models v$, $v = v' \cdot (\dagger, \mathfrak{C})$ and $a \in \mathfrak{C} \cup \Sigma_u \cup \{\varepsilon\}$,
 - (b) $0 < d_{n-1} < 1$, $v = v' \cdot (1, \mathfrak{C}_0) \cdot (0^+, \mathfrak{C}_1) \cdots (0^+, \mathfrak{C}_{m-1}) (\dagger, \mathfrak{C}_m)$, $a \in \mathfrak{C}_m \cup \Sigma_u \cup \{\varepsilon\}$, there exists i , $0 \leq i \leq m$, $\rho' \models v' \cdot (1, \mathfrak{C}_0) \cdot (0^+, \mathfrak{C}_1) \cdots (\dagger, \mathfrak{C}_i)$, $\dagger = 0^+$ if $i < m$ and $\dagger = \dagger$ otherwise, and either
 - $\text{fr}(\mu_{n-1}(z)) \neq 0$, $\text{fr}(\mu_n(z)) \neq 0$, and $\dagger = 0^+$, or
 - $\text{fr}(\mu_{n-1}(z)) = 0$ or $\text{fr}(\mu_n(z)) = 0$ and $\dagger = 1$,
 - (c) $d_{n-1} = 1$, $a \in \mathfrak{C} \cup \Sigma_u \cup \{\varepsilon\}$, $v = v' \cdot (1, \mathfrak{C}_0) \cdot (0^+, \mathfrak{C}_1) \cdots (0^+, \mathfrak{C}_m) \cdot (1, \mathfrak{C})$ with $0 \leq m$ and such that $\rho' \models v'$.

Given a \mathfrak{b} -strategy α , ρ is feasible in $\mathcal{B}_{\mathcal{A}}^\alpha$ when there exist $v \in (\mathfrak{A}^{\mathcal{B}\mathcal{A}})^*$ and a belief $\mathfrak{b} \in \mathfrak{B}^{\mathcal{B}\mathcal{A}}$ such that $\rho \models v$ and (v, \mathfrak{b}) is reachable in $\mathcal{B}_{\mathcal{A}}^\alpha$.

Example 8. Let \mathcal{A}_2 be the TA in Fig. 2. Let σ be a strategy such that \mathcal{A}_2 is fully ET-opaque defined as follows:

$$\sigma(\tau) = \begin{cases} \{a\} & \text{if } \tau \in \mathbb{N} \\ \emptyset & \text{otherwise} \end{cases}$$

Let α be a \mathfrak{b} -strategy for $\mathcal{B}_{\mathcal{A}_2}$ such that $\sigma \vdash \alpha$:

$$\alpha(v) = \begin{cases} (0, \{a\}) & \text{if } v = \varepsilon \\ (1, \emptyset) & \text{if } v \in (0, \{a\}) \cdot ((1, \emptyset) \cdot (1, \{a\}))^* \\ (1, \{a\}) & \text{if } v \in (0, \{a\}) \cdot (1, \emptyset) \cdot ((1, \{a\}) \cdot (1, \emptyset))^* \end{cases}$$

Then, the first states of the automaton $\mathcal{B}_{\mathcal{A}_2}^\alpha$ are depicted in Fig. 3.

Runs ρ_1 and ρ_2 are feasible in $\mathcal{B}_{\mathcal{A}_2}^\alpha$:

$$\begin{aligned} \rho_1 &= (\ell_0, 0), (1, \mathfrak{e}_1), (\ell_0, 0), (0, \mathfrak{e}_2), (\ell_{priv}, 0), (0, \mathfrak{e}_3), (\ell_f^p, 0) \\ \rho_2 &= (\ell_0, 0), (1, \mathfrak{e}_1), (\ell_0, 0), (0, \mathfrak{e}_4), (\ell_f, 0) \end{aligned}$$

For $v = (0, \{a\}) \cdot (1, \emptyset) \cdot (1, \{a\})$ a sequence of actions in the automaton $\mathcal{B}_{\mathcal{A}_2}^\alpha$, we have $\rho_1 \models v$ and $\rho_2 \models v$.

We can now link a strategy and a \mathbf{b} -strategy, from the point of view of runs. The proof of the following lemma can be found in [Appendix B.2](#).

Lemma 3 (Strategies and runs acceptance). *Let \mathcal{A} be a TA, ρ a run of \mathcal{A}^{dup} , σ a finitely-varying strategy and α a finitely-varying \mathbf{b} -strategy such that $\sigma \vdash \alpha$. We have: ρ is σ -compatible iff ρ is feasible in $\mathcal{B}_{\mathcal{A}}^{\alpha}$.*

5 Solving ET-opacity problems through the belief automaton

Finally, we can prove that reasoning on the TA is equivalent to reasoning on the belief automaton.

Theorem 1. *Let \mathcal{A} be a TA. There exists a finitely-varying strategy σ such that \mathcal{A} is fully ET-opaque with σ iff there exists a finitely-varying \mathbf{b} -strategy α such that $\sigma \vdash \alpha$ and there is no reachable belief in $\mathcal{B}_{\mathcal{A}}^{\alpha}$ that is bad for full ET-opacity.*

Proof. Let \mathcal{A} be a TA and σ a finitely-varying strategy such that \mathcal{A} is fully ET-opaque with σ . By [Lemma 2](#), the existence of the finitely-varying strategy σ is equivalent to the existence of α a finitely-varying \mathbf{b} -strategy with $\sigma \vdash \alpha$. By [Lemma 3](#), we know that it is equivalent for a run ρ to be σ -compatible or feasible in $\mathcal{B}_{\mathcal{A}}^{\alpha}$. By [Definition 12](#), this means that there is a sequence v of actions of $\mathcal{B}_{\mathcal{A}}^{\alpha}$ such that $\rho \models v$ and there is a belief $\mathbf{b} = \mathbf{b}_{dur(\rho)}^{\sigma} \in \mathbb{B}_{\mathcal{A}}^{\sigma}$ such that $(v \cdot \mathbf{b})$ is reachable in $\mathcal{B}_{\mathcal{A}}^{\alpha}$. This results that the set of reachable beliefs in $\mathcal{B}_{\mathcal{A}}^{\alpha}$ is equal to the set $\mathbb{B}_{\mathcal{A}}^{\sigma}$. Finally, by [Lemma 1](#), there is no belief bad for full ET-opacity reachable in $\mathbb{B}_{\mathcal{A}}^{\sigma}$, and therefore, as the two sets are equal, neither in $\mathcal{B}_{\mathcal{A}}^{\alpha}$. \square

Intuitively, finding a \mathbf{b} -strategy which avoids bad beliefs within $\mathcal{B}_{\mathcal{A}}$ amounts to solving a one-player safety game on a finite arena where the bad beliefs are the states to avoid. However, this approach would not necessarily produce a finitely-varying strategy. We will therefore rely on one-player Büchi games instead. More precisely, a one-player Büchi game can be defined by a tuple $\mathcal{G} = (Q, q_0, \Sigma, \delta^{\mathcal{G}}, G)$ where Q is a set of states, $q_0 \in Q$ is the initial state, Σ is a set of actions, $\delta^{\mathcal{G}} \subseteq Q \times \Sigma \times Q$ describes the transitions, and $G \subseteq \Sigma$ is a set of “good” actions. Starting from q_0 , at each step, the player selects a transition from $\delta^{\mathcal{G}}$ to reach a new state. The player wins if transitions labeled by actions from G are taken infinitely often (note that Büchi games usually require that a set of “good” states is visited infinitely often instead of actions, but both frameworks are trivially equivalent).

Lemma 4 ([28]). *Deciding the existence of a winning strategy in a one-player Büchi game can be done in NLOGSPACE. Moreover this strategy, if it exists, can be constructed in polynomial time.*

As shown in [\[9\]](#), the full ET-opacity problem for timed automata (without control) is decidable in NEXPTIME. The ability to control the system slightly increases the complexity:

Theorem 2. *The full ET-opacity finitely-varying controller emptiness problem is decidable in EXPSPACE; and the full ET-opacity finitely-varying controller synthesis problem is solvable in 2-EXPTIME.*

See Appendix C.1 for the proof of this result.

6 Extension: robust definitions of ET-opacity

So far, the attacker needed to measure the execution time with an infinite precision—this is often unrealistic in practice [22,18]. We therefore consider variants of opacity where intervals of non-opaque execution times can be considered acceptable as long as they are of size 0, i.e., reduced to a point (note that there can be an infinite number of them).

In order to formally define these two new notions, we introduce new notations: given a set S , then let $\llbracket S \rrbracket$ denote the *closure* of S (i.e., the smallest closed set containing S) and let $\langle S \rangle$ denote the *interior* of S (i.e., the largest open set contained in S). Let \oplus denotes the exclusive OR operator such that, for two sets A and B , $A \oplus B = \{v \mid v \in (A \cup B) \setminus (A \cap B)\}$.

Let us first define *almost full ET-opacity*, where every punctual opacity violation is ignored. That is, a TA is almost fully ET-opaque whenever all non-opaque durations are isolated from each other; that is, since these non-opaque durations must be punctual, then the interior of intervals of non-opaque durations must be empty.

Definition 13 (Almost full ET-opacity). *A TA \mathcal{A} is almost fully ET-opaque when $\langle DVisit^{priv}(\mathcal{A}) \oplus DVisit^{priv}(\mathcal{A}) \rangle = \emptyset$.*

Let us now define *closed full ET-opacity*, where a punctual violation is ignored only if it is followed or preceded by an opaque interval. That is, we say a TA is closed fully ET-opaque when the closure of the private durations equals the closure of the public durations.

Definition 14 (Closed full ET-opacity). *A TA \mathcal{A} is closed fully ET-opaque when $\llbracket DVisit^{priv}(\mathcal{A}) \rrbracket = \llbracket DVisit^{priv}(\mathcal{A}) \rrbracket$.*

A difference between both definitions is, for example, whenever a non-opaque duration is such that the immediately neighboring durations do not correspond to any accepting run (neither public nor private). In that case, this non-opaque

duration will be left out by almost full ET-opacity, but will still be considered non-opaque by closed full ET-opacity.

Example 9. Let \mathcal{A}_3 be a TA such that $DVisit^{priv}(\mathcal{A}_3) = (0, 1) \cup \{2\}$ and $DVisit^{pub}(\mathcal{A}_3) = (0, 1)$. \mathcal{A}_3 is not fully ET-opaque. $(DVisit^{priv}(\mathcal{A}_3) \oplus DVisit^{pub}(\mathcal{A}_3)) = \emptyset$ so \mathcal{A}_3 is almost fully ET-opaque. $\llbracket DVisit^{priv}(\mathcal{A}_3) \rrbracket = [0, 1] \cup \{2\} \neq \llbracket DVisit^{pub}(\mathcal{A}_3) \rrbracket = [0, 1]$ so \mathcal{A}_3 is not closed fully ET-opaque.

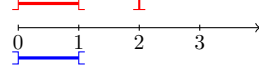


Fig. 4: Representation of private (above in red) and public (below in blue) durations in \mathcal{A}_3

We are interested in the same problems as before. In particular, our aim is to build a controller to make a TA closed ET-opaque with a finitely-varying strategy. Reasoning and results similar to the following are presented for almost full ET-opacity in [Appendices D and E.2](#).

Characterization A single belief is not sufficient to characterize a TA that is not almost ET-opaque (resp. closed). Indeed, suppose a time t such that \mathbf{b}_t is bad for full ET-opacity. This means that a punctual violation of opacity exists. This kind of violation can be allowed in the context of almost and closed full ET-opacity. It is problematic if the times around it are also a violation of opacity.

More specifically, a violation to almost full ET-opacity corresponds to a succession of bad beliefs, i.e., every punctual violation is ignored. On the other hand, a violation to closed full ET-opacity corresponds either to a succession of bad beliefs, or to a unique bad belief surrounded by beliefs that do not contain any final region. Intuitively, a punctual violation is ignored if it belongs to an interval where private and public final states can be reached.

The formal definitions of bad belief for almost ET-opacity ([Definition 15](#)) and closed ET-opacity ([Definition 16](#)) can be found in [Appendix D](#).

Theorem 3. *The closed full ET-opacity finitely-varying controller emptiness problem is decidable; the closed full ET-opacity finitely-varying controller synthesis problem is solvable.*

As for full ET-opacity, this result is due to the equivalence between finding a strategy for a TA and finding a strategy in the corresponding belief automaton, and to the fact that such a strategy corresponds to a winning strategy in a one-player safety game. Intermediate results and proofs are available in [Appendix E.1](#) while results for almost full ET-opacity are presented in [Appendix E.2](#).

7 Conclusion

We addressed here the control of a system modeled by a TA to make it fully ET-opaque. We showed that the controller synthesis problem can be effectively solvable for these two definitions. We also addressed two extensions (closed full

ET-opacity and almost full ET-opacity) which can relate to a *robust* setting where the attacker cannot have an infinite precision.

In addition, we studied two other versions of opacity [9], namely weak ET-opacity (in which it is harmless that the attacker deduces that the private location was *not* visited) and \exists -ET-opacity (in which we are simply interested in the *existence* of one execution time for which opacity is ensured); due to space constraints, and to simplify the exposé, these extensions are omitted, but can be found in Appendix F.

Future works A natural next step will be to introduce timing parameters *à la* [9], and address control in that setting.

Addressing the control for the definition of opacity (based on languages) as in [20] would be interesting in two settings: 1) the general setting, where the controller synthesis will be undecidable but may terminate for some semi-algorithms, and 2) decidable subclasses that remain to be exhibited, presumably one-clock TAs.

Finally, the implementation of this work is on our agenda. While implementing the beliefs directly would be straightforward, it would probably result in an unnecessary blowup, and therefore an adaptation with structures such as zones (which does not seem immediate) should be designed.

References

- [1] Alur, R., Dill, D.L.: A theory of timed automata. *TCS* **126**(2), 183–235 (1994). [https://doi.org/10.1016/0304-3975\(94\)90010-8](https://doi.org/10.1016/0304-3975(94)90010-8)
- [2] Alur, R., Fix, L., Henzinger, T.A.: Event-clock automata: A determinizable class of timed automata. *TCS* **211**(1-2), 253–273 (1999). [https://doi.org/10.1016/S0304-3975\(97\)00173-4](https://doi.org/10.1016/S0304-3975(97)00173-4)
- [3] Alur, R., Henzinger, T.A., Vardi, M.Y.: Parametric real-time reasoning. In: *STOC*. pp. 592–601. ACM (1993). <https://doi.org/10.1145/167088.167242>
- [4] Ammar, I., El Touati, Y., Yeddes, M., Mullins, J.: Bounded opacity for timed systems. *Journal of Information Security and Applications* **61**, 1–13 (2021). <https://doi.org/10.1016/j.jisa.2021.102926>
- [5] An, J., Gao, Q., Wang, L., Zhan, N., Hasuo, I.: The opacity of timed automata. In: Platzer, A., Rozier, K.Y., Pradella, M., Rossi, M. (eds.) *FM*. LNCS, vol. 14933, pp. 620–637. Springer (2024). https://doi.org/10.1007/978-3-031-71162-6_32
- [6] André, É., Bolat, S., Lefauchaux, E., Marinho, D.: strategFTO: Untimed control for timed opacity. In: *FTSCS*. pp. 27–33. ACM (2022). <https://doi.org/10.1145/3563822.3568013>
- [7] André, É., Dépernet, S., Lefauchaux, E.: The bright side of timed opacity. In: Ogata, K., Sun, M., Méry, D. (eds.) *ICFEM*. LNCS, vol. 15394, pp. 51–69. Springer (Dec 2024). https://doi.org/10.1007/978-981-96-0617-7_4
- [8] André, É., Kryukov, A.: Parametric non-interference in timed automata. In: *ICECCS*. pp. 37–42 (2020). <https://doi.org/10.1109/ICECCS51672.2020.00012>
- [9] André, É., Lefauchaux, E., Lime, D., Marinho, D., Sun, J.: Configuring timing parameters to ensure execution-time opacity in timed automata. In: *TiCSA*. *Electronic Proceedings in Theoretical Computer Science*, vol. 392, pp. 1–26 (2023). <https://doi.org/10.4204/EPTCS.392.1>, invited paper.
- [10] André, É., Lime, D., Marinho, D., Sun, J.: Guaranteeing timed opacity using parametric timed model checking. *ACM Transactions on Software Engineering and Methodology* **31**(4), 1–36 (2022). <https://doi.org/10.1145/3502851>
- [11] Arcile, J., André, É.: Timed automata as a formalism for expressing security: A survey on theory and practice. *ACM Computing Surveys* **55**(6), 1–36 (2023). <https://doi.org/10.1145/3534967>
- [12] Asarin, E., Maler, O., Pnueli, A., Sifakis, J.: Controller synthesis for timed automata. *IFAC Proceedings Volumes* **31**(18), 447–452 (1998). [https://doi.org/10.1016/S1474-6670\(17\)42032-5](https://doi.org/10.1016/S1474-6670(17)42032-5), proceedings of the 5th IFAC Conference on System Structure and Control (SSC 1998)
- [13] Barbuti, R., Francesco, N.D., Santone, A., Tesei, L.: A notion of non-interference for timed automata. *FI* **51**(1-2), 1–11 (2002)
- [14] Barbuti, R., Tesei, L.: A decidable notion of timed non-interference. *FI* **54**(2-3), 137–150 (2003)
- [15] Benattar, G., Cassez, F., Lime, D., Roux, O.H.: Control and synthesis of non-interferent timed systems. *International Journal of Control* **88**(2), 217–236 (2015). <https://doi.org/10.1080/00207179.2014.944356>
- [16] Bertrand, N., Fabre, E., Haar, S., Haddad, S., Hélouët, L.: Active diagnosis for probabilistic systems. In: Muscholl, A. (ed.) *FoSSaCS*. LNCS, vol. 8412, pp. 29–42. Springer (2014). https://doi.org/10.1007/978-3-642-54830-7_2
- [17] Bouyer, P., Fang, E., Markey, N.: Permissive strategies in timed automata and games. *Electronic Communication of the European Association of Software Science and Technology* **72** (2015). <https://doi.org/10.14279/tuj.eceasst.72.1015>

- [18] Bouyer, P., Markey, N., Sankur, O.: Robustness in timed automata. In: RP. LNCS, vol. 8169, pp. 1–18. Springer (2013). https://doi.org/10.1007/978-3-642-41036-9_1, invited paper
- [19] Bruyère, V., Dall’Olio, E., Raskin, J.F.: Durations and parametric model-checking in timed automata. *ACM Transactions on Computational Logic* **9**(2), 12:1–12:23 (2008). <https://doi.org/10.1145/1342991.1342996>
- [20] Cassez, F.: The dark side of timed opacity. In: ISA. LNCS, vol. 5576, pp. 21–30. Springer (2009). https://doi.org/10.1007/978-3-642-02617-1_3
- [21] Chen, A., Hong, C., Shang, X., Jing, H., Xu, S.: Timing leakage to break SM2 signature algorithm. *Journal of Information Security and Applications* **67**, 103210 (2022). <https://doi.org/10.1016/J.JISA.2022.103210>
- [22] De Wulf, M., Doyen, L., Markey, N., Raskin, J.F.: Robustness and implementability of timed automata. In: FORMATS and FTRTFT. LNCS, vol. 3253, pp. 118–133. Springer (2004). https://doi.org/10.1007/978-3-540-30206-3_10
- [23] Dima, C.: Real-time automata. *Journal of Automata, Languages and Combinatorics* **6**(1), 3–23 (2001). <https://doi.org/10.25596/jalc-2001-003>
- [24] Jurdzinski, M., Trivedi, A.: Reachability-time games on timed automata. In: ICALP. LNCS, vol. 4596, pp. 838–849. Springer (2007). https://doi.org/10.1007/978-3-540-73420-8_72
- [25] Klein, J., Kogel, P., Glesner, S.: Verifying opacity of discrete-timed automata. In: FormaliSE. pp. 55–65. ACM (2024). <https://doi.org/10.1145/3644033.3644376>
- [26] Li, J., Lefebvre, D., Hadjicostis, C.N., Li, Z.: Observers for a class of timed automata based on elapsed time graphs. *IEEE Transactions on Automatic Control* **67**(2), 767–779 (2022). <https://doi.org/10.1109/TAC.2021.3064542>
- [27] Standaert, F.: Introduction to side-channel attacks. In: *Secure Integrated Circuits and Systems*, pp. 27–42. *Integrated Circuits and Systems*, Springer (2010). https://doi.org/10.1007/978-0-387-71829-3_2
- [28] Vardi, M.Y., Wolper, P.: Reasoning about infinite computations. *Information and Computation* **115**(1), 1–37 (1994). <https://doi.org/10.1006/INCO.1994.1092>
- [29] Wang, L., Zhan, N.: Decidability of the initial-state opacity of real-time automata. In: *Symposium on Real-Time and Hybrid Systems - Essays Dedicated to Professor Chaochen Zhou on the Occasion of His 80th Birthday*, LNCS, vol. 11180, pp. 44–60. Springer (2018). https://doi.org/10.1007/978-3-030-01461-2_3
- [30] Wang, L., Zhan, N., An, J.: The opacity of real-time automata. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **37**(11), 2845–2856 (2018). <https://doi.org/10.1109/TCAD.2018.2857363>
- [31] Zhang, K.: State-based opacity of labeled real-time automata. *TCS* **987**, 114373 (2024). <https://doi.org/10.1016/J.TCS.2023.114373>

A Notation table

Timed automaton	
\mathcal{A}	a timed automaton
Σ	a finite set of actions of a TA
L	a finite set of locations of a TA
ℓ_0	the initial location of a TA
ℓ_{priv}	the private location of a TA
F	the set of final locations of a TA
X	a finite set of clocks
x_i	the i^{th} clock
H	the number of clocks
$I(\ell)$	the invariant of location ℓ
E	the finite set of edges of a TA
e	an edge
R	a set of clocks to be reset
g	a guard
μ	a clock valuation
d	a delay
z	extra clock
c_i	largest constant for clock x_i

Semantics	
$TTS_{\mathcal{A}}$	the semantics of TA \mathcal{A}
S	the set of states in $TTS_{\mathcal{A}}$
s_0	the initial state in $TTS_{\mathcal{A}}$
s	a state in $TTS_{\mathcal{A}}$
δ	transition function of $TTS_{\mathcal{A}}$
\xrightarrow{e}	a discrete transition with edge e
\xrightarrow{d}	a delay transition with delay d
ρ	a run
$last(\rho)$	last state of run ρ

Regions	
r	a region
$[s]$	equivalence class of s
$\mathcal{R}_{\mathcal{A}}$	regions set of \mathcal{A}
$\mathcal{R}_{\mathcal{A}}^F$	set of final regions of \mathcal{A}
$\mathcal{R}_{\mathcal{A}}$	region automaton of \mathcal{A}
$\Sigma^{\mathcal{R}}$	set of actions of $\mathcal{R}_{\mathcal{A}}$
$\delta^{\mathcal{R}}$	transition function of $\mathcal{R}_{\mathcal{A}}$
$\rightarrow_{\mathcal{R}}$	a transition in $\mathcal{R}_{\mathcal{A}}$

Opacity	
$\overline{Visit^{priv}}(\mathcal{A})$	set of private runs of \mathcal{A}
$Visit^{priv}(\mathcal{A})$	set of public runs of \mathcal{A}
$D\overline{Visit^{priv}}(\mathcal{A})$	set of durations of private runs of \mathcal{A}
$DVisit^{priv}(\mathcal{A})$	set of durations of public runs of \mathcal{A}
Strategy	
Σ_c	controllable actions
Σ_u	uncontrollable actions
σ	a strategy
α	a \mathfrak{b} -strategy
\mathcal{A}^σ	\mathcal{A} controlled by strategy σ
δ^σ	transition function of $TTS_{\mathcal{A}^\sigma}$
\mapsto_σ	a discrete or delay transition in $TTS_{\mathcal{A}^\sigma}$
$\xrightarrow{d,e}_\sigma$	a transition with delay d and edge e in $TTS_{\mathcal{A}^\sigma}$
\mathfrak{C}	set of activated controllable actions
Duplicated TA	
\mathcal{A}^{dup}	a duplicated TA
L_{priv}	set of private states
L_{pub}	set of public states
ℓ_{priv}	a private state
Beliefs	
\mathfrak{b}_t	belief for time t
$\mathcal{B}_{\mathcal{A}}$	belief automaton of \mathcal{A}
$\mathfrak{S}^{\mathcal{B}_{\mathcal{A}}}$	states of $\mathcal{B}_{\mathcal{A}}$
$\mathfrak{A}^{\mathcal{B}_{\mathcal{A}}}$	actions of $\mathcal{B}_{\mathcal{A}}$
v	a sequence of actions of $\mathcal{B}_{\mathcal{A}}$
$\mathfrak{D}^{\mathcal{B}_{\mathcal{A}}}$	transition function of $\mathcal{B}_{\mathcal{A}}$
$\text{Secret}_{\mathcal{A}}$	set of regions reachable on a run visiting ℓ_{priv}
$\overline{\text{Secret}}_{\mathcal{A}}$	set of regions not reachable on a run visiting ℓ_{priv}
$\mathbb{B}_{\mathcal{A}}^\sigma$	set of beliefs reachable by a strategy σ
$\mathcal{B}_{\mathcal{A}}^\alpha$	controlled belief automaton of \mathcal{A} and \mathfrak{b} -strategy α
$\mathfrak{S}^{\mathcal{B}_{\mathcal{A}}^\alpha}$	states of controlled belief automaton
$\mathfrak{A}^{\mathcal{B}_{\mathcal{A}}^\alpha}$	actions of controlled belief automaton
$\mathfrak{D}^{\mathcal{B}_{\mathcal{A}}^\alpha}$	transition function of controlled belief automaton

B Proofs of Section 4

B.1 Proof of Lemma 1

Lemma 1 (Beliefs characterization for full ET-opacity). *A TA \mathcal{A} is fully ET-opaque with a strategy σ iff, for all $\mathfrak{b} \in \mathbb{B}_{\mathcal{A}}^\sigma$, \mathfrak{b} is not bad for full ET-opacity.*

Proof. \Rightarrow Let \mathcal{A} be a TA that is fully ET-opaque with strategy σ . Let $\mathbf{b} \in \mathbb{B}_{\mathcal{A}}^{\sigma}$. Suppose w.l.o.g. that $\mathbf{b} \cap \mathbb{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}} \neq \emptyset$. Let $r \in \mathbf{b} \cap \mathbb{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}}$. Then there is a σ -compatible run $\rho \in \text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})$ such that $\overline{\text{last}(\rho)} \in r$. \mathcal{A} being fully ET-opaque with strategy σ , $D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A}) = \overline{D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})}$. Thus, there exists $\rho' \in \text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})$ such that $\text{dur}(\rho') = \text{dur}(\rho)$. Denoting $r' = [\text{last}(\rho')]$, we have by definition that $r' \in \mathbf{b} \cap \mathbb{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}}$. (Note that $r' \in \mathbb{R}_{\mathcal{A}}^F$ as a run ρ belongs to $\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A}) \cup \overline{\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})}$ only if $[\text{last}(\rho)] \in \mathbb{R}_{\mathcal{A}}^F$.) Therefore $\mathbf{b} \cap \mathbb{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}} \neq \emptyset$.

\Leftarrow If $D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A}) \cup \overline{D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})} = \emptyset$, \mathcal{A} is fully ET-opaque with σ by definition. We thus assume otherwise and select $t \in D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A}) \cup \overline{D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})}$. W.l.o.g. we can assume that $t \in D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})$. There thus exists a private σ -compatible run ρ of duration t . In particular, $[\text{last}(\rho)] \in \mathbf{b}_t^{\sigma}$. By hypothesis, as $\mathbf{b}_t^{\sigma} \cap \mathbb{R}_{\mathcal{A}}^F \neq \emptyset$ (because of ρ), we have that $\mathbf{b}_t^{\sigma} \cap \mathbb{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}} \neq \emptyset$. Thus, there exists a σ -compatible run ρ' such that $\text{dur}(\rho') = t$ and $[\text{last}(\rho')] \in \mathbf{b}_t^{\sigma} \cap \mathbb{R}_{\mathcal{A}}^F \cap \overline{\text{Secret}}_{\mathcal{A}}$. By definition of these sets, this implies that $t \in D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})$. Hence, $D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A}) = \overline{D\text{Visit}_{\sigma}^{\text{priv}}(\mathcal{A})}$ and \mathcal{A} is fully ET-opaque with σ . \square

B.2 Proof of Lemma 3

Lemma 3 (Strategies and runs acceptance). *Let \mathcal{A} be a TA, ρ a run of \mathcal{A}^{dup} , σ a finitely-varying strategy and α a finitely-varying \mathbf{b} -strategy such that $\sigma \vdash \alpha$. We have: ρ is σ -compatible iff ρ is feasible in $\mathcal{B}_{\mathcal{A}}^{\alpha}$.*

Proof. \Rightarrow Let $\rho = (\ell_0, \mathbf{0}), (d_0, \mathbf{e}_0), \dots, (d_{n-1}, \mathbf{e}_{n-1}), (\ell_n, \mu_n)$ be a σ -compatible run of \mathcal{A}^{dup} . By definition of a run, for all $0 \leq i < n$, there exists a sequence of transitions $(\ell_i, \mu_i) \xrightarrow{d_i} (\ell_i, \mu'_i) \xrightarrow{\mathbf{e}_i} (\ell_{i+1}, \mu_{i+1})$ in $TTS_{\mathcal{A}}$ with $\mathbf{e}_i = (\ell_i, g, a_i, R, \ell_{i+1})$ s.t. $a_i \in \sigma \left(\sum_{j \leq i} d_j \right) \cup \Sigma_u \cup \{\varepsilon\}$.

For all transitions in the $TTS_{\mathcal{A}}$, there exist $\mathbf{b}_{i_1}, \mathbf{b}_{i_m}$ such that the initial state of the transition belongs to \mathbf{b}_{i_1} and the target state belongs to \mathbf{b}_{i_m} , and such that either $\mathbf{b}_{i_1} = \mathbf{b}_{i_m}$ or there is a transition between them in the belief automaton. More exactly,

1. if $[\mu_i(z)] = [\mu_{i+1}(z)]$, and $\text{fr}(\mu_i(z)) = 0$ iff $\text{fr}(\mu_{i+1}(z)) = 0$, then there exist $\mathbf{b}_{i_1}, \dots, \mathbf{b}_{i_m}$, $1 \leq m$, s.t. $[(\ell_i, \mu_i)] \in \mathbf{b}_{i_1}$, $[(\ell_{i+1}, \mu_{i+1})] \in \mathbf{b}_{i_m}$, $a_i \in (\mathcal{C}_{m-1} \cup \Sigma_u \cup \{\varepsilon\})$ and $(\mathbf{b}_{i_1}, (\dagger, \mathcal{C}_1), \mathbf{b}_{i_2}), \dots, (\mathbf{b}_{i_{m-1}}, (\dagger, \mathcal{C}_{m-1}), \mathbf{b}_{i_m}) \in \delta^{\mathcal{B}_{\mathcal{A}}}$, $\dagger \in \{0, 0^+\}$,
2. otherwise, there exist $\mathbf{b}_{i_1}, \dots, \mathbf{b}_{i_m}$ s.t. $[(\ell_i, \mu_i)] \in \mathbf{b}_{i_1}$, $[(\ell_{i+1}, \mu_{i+1})] \in \mathbf{b}_{i_m}$ and $(\mathbf{b}_{i_1}, (\dagger, \mathcal{C}_1), \mathbf{b}_{i_2}), \dots, (\mathbf{b}_{i_{m-1}}, (\dagger, \mathcal{C}_{m-1}), \mathbf{b}) \in \delta^{\mathcal{B}_{\mathcal{A}}}$, $\dagger \in \{0^+, 1\}$.

Finally, as $a_i \in \sigma \left(\sum_{j \leq i} d_j \right) \cup \Sigma_u \cup \{\varepsilon\}$, for all $0 \leq i < n$, and $\sigma \vdash \alpha$, the transition associated to each action will belongs to \mathcal{C}_{m-1} and ρ is feasible in $\mathcal{B}_{\mathcal{A}}^{\alpha}$.

\Leftarrow Let $\rho = (\ell_0, \mathbf{0}), (d_0, \mathbf{e}_0), \dots, (d_{n-1}, \mathbf{e}_{n-1}), (\ell_n, \mu_n)$ feasible in $\mathcal{B}_{\mathcal{A}}^{\alpha}$. Then, there is $v = v_1, \dots, v_n \in \mathfrak{A}^{\mathcal{B}_{\mathcal{A}}^{\alpha}}$ such that, for all $0 \leq i < n$, $(\ell_0, \mathbf{0}), (d_0, \mathbf{e}_0), \dots, (d_{i-1}, \mathbf{e}_{i-1}), (\ell_i, \mu_i) \models v_1, \dots, v_i$ and there is \mathbf{b}_i , such that $[(\ell_i, \mu_i)]$, reachable by v_1, \dots, v_i . For all $0 \leq i < n$,

- if $d_i = 0$, then $a_i \in \mathfrak{C} \cup \Sigma_u \cup \{\varepsilon\}$ with $\mathfrak{C} = \alpha(v_1, \dots, vi)$,
- otherwise, there is $(\dagger_{i_1}, \mathfrak{C}_{i_1}), \dots, (\dagger_{i_m}, \mathfrak{C}_{i_m})$ such that \mathfrak{b}_i is reachable by $v_1, \dots, vi, (\dagger_{i_1}, \mathfrak{C}_{i_1}), \dots, (\dagger_{i_m}, \mathfrak{C}_{i_m})$, and, in particular, $(\mathfrak{b}_{i_m}, (\dagger, \mathfrak{C}), \mathfrak{b}_i)$ such that $a_i \in \mathfrak{C} \cup \Sigma_u \cup \{\varepsilon\}$ with $\mathfrak{C} = \alpha(v_1, \dots, vi, (\dagger_{i_1}, \mathfrak{C}_{i_1}), \dots, (\dagger_{i_m}, \mathfrak{C}_{i_m}))$.

In both cases, as $\sigma \Vdash \alpha$, $a_i \in \sigma(\sum_{j \leq i} d_j)$ and then ρ is σ -compatible. \square

C Proof of Section 5

C.1 Proof of Theorem 2

Theorem 2. *The full ET-opacity finitely-varying controller emptiness problem is decidable in EXPSPACE; and the full ET-opacity finitely-varying controller synthesis problem is solvable in 2-EXPTIME.*

Proof. We define the one-player Büchi game $\mathcal{G} = (\mathfrak{S}^{\mathcal{B}_{\mathcal{A}}}, (\varepsilon, \perp), \mathfrak{A}^{\mathcal{B}_{\mathcal{A}}}, \delta_{\mathfrak{D}^{\mathcal{B}_{\mathcal{A}}}}, G)$ where $G = \{(\dagger, \mathfrak{C}) \in \mathfrak{A}^{\mathcal{B}_{\mathcal{A}}} \mid \dagger = 1\}$ and $\delta_{\mathfrak{D}^{\mathcal{B}_{\mathcal{A}}}} = \{(\mathfrak{b}, (\dagger, \mathfrak{C}), \mathfrak{b}') \in \mathfrak{D}^{\mathcal{B}_{\mathcal{A}}} \mid \mathfrak{b} \text{ is not a bad belief for full ET-opacity}\}$ (in other words, we cut transitions leaving the bad beliefs).

First, for a TA \mathcal{A} , we know by Theorem 1 that if there is a finitely-varying strategy σ such that \mathcal{A} is fully ET-opaque with σ then there is a finitely-varying \mathfrak{b} -strategy α such that no bad belief is reachable in $\mathcal{B}_{\mathcal{A}}^\alpha$. The strategy α implies a winning strategy for \mathcal{G} as, being finitely-varying, it selects infinitely often a good transition and, not visiting a bad belief, the removal of transitions leaving those beliefs does not affect the strategy. Conversely, a winning strategy of \mathcal{G} directly entails a \mathfrak{b} -strategy α of $\mathcal{B}_{\mathcal{A}}$ which, being winning in \mathcal{G} does not visit a bad belief and is finitely-varying, hence by Theorem 1 there is a finitely-varying strategy σ such that \mathcal{A} is fully ET-opaque with σ .

By Lemma 4, the existence of a winning strategy in this game is decidable in NLOGSPACE in the size of the game, hence it is in EXPSPACE (the non-determinism can freely be removed thanks to Savitch theorem, which implies that EXPSPACE = NEXPSPACE). Moreover, through the computation of a strategy in \mathcal{G} , one directly obtains that the full ET-opacity finitely-varying controller synthesis problem is solvable in 2-EXPTIME. \square

D Full definitions of Section 6

We define formally bad belief for almost full ET-opacity and closed full ET-opacity.

Definition 15 (Bad belief for almost full ET-opacity). *A belief \mathfrak{b}_t is bad for almost full ET-opacity when*

- *it is bad for full ET-opacity and*
- *there is $d \in \mathbb{R}_{>0}$ such that, either*
 - *for all $\epsilon \in \mathbb{R}_{>0}$, $\epsilon \leq d$, $\mathfrak{b}_{t+\epsilon}$ is bad for full ET-opacity, or*

- for all $\epsilon \in \mathbb{R}_{<0}, \epsilon \geq -d$, $\mathbf{b}_{t+\epsilon}$ is bad for full ET-opacity.

Definition 16 (Bad belief for closed full ET-opacity). A belief \mathbf{b}_t is bad for closed full ET-opacity when

- it is bad for full ET-opacity and
- there is $d \in \mathbb{R}_{>0}$ such that either
 - for all $\epsilon \in \mathbb{R}_{>0}, \epsilon \leq d$, $\mathbf{b}_{t+\epsilon}$ is bad for full ET-opacity, or
 - for all $\epsilon \in \mathbb{R}_{<0}, \epsilon \geq -d$, $\mathbf{b}_{t+\epsilon}$ is bad for full ET-opacity, or
 - for all $\epsilon \in \mathbb{R}_{\neq 0}, -d \leq \epsilon \leq d$, $\mathbf{b}_{t+\epsilon} \cap \mathbb{R}_{\mathcal{A}}^F = \emptyset$.

Now, we formally define weak variants of almost and closed opacity. We give also the corresponding definition of bad beliefs for each notion.

Definition 17 (Almost weak ET-opacity). A TA \mathcal{A} is almost weakly ET-opaque when $(\llbracket DVisit^{priv}(\mathcal{A}) \rrbracket) \setminus (\llbracket DVisit^{\overline{priv}}(\mathcal{A}) \rrbracket) = \emptyset$.

Definition 18 (Bad belief for almost weak ET-opacity). A belief \mathbf{b}_t is bad for almost weak ET-opacity when

- it is bad for weak ET-opacity and
- there is $d \in \mathbb{R}_{>0}$ such that, either
 - for all $\epsilon \in \mathbb{R}_{>0}, \epsilon \leq d$, $\mathbf{b}_{t+\epsilon}$ is bad for weak ET-opacity, or
 - for all $\epsilon \in \mathbb{R}_{<0}, \epsilon \geq -d$, $\mathbf{b}_{t+\epsilon}$ is bad for weak ET-opacity.

Definition 19 (Closed weak ET-opacity). A TA \mathcal{A} is closed weakly ET-opaque when $\llbracket DVisit^{priv}(\mathcal{A}) \rrbracket \subseteq \llbracket DVisit^{\overline{priv}}(\mathcal{A}) \rrbracket$.

Definition 20 (Bad belief for closed weak ET-opacity). A belief \mathbf{b}_t is bad for closed weak ET-opacity when

- it is bad for weak ET-opacity and
- there is $d \in \mathbb{R}_{>0}$ such that either
 - for all $\epsilon \in \mathbb{R}_{>0}, \epsilon \leq d$, $\mathbf{b}_{t+\epsilon}$ is bad for weak ET-opacity, or
 - for all $\epsilon \in \mathbb{R}_{<0}, \epsilon \geq -d$, $\mathbf{b}_{t+\epsilon}$ is bad for weak ET-opacity, or
 - for all $\epsilon \in \mathbb{R}_{\neq 0}, -d \leq \epsilon \leq d$, $\mathbf{b}_{t+\epsilon} \cap \mathbb{R}_{\mathcal{A}}^F = \emptyset$.

E Detailed proofs for Section 6

E.1 Closed ET-opacity

Lemma 5 (Beliefs characterization for closed full ET-opacity). A TA \mathcal{A} is closed fully ET-opaque with a strategy σ iff, for all $\mathbf{b} \in \mathbb{B}_{\mathcal{A}}^{\sigma}$, \mathbf{b} is not bad for closed full ET-opacity.

Proof. \Rightarrow Let \mathcal{A} be a TA closed fully ET-opaque with strategy σ . Let $\mathbf{b}_t \in \mathbb{B}_{\mathcal{A}}^{\sigma}$.

- Either $t \in DVisit_{\sigma}^{priv}(\mathcal{A}) \cap DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$ then \mathbf{b}_t is not bad for full ET-opacity, and thus cannot be bad for closed full ET-opacity,

- or, similarly, $t \notin D\overline{Visit}_\sigma^{priv}(\mathcal{A}) \cup D\overline{Visit}_\sigma^{priv}(\mathcal{A})$ and is not bad for closed full ET-opacity,
- or finally, suppose w.l.o.g. that $t \in D\overline{Visit}_\sigma^{priv}(\mathcal{A}) \setminus D\overline{Visit}_\sigma^{priv}(\mathcal{A})$. As \mathcal{A} is closed fully ET-opaque with σ , $[[D\overline{Visit}_\sigma^{priv}(\mathcal{A})]] = [[D\overline{Visit}_\sigma^{priv}(\mathcal{A})]]$, and so this means that there exists $d \in \mathbb{R}_{>0}$ such that either
 1. for all $\epsilon \in \mathbb{R}_{>0}$, $\epsilon \leq d$, $t + \epsilon \in D\overline{Visit}_\sigma^{priv}(\mathcal{A}) \cap D\overline{Visit}_\sigma^{priv}(\mathcal{A})$, or
 2. for all $\epsilon \in \mathbb{R}_{<0}$, $\epsilon \geq -d$, $t + \epsilon \in D\overline{Visit}_\sigma^{priv}(\mathcal{A}) \cap D\overline{Visit}_\sigma^{priv}(\mathcal{A})$, or
 3. for all $\epsilon \in \mathbb{R}_{\neq 0}$, $-d \leq \epsilon \leq d$, $\mathbf{b}_{t+\epsilon} \cap R_{\mathcal{A}}^F = \emptyset$.

In other words, for Items 1 and 2, for all ϵ there are σ -compatible runs $\rho_\epsilon, \rho'_\epsilon$ such that $\text{dur}(\rho_\epsilon) = \text{dur}(\rho'_\epsilon) = t + \epsilon$ and $\rho_\epsilon \in D\overline{Visit}_\sigma^{priv}(\mathcal{A})$, $\rho'_\epsilon \in D\overline{Visit}_\sigma^{priv}(\mathcal{A})$. Then, $\{\text{last}(\rho_\epsilon), \text{last}(\rho'_\epsilon)\} \subseteq \mathbf{b}_{t+\epsilon}$ and so $\mathbf{b}_{t+\epsilon}$ is not bad for full ET-opacity. For Item 3, for all ϵ , there are no run ρ_ϵ such that $\text{dur}(\rho_\epsilon) = t$. So $\mathbf{b}_{t+\epsilon} \cap R_{\mathcal{A}}^F \neq \emptyset$. As a consequence, \mathbf{b}_t is not a bad belief for closed full ET-opacity.

Therefore, no bad belief for closed full ET-opacity belongs to $\mathbb{B}_{\mathcal{A}}^\sigma$.

⇐ Let \mathcal{A} be a TA such that there is no bad belief for closed full ET-opacity in $\mathbb{B}_{\mathcal{A}}^\sigma$. For every time $t \in \mathbb{R}_{\geq 0}$, let \mathbf{b}_t be the associated belief. \mathbf{b}_t is not bad for closed full ET-opacity, thus one of the following holds:

- \mathbf{b}_t is not bad for full ET-opacity so no violation of opacity occurs at time t ,
- \mathbf{b}_t is bad for full ET-opacity and there exists $d \in \mathbb{R}_{>0}$ such that either
 1. for all $\epsilon \in \mathbb{R}_{>0}$, $-d \leq \epsilon \leq d$, $\mathbf{b}_{t+\epsilon}$ is not bad for full ET-opacity, and
 2. for all $\epsilon \in \mathbb{R}_{\neq 0}$, $-d \leq \epsilon \leq d$, $\mathbf{b}_{t+\epsilon} \cap R_{\mathcal{A}}^F \neq \emptyset$.

Thus, for all ϵ , there are σ -compatible runs $\rho_\epsilon, \rho'_\epsilon$ such that $\rho_\epsilon \in \overline{Visit}_\sigma^{priv}(\mathcal{A})$, $\rho'_\epsilon \in \overline{Visit}_\sigma^{priv}(\mathcal{A})$ with $t + \epsilon = \text{dur}(\rho_\epsilon) = \text{dur}(\rho'_\epsilon)$. This means that $t + \epsilon \in D\overline{Visit}_\sigma^{priv}(\mathcal{A}) \cap D\overline{Visit}_\sigma^{priv}(\mathcal{A})$ and $[[D\overline{Visit}_\sigma^{priv}(\mathcal{A})]] = [[D\overline{Visit}_\sigma^{priv}(\mathcal{A})]]$.

Therefore, \mathcal{A} is closed fully ET-opaque with strategy σ . □

Lemma 6 (Beliefs characterization for closed weak ET-opacity). *A TA \mathcal{A} is closed weakly ET-opaque with a strategy σ iff, for all $\mathbf{b} \in \mathbb{B}_{\mathcal{A}}^\sigma$, \mathbf{b} is not bad for closed weak ET-opacity.*

Proof. This proof can be achieved similarly to the proof of Lemma 5. □

We can now proceed to the proof of Theorem 3.

Theorem 3. *The closed full ET-opacity finitely-varying controller emptiness problem is decidable; the closed full ET-opacity finitely-varying controller synthesis problem is solvable.*

Proof. First, as for full ET-opacity with Theorem 1, we can prove that, for a given TA \mathcal{A} and a strategy σ , \mathcal{A}^σ is closed fully ET-opaque (resp. closed weakly ET-opaque) iff there exists α , $\sigma \vdash \alpha$ such that there is no bad belief for closed full ET-opacity (resp. for closed weak ET-opacity) reachable in $\mathcal{B}_{\mathcal{A}}^\alpha$. The proof is similar to the one of Theorem 1.

Then, as within the proof of Theorem 2, based on Lemma 4 and the relation between our problems and their correspondence with a safety game, we can conclude that closed full ET-opacity finitely-varying controller emptiness problem is decidable and the closed full ET-opacity finitely-varying controller synthesis problem is solvable. \square

Theorem 4. *The closed weak ET-opacity finitely-varying controller emptiness problem is decidable; the closed weak ET-opacity finitely-varying controller synthesis problem is solvable.*

Proof. This proof can be achieved similarly to the proof of Theorem 3. \square

E.2 Almost ET-opacity

Lemma 7 (Beliefs characterization for almost full ET-opacity). *A TA \mathcal{A} is almost fully ET-opaque with a strategy σ iff, for all $\mathbf{b} \in \mathbb{B}_{\mathcal{A}}^{\sigma}$, \mathbf{b} is not bad for almost full ET-opacity.*

Proof. \Rightarrow Let \mathcal{A} be a TA almost fully ET-opaque with strategy σ . Let $\mathbf{b}_t \in \mathbb{B}_{\mathcal{A}}^{\sigma}$.

- Either $t \in DVisit_{\sigma}^{priv}(\mathcal{A}) \cap DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$, then \mathbf{b}_t is not bad for full ET-opacity, and thus cannot be bad for almost full ET-opacity,
- or, similarly, $t \notin DVisit_{\sigma}^{priv}(\mathcal{A}) \cup DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$ and is not bad for almost full ET-opacity,
- or finally, suppose w.l.o.g. that $t \in DVisit_{\sigma}^{priv}(\mathcal{A}) \setminus DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$. As \mathcal{A} is almost fully ET-opaque with σ , $(DVisit_{\sigma}^{priv}(\mathcal{A}) \oplus DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})) = \emptyset$, and so this means that there exists $d \in \mathbb{R}_{>0}$ such that either
 - * for all $\epsilon \in \mathbb{R}_{>0}, \epsilon \leq d, t + \epsilon \in DVisit_{\sigma}^{priv}(\mathcal{A}) \cap DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$, or
 - * for all $\epsilon \in \mathbb{R}_{<0}, \epsilon \geq -d, t + \epsilon \in DVisit_{\sigma}^{priv}(\mathcal{A}) \cap DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$.

In other words, for all ϵ there are σ -compatible runs $\rho_{\epsilon}, \rho'_{\epsilon}$ such that $dur(\rho_{\epsilon}) = dur(\rho'_{\epsilon}) = t + \epsilon$ and $\rho_{\epsilon} \in DVisit_{\sigma}^{priv}(\mathcal{A}), \rho'_{\epsilon} \in DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$. Then, $\{[last(\rho_{\epsilon})], [last(\rho'_{\epsilon})]\} \subseteq \mathbf{b}_{t+\epsilon}$. In both case, $\mathbf{b}_{t+\epsilon}$ is not bad for full ET-opacity. As a consequence, \mathbf{b}_t is not a bad belief for almost full ET-opacity.

Therefore, no bad belief for almost full ET-opacity belongs to $\mathbb{B}_{\mathcal{A}}^{\sigma}$.

\Leftarrow Let \mathcal{A} be a TA such that there is no bad belief for almost full ET-opacity in $\mathbb{B}_{\mathcal{A}}^{\sigma}$. For every time $t \in \mathbb{R}_{\geq 0}$, let \mathbf{b}_t be the associated belief. \mathbf{b}_t is not bad for almost full ET-opacity, thus one of the following holds:

- \mathbf{b}_t is not bad for full ET-opacity so no violation of opacity occurs at time t ,
- \mathbf{b}_t is bad for full ET-opacity and there exists $d \in \mathbb{R}_{>0}$ such that for all $\epsilon \in \mathbb{R}_{>0}, -d \leq \epsilon \leq d, \mathbf{b}_{t+\epsilon}$ is not bad for full ET-opacity. Thus, for all ϵ , either
 - * there are σ -compatible runs $\rho_{\epsilon}, \rho'_{\epsilon}$ such that $\rho_{\epsilon} \in Visit_{\sigma}^{priv}(\mathcal{A}), \rho'_{\epsilon} \in Visit_{\sigma}^{\overline{priv}}(\mathcal{A})$ with $t + \epsilon = dur(\rho_{\epsilon}) = dur(\rho'_{\epsilon})$. This means that $t + \epsilon \in DVisit_{\sigma}^{priv}(\mathcal{A}) \cap DVisit_{\sigma}^{\overline{priv}}(\mathcal{A})$, or

* there are no run ρ_ϵ such that $\text{dur}(\rho_\epsilon) = t + \epsilon$. This means that $t + \epsilon \notin D\text{Visit}_\sigma^{\text{priv}}(\mathcal{A}) \cup \overline{D\text{Visit}_\sigma^{\text{priv}}(\mathcal{A})}$.

Finally, $(D\text{Visit}_\sigma^{\text{priv}}(\mathcal{A}) \oplus \overline{D\text{Visit}_\sigma^{\text{priv}}(\mathcal{A})})$.

Therefore, \mathcal{A} is almost fully ET-opaque with strategy σ . \square

Lemma 8 (Beliefs characterization for almost weak ET-opacity). *A TA \mathcal{A} is almost weakly ET-opaque with a strategy σ iff, for all $\mathbf{b} \in \mathbb{B}_{\mathcal{A}}^\sigma$, \mathbf{b} is not bad for almost weak ET-opacity.*

Proof. This proof can be achieved similarly to the proof of Lemma 7. \square

Theorem 5. *The almost full (resp. weak) ET-opacity finitely-varying controller emptiness problem is decidable; the almost full (resp. weak) ET-opacity finitely-varying controller synthesis problem is solvable.*

Proof. This proof can be achieved similarly to the proof of Theorem 3. \square

F Weak and existential ET-opacity

Here, we study two other versions of opacity [9], namely weak ET-opacity (in which it is harmless that the attacker deduces that the private location was *not* visited) and \exists -ET-opacity (in which we are simply interested in the *existence* of one execution time for which opacity is ensured).

F.1 Definitions

We recall definitions of weak and existential ET-opacity from [9].

Definition 21 (Weak and existential ET-opacity). *A TA \mathcal{A} is*

- weakly ET-opaque *when* $D\text{Visit}^{\text{priv}}(\mathcal{A}) \subseteq \overline{D\text{Visit}^{\text{priv}}(\mathcal{A})}$,
- existentially ET-opaque *when* $D\text{Visit}^{\text{priv}}(\mathcal{A}) \cap \overline{D\text{Visit}^{\text{priv}}(\mathcal{A})} \neq \emptyset$.

That is, if for any run of duration d reaching a final location after visiting ℓ_{priv} , there exists another run of the same duration reaching a final location but not visiting the private location, the system is weakly ET-opaque. Finally, when there is at least one private run such that there exists a public run of the same duration, the system is existentially ET-opaque, denoted \exists -ET-opaque in the following.

Example 10. We have seen that \mathcal{A}_1 is not fully ET-opaque Example 3. However, since we can reach ℓ_f at any time without visiting ℓ_{priv} , clearly $D\text{Visit}^{\text{priv}}(\mathcal{A}_1) \subseteq \overline{D\text{Visit}^{\text{priv}}(\mathcal{A}_1)}$ and \mathcal{A}_1 is weakly ET-opaque (and \exists -ET-opaque too).

As for the other notions of ET-opacity, we can define a bad belief for weak ET-opacity.

Definition 22 (Bad belief for weak ET-opacity). *A belief \mathbf{b} is bad for weak ET-opacity when*

- $(\mathbf{b} \cap R_{\mathcal{A}}^F \cap \text{Secret}_{\mathcal{A}} \neq \emptyset)$ and
- $(\mathbf{b} \cap R_{\mathcal{A}}^F \cap \overline{\text{Secret}_{\mathcal{A}}} = \emptyset)$.

F.2 Beliefs characterization and results for weak ET-opacity

Lemma 9 (Beliefs characterization for weak ET-opacity). *A TA \mathcal{A} is weakly ET-opaque with a strategy σ iff, for all $\mathbf{b} \in \mathbb{B}_{\mathcal{A}}^{\sigma}$, \mathbf{b} is not bad for weak ET-opacity.*

Proof. This proof can be achieved similarly to the proof of Lemma 1. \square

Theorem 6. *Let \mathcal{A} be a TA. Given a finitely-varying strategy σ such that \mathcal{A} is weakly ET-opaque with σ , there exists $\sigma \vdash \alpha$ such that there is no belief that is bad for weak ET-opacity reachable in $\mathcal{B}_{\mathcal{A}}^{\alpha}$. Reciprocally, given a \mathbf{b} -strategy α such that there is no belief that is bad for weak ET-opacity reachable in $\mathcal{B}_{\mathcal{A}}^{\alpha}$, there exists $\sigma \vdash \alpha$ such that \mathcal{A} is weakly ET-opaque with σ .*

Proof. This proof can be achieved similarly to the proof of Theorem 1. \square

Theorem 7. *The weak ET-opacity finitely-varying controller emptiness problem is decidable; and the weak ET-opacity finitely-varying controller synthesis problem is solvable.*

Proof. This proof can be achieved similarly to the proof of Theorem 2. \square

F.3 Results for \exists -ET-opacity

\exists -ET-opacity is a special case where control, as understood here, is useless. In fact, as the strategy of the controller can only prevent some behavior and remove possible executions. However, \exists -ET-opacity wonder if there exists an opaque time in the TA, so adding a controller cannot change the result. The following theorem proves this claim. The remained of the paper focuses on full and weak ET-opacity.

Theorem 8. *Let \mathcal{A} be a TA. There exists a finitely-varying strategy σ such that \mathcal{A}^{σ} is \exists -ET-opaque iff \mathcal{A} is \exists -ET-opaque.*

Proof. \Rightarrow Let \mathcal{A} be a TA and σ a strategy such that \mathcal{A}^{σ} is \exists -ET-opaque. That is, there exists a duration d such that there exist a private run ρ and a public run ρ' , both of duration d . Let $\rho = (\ell_0, \mathbf{0}), (d_0, \mathbf{e}_0), \dots, (d_{n-1}, \mathbf{e}_{n-1}), (\ell_n, \mu)$. For all $0 \leq i < n$, $e_i = (\ell_i, g_i, a_i, R_i, \ell'_i)$, with $a_i \in \sigma(\sum_{j=0}^{j \leq i} d_j) \cup \Sigma_u$. As $\sigma(\sum_{j=0}^{j \leq i} d_j) \cup \Sigma_u \subseteq \Sigma_c \cup \Sigma_u = \Sigma$, the action a_i is available when \mathcal{A} is not controlled and $\rho \in \text{Visit}^{\text{priv}}(\mathcal{A})$. With the same reasoning for ρ' , $\rho' \in \overline{\text{Visit}^{\text{priv}}}(\mathcal{A})$ and \mathcal{A} is \exists -ET-opaque.

\Leftarrow Assume that \mathcal{A} is \exists -ET-opaque. Then, \mathcal{A}^{σ} is \exists -ET-opaque for a controller with a strategy σ such that, for all $t \in \mathbb{R}_{\geq 0}$, $\sigma(t) = \Sigma_c$.

G Additional examples

If there is more than one clock, we extend our abuse of notation for regions to $(\ell, \tau_1, \dots, \tau_H)$, where each τ_i is either an interval or an integer. Note that this notation does not take into account the comparison between clocks but this is acceptable in the following example as the clocks always have the same fractional part.

Example 11. Let \mathcal{A}_3 the TA depicted in Fig. 5a. With the invariant $x \leq 1$ for ℓ_0 and $y \leq 2$ for ℓ_{priv} , we have the following beliefs. Each region is written $(\ell, \tau_1, \tau_2, \tau_3)$ with τ_1 for x , τ_2 for y and τ_3 for z . The corresponding belief automaton is depicted in Fig. 5b.

$$\begin{aligned}
\mathfrak{b}_0 &= \{(\ell_0, 0, 0, 0), (\ell_{priv}, 0, 0, 0), (\ell_f, 0, 0, 0)\} \\
\mathfrak{b}'_0 &= \{(\ell_0, 0, 0, 0), (\ell_{priv}, 0, 0, 0)\} \\
\mathfrak{b}_{(0,1)} &= \{(\ell_0, (0, 1), (0, 1), (0, 1)), (\ell_{priv}, (0, 1), (0, 1), (0, 1)), (\ell_f, (0, 1), (0, 1), (0, 1))\} \\
\mathfrak{b}'_{(0,1)} &= \{(\ell_0, (0, 1), (0, 1), (0, 1)), (\ell_{priv}, (0, 1), (0, 1), (0, 1))\} \\
\mathfrak{b}_1 &= \{(\ell_0, 1, 1, 1), (\ell_0, 1, 1, 0), (\ell_0, 0, 1, 1), (\ell_0, 0, 1, 0), (\ell_{priv}, 1, 1, 1), (\ell_{priv}, 1, 1, 0), \\
&\quad (\ell_{priv}, 0, 1, 1), (\ell_{priv}, 0, 1, 0), (\ell_f, 1, 1, 1), (\ell_f, 1, 1, 0), (\ell_f, 0, 1, 1), (\ell_f, 0, 1, 0)\} \\
\mathfrak{b}'_1 &= \{(\ell_0, 1, 1, 1), (\ell_0, 1, 1, 0), (\ell_0, 0, 1, 1), (\ell_0, 0, 1, 0), (\ell_{priv}, 0, 1, 1), (\ell_{priv}, 0, 1, 0), \\
&\quad (\ell_{priv}, 1, 1, 1), (\ell_{priv}, 1, 1, 0)\} \\
\mathfrak{b}_{(1,2)} &= \{(\ell_0, (0, 1), (1, 2), (0, 1)), (\ell_f, (0, 1), (1, 2), (0, 1)), (\ell_{priv}, (0, 1), (1, 2), (0, 1)), \\
&\quad (\ell_{priv}, (1, +\infty), (1, 2), (0, 1))\} \\
\mathfrak{b}'_{(1,2)} &= \{(\ell_0, (0, 1), (1, 2), (0, 1)), (\ell_{priv}, (0, 1), (1, 2), (0, 1)), (\ell_{priv}, (1, +\infty), (1, 2), (0, 1))\} \\
\mathfrak{b}_2 &= \{(\ell_0, 0, 2, 1), (\ell_0, 0, 2, 0), (\ell_0, 1, 2, 1), (\ell_0, 1, 2, 0), (\ell_{priv}, 0, 2, 1), (\ell_{priv}, 0, 2, 0), \\
&\quad (\ell_{priv}, 1, 2, 1), (\ell_{priv}, 1, 2, 0), (\ell_{priv}, (1, +\infty), 2, 1), (\ell_{priv}, (1, +\infty), 2, 0), \\
&\quad (\ell_f, 1, 2, 1), (\ell_f, 1, 2, 0), (\ell_f^p, 0, 2, 1), (\ell_f^p, 0, 2, 0)\} \\
\mathfrak{b}'_2 &= \{(\ell_0, 1, 2, 1), (\ell_0, 1, 2, 0), (\ell_0, 0, 2, 1), (\ell_0, 0, 2, 0), (\ell_{priv}, 0, 2, 1), (\ell_{priv}, 0, 2, 0), \\
&\quad (\ell_{priv}, (1, +\infty), 2, 1), (\ell_{priv}, (1, +\infty), 2, 0), (\ell_{priv}, 1, 2, 1), (\ell_{priv}, 1, 2, 0), \\
&\quad (\ell_f^p, 0, 2, 1), (\ell_f^p, 0, 2, 0)\} \\
\mathfrak{b}_{(2,3)} &= \{(\ell_0, (0, 1), (2, +\infty), (0, 1)), (\ell_f, (0, 1), (2, +\infty), (0, 1))\} \\
\mathfrak{b}'_{(2,3)} &= \{(\ell_0, (0, 1), (2, +\infty), (0, 1))\} \\
\mathfrak{b}_3 &= \{(\ell_0, 1, (2, +\infty), 1), (\ell_0, 1, (2, +\infty), 0), (\ell_0, 0, (2, +\infty), 1), (\ell_0, 0, (2, +\infty), 0), \\
&\quad (\ell_f, 0, (2, +\infty), 1), (\ell_f, 0, (2, +\infty), 0), (\ell_f, 1, (2, +\infty), 1), (\ell_f, 1, (2, +\infty), 0)\} \\
\mathfrak{b}'_3 &= \{(\ell_0, 1, (2, +\infty), 1), (\ell_0, 1, (2, +\infty), 0), (\ell_0, 0, (2, +\infty), 1), (\ell_0, 0, (2, +\infty), 0)\}
\end{aligned}$$

The following example show a TA opaque with a non-finitely-varying strategy and why we cannot manage it with our beliefs construction.

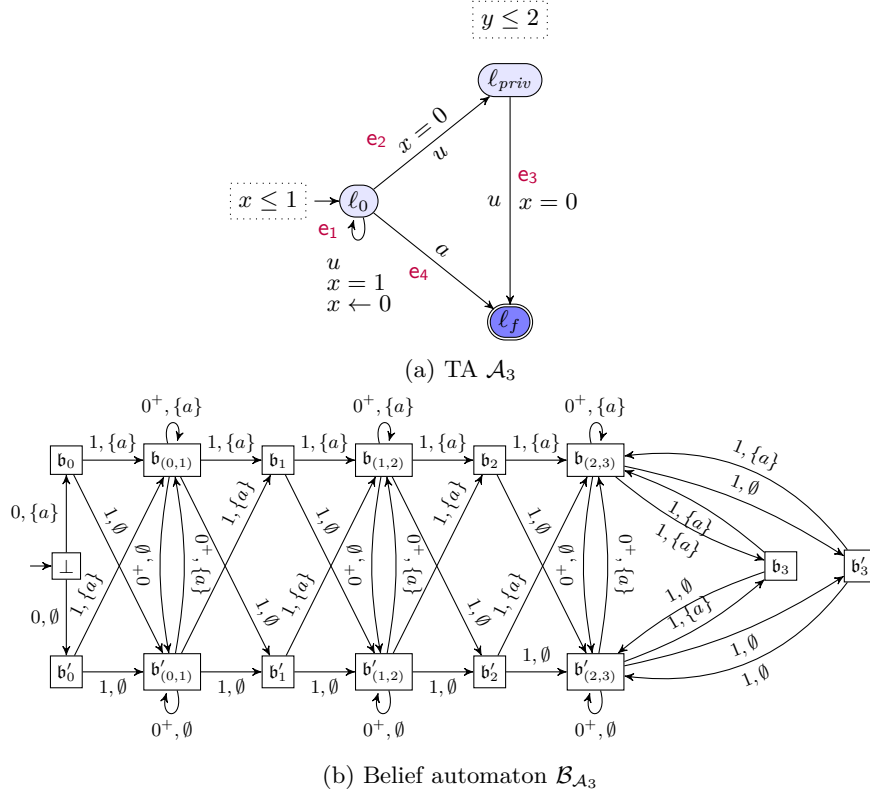


Fig. 5: \mathcal{A}_3 and the corresponding belief automaton

Example 12 (A non-finitely-varying strategy). Let \mathcal{A}_4 be the automaton in Fig. 6, with a global invariant $x, y \leq 1$. Let $\Sigma_u = \{u\}$ and $\Sigma_c = \{a, b\}$, we can build a strategy σ (depicted in red boxes) to make it opaque where

- $\forall \tau \in \mathbb{Q}, \sigma(\tau) = \{a\}$
- $\forall \tau \notin \mathbb{Q}, \sigma(\tau) = \{b\}$.

Let us see if we can find an corresponding **b**-strategy α for the belief automaton. In the belief automaton, whatever the subset of controllable actions chosen in \perp , \mathbf{b}_0 contains the region with location ℓ_0 and $x = 0$. Different beliefs are then reachable from \mathbf{b}_0 according to the chosen (finitely-varying) strategy:

- $\forall \tau \in (0, 1)$, if $\alpha(\tau) = \{a, b\}$, then we can reach the final location by the edge \mathbf{e}_3 (private way) and by \mathbf{e}_4 (public way). However, we can also reach location ℓ_3 , which will allow us to reach the final location by \mathbf{e}_7 when $x = 1$. As we cannot reach ℓ_f by a private way when $x = 1$, we do not want a and b are available at the same time when $x \leq 1$ to prevent to take edge \mathbf{e}_7 . So this strategy is not good.

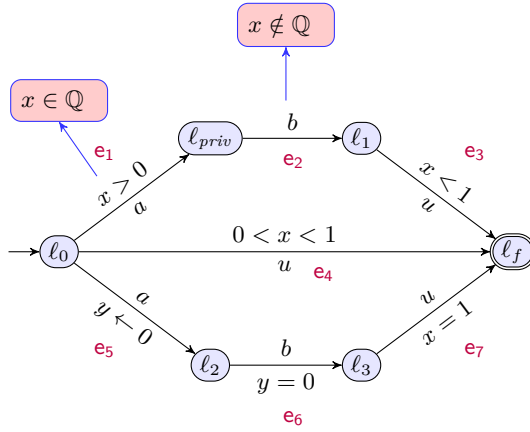


Fig. 6: Automaton \mathcal{A}_4

– $\forall \tau \in (0, 1)$, if $\alpha(\tau) = \{a\}$, $\alpha(\tau) = \{b\}$, or $\alpha(\tau) = \emptyset$ then we can reach the final location by e_4 , but not by e_3 . None of these strategies makes \mathcal{A}_3 opaque.

As σ is not finitely-varying and there is no other finitely-varying strategy such that \mathcal{A}_3 is opaque, we cannot find a \mathfrak{b} -strategy in the belief automaton.