Optimizing Performance of Continuous-Time Stochastic Systems using Timeout Synthesis*

Tomáš Brázdil[1], L'uboš Korenčiak[1], Jan Krčál[2], Petr Novotný[3], and Vojtěch Řehák[1]

[1] Faculty of Informatics, Masaryk University, Brno, Czech Republic
{brazdil, korenciak, rehak}@fi.muni.cz
[2] Saarland University – Computer Science, Saarbrücken, Germany
krcal@cs.uni-saarland.de
[3] IST Austria, Klosterneuburg, Austria
petr.novotny@ist.ac.at

Abstract. We consider parametric version of fixed-delay continuous-time Markov chains (or equivalently deterministic and stochastic Petri nets, DSPN) where fixed-delay transitions are specified by parameters, rather than concrete values. Our goal is to synthesize values of these parameters that, for a given cost function, minimise expected total cost incurred before reaching a given set of target states. We show that under mild assumptions, optimal values of parameters can be effectively approximated using translation to a Markov decision process (MDP) whose actions correspond to discretized values of these parameters. To this end we identify and overcome several interesting phenomena arising in systems with fixed delays.

1 Introduction

Continuous-time Markov chains (CTMC) are a fundamental model of stochastic systems with discrete state-spaces that evolve in continuous-time. Several higher level modelling formalisms, such as stochastic Petri nets and stochastic process algebras, use CTMC as their semantics. As such, CTMC have been applied in performance and dependability analysis in various contexts ranging from aircraft communication protocols (see, e.g. [35]) to models of biochemical systems (see, e.g. [22]).

There are several equivalent definitions of CTMC (see, e.g. [15, 31]). We may define a (uniformized, finite-state) CTMC to consist of a finite set of states S coupled with a common rate λ and a stochastic matrix $\mathrm{P} \in \mathbb{R}_{\geq 0}^{S \times S}$ specifying probabilities of transitions between states. An execution starts in a given initial state. In every step, the CTMC waits for a duration that is selected randomly according to the exponential distribution with the rate λ, and then moves to a state s' randomly chosen with probability $\mathrm{P}(s, s')$.

The practical interpretation of the above semantics is that in every state the system waits for an event to occur and then reacts by changing its state. A typical example is a model of

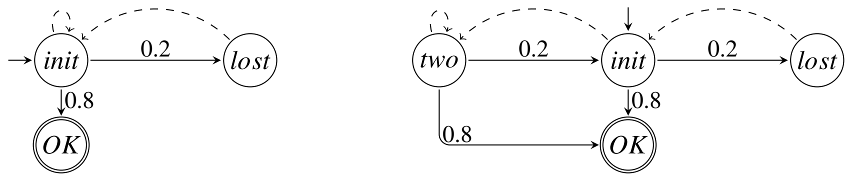
* The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement n° [291734]. This work is partly supported by the German Research Council (DFG) as part of the Transregional Collaborative Research Center AVACS (SFB/TR 14), by the EU 7th Framework Programme under grant agreement no. 295261 (MEALS) and 318490 (SENSATION), by the Czech Science Foundation, grant No. 15-17564S, and by the CAS/SAFEA International Partnership Program for Creative Research Teams.

a simple queue to which new customers come in random intervals and are also served in random intervals. However, in practice, events are usually not exponentially distributed, and, in fact, their distributions may be quite far from being exponential. To deal with such events, phase-type approximation technique [30] is usually applied. Unfortunately, as already noted in [30], some distributions cannot be efficiently fit with phase-type approximation. In particular, degenerate distributions of events with fixed delays, i.e., events that occur after a fixed amount of time with probability 1, form a distinguished example of this phenomenon (for more details see [25]). However, as events with fixed delays play a crucial role in many systems, especially in communication protocols [32], time-driven real-time scheduling [34], etc., they should be handled faithfully in modelling and analysis.

Inspired by deterministic and stochastic Petri nets [28] and delayed CTMC [16] with at most one non-exponential transition enabled in any time, we study fixed-delay CTMC (fdCTMC), the CTMC extended with *fixed-delay transitions*. More concretely, we specify a set of states $S_{\mathrm{fd}} \subseteq S$ where fixed-delay transitions are enabled and add a stochastic matrix $\mathrm{F} \in \mathbb{R}_{\geq 0}^{S_{\mathrm{fd}} \times S}$ specifying probabilities of fixed-delay transitions between states. In addition, we consider a *delay function* $\mathbf{d} : S_{\mathrm{fd}} \to \mathbb{R}_{>0}$. The semantics can be intuitively described as follows. Imagine a CTMC extended with an alarm clock. At the beginning of an execution, the alarm clock is turned off and the process behaves as the original CTMC. Whenever a state s of S_{fd} is visited and the alarm clock is off at the time, it is turned on and set to ring after $\mathbf{d}(s)$ time units. Subsequently, the process keeps behaving as the original CTMC until either a state of $S \setminus S_{\mathrm{fd}}$ is visited (in which case the alarm clock is turned off), or the alarm clock rings in a state s' of S_{fd}. In the latter case, a fixed-delay transition takes place, which means that the process changes the state randomly according to the distribution $\mathrm{F}(s', \cdot)$, and the alarm clock is either turned off or newly set (when entering a state of S_{fd}).

In most practical applications mentioned above, fixed-delay transitions are determined by the design of the system and often strongly influence performance of the system. Indeed, both timeouts in network protocols as well as scheduling intervals in real-time systems directly influence performance of the respective systems and their manual setting usually requires considerable effort and expertise. This motivates us to consider the fixed-time delays $\mathbf{d}(s)$ as *free parameters* of the model, and develop techniques for their optimization with respect to a given performance measure.

Example 1. We demonstrate the concept on two different models of sending *one* segment of data in the *alternating bit protocol*. In the protocol, each segment of data is retransmitted until an acknowledgement is received. The delay between retransmissions has impact on throughput of the protocol as well as on network congestion. In the simpler model below on the left, the data is sent in state *init*. The exp-delay transitions, drawn as solid arrows, model message loss (with probability 0.2) and delivery (with probability 0.8). For simplicity we use rate 1 and omit self loops of exponential transitions in all examples. The fixed-delay transitions, drawn as dashed arrows, cause the data to be retransmitted. Note that whenever the data is retransmitted, the previous message with the data is canceled in this model.



The more faithful model on the right models up to two messages with the data segment being delivered concurrently. For choosing an optimal delay between retransmissions, we need to formalize how to express performance of the protocol.

To express performance properties, we use standard cost (or reward) structures (see, e.g. [33]) that assign numerical rewards to states and transitions. More precisely, we consider the following three cost functions: $\mathcal{R} : S \to \mathbb{R}_{\geq 0}$, which assigns a cost rate $\mathcal{R}(s)$ to every state s so that the cost $\mathcal{R}(s)$ is paid for every unit of time spent in the state s, and functions $\mathcal{I}_{\mathrm{P}}, \mathcal{I}_{\mathrm{F}} : S \times S \to \mathbb{R}_{\geq 0}$ that assign to each exp-delay and fixed-delay transition, respectively, the cost that is immediately paid when the transition is taken. Note that $\mathcal{R}$ is usually used to express time spent in individual states, while the other two cost functions are used to quantify the difficulty of dealing with events corresponding to transitions. The performance measure itself is the *expected total cost incurred before reaching a given set of states G starting in a given initial state* s_{in}. For this moment, let us denote this measure by $E_{\mathbf{d}}$, stressing the fact that it depends on the delay function $\mathbf{d}$ which is the only variable quantity in our optimization task:

Problem 1 (Cost optimization). For a subspace of delay functions $D \subseteq (\mathbb{R}_{>0})^{S_{\mathrm{fd}}}$ and a given approximation error $\varepsilon > 0$, compute a delay function $\mathbf{d} \in D$ that is *ε-optimal within D*, i.e.

$$\left| \inf_{\mathbf{d}' \in D} E_{\mathbf{d}'} - E_{\mathbf{d}} \right| \quad < \quad \varepsilon.$$

Example 1 (cont.) We can model the expected cost of sending one data segment in our examples as follows. To take into account the expected time of data delivery, we set the cost rate of each state to, e.g., 1. To take into account the expected number of retransmissions, we set the cost of each fixed-delay transition, e.g., to 3. The cost of each exp-delay transition is set to 0. Now the goal for the left model is to find a delay $\mathbf{d}(init)$ optimizing the expected total cost incurred before reaching the state OK. Note that $\mathbf{d}$ is never set in the state $lost$. The goal is the same for the model on the right where $\mathbf{d}$ is set also in the state two. Note that it makes no sense to synthesize different delays $\mathbf{d}(init)$ and $\mathbf{d}(two)$ as the states $init$ and two are indistinguishable in the implementation of the protocol. Therefore, we need to require that the synthesised delay function satisfies $\mathbf{d}(init) = \mathbf{d}(two)$.

Our contribution: We consider fixed-delay CTMC as a natural extension of CTMC suitable for algorithmic synthesis of fixed timeouts. Upon this model, we investigate algorithmic complexity of the cost optimization problem. This is, to the best of our knowledge, the most general attempt at fully automatic synthesis of timeouts in continuous-time stochastic systems. We provide algorithms for solving the following two special cases of the cost optimization problem under the assumption that the reward rate $\mathcal{R}(s)$ is *positive* in every state s:

1. **Unconstrained optimization** where we demand $D = (\mathbb{R}_{>0})^{S_{\mathrm{fd}}}$, i.e. the set of all delay functions. We solve this problem by reduction to a finite Markov decision process (MDP) whose actions correspond to *discretized* (i.e. rounded onto a finite mesh) values of delays in the individual states, and then apply standard polynomial time algorithms for synthesis of the delays (note that a brute force search through a "discretized" subset of D would be exponentially worse). The most non-trivial part is to prove

that the delays may be discretized. We show that a naïve rounding of a near-optimal delay function may cause arbitrarily high *absolute* error. Our solution, based on rather non-trivial insights into the structure of fdCTMCs, avoids this obstacle by identifying "safe" delay functions that may be rounded with an error bounded (exponentially) in the size of the system. This leads to an exponential time algorithm for solving the cost optimization problem.

2. **Bounded optimization under partial observation** where we introduce bounds $\underline{d}, \overline{d} > 0$ together with an equivalence relation $\equiv$ on S_{fd} and demand D to be the set of all delay functions $\mathbf{d}$ satisfying the following conditions:
   - $\underline{d} \leq \mathbf{d}(s) \leq \overline{d}$ for all $s \in S_{\mathrm{fd}}$,
   - $\mathbf{d}(s) = \mathbf{d}(s')$ whenever $s \equiv s'$.

   Like in the Example 1, the equivalence $\equiv$ can be used to hide information about detailed internal structure of states which is often needed in practical applications. In this paper, we show that the bounded optimization under partial observation can be solved in time doubly exponential in $\overline{d}$ and exponential in all other parameters.

   We also consider the corresponding approximate threshold variant: For a given x decide whether $\inf_{\mathbf{d}' \in D} E_{\mathbf{d}'} > x + \varepsilon$, or $\inf_{\mathbf{d}' \in D} E_{\mathbf{d}'} < x - \varepsilon$ (for $\inf_{\mathbf{d}' \in D} E_{\mathbf{d}'} \in [x - \varepsilon, x + \varepsilon]$ an arbitrary answer may be given). We show that this bounded optimization problem is NP-hard, thus a polynomial time solution of the bounded optimization under partial observation is unlikely.

   The assumption that all delays are between fixed thresholds $\underline{d}$ and $\overline{d}$ is crucial in our approach. As we discuss in Section 4, without this assumption the optimization under partial observation becomes much trickier and we leave its solution for future work.

Related work. Various forms of continuous-time stochastic processes with fixed-delay transitions have already been studied, see e.g. [28, 13, 1, 9, 4]. In particular, as noted above, our definition of fdCTMC is closely related to the original definition of deterministic and stochastic Petri nets [28]. Papers on verification of continuous-time systems with timed automata (TA) specifications [12, 4, 5] are also related to our work as the constraints in timed automata resemble fixed-delay transitions. None of these works, however, considers synthesis of fixed-delays (or other parameters).

Parameter synthesis techniques have been developed for several models, such as parametric timed automata [2], parametric one-counter automata [17], parametric Markov models [18], etc. In continuous-time stochastic systems, [19, 22] study synthesis of rates in CTMC which is a problem orthogonal to timeouts. Furthermore, optimal control of continuous-time (Semi)-Markov decision processes [29, 10, 8, 6] can be viewed as synthesis of *discrete* parameters in continuous-time systems.

The problem of synthesizing timeouts as *continuous* parameters has been studied in variety of engineering contexts such as vehicle communication systems [24] and avionic subsystems [3, 35]. To the best of our knowledge, no generic framework for synthesis of timeouts in stochastic continuous-time systems has been developed so far. In theoretical literature, only simpler cases have been addressed. For instance [11, 36] consider a finite test case, a sequence of input and output actions, and synthesize times for input actions that maximise the probability of executing this acyclic sequence. Allowing cycles in fdCTMC makes the timeout synthesis problem much more demanding, e.g., due to potentially unbounded number of stochastic events between timeouts. Instead of static timeouts, [26, 21] consider synthesis of "dynamic" timeouts where the delay is chosen based on the history of the execution so far. Consequently, the delay can be changed *while it is elapsing* whenever stochastic events occur. This makes it much simpler to solve and also adequate for a different application domain.

Section 2 introduces fixed-delay CTMC and cost structures. Section 3 and Section 4 are devoted to unconstrained optimization and bounded optimization under partial observation, respectively. Due to space constraints, full proofs are in [7].

2 Preliminaries

We use $\mathbb{N}_0$, $\mathbb{R}_{\geq 0}$, and $\mathbb{R}_{>0}$ to denote the set of all non-negative integers, non-negative real numbers, and positive real numbers, respectively. Furthermore, for a countable set A, we denote by $\mathcal{D}(A)$ the set of discrete probability distributions over A, i.e. functions $\mu : A \to \mathbb{R}_{\geq 0}$ such that $\sum_{a \in A} \mu(a) = 1$. Encoding size of an object O is denoted by $\|O\|$.

Definition 1. *A* fixed-delay CTMC structure *(fdCTMC structure) C is a tuple $(S, \lambda, \mathrm{P}, S_{\mathrm{fd}}, \mathrm{F}, s_{in})$ where*

- *S is a finite set of states,*
- *$\lambda \in \mathbb{R}_{>0}$ is a (common) rate of exp-delay transitions,*
- *$\mathrm{P} : S \times S \to \mathbb{R}_{\geq 0}$ is a stochastic matrix specifying probabilities of exp-delay transitions,*
- *$S_{\mathrm{fd}} \subseteq S$ is a set of states where fixed-delay transitions are enabled,*
- *$\mathrm{F} : S_{\mathrm{fd}} \times S \to \mathbb{R}_{\geq 0}$ is a stochastic matrix specifying probabilities of fixed-delay transitions, and*
- *$s_{in} \in S$ is an initial state.*

A fixed-delay CTMC *(fdCTMC) is a pair $C(\mathbf{d}) = (C, \mathbf{d})$ where C is a fdCTMC structure and $\mathbf{d} : S_{\mathrm{fd}} \to \mathbb{R}_{>0}$ is a* delay function *which to every state where fixed-delay transitions are enabled assigns a positive delay.*

A *configuration* of a fdCTMC is a pair (s, d) where $s \in S$ is the current state and $d \in \mathbb{R}_{>0} \cup \{\infty\}$ is the remaining time before a fixed-delay transition takes place. We assume that $d = \infty$ iff $s \notin S_{\mathrm{fd}}$. To simplify notation, we similarly extend any delay function $\mathbf{d}$ to all states S by assuming $\mathbf{d}(s) = \infty$ iff $s \notin S_{\mathrm{fd}}$.

An execution of $C(\mathbf{d})$ starts in the configuration (s_0, d_0) with $s_0 = s_{in}$ and $d_0 = \mathbf{d}(s_{in})$. In every step, assuming that the current configuration is (s_i, d_i), the fdCTMC waits for some time t_i and then moves to a next cofiguration (s_{i+1}, d_{i+1}) determined as follows:

- First, a waiting time t_{exp} for exp-delay transitions from s_i is chosen randomly according to the exponential distribution with the rate λ.
- Then
  - If $t_{exp} < d_i$, then an exp-delay transition occurs, which means that $t_i = t_{exp}$, s_{i+1} is chosen randomly with probability $P(s_i, s_{i+1})$, and d_{i+1} is determined by

$$d_{i+1} = \begin{cases} d_i - t_{exp} & \text{if } s_{i+1} \in S_{\mathrm{fd}} \text{ and } s_i \in S_{\mathrm{fd}} \text{ (previous delay remains)}, \\ \mathbf{d}(s_{i+1}) & \text{if } s_{i+1} \notin S_{\mathrm{fd}} \text{ or } s_i \notin S_{\mathrm{fd}} \text{ (delay is newly set or disabled)}. \end{cases}$$

  - If $t_{exp} \geq d_i$, then a fixed-delay transition occurs, which means that $t_i = d_i$, s_{i+1} is chosen randomly with probability $F(s_i, s_{i+1})$, and $d_{i+1} = \mathbf{d}(s_{i+1})$.

This way, the execution of a fdCTMC forms a *run*, an alternating sequence of configurations and times $(s_0, d_0)t_0(s_1, d_1)t_1 \cdots$. The probability measure $\mathrm{Pr}_{C(\mathbf{d})}$ over runs of $C(\mathbf{d})$ is formally defined in [7].

Total cost before reaching a goal To allow formalization of performance properties, we enrich the model in a standard way (see, e.g. [33]) with costs (or rewards). A *cost structure* over a fdCTMC structure C with state space S is a tuple $Cost = (G, \mathcal{R}, \mathcal{I}_{\mathrm{P}}, \mathcal{I}_{\mathrm{F}})$ where $G \subseteq S$ is a set of goal states, $\mathcal{R} : S \to \mathbb{R}_{\geq 0}$ assigns a cost rate to every state, and $\mathcal{I}_{\mathrm{P}}, \mathcal{I}_{\mathrm{F}} : S \times S \to \mathbb{R}_{\geq 0}$ assign an impulse cost to every exp-delay and fixed-delay transition, respectively. Slightly abusing the notation, we denote by *Cost* also the random variable assigning to each run $\omega = (s_0, d_0)t_0 \cdots$ the *total cost before reaching G* (in at least one transition), given by

$$Cost(\omega) = \begin{cases} \sum_{i=0}^{n-1} (t_i \cdot \mathcal{R}(s_i) + \mathcal{I}_i(\omega)) & \text{for minimal } n > 0 \text{ such that } s_n \in G, \\ \infty & \text{if there is no such } n, \end{cases}$$

where $\mathcal{I}_i(\omega)$ equals $\mathcal{I}_{\mathrm{P}}(s_i, s_{i+1})$ for an exp-delay transition, i.e. when $t_i < d_i$, and equals $\mathcal{I}_{\mathrm{F}}(s_i, s_{i+1})$ for a fixed-delay transition, i.e. when $t_i = d_i$.

We denote the expectation of *Cost* with respect to $\mathrm{Pr}_{C(\mathbf{d})}$ simply by $E_{C(\mathbf{d})}$, or by $E^{Cost}_{C(\mathbf{d})}$ when *Cost* is not clear from context. Our aim is to (approximatively) minimize the expected cost, i.e. to find a delay function $\mathbf{d}$ such that $E_{C(\mathbf{d})} \leq Val[C] + \varepsilon$ where $Val[C]$ denotes the optimal cost $\inf_{\mathbf{d}'} E_{C(\mathbf{d}')}$.

Non-parametric analysis Due to [27], we can easily analyze a fdCTMC where the delay function is *fixed*. Hence, both the expected total cost before reaching G and the reaching probabilities of states in G can be efficiently approximated.

Proposition 1. *There is a polynomial-time algorithm that for a given fdCTMC $C(\mathbf{d})$, cost structure Cost with goal states G, and an approximation error $\varepsilon > 0$ computes $x \in \mathbb{R}_{>0} \cup \{\infty\}$ and $p_s \in \mathbb{R}_{>0}$, for each $s \in G$, such that*

$$\left| E_{C(\mathbf{d})} - x \right| < \varepsilon \quad \text{and} \qquad \left| \mathrm{Pr}_{C(\mathbf{d})}(\Diamond^s_G) - p_s \right| < \varepsilon$$

where $\Diamond^s_G$ is the set of runs that reach s as the first state of G (after at least one transition).

Markov decision processes. In Section 3 we use a reduction of fdCTMC to discrete-time Markov decision processes (DTMDP, see e.g. [33]) with uncountable space of actions.

Definition 2. *A* DTMDP *is a tuple $\mathcal{M} = (V, Act, T, v_{in}, V')$, where V is a finite set of* vertices*, Act is a (possibly uncountable) set of* actions*, $T : V \times Act \to \mathcal{D}(V) \cup \{\perp\}$ is a* transition function*, $v_{in} \in V$ is an initial vertex, and $V' \subseteq V$ is a set of* goal vertices.

An action a is *enabled* in a vertex v if $T(v, a) \neq \perp$. A *strategy* is a function $\sigma : V \to Act$ which assigns to every vertex v an action enabled in v. The behaviour of $\mathcal{M}$ with a fixed strategy σ can be intuitively described as follows: A run starts in the vertex v_{in}. In every step, assuming that the current vertex is v, the process moves to a new vertex v' with probability $T(v, \sigma(v))(v')$. Every strategy σ uniquely determines a probability measure $\mathrm{Pr}_{\mathcal{M}(\sigma)}$ on the set of *runs*, i.e. infinite alternating sequences of vertices and actions $v_0 a_1 v_1 a_2 v_2 \cdots \in (V \cdot Act)^\omega$; see [7] for details.

Analogously to fdCTMC, we can endow a DTMDP with a *cost function* $A : V \times Act \to \mathbb{R}_{\geq 0}$. We then define for each run $v_0 a_1 v_1 a_2 \ldots$, the *total cost incurred before reaching V'* as $\sum_{i=0}^{n-1} A(v_i, a_{i+1})$ if there is a minimal $n > 0$ such that $v_n \in V'$, and as ∞ otherwise. The expectation of this cost w.r.t. $\Pr_{\mathcal{M}(\sigma)}$ is similarly denoted by $E_{\mathcal{M}(\sigma)}$ or by $E^{A}_{\mathcal{M}(\sigma)}$ if the cost function is not clear from context.

Given $\varepsilon \geq 0$, we say that a strategy σ is *ε-optimal* in $\mathcal{M}$ if $E_{\mathcal{M}(\sigma)} \leq \mathit{Val}[\mathcal{M}] + \varepsilon$ where $\mathit{Val}[\mathcal{M}] = \inf_{\sigma'} E_{\mathcal{M}(\sigma')}$; we call it *optimal* if it is 0-optimal. For any $s \in S$, let us denote by $\mathcal{M}[s]$ the DTMDP obtained from $\mathcal{M}$ by replacing the initial state by s. We call a strategy *globally (ε-)optimal* if it is (ε-)optimal in $\mathcal{M}[s]$ for every $s \in S$. Sometimes, we restrict to a subset D of all strategies and denote by $\mathit{Val}[\mathcal{M}, D]$ the restricted infimum $\inf_{\sigma' \in D} E_{\mathcal{M}(\sigma')}$.

3 Unconstrained Optimization

Theorem 1. *There is an algorithm that given a fdCTMC structure C, a cost structure Cost with $\mathcal{R}(s) > 0$ for all $s \in S$, and $\varepsilon > 0$ computes in exponential time a delay function* $\mathbf{d}$ *with*

$$\left| E_{C(\mathbf{d})} - \inf_{\mathbf{d}'} E_{C(\mathbf{d}')} \right| < \varepsilon.$$

The rest of this section is devoted to a proof of Theorem 1, which consists of two parts. First, we reduce the optimization problem in the fdCTMC to an optimization problem in a *discrete time* Markov decision process (DTMDP) with uncountably many actions. Second, we present the actual approximation algorithm based on a straightforward discretization of the space of actions of the DTMDP. However, the proof of its error bound is actually quite intricate. The time complexity is exponential because the discretized DTMDP needs exponential size in the worst case. We provide more detailed complexity analysis with respect to various parameters in [7].

For the rest of this section we fix a fdCTMC structure $C = (S, \lambda, \mathrm{P}, S_{\mathrm{fd}}, \mathrm{F}, s_{in})$, a cost structure $\mathit{Cost} = (G, \mathcal{R}, \mathcal{I}_{\mathrm{P}}, \mathcal{I}_{\mathrm{F}})$, and $\varepsilon > 0$. We assume that $\mathit{Val}[C] < \infty$. The opposite case can be easily detected by fixing an arbitrary delay function $\mathbf{d}$ and finding out whether $E_{C(\mathbf{d})} = \infty$ by Proposition 1. This is equivalent to $\mathit{Val}[C] = \infty$ by the following observation.

Lemma 1. *For any delay functions* $\mathbf{d}, \mathbf{d}'$ *we have $E_{C(\mathbf{d})} = \infty$ if and only if $E_{C(\mathbf{d}')} = \infty$.*

To further simplify our presentation, we assume that each state s of S_{fd} directly encodes whether the delay needs to be reset upon entering s. Formally, we assume $S_{\mathrm{fd}} = S^{reset} \uplus S^{keep}$ where $s' \in S^{reset}$ if $P(s, s') > 0$ for some $s \in S \setminus S_{\mathrm{fd}}$ or if $F(s, s') > 0$ for some $s \in S_{\mathrm{fd}}$; and $s' \in S^{keep}$ if $P(s, s') > 0$ for some $s \in S_{\mathrm{fd}}$. We furthermore assume that $s_{in} \in S^{reset}$ if $s_{in} \in S_{\mathrm{fd}}$. Note that each fdCTMC structure can be easily transformed to satisfy this assumption in polynomial time by duplication of states S_{fd}, see, e.g., Example 2.

3.1 Reduction to DTMDP $\mathcal{M}$ with Uncountable Space of Actions.

We reduce the problem into a *discrete-time* problem by capturing the evolution of the fdCTMC only at *discrete moments* when a transition is taken after which the fixed-delay is (a) newly set, or (b) switched off, or (c) irrelevant as the goal set is reached. This happens exactly when one of the states of $S' = S^{reset} \cup (S \setminus S_{\mathrm{fd}}) \cup G$ is reached. We define a DTMDP $\mathcal{M} = (S', \mathit{Act}, T, s_{in}, G)$ with a cost function $€$:

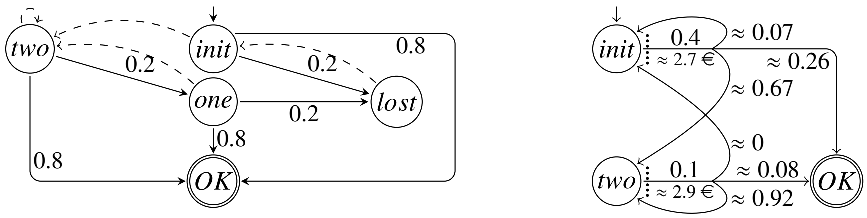
- $\mathit{Act} := \mathbb{R}_{>0} \cup \{\infty\}$; where actions $\mathbb{R}_{>0}$ are enabled in $s \in S^{reset}$ and action ∞ is enabled in $s \in S \setminus S_{\mathrm{fd}}$.

– Let $s \in S'$ and d be an action of $\mathcal{M}$. Intuitively, we define $T(s,d)$ and $€(s,d)$ to summarize the behaviour of the fdCTMC starting in the configuration (s,d) until the first moment when S' is reached again.
Formally, let $C[s](d)$ denote a fdCTMC obtained from C by changing initial state to s and fixing a delay function that assigns d to s (and arbitrary values elsewhere). We define $€(s,d)$ as the cost accumulated before reaching another state of S' and $T(s,d)(s')$ as the probability that s' is the first such a reached state of S'. That is,

$$€(s,d) = E_{C[s](d)}^{Cost[S']} \qquad \text{and} \qquad T(s,d)(s') = \Pr_{C[s](d)}(\Diamond_{S'}^{s'})$$

where $Cost[S']$ is obtained from $Cost$ by changing the set of goal states to S'. Note that the definition is correct as it does not depend on the delay function apart from its value d in the initial state s.

Example 2. Let us illustrate the construction on the fdCTMC from Section 1. The model, depicted on the left is modified to satisfy the assumption $S_{\text{fd}} = S^{reset} \uplus S^{keep}$: we duplicate the state *init* into another state $one \in S^{keep}$; the states from S^{reset} are then depicted in the top row. As in Section 1, we assign cost rate 1 to all states and impulse cost 3 to every fixed-delay transition (and zero to exp-delay transitions).



On the right, there is an excerpt of the DTMDP $\mathcal{M}$ and of the cost function €. For each non-goal state, we depict only one action out of uncountably many: for state *two* it is action 0.1 with cost ≈ 2.9, for state *init* it is action 0.4 with cost ≈ 2.7. The costs are computed in PRISM. □

Note that there is a one-to-one correspondence between the delay functions in C and strategies in $\mathcal{M}$. Thus we use $\mathbf{d}, \mathbf{d}', \ldots$ to denote strategies in $\mathcal{M}$. Finally, let us state correctness of the reduction.

Proposition 2. *For any delay function* $\mathbf{d}$ *it holds* $E_{C(\mathbf{d})} = E_{\mathcal{M}(\mathbf{d})}$. *Hence,*

$$Val[C] = Val[\mathcal{M}]\,.$$

In particular, in order to solve the optimization problem for C it suffices to find an ε-optimal strategy (i.e., a delay function) $\mathbf{d}$ in $\mathcal{M}$.

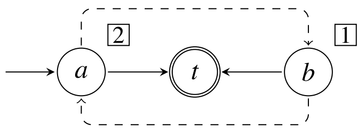
3.2 Discretization of the Uncountable MDP $\mathcal{M}$

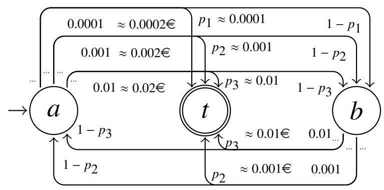
Since the MDP $\mathcal{M}$ has uncountably many actions, it is not directly suitable for algorithmic solutions. We proceed in two steps. In the first and technically demanding step, we show that we can restrict to actions on a finite mesh. Second, we argue that we can also approximate all transition probabilities and costs by rational numbers of small bit length.

Restricting to a Finite Mesh. For positive reals $\delta > 0$ and $\overline{d} > 0$, we define a subset of delay functions $D(\delta, \overline{d}) = \{\mathbf{d} \mid \forall s \in S' \ \exists k \in \mathbb{N} : \mathbf{d}(s) = k\delta \leq \overline{d}\}$. Here, all delays are multiples of δ bounded by $\overline{d}$.

We need to argue that for the fixed $\varepsilon > 0$ there are some appropriate values δ and $\overline{d}$ such that $D(\delta, \overline{d})$ contains an ε-optimal delay function. A naïve approach would be to take any, say $\varepsilon/2$-optimal delay function $\mathbf{d}$, round it to closest delay function $\mathbf{d}^\star \in D(\delta, \overline{d})$ on the mesh, and show that the expected costs of these two functions do not differ by more than $\varepsilon/2$. However, this approach does not work as shown by the following example.

Example 3. Let us fix the fdCTMC structure C on the left (with cost rates in small boxes and zero impulse costs). An excerpt of $\mathcal{M}$ and € is shown on the right (where only a few actions are depicted).





First, we point out that $Val\,[C] = 1$ as one can make sure that nearly all time before reaching t is spent in the state b that has a lower cost rate 1. Indeed, this is achieved by setting a very short delay in a and a long delay in b.

We claim that for any $\delta > 0$ there is a near-optimal delay function $\mathbf{d}$ such that rounding its components to the nearest integer multiples of δ yields a large error independent of δ. Indeed, it suffices to take a function $\mathbf{d}$ with $\mathbf{d}(b) = \delta$ and $\mathbf{d}(a)$ an arbitrary number significantly smaller than $\mathbf{d}(b)$, say $\mathbf{d}(a) = 0.01 \cdot \mathbf{d}(b)$. The error produced by the rounding can then be close to 0.5. For instance, given $\delta = 0.01$ we take a function with $\mathbf{d}(a) = 0.0001$ and $\mathbf{d}(b) = 0.01$, whose rounding to the closest delay function on the finite mesh yields a constant function $\mathbf{d}^\star = (0.01, 0.01)$. Then $E_{C(\mathbf{d})} \approx 1.01$ and $E_{C(\mathbf{d}^\star)} \approx 1.5$, even though the rounding does not change any transition probability or cost by more than 0.02!

The reason why the delay function $\mathbf{d}$ is so sensitive to small perturbations is that it makes a very large number of steps before reaching t (around 200 on average) and thus the small one-step errors caused by a perturbation accumulate into a large global error. The number of steps of an ε-optimal delay functions is not bounded, in general. By multiplying both $\mathbf{d}(a)$ and $\mathbf{d}(b)$ by the same infinitesimally small factors we obtain an ε-optimal delay functions that make an arbitrarily high expected number of steps before reaching t. $\square$

A crucial observation is that we do not have to show that the "naïve" rounding works for *every* near-optimal delay function. To prove that $D(\delta, \overline{d})$ contains an ε-optimal function, it suffices to show that there is *some* $\varepsilon/2$-optimal function whose rounding yields error at most $\varepsilon/2$. Proving the existence of such well-behaved functions forms the technical core of our discretization process which is formalized below.

We start by formalizing the concept of "small perturbations". We say that a delay function $\mathbf{d}^\star$ is *α-bounded* by a delay function $\mathbf{d}$ if for all states $s, t \in S'$ we have:

1. $|T(s, \mathbf{d}^\star(s))(t) - T(s, \mathbf{d}(s))(t)| \leq \alpha$ and
2. $\text{€}(s, \mathbf{d}^\star(s)) - \text{€}(s, \mathbf{d}(s)) \leq \alpha$;

and furthermore, $T(s, \mathbf{d}(s))(t) = 0$ iff $T(s, \mathbf{d}^\star(s))(t) = 0$, i.e. the qualitative transition structure is preserved. (Note that $\mathbf{d}^\star$ may incur much smaller one-step costs than $\mathbf{d}$, but not significantly higher).

Using standard techniques of numerical analysis, we express the increase in accumulated cost caused by a bounded perturbation as a function of the worst-case (among all possible initial states) expected cost and expected number of steps before reaching the target. The number of steps is essential as discussed in Example 3 and can be easily expressed by a cost function # that assigns 1 to every action in every state. To express the worst-case expectation of some cost function \$, we denote $\mathit{Bound}[\$, \mathbf{d}] := \max_{s \in S'} E^{\$}_{\mathcal{M}[s](\mathbf{d})}$.

Lemma 2. *Let $\alpha \in [0, 1]$ and let, $\mathbf{d}'$ be a delay function that is α-bounded by another delay function $\mathbf{d}$. If $\alpha \leq \frac{1}{2 \cdot \mathit{Bound}[€, \mathbf{d}] \cdot |S'|}$, then*

$$E_{\mathcal{M}(\mathbf{d}')} \leq E_{\mathcal{M}(\mathbf{d})} + 2 \cdot \alpha \cdot \mathit{Bound}[\#, \mathbf{d}] \cdot (1 + \mathit{Bound}[€, \mathbf{d}] \cdot |S'|).$$

The next lemma shows how to set the parameters δ and $\overline{d}$ to make the finite mesh $D(\delta, \overline{d})$ "dense" enough, i.e. to ensure that for any $\mathbf{d}$, $D(\delta, \overline{d})$ contains a delay function that is α-bounded by $\mathbf{d}$.

Lemma 3. *There are positive numbers $D_1, D_2 \in \exp(||C||^{O(1)})$ computable in time polynomial in $||C||$ such that the following holds for any $\alpha \in [0, 1]$ and any delay function $\mathbf{d}$: If we put*

$$\delta := \alpha / D_1 \quad \textit{and} \quad \overline{d} := |\log(\alpha)| \cdot D_2 \cdot \mathit{Bound}[€, \mathbf{d}],$$

then $D(\delta, \overline{d})$ contains a delay function which is α-bounded by $\mathbf{d}$.

Proof (Sketch). Computing the value of δ is easy as the derivatives of the probabilities and costs are bounded from above by the rate λ and the maximal cost rate, respectively. For $\overline{d}$ we need additional technical observations, see [7] for further details.

Unfortunately, as shown in Example 3, the value $\mathit{Bound}[\#, \mathbf{d}]$ can be arbitrarily high, even for near-optimal functions $\mathbf{d}$. Hence, we cannot use Lemma 2 right away to show that a delay function in $D(\delta, \overline{d})$ that is α-bounded by some near-optimal $\mathbf{d}$ is also near-optimal. The crucial insight is that for any $\varepsilon' > 0$ there are (globally) ε'-optimal delay functions that use number of steps that is *proportional* to their expected cost.

Lemma 4. *There is a positive number $N \in \exp(||C||^{O(1)})$ computable in time polynomial in $||C||$ such that the following holds: for any $\varepsilon' > 0$, there is a globally $\varepsilon'/2$-optimal delay function $\mathbf{d}'$ with*

$$\mathit{Bound}[\#, \mathbf{d}'] \leq \frac{\mathit{Bound}[€, \mathbf{d}']}{\varepsilon'} \cdot N. \tag{1}$$

Proof (Sketch). After proving the existence of globally near-optimal strategies, we suitably define the number N and take an arbitrary globally ε''-optimal delay function $\mathbf{d}''$, where $\varepsilon'' << \varepsilon'$. If this function *does not* satisfy (1), we conclude that it must induce the following pathological behaviour in C: the system stays for a long time in a component of its state space such that a) fixed-delay transitions are active in each state of the component, each such transition within the component having zero impulse cost; and b) function $\mathbf{d}''$ assigns very small (in a well-defined sense) delays to all states of the component. We call such a component a *bad sink*. Intuitively, inside a bad sink the system rapidly performs one fixed-delay transition after another, incurring only a tiny cost between two successive transitions. This allows the delay function to perform many steps while staying ε''-optimal. (In Example 3, $\{a, b\}$ would be a bad sink for $\mathbf{d}$, as with high probability the cycle on these two states is completed every 0.0101 units of time, with cost 0.0102 incurred per cycle.)

To obtain a globally ε'-optimal delay function satisfying (1), we carefully modify $\mathbf{d}''$ so as to remove all bad sinks. This is done by selecting a suitable state in each bad sink and "inflating" its delay to a sufficiently high threshold. Choosing the right state and threshold is a rather delicate process, since an improper choice might significantly increase the incurred cost. Also note that Lemma 2 cannot be used to bound the increase in cost caused by the modification, as we do not know the value of $\mathit{Bound}[\#, \mathbf{d}'']$. Instead, we utilize non-trivial insights into the structure of C and $\mathcal{M}$. $\square$

By using these proportional delay functions, we reduce the perturbation error of Lemma 2 only to a function of $\mathit{Bound}[€, \mathbf{d}]$. Combining this with Lemma 3, we obtain that the delay functions in $D(\delta, \overline{d})$ approximate all the proportional delay functions $\mathbf{d}$ of Lemma 4, and thus $\mathit{Val}\left[\mathcal{M}, D(\delta, \overline{d})\right]$ approximates $\mathit{Val}[\mathcal{M}]$. The parameters $\delta, \overline{d}$ depend on ε and $\mathit{Bound}[€, \mathbf{d}]$ of any such $\mathbf{d}$ from Lemma 4. As these delay functions are *globally* ε-optimal, all such $\mathit{Bound}[€, \mathbf{d}]$ can be ε-approximated by $\overline{\mathit{Val}}[\mathcal{M}] := \max_{s \in S'} \mathit{Val}[\mathcal{M}[s]]$.

Proposition 3. *For N from Lemma 4, D_1 and D_2 from Lemma 3, it holds that*

$$\left| \mathit{Val}[\mathcal{M}] - \mathit{Val}\left[\mathcal{M}, D(\delta, \overline{d})\right] \right| \leq \frac{\varepsilon}{2}$$

where $\delta := \dfrac{\alpha}{D_1}, \ \overline{d} := |\log(\alpha)| \cdot D_2 \cdot (\overline{\mathit{Val}}[\mathcal{M}] + \varepsilon), \ \alpha := \dfrac{\varepsilon^2}{64N \cdot |S'| \cdot (1 + \overline{\mathit{Val}}[\mathcal{M}])^2}.$

Bounding $\overline{\mathit{Val}}[\mathcal{M}]$ In Proposition 3, the allowed perturbation α and hence the fineness of the mesh δ needed to obtain the required precision depend on the bound $\overline{\mathit{Val}}[\mathcal{M}]$. We first provide the following theoretical worst-case bound.

Lemma 5. *There is a number $M \in \exp(\|C\|^{O(1)})$ computable in time polynomial in $\|C\|$ such that $\overline{\mathit{Val}}[\mathcal{M}] \leq M$.*

In practice, one can obtain better bounds by computing $\max_{s \in S} E_{C[s](\mathbf{d})}$ for an arbitrary $\mathbf{d}$ as $\max_{s \in S} E_{C[s](\mathbf{d})} \geq \max_{s \in S} \inf_{\mathbf{d}'} E_{C[s](\mathbf{d}')} = \overline{\mathit{Val}}[\mathcal{M}]$. One can set $\mathbf{d}$ by some heuristics (e.g. to the constant function $1/\lambda$) or randomly. One can even use the minimum from a series of such computations. We believe that in most cases, this yields a significant improvement. For instance, for the 3-state model from Section 1, we get a bound $\max_{s \in S} E_{C[s](1/\lambda)} \approx 4.3$ instead of the theoretical bound $\overline{\mathit{Val}}[\mathcal{M}] \approx 55000$.

Representing the Finite Mesh Since one-step costs and probabilities produced by delay functions in $D(\delta, \overline{d})$ may be irrational, we need to approximate them by rational numbers. So let us fix δ and $\overline{d}$ from Proposition 3. For any $\kappa > 0$ we define DTMDP $\mathcal{M}_\kappa = (S', \mathit{Act}_\kappa, T_\kappa, s_{in}, G)$ with a cost function $€_\kappa$ where

- the strategies are exactly delay functions from $D(\delta, \overline{d})$, i.e. $\mathit{Act}_\kappa = \{k\delta \mid k \in \mathbb{N}, \delta \leq k\delta \leq \overline{d}\} \cup \{\infty\}$ where again ∞ is enabled in $s \in S' \setminus S_{\mathrm{fd}}$ and the rest is enabled in $s \in S_{\mathrm{fd}}$; and

– for all $(s, \mathbf{d}) \in S' \times Act_\kappa$ the transition probabilities in $T_\kappa(s, \mathbf{d})$ and costs in $€_\kappa(s, \mathbf{d})$ are obtained by rounding the corresponding numbers in $T(s, \mathbf{d})$ and $€(s, \mathbf{d})$ up (using the algorithm of Proposition 1) to the closest multiple of κ.[4]

Proposition 4. *Let $\varepsilon > 0$ and fix $\kappa = (\varepsilon \cdot \delta \cdot minR)/(2 \cdot |S'| \cdot (1 + \overline{Val}[\mathcal{M}])^2)$, where minR is a minimal cost rate in C. Then it holds*

$$\left| Val\left[\mathcal{M}, D(\delta, \overline{d})\right] - Val[\mathcal{M}_\kappa] \right| \leq \frac{\varepsilon}{2}.$$

Proof (Sketch). We use similar technique as in Lemma 2, taking advantage of the fact that probabilities and costs of each action are changed by at most κ by the rounding. □

The Algorithm for Theorem 1 First the discretization step δ, maximal delay $\overline{d}$, and rounding error κ are computed. Then the discretized DTMDP $\mathcal{M}_\kappa$ is constructed according to the above-mentioned finite mesh representation. Finally the globally optimal delay function from $\mathcal{M}_\kappa$ is chosen using standard polynomial algorithms for finite MDPs [33, 14]. From Propositions 3 and 4 it follows that this delay function is ε-optimal in $\mathcal{M}$, and thus also in C (Proposition 2).

The size of $\mathcal{M}_\kappa$ (and its construction time) can be stated in terms of a polynomial in $||C||$, $\overline{Val}[\mathcal{M}]$, $1/\delta$, $\overline{d}$, and $1/\kappa$. Examining the definitions of these parameters in Propositions 3 and 4, as well as the bound on $\overline{Val}[\mathcal{M}]$ from Lemma 5, we conclude that the size of $\mathcal{M}_\kappa$ and the overall running time of our algorithm are exponential in $||C||$ and polynomial in $1/\varepsilon$. The pseudo-code of the whole algorithm is given in [7].

4 Bounded Optimization Under Partial Observation

In this section, we address the cost optimization problem for delay functions chosen under partial observation. For an equivalence relation $\equiv$ on S_{fd} specifying observations, and $\underline{d}, \overline{d} > 0$, we define $D(\underline{d}, \overline{d}, \equiv) = \{\mathbf{d} \mid \forall s, s' : \underline{d} \leq \mathbf{d}(s) \leq \overline{d}, s \equiv s' \Rightarrow \mathbf{d}(s) = \mathbf{d}(s')\}$.

Theorem 2. *There is an algorithm that for a fdCTMC structure C, a cost structure Cost with $\mathcal{R}(s) > 0$ for all $s \in S$, an equivalence relation $\equiv$ on S_{fd}, $\underline{d}, \overline{d} > 0$, and $\varepsilon > 0$ computes in time exponential in $||C||$, $||\underline{d}||$, and $\overline{d}$ a delay function $\mathbf{d}$ such that*

$$\left| \inf_{\mathbf{d}' \in D(\underline{d}, \overline{d}, \equiv)} E_{C(\mathbf{d}')} - E_{C(\mathbf{d})} \right| < \varepsilon.$$

Also, one cannot hope for polynomial complexity as the corresponding threshold problem is NP-hard, even if we restrict to instances where $\overline{d}$ is of magnitude polynomial in $||C||$.

Theorem 3. *For a fdCTMC structure C, a cost structure Cost with $\mathcal{R}(s) > 0$ for all $s \in S$, an equivalence relation $\equiv$ on S_{fd}, $\underline{d}, \overline{d} > 0$, $\varepsilon > 0$, and $x \in \mathbb{R}_{\geq 0}$, it is NP-hard to decide*

$$\text{whether} \quad \inf_{\mathbf{d} \in D(\underline{d}, \overline{d}, \equiv)} E_{C(\mathbf{d})} > x + \varepsilon \quad \text{or} \quad \inf_{\mathbf{d} \in D(\underline{d}, \overline{d}, \equiv)} E_{C(\mathbf{d})} < x - \varepsilon$$

(if the optimal cost lies in the interval $[x - \varepsilon, x + \varepsilon]$, an arbitrary answer may be given). The problem remains NP-hard even if $\overline{d}$ is given in unary encoding.

[4] More precisely, all but the largest probability in $T(s, \mathbf{d})$ are rounded up, the largest probability is suitably rounded down so that the resulting vector adds up to 1.

For $\overline{d}$ given in unary we get a matching upper bound.

Theorem 4. *The approximate threshold problem of Theorem 3 is in NP, provided that $\overline{d}$ is given in unary.*

We leave the task of settling the exact complexity of the general problem (where $\overline{d}$ is given in binary) to future work.

For the rest of this section we fix a fdCTMC structure $C = (S, \lambda, \mathrm{P}, S_{\mathrm{fd}}, \mathrm{F}, s_{in})$, a cost structure $Cost = (G, \mathcal{R}, \mathcal{I}_{\mathrm{P}}, \mathcal{I}_{\mathrm{F}})$, $\varepsilon > 0$, and an equivalence relation $\equiv$ on S_{fd}, $\underline{d}, \overline{d} > 0$. We simply write D instead of $D(\underline{d}, \overline{d}, \equiv)$ and again assume that $Val\,[C, D] < \infty$.

4.1 Approximation Algorithm

In this Section, we address Theorem 2. First observe, that the MDP $\mathcal{M}$ introduced in Section 3 can be due to Proposition 2 also applied in the bounded partial-observation setting. Indeed, $E_{C(\mathbf{d})} = E_{\mathcal{M}(\mathbf{d})}$ for each $\mathbf{d} \in D$ and thus, $Val\,[C, D] = Val\,[\mathcal{M}, D]$ (where analogously $Val\,[C, D]$ denotes $\inf_{\mathbf{d} \in D} E_{C(\mathbf{d})}$). Furthermore, by fixing a mesh δ and a round-off error κ, we define a finite DTMDP $\mathcal{M}^{\star}_{D}$ where

- actions are restricted to a finite mesh of multiples of δ within the bounds $\underline{d}$ and $\overline{d}$; and
- probabilities and costs are rounded to multiples of κ as in Section 3.

To show that $\mathcal{M}^{\star}_{D}$ suitably approximates $\mathcal{M}$ we use similar techniques as in Section 3. However, thanks to the constraints $\underline{d}$ and $\overline{d}$ we can show that for *every* delay function $\mathbf{d} \in D$ the values $Bound\,[\#, \mathbf{d}]$ and $Bound\,[€, \mathbf{d}]$, which feature in Lemma 2, are bounded by a function of $\|C\|$, $\underline{d}$ and $\overline{d}$ (in particular, the bound is independent of $\mathbf{d}$). This substantially simplifies the analysis. We state just the final result.

Proposition 5. *There is a number $B \in \exp((\|C\| \cdot \|\underline{d}\| \cdot \overline{d})^{O(1)})$ such that for $\delta = \varepsilon/B$ and $\kappa = (\varepsilon \cdot \delta)/B$ it holds $\left| Val\,[\mathcal{M}, D] - Val\left[\mathcal{M}^{\star}_{D}\right] \right| < \varepsilon$.*

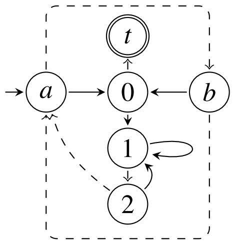
The proof of Theorem 2 is finished by the following algorithm.

- For δ and κ from Proposition 5, the algorithm first constructs (in the same fashion as in Section 3) in 2-exponential time the MDP $\mathcal{M}^{\star}_{D}$.
- Then it finds an optimal strategy $\mathbf{d}$ (which also satisfies $|E_{C(\mathbf{d})} - \inf_{\mathbf{d}'} E_{C(\mathbf{d}')}| < \varepsilon$) by computing $E_{\mathcal{M}'_{\varepsilon}(\mathbf{d})}$ for every (MD) strategy $\mathbf{d}$ of $\mathcal{M}'_{\varepsilon}$ in the set D.

The algorithm runs in 2-EXPTIME because there are $\leq |Act_{\varepsilon}|^{|S|}$ strategies which is exponential in $\|C\|$, $\|\underline{d}\|$, and $\overline{d}$ as $|Act_{\varepsilon}|$ is exponential in these parameters. The correctness follows from Propositions 2, 5, proving Theorem 2.

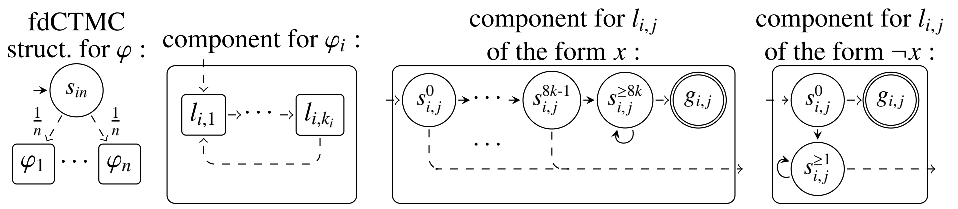
Challenges of Unbounded Optimization The proof of Proposition 5 is simpler than the techniques from Section 3 because we work with the compact space bounded by $\underline{d}$ and $\overline{d}$. This restriction is not easy to lift; the techniques from Section 3 cannot be easily adapted to *unbounded* optimization under partial observation.

The reason is that local adaptation of the delay function (heavily applied in the proof of Lemma 4) is not possible as the delays are not independent. Consider on the right a variant of Example 3 with components a and b being switched by fixed-delay transitions. All states have cost rate 1 and all transitions have cost 0; furthermore, all states are in one class of equivalence of $\equiv$. If in state a or b more than one exp-delay transition is taken before a fixed-delay transition, a long detour via state 1 is taken. In order to avoid it and to optimize the cost, one needs to set the one common delay as close as possible to 0. Contrarily, in order to decrease the expected number of visits from a to b from a before reaching t which is crucial for the error bound, one needs to increase the delay.



4.2 Complexity of the Threshold Problem

We now turn our attention to Theorem 3. We show the hardness by reduction from SAT. Let $\varphi = \varphi_1 \wedge \cdots \wedge \varphi_n$ be a propositional formula in conjunctive normal form (CNF) with $\varphi_i = (l_{i,1} \vee \cdots \vee l_{i,k_i})$ for each $1 \le i \le n$ and with the total number of literals $k = \sum_{i=1}^{n} k_i$. As depicted in the following figure, the fdCTMC structure C_φ is composed of n components (one per clause), depicted by rectangles. The component of each clause is formed by a cycle of sub-components (one per literal) connected by fixed-delay transitions. Positive literals are modelled differently from negative literals.



The cost structure $Cost_\varphi$ assigns rate cost 1 to every state, and impulse cost 0 to every transition; the goal states are depicted by double circles and exp-delay transitions are depicted with heavier heads to distinguish from the dashed fixed-delay transitions. We require $s^0_{i,j} \equiv s^0_{i',j'}$ iff the literals $l_{i,j}$ and $l_{i',j'}$ have the same variable. Furthermore, let D denote $D(0.01, 16k, \equiv)$. Note that $\overline{d} = 16k$ is linear in $||\varphi||$ and thus it can be encoded in unary. We obtain the following:

Proposition 6. *For a formula φ in CNF with k literals, C_φ and $Cost_\varphi$ are constructed in time polynomial in k and, furthermore,*

$$Val\left[C_\varphi, D\right] < 17k^2 \text{ \emph{if} } \varphi \text{ \emph{is satisfiable} } \quad \text{\emph{and}} \quad Val\left[C_\varphi, D\right] > 17k^2 + 1, \text{ \emph{otherwise.}}$$

Proof (Sketch). The proof is based on the facts that the probability to take no exponential transition within time 0.01 is > 0.99 and the probability to take at least $8k$ exponential transitions within time $16k$ is > 0.99 and that $(16k \cdot k)/0.99 < 17k^2$. ☐

The reduction proves NP-hardness as it remains to set $x := 2k^2 + \frac{1}{2}$ and $\varepsilon := \frac{1}{2}$.

NP Membership for Unary $\overline{d}$ To prove Theorem 4 we give an algorithm which, for a given approximate threshold $x > 0$, consists of

- *first* guessing the delay function $\mathbf{d}$ of $\mathcal{M}^\star_D$ that is in the set D such that $E_{\mathcal{M}^\star_D(\mathbf{d})} < x$,
- *then* constructing just the fragment $\mathcal{M}_\mathbf{d}$ of $\mathcal{M}^\star_D$ used by the guessed function $\mathbf{d}$. Here $\mathcal{M}_\mathbf{d} = (S', \{\infty\}, T_\mathbf{d}, G, €_\mathbf{d})$ where the transition probabilities and costs coincide with $\mathcal{M}$ for states in $S' \setminus S_{\mathrm{fd}}$ and in any state $s \in S_{\mathrm{fd}}$ are defined by $T_\mathbf{d}(s, \infty) = T^\star(s, \mathbf{d}(s))$ and $€_\mathbf{d}(s, \infty) = €^\star(s, \mathbf{d}(s))$ (here $T^\star(s, \mathbf{d}(s))$ and $€^\star(s, \mathbf{d}(s))$ are as in $\mathcal{M}^\star_D$).
- Last, for $\sigma : s \mapsto \infty$, the algorithm computes $y = E_{\mathcal{M}_\mathbf{d}(\sigma)}$ by standard methods and *accepts* iff $y < x$.

Note that when $\overline{d}$ is encoded in unary, both $\mathbf{d}$ and $\mathcal{M}_\mathbf{d}$ are of bit size that is polynomial in the size of the input. Hence, $\mathbf{d}$ and $\mathcal{M}_\mathbf{d}$ can be constructed in non-deterministic polynomial time (although the whole $\mathcal{M}^\star_D$ is of exponential size in this unary case). The expected total cost x in $\mathcal{M}_\mathbf{d}(\sigma)$ that has polynomial size can be also computed in polynomial time. The correctness of the algorithm easily follows from Proposition 5; for an explicit proof see [7].

5 Conclusions

In this paper, we introduced the problem of synthesising timeouts for fixed-delay CTMC. We study two variants of this problem, show that they are effectively solvable, and obtain provable worst-case complexity bounds. First, for *unconstrained optimization*, we present an approximation algorithm based on a reduction to a discrete-time Markov decision process and a standard optimization algorithm for this model. Second, we approximate the case of *bounded optimization under partial observation* also by a MDP. However, a restriction of the class of strategies twists it basically into a partial-observation MDP (where only memoryless deterministic strategies are considered). We give a 2-exponential approximation algorithm (which becomes exponential if one of the constraints is given in unary) and show that the corresponding decision problem is NP-hard.

The correctness of our algorithms stems from non-trivial insights into the behaviour of fdCTMC that we deem to be interesting in their own right. Hence, we believe that techniques presented in this paper lay the ground for further development of performance optimization via timeout synthesis.

References

1. R. Alur, C. Courcoubetis, and D.L. Dill. Model-checking for probabilistic real-time systems. In *Proceedings of ICALP'91*, volume 510 of *LNCS*, pages 115–136. Springer, 1991.
2. R. Alur, T.A. Henzinger, and M.Y. Vardi. Parametric real-time reasoning. In *STOC*, pages 592–601. ACM, 1993.
3. N.C. Audsley and A. Grigg. Timing analysis of the ARINC 629 databus for real-time applications. *Microprocessors and Microsystems*, 21(1):55–61, 1997.
4. C. Baier, N. Bertrand, P. Bouyer, T. Brihaye, and M. Größer. Probabilistic and topological semantics for timed automata. In *FSTTCS*, pages 179–191. Springer, 2007.
5. C. Baier, N. Bertrand, P. Bouyer, T. Brihaye, and M. Größer. Almost-sure model checking of infinite paths in one-clock timed automata. In *LICS*, pages 217–226. IEEE, 2008.
6. T. Brázdil, V. Forejt, J. Krcál, J. Kretínský, and A. Kucera. Continuous-time stochastic games with time-bounded reachability. *Inf. Comput.*, 224:46–70, 2013.
7. T. Brázdil, L'. Korenčiak, J. Krčál, P. Novotný, and V. Řehák. Optimizing performance of continuous-time stochastic systems using timeout synthesis. *CoRR*, abs/1407.4777, 2014.

8. T. Brázdil, J. Krčál, J. Křetínský, A. Kučera, and V. Řehák. Stochastic real-time games with qualitative timed automata objectives. In *CONCUR*, pages 207–221, 2010.
9. T. Brázdil, J. Krčál, J. Křetínský, and V. Řehák. Fixed-delay events in generalized semi-markov processes revisited. In *CONCUR*, pages 140–155, 2011.
10. P. Buchholz, E.M. Hahn, H. Hermanns, and L. Zhang. Model checking algorithms for CTMDPs. In *CAV*, volume 6806 of *LNCS*, pages 225–242. Springer, 2011.
11. L. Carnevali, L. Ridi, and E. Vicario. A quantitative approach to input generation in real-time testing of stochastic systems. *IEEE Trans. Softw. Eng.*, 39(3):292–304, 2013.
12. T. Chen, T. Han, J.-P. Katoen, and A. Mereacre. Quantitative model checking of continuous-time Markov chains against timed automata specifications. In *LICS*, pages 309–318. IEEE, 2009.
13. H. Choi, V. G. Kulkarni, and K. S. Trivedi. Transient analysis of deterministic and stochastic petri nets. In *Application and Theory of Petri Nets*, pages 166–185, 1993.
14. K. Etessami, D. Wojtczak, and M. Yannakakis. Recursive stochastic games with positive rewards. In *Proceedings of ICALP 2008, Part I*, volume 5125 of *LNCS*, pages 711–723. Springer, 2008.
15. W. Feller. *An Introduction to Probability Theory and Its Applications, Vol. 1*. Wiley, 1968.
16. C. Guet, A. Gupta, T. Henzinger, M. Mateescu, and A. Sezgin. Delayed continuous-time Markov chains for genetic regulatory circuits. In *CAV*, pages 294–309. Springer, 2012.
17. C. Haase, S. Kreutzer, J. Ouaknine, and J. Worrell. Reachability in succinct and parametric one-counter automata. In *CONCUR*, volume 5710 of *LNCS*, pages 369–383. Springer, 2009.
18. E. M. Hahn, H. Hermanns, and L. Zhang. Probabilistic reachability for parametric markov models. *STTT*, 13(1):3–19, 2011.
19. T. Han, J.P. Katoen, and A. Mereacre. Approximate parameter synthesis for probabilistic time-bounded reachability. In *Real-Time Systems Symposium*, pages 173–182. IEEE, 2008.
20. E. Isaacson and H.B. Keller. *Analysis of Numerical Methods*. Wiley, 1966.
21. P.G. Jensen and J.H. Taankvist. Learning optimal scheduling for time uncertain settings. Student project, Aalborg University, 2014.
22. S.K. Jha and C.J. Langmead. Synthesis and infeasibility analysis for stochastic models of biochemical systems using statistical model checking and abstraction refinement. *TCS*, 412(21):2162–2187, 2011.
23. J.G. Kemeny and J.L. Snell. *Finite Markov chains*. D. Van Nostrand Company, 1960.
24. M. Khaksari and C. Fischione. Performance analysis and optimization of the joining protocol for a platoon of vehicles. In *ISCCSP*, pages 1–6. IEEE, 2012.
25. L'. Korenčiak, J. Krčál, and V. Řehák. Dealing with zero density using piecewise phase-type approximation. In *EPEW*, volume 8721 of *LNCS*, pages 119–134. Springer, 2014.
26. M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston. Verifying quantitative properties of continuous probabilistic timed automata. In *CONCUR*, pages 123–137. Springer, 2000.
27. Ch. Lindemann. An improved numerical algorithm for calculating steady-state solutions of deterministic and stochastic petri net models. *Perform. Eval.*, 18(1):79–95, 1993.
28. M. A. Marsan and G. Chiola. On Petri nets with deterministic and exponentially distributed firing times. In *Advances in Petri Nets*, pages 132–145. Springer, 1987.
29. M.R. Neuhäusser and L. Zhang. Time-bounded reachability probabilities in continuous-time Markov decision processes. In *QEST*, pages 209–218. IEEE, 2010.
30. M.F. Neuts. *Matrix-geometric Solutions in Stochastic Models: An Algorithmic Approach*. Courier Dover Publications, 1981.
31. J.R. Norris. *Markov Chains*. Cambridge University Press, 1998.
32. R. Obermaisser. *Time-Triggered Communication*. CRC Press, 1st edition, 2011.
33. M.L. Puterman. *Markov Decision Processes*. Wiley, 1994.
34. K. Ramamritham and J.A Stankovic. Scheduling algorithms and operating systems support for real-time systems. *Proceedings of the IEEE*, 82(1):55–67, 1994.
35. K.B. Tiassou. *Aircraft operational reliability – A Model-based approach and case studies*. PhD thesis, Universié de Toulouse, 2013.
36. N. Wolovick, P. R. D'Argenio, and H. Qu. Optimizing probabilities of real-time test case execution. In *ICST*, pages 446–455. IEEE, 2009.