Najla Chamseddine,
Marie Duflot,
Laurent Fribourg and
Claudine Picaronny

Determinate Probabilistic Timed Automata as Markov Chains with Parametric Costs

Research Report LSV-07-21

May 2007

Laboratoire Spécification et Vérification

CENTRE NATIONAL DE LA RECHERCHE

Ecole Normale Supérieure de Cachan
61, avenue du Président Wilson
94235 Cachan Cedex France

Determinate Probabilistic Timed Automata As Markov Chains with Parametric Costs

N. Chamseddine [1], M. Duflot [2], L. Fribourg [1] and C. Picaronny [1]
[1] LSV - ENS Cachan & CNRS, France
[2] LACL - University Paris 12, France

Abstract

We consider probabilistic systems modeled under the form of a special class of probabilistic timed automata. Such automata have "no choice":

- when the automaton arrives at a node, the time at which it will leave it is determined;

- when the automaton leaves the node, there is just one distribution of target nodes.

In the paper, we give a method for computing the expected time $\mathcal{A}$ for the automaton to reach an "absorbing" node. Roughly speaking, the method consists in putting the automaton under the form of a Markov chain with costs (corresponding to durations). Under certain conditions, the method is parametric *in the sense that $\mathcal{A}$ is computed as a function of the constants appearing in the outgoing conditions and the invariants of nodes, but does not assume known their explicit values.*

We illustrate the method on the CSMA/CD protocol.

1 Introduction

We consider probabilistic systems modeled under the form of a special class of probabilistic timed automata [9]. In such a class, called "Determinate Probabilistic Timed Automaton (DPTA)", there is *no choice*:

- when the automaton arrives at a node, the time at which it will leave it is determined;
- when the automaton leaves the node, there is just one distribution of target nodes.

Such automata thus behave as discrete Markov chains on an infinite state space (set of nodes augmented with dense-time information). We will be interested in computing the expected time $\mathcal{A}$ for this Markov chain to reach *End* for the first time, where *End* is an "absorbing" node (a node reached with probability 1, then never left). We will give a method for computing $\mathcal{A}$ for DPTA. Roughly speaking, this consists in transforming the DPTA under the form of a Markov chain with costs [3], where the costs of the steps correspond to their durations. Under certain conditions, the method is *parametric* in the sense that $\mathcal{A}$ is computed as a function of the constants a's appearing in the outgoing conditions and the invariants of nodes, but does not assume known their explicit values.

We illustrate the method on the CSMA/CD protocol [8, 11].

2 DPTA As Markov Chains

Probabilistic timed automata [9] are an extension of the timed automata model of [1] with discrete probability distributions. These models combine nondeterminism and probabilistic distributions, and can be seen as a variant of Markov decision processes: the selection of a probability distribution is made by an *adversary* (or scheduler). With each adversary one can associate a sequential Markov chain [2, 5, 7, 10]. In this paper, we consider a restricted model of timed automata where there is no nondeterminism: The adversary is unique and implicit. The system thus behaves as a discrete Markov chain over a finite set of nodes (or locations), augmented with (real-valued) time information. The model is formally described in the rest of the Section.

Clocks and clock valuation. A *clock* is a real-valued variable which increases at the same rate as real-lime. Let $\mathcal{X} = \{x_1, ..., x_n\}$ be a set of clocks, and let $\nu : \mathcal{X} \rightarrow \mathcal{R}$ be a function assigning a real value to each of the clocks in this set. Such a function is called a *clock valuation*. We denote the set of all clock valuations of $\mathcal{X}$ by $\mathcal{R}^{\mathcal{X}}$. For some $X \subseteq \mathcal{X}$, we write $\nu[X := 0]$ for the clock valuation that assigns 0 to clocks in X, and agrees with ν for all clocks in $\mathcal{X}\backslash X$. For every $t \in \mathcal{R}$, $\nu + t$ denotes the clock valuation for which all clocks $x \in \mathcal{X}$ take the value $\nu(x)+t$.

Constraints. A *inequality (resp. equality) constraint* over $\mathcal{X}$ is an expression of the form $x_k \leq a$ (resp. $x_k = a$), with $x_k \in \mathcal{X}, a \in \mathcal{N}$. A clock valuation ν satisfies a constraint $x_k \leq a$ (resp. $x_k = a$) iff $\nu(x_k) \leq a$ (resp. $\nu(x_k) = a$). Let $\mathbf{I}_{\mathcal{X}}$ (resp. $\mathbf{E}_{\mathcal{X}}$) be the set of all the inequality (resp. equality) constraints of $\mathcal{X}$.

Determinate Probabilistic Timed Automata.

Definition 1. *A determinate probabilistic timed automaton (DPTA) is a tuple* $\Delta = (\mathcal{S}, 0, \mathcal{X}, inv, p, \alpha)$ *which contains:*

- *a finite set* $\mathcal{S}$ *of nodes,*
- *a start node* 0*,*
- *a finite set* $\mathcal{X}$ *of clocks,*
- *a function* $inv : \mathcal{S} \rightarrow \mathbf{I}_{\mathcal{X}}$ *assigning to each node* s *an invariant condition of the form* $x_{k_s} \leq a_s$*,*
- *a function* $\alpha : \mathcal{S} \rightarrow \mathbf{E}_{\mathcal{X}}$ *assigning to each node* s *an outgoing condition of the form* $x_{k_s} = a_s$*,*
- *a function* $\mathbf{p} : \mathcal{S} \rightarrow \mu(\mathcal{S})$ *assigning to each node* s *a discrete probability distribution, denoted by* $\mathbf{p}(s)$ *or more simply by* $\mathbf{p}_s$*, on* $\mathcal{S}$*,*
- *a function* $R : \mathcal{S} \times \mathcal{S} \rightarrow 2^{\mathcal{X}}$ *assigning to each edge* (s, s') *(with* $\mathbf{p}_s(s') > 0$*) a set* X *of clocks to reset.*

Schematically, a DPTA is represented as an oriented graph with nodes in $\mathcal{S}$ (associated with invariants), where each edge from s to s' is labelled with $Prob = \mathbf{p}_s(s')$, the set $\{x_k := 0\}_k$ with $x_k \in R(s, s')$, and the outgoing condition $\alpha(s) : (x_{k_s} = a_s)$.

The *untimed graph of the DPTA* is the graph where the invariant has been omitted from nodes, and edges are labeled only with their probability $Prob$ (no outgoing conditions, no reset clocks). This define a finite Markov chain on $\mathcal{S}$, denoted by $U(\Delta)$, called *untimed version of* Δ.

The constants a_s appearing in the invariants and in the outgoing conditions of Δ will be often referre to as "costs".

Example 1. *Let us consider the following DPTA depicted on Figure 1.*[1] *There are 5 nodes* $0, 1, 2, 3$ *and End, and 3 clocks* x, u *and* v*. Invariant* $x \leq 0$ *is attached to* 0*. Invariant* $u \leq TD$ *is attached to* 1*,* $v \leq TD$ *to* 3*, and* $x \leq T1$ *to* 2*. End is an absorbing node. The initial node is* 0*. There are 4 transitions (besides a transition looping on End with probability 1, not represented on the Figure):*

1. A transition I, with outgoing condition $x = 0$*, goes from* 0 *to* 1*, which resets all the clocks to 0.*

2. A transition G, with outgoing condition $u = TD$*, leaves* 1 *and goes with prob.* p *to* 3*, or with prob.* $1 - p$ *to* 2*. In the first case,* v *is also reset to 0.*

3. A transition B, with outgoing condition $v = TD$*, leaves* 3 *and goes with prob.* q *to End, or with prob.* $1 - q$ *to* 2*.*

4. A transition F, with outgoing condition $x = T1$*, goes from 2 to 1, and resets* u *and* x *to 0.*

A *state* is a pair $\langle s, \nu \rangle$, where $s \in \mathcal{S}$ is a node and $\nu \in \mathcal{R}^{\mathcal{X}}$ is a valuation such that ν satisfies $inv(s)$.[2] Let Q_Δ (or more simply Q) be the set of states of Δ.

The behaviour of the system takes the form of transitions between states, which can be seen as a result of the passage of time followed by the execution of a discrete transition. The role of the invariant condition is to describe the set of *admissible states* of the probabilistic timed automaton; therefore, we forbid transitions to inadmissible states.

The system starts in node 0 with all its clocks initialized to 0. The values of all the clocks increase uniformly with time. When the value of the clock x_{k_s} reaches a_{k_s}, then the system makes a discrete probabilistic transition according to $\mathbf{p}_s$: for any $s' \in \mathcal{S}$, the probability that the system will make a state transition to node s', is given by $\mathbf{p}_s(s')$. The clock of X are reset accordingly, with $X = R(s, s')$.

In the following, the invariants and outgoing conditions are subject to the following assumption (called *admissible targets* in [9]:

(H0) We assume that it is never possible to perform a discrete transition to a node for which the invariant condition is not satisfied by the current values of the clocks. Formally, for each $q = \langle s, \nu \rangle$ with $\nu(x_{k_s}) = a_s$, assume that, for all $s' \in \mathcal{S}$ such that $\mathbf{p}_s(s') > 0$, it is the case that $\nu[X := 0]$ satisfies $inv(s')$ with $X = R(s, s')$.

Given two nodes s, s', two valuations of clocks ν, ν', we say that there is a *step of computation* (or a transition) between $q = \langle s, \nu \rangle$ and $q' = \langle s', \nu' \rangle$ of probability p and duration t (upon outgoing condition $\alpha(s)$), and

[1] This automaton is inspired from a simplified version of the sender in the BRP model of [6].

[2] Given a node s, and a clock valuation ν, we say that ν is *admissible at* s if the valuation ν satisfies the invariant of node s.

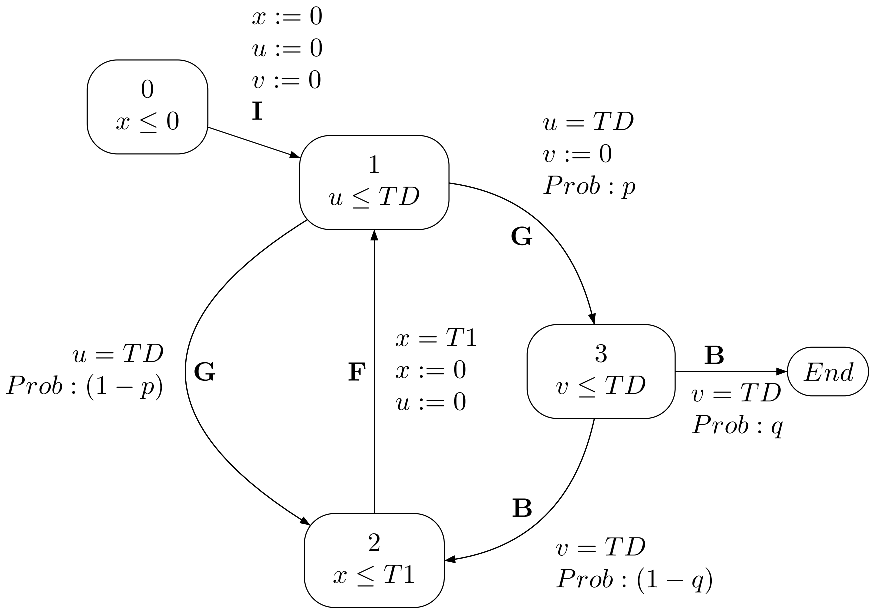

Figure 1. Timed Automaton of Example 1.

write:

$$\langle s, \nu \rangle \rightarrow^{t,p}_{\alpha(s)} \langle s', \nu' \rangle$$

if:

- The outgoing condition $\alpha(s)$ is of the form $x_{k_s} = a_s$.
- ν is admissible at s, i.e.: $\nu(x_{k_s}) \leq a_s$. Furthermore, the duration t is given by: $t = a_s - \nu(x_{k_s})$.
- The probability distribution assigned to s is positive in s' ($\mathbf{p}_s(s') > 0$). Furthermore, the probability p is given by: $p = \mathbf{p}_s(s')$.
- $\nu' = (\nu + t)[X := 0]$, where $X = R(s, s')$. Furthermore ν' is admissible at s' (see assumption **(H0)**).

Note that, in step $\langle s, \nu \rangle \rightarrow \langle s', \nu' \rangle$, the valuation ν' is uniquely determined, once $q = \langle s, \nu \rangle$, and s' are known: It corresponds to the valuation of clocks immediately after the incoming of transition from q at node s'.

The value t corresponds to the duration elapsed in s from $q = \langle s, \nu \rangle$ (until the firing of the outgoing transition). The value p corresponds to the probability of performing step $q = \langle s, \nu \rangle \rightarrow q' = \langle s', \nu' \rangle$.

Such a step of computation can be iterated (under assumption **(H0)**). A DPTA thus behaves as a discrete *Markov chain* $\{q_n\}_{n \geq 0}$ on the (infinite) space Q_Δ.

3 Absorption Time for DPTA

3.1 Paths As Sequences of Transitions

A *(finite) path of computation via* Δ is a sequence:

$$\omega = q_0 \rightarrow^{t_1,p_1}_{\alpha_1} q_1 \rightarrow^{t_2,p_2}_{\alpha_2} \cdots q_{m-1} \rightarrow^{t_m,p_m}_{\alpha_m} q_m$$

where, for all $0 \leq i \leq m$, $q_i = \langle s_i, \nu_i \rangle \in Q$, and α_i (resp. t_i, p_i) are abbreviations for $\alpha(s_i)$ (resp. $a_{s_i} - \nu_i(x_{k_i}), \mathbf{p}_{s_i}(s_{i+1})$), as defined in Sect. 2. The *length* of such a path is m.

In the following, we will mainly focus on paths starting at the initial state $q_0 = \langle 0, \overline{0} \rangle$, where $\overline{0}$ is the null valuation, and 0 the start node. By convention, the state q_0 is considered as a path of *null length*.

Besides, we suppose that the system leaves the start node intantaneously, and never returns to it. This is ensured by assuming that:

- the invariant associated to 0 is of the form $x_k \leq 0$, for some k,

- the probability $\mathbf{p}_s(0) = 0$, for all $s \in \mathcal{S}$.

It is also convenient to consider that all the clocks are reset to 0, when the system leaves the start node. Without loss of understanding, we will often abbreviate paths starting at q_0 under the form:

$\omega : q_0 \rightarrow_{\alpha_1} q_1 \rightarrow_{\alpha_2} q_2 \rightarrow \cdots$, or

$\omega : 0 \rightarrow_{\alpha_1} s_1 \rightarrow_{\alpha_2} s_2 \rightarrow \cdots$,

and will say that α_k is the kth transition from q_{k-1} to

q_k (or from s_{k-1} to s_k).[3]

It follows from the above observations, that such a path is fully determined by the sequence of nodes $s_1, ..., s_m$: Given $s_1, ..., s_m$, the duration t_{k+1} elapsed in each s_k, the valuation ν_{k+1}, and the probability p_{k+1} of each step ($0 \leq k \leq m-1$) are uniquely determined.

Let us define the notion of duration and probability associated to such paths.

Definition 2. *Let us consider a (finite) path of the form:*

$\omega : 0 \rightarrow_{\alpha_1}^{t_1,p_1} q_1 \rightarrow_{\alpha_2}^{t_2,p_2} q_2 \cdots \rightarrow_{\alpha_{m-1}}^{t_{m-1},p_{m-1}} q_{m-1} \rightarrow_{\alpha_m}^{t_m,p_m} q_m.$

The probability of ω*, denoted by* $Pr(\omega)$*, is defined by:*

$$Pr(\omega) = \prod_{k=1}^{m} p_k.$$

The duration of ω*, denoted by* $Cost(\omega)$*, is defined by:*

$$Cost(\omega) = \sum_{k=1}^{m} t_k.$$

By convention, the probability (resp. duration) of a path of null length (i.e., reduced to state 0) is equal to 1 (resp. 0).

The paths of computation via a DPTA Δ can be mapped 1-to-1 on the paths of the untimed version $U(\Delta)$.

Proposition 1. *Let* Δ *be a DPTA and* $U(\Delta)$ *its untimed version. We have, for all path of* $U(\Delta)$ *of the form:*

$$0 \rightarrow^{p_1} s_1 \rightarrow^{p_2} s_2 \cdots \rightarrow^{p_m} s_m,$$

there is a (unique) path of computation via Δ

$$0 \rightarrow_{\alpha_1}^{t_1,p_1} \langle s_1, \nu_1 \rangle \rightarrow_{\alpha_2}^{t_2,p_2} \langle s_2, \nu_2 \rangle \cdots \rightarrow_{\alpha_m}^{t_m,p_m} \langle s_m, \nu_m \rangle,$$

of same probability, for some (unique) sequences $t_1, t_2, t_3, ...$ *and* $\nu_1, \nu_2, \nu_3, ...$ *(and some transitions* $\alpha_1, ..., \alpha_m$*).*

Conversely, for any path of computation via Δ

$$0 \rightarrow_{\alpha_1}^{t_1,p_1} \langle s_1, \nu_1 \rangle \rightarrow_{\alpha_2}^{t_2,p_2} \langle s_2, \nu_2 \rangle \cdots \rightarrow_{\alpha_m}^{t_m,p_m} \langle s_m, \nu_m \rangle,$$

there is a (unique) path of $U(\Delta)$*:*

$$0 \rightarrow^{p_1} s_1 \rightarrow^{p_2} s_2 \cdots \rightarrow^{p_m} s_m$$

of same probability.

It follows that the probability to reach a node $s \in \mathcal{S}$ from 0 are the same in Δ and $U(\Delta)$.

[3] Strictly speaking, this is an abuse of notation, since we assimilate outgoing condition $\alpha(s)$ and (outgoing) transition. This is harmless, because there is a single outgoing transition per node

3.2 Absorption Time

We assume that there is a special terminal node, called *End*, which is "absorbing".

Definition 3. *We say that End is* absorbing *if:*

- the transition outgoing from End, say α_{abs}*, goes back to End with probability 1,*

- the probability that a path (starting at 0) of length m *contains End tends to 1 when* m *tends to* ∞*.*

In the following, the set of (finite) paths that we consider do not use α_{abs}. Such paths either do not contain *End*, or contain *End* at their last position. (We do not consider paths looping at *End* via α_{abs}.)

We are interested in computing the *expected absorbing time of* Δ, denoted by $\mathcal{A}$, i.e., the expected time for Δ to first hit *End*.

Let Ω_i be the set of finite paths having i at their last position (i.e., paths of the form $\omega : 0 \rightarrow_{\alpha_1} s_1 \rightarrow_{\alpha_2} s_2 \cdots \rightarrow_{\alpha_{m-1}} s_{m-1} \rightarrow_{\alpha_m} i$, for some nodes $s_1, ..., s_{m-1}$ and some transitions $\alpha_1, ..., \alpha_m$ distinct from α_{abs}).

For all transition $\alpha \neq \alpha_{abs}$, we will also consider the subset Ω_{α_i} of the paths of Ω_i of last transition α (i.e., paths of the form $\omega : 0 \rightarrow_{\alpha_1} s_1 \rightarrow_{\alpha_2} s_2 \cdots \rightarrow_{\alpha_{m-1}} s_{m-1} \rightarrow_{\alpha} i$, for some nodes $s_1, ..., s_{m-1}$ and some transitions $\alpha_1, ..., \alpha_{m-1}$ distinct from α_{abs}).

The expected absorbing time $\mathcal{A}$ can be expressed as follows:

$$\mathcal{A} = \sum_{\omega \in \Omega_{End}} Pr(\omega) Cost(\omega).$$

4 Computation of the Absorption Time

We now explain how to compute $\mathcal{A} = \Sigma_{\omega \in \Omega_{End}} Pr(\omega) Cost(\omega)$. The idea is to decompose every path of computation of Ω_{End} into a certain number of "macro-steps", where each macro-step groups together several consecutive transitions. The interest of a macro-step is that its duration is known. (It corresponds to a "cost" a.)

4.1 Macro-Step

The transitions that we will consider in the following are implicitly assumed to be distinct from α_{abs}.

Given a state r (resp. q) and a transition α (resp. β) arriving to r (resp. q), let us focus on finite sequences between q (just after the firing of β) and r (just after the firing of α), of the form:

$$q \rightarrow_{\alpha_1} q_1 \cdots \rightarrow_{\alpha_{m-1}} q_{m-1} \rightarrow_{\alpha} r,$$

where:

- the outgoing condition of α is of the form $x = a_\alpha$,
- clock x is reset to 0 by β, but by no α_k $(1 \le k \le m-1)$.[4]

The interest of such a sequence is that it always takes a_α units of time (since x is 0 at the beginning, x is never reset before the last transition, and the outgoing transition of last transition is $x = a_\alpha$.).

A *point*, denoted by ${}^\beta j$, is a couple (β, j) where j is a node, and β is a transition of target node j.

Additionally, we will consider the start node 0 as a point (without associated incoming transition β). The valuation implicitly associated to 0 is the null valuation $\overline{0}$.

Definition 4. *(*Macro-step via σ*).*
Let ${}^\alpha i, {}^\beta j$ be two points. Suppose that the outgoing condition of α is of the form $x = a_\alpha$, and that β resets x to 0. Let $\alpha_1, ..., \alpha_{m-1}$ be transitions not *resetting x to 0. For $\sigma : (\alpha_1, s_1, \alpha_2, s_2, ..., \alpha_{m-1}, s_{m-1})$, we say that there is a* macro-step between ${}^\beta j$ and ${}^\alpha i$ via σ, *and write ${}^\beta j \Rightarrow_\sigma {}^\alpha i$, if there exist a valuation ν and real numbers $t_1, ..., t_m$, $p_1, ..., p_m$ such that:*

$$(j, \nu) \to_{\alpha_1}^{t_1, p_1} s_1 \cdots \to_{\alpha_{m-1}}^{t_{m-1}, p_{m-1}} s_{m-1} \to_{\alpha}^{t_m, p_m} i.$$

The probability of the macro-step between ${}^\beta j$ and ${}^\alpha i$ via σ, *denoted by $P_\sigma({}^\beta j, {}^\alpha i)$, is given by:*

$$P_\sigma({}^\beta j, {}^\alpha i) = \prod_{k=1}^{m} \mathbf{p_k},$$

with $s_0 = j$ and $s_m = i$.

The duration associated to such a macro-step via σ, *denoted by $T_\sigma({}^\beta j, {}^\alpha i)$, is equal to a_α.*

Example 2. *Consider the DPTA of Example 1. There is a sequence of transitions going from 1 to 1 of the form: $1 \to_G 2 \to_F 1$. The outgoing condition of F is: $x = T1$. Clock x is reset to* 0 *by transition I arriving at state 1, but not by G. It follows that there is a macro-step via σ_1 of the form: ${}^I 1 \Rightarrow_{\sigma_1}^F 1$, where $\sigma_1 = (G, 2)$. We have $P_{\sigma_1}({}^I 1, {}^F 1) = 1 - p$ and $T_{\sigma_1}({}^I 1, {}^F 1) = T1$.*

Likewise, there is a sequence of transitions going from 1 to 1 of the form: $1 \to_G 3 \to_B 2 \to_F 1$. The outgoing condition of F is: $x = T1$. Clock x is reset to 0 *by transition I arriving at state 1, but neither by G nor by B. Hence, there is a macro-step via σ_2 of the form: ${}^I 1 \Rightarrow_{\sigma_2}^F 1$, where $\sigma_2 = (G, 3, B, 2)$, with $P_{\sigma_2}({}^I 1, {}^F 1) = p(1-q)$ and $T_{\sigma_2}({}^I 1, {}^F 1) = T1$.*

A macro-step via σ can be seen as a portion of path ω, as defined in Sect. 3.1. In particular, the associated probabilities and durations are consistent.

[4] Note that α may reset x to 0 (or not).

Proposition 2. *Consider a macro-step via σ, as defined in Def. 4. We have:*
$P_\sigma({}^\beta j, {}^\alpha i) = p_1 \times \cdots \times p_m$, and
$T_\sigma({}^\beta j, {}^\alpha i) = a_\alpha$.

Let us make the following assumption:

(H1) There is no cyclic sequence of transitions of null duration, i.e, there is no path

$$0 \to_{\alpha_1}^{t_1, p_1} s_1 \to_{\alpha_2}^{t_2, p_2} s_2 \cdots \to_{\alpha_{m-1}}^{t_{m-1}, p_{m-1}} s_{m-1} \to_{\alpha_m}^{t_m, p_m} s_m \cdots$$

such that $s_k = s_\ell$ and $t_{k+1} + \cdots + t_\ell = 0$, for some k, ℓ with $k < \ell$.[5]

We have:

Proposition 3. *Consider a macro-step ${}^\beta j \Rightarrow_\sigma {}^\alpha i$ via σ. Under* **(H1)***, the length of such a macro-step is bounded above by $N!$, with $N = |\mathcal{S}|$.*

Proof. (Sketch). Let us show that σ cannot contain two occurrences of some $s \in \mathcal{S}$ (no cycle). By reductio ad absurdum: suppose that there exist a sequence σ_1 between ${}^\beta j$ and ${}^\alpha i$ which contains one cycle. There would be a sequence σ_2 associated to a macro-step between ${}^\beta j$ and ${}^\alpha i$, which contains two such cycles, then a sequence σ_3 with 3 cycles, etc. Such cycles are possible by supposing the hypothesis **(H0)**. This leads to a contradiction because the total duration of any macro-step via σ_k is fixed (equal to a_α), while the total duration of the repeated included cycles increases (by **(H1)**). □

Given two points ${}^\alpha i, {}^\beta j$, we define the set of all sequence σ of transitions forming a macro-step between ${}^\beta j$ to ${}^\alpha i$, and write $S({}^\beta j, {}^\alpha i)$, the set:
$S({}^\beta j, {}^\alpha i) = \{\sigma \mid {}^\beta j \Rightarrow_\sigma {}^\alpha i$ occurs with prob. $> 0\}$.

For any couple of points ${}^\beta j, {}^\alpha i$, Prop. 3 ensures that every σ of $S({}^\beta j, {}^\alpha i)$ is of bounded length, hence the set $S({}^\beta j, {}^\alpha i)$ is finite.

Definition 5. *(*Macro-step*).*
Let ${}^\alpha i, {}^\beta j$ be two points. Suppose that the outgoing transition of α is of the form $x = a_\alpha$, and that β resets x to 0. We say that there is a macro-step between ${}^\beta j$ and ${}^\alpha i$, *and write ${}^\beta j \Rightarrow {}^\alpha i$, if there is a sequence σ (as defined in Def. 4) such that: ${}^\beta j \Rightarrow_\sigma {}^\alpha i$.*
The probability of a macro-step between ${}^\beta j$ and ${}^\alpha i$, *denoted by $P({}^\beta j, {}^\alpha i)$, is given by:*

$$P({}^\beta j, {}^\alpha i) = \sum_{\sigma \in S({}^\beta j, {}^\alpha i)} P_\sigma({}^\beta j, {}^\alpha i).$$

[5] This ensures that all the paths are *divergent* (see [9]).

The duration associated to such a macro-step*, denoted by $T(^{\beta}j,^{\alpha}i)$, is equal to a_{α}.*

Example 3. *Let us consider again the DPTA depicted on Fig.1, and the macro-steps $^{I}1 \Rightarrow_{\sigma_1} {}^{F}1$ and $^{I}1 \Rightarrow_{\sigma_2} {}^{F}1$ from $^{I}1$ to $^{F}1$ (see Example 2). It can be seen that, besides σ_1 and σ_2, there is no macro-step between $^{I}1$ and $^{F}1$ via any other sequence σ. We have: $S(^{I}1,^{F}1) = \{\sigma_1, \sigma_2\}$. Hence, there is a macro-step $^{I}1 \Rightarrow^{F} 1$ with $P(^{I}1,^{F}1) = 1 - p + p(1-q) = 1 - pq$ and $T(^{I}1,^{F}1) = T1$.*

4.2 Graph of Macro-steps

Intuitively, each macro-step corresponds to a set of sequences of transitions. The idea is now to transform the graph of the DPTA into the "compact" form of a graph of macro-steps. In order to construct such a graph, there are *a priori* two possible manners: a forward manner, which, given a point $^{\beta}j$, constructs all the possible "successors" $^{\alpha}i$ and all the sequences σ such that $^{\beta}j \Rightarrow_{\sigma}^{\alpha} i$, or a backward manner, which, given a point $^{\alpha}i$, constructs all the possible "predecessors" $^{\beta}j$ and all the sequences σ such that $^{\beta}j \Rightarrow_{\sigma}^{\alpha} i$.

In a backward manner, $^{\alpha}i$ is given, so the guard of α, say $x_{\alpha} = a$ – which is *unique*, is known: one constructs backwards all the sequences σ until one gets a transition resetting x_{α} to 0.

In a forward manner, $^{\beta}j$ is given, so the clocks reset by β, say $y_{\beta}, z_{\beta}, ...$ – which are *(possibly) multiple*, are known: one constructs forwards all the sequences σ until one gets a transition of guard of the form $y_{\beta} = b$ *or* of the form $z_{\beta} = c$,...

For each step of predecessor generation, the stopping condition is therefore unique while it is (possibly) multiple for a step of successor generation. It is therefore simpler to proceed backward rather than forward.

We define the set $Pred(^{\alpha}i)$ of *predecessors of* $^{\alpha}i$ as:
$Pred(^{\alpha}i) = \{^{\beta}j \mid {}^{\beta}j \Rightarrow {}^{\alpha}i \text{ occurs (with prob. } > 0)\}$.
Alternatively, we have:
$Pred(^{\alpha}i) = \{^{\beta}j \mid {}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i \quad \text{for some } \sigma \in S(^{\beta}j,^{\alpha}i)\}$.
For each $^{\alpha}i$, we will have to find all the sequences σ of $S(^{\beta}j,^{\alpha}i)$ for some $^{\beta}j$.

We are able to construct in finite time the (oriented) graph of the "iterated predecessors" of $^{\gamma}End$. More precisely, we compute iteratively the sets of iterated predecessors $\cup_{m=1}^{k} Pred^{m}(^{\gamma}End)$, until no new $^{\beta}j$ is produced
($\cup_{m=1}^{k+1} Pred^{m}(^{\gamma}End) = \cup_{m=1}^{k} Pred^{m}(^{\gamma}End)$). The set of nodes of this graph, denoted by $\mathcal{V}$, is the set of iterated predecessors of $^{\gamma}End$. There is an edge between two nodes if there is a macro-step between them.

Using this graph, one can define "macro-paths", which are sequences of macro-steps. A macro-path τ is represented under the form:

$$0 \Rightarrow {}^{\beta_1}j_1 \Rightarrow {}^{\beta_2}j_2 \cdots \Rightarrow {}^{\beta_m}j_m,$$

where $^{\beta_1}j_1, ...,^{\beta_m}j_m$ are points. Let $\mathcal{C}(0,^{\alpha}i)$ be the set of macro-paths from 0 to $^{\alpha}i$. Henceforth, we focus on the subset $\mathcal{V}'$ of points $^{\alpha}i$ of $\mathcal{V}$ which are either equal to 0 or are "connected to 0", i.e., for which the set $\mathcal{C}(0,^{\alpha}i)$ is non empty.

Let $Pred'$ be defined, for all $^{\alpha}i \in \mathcal{V}'$ by:
$Pred'(^{\alpha}i) = Pred(^{\alpha}i) \cap \mathcal{V}'$.
The set $Pred'(^{\alpha}i)$ corresponds to the subset of predecessors of $^{\alpha}i$ connected to 0.

Henceforth, the elements of $\mathcal{V}'$ are supposed ordered (with 0 as least element and points $^{\gamma}End$ as greatest elements), and the set $\mathcal{V}'$ viewed as a vector (with 0 as 1st component, and $^{\gamma}End$ as last components).

Consider the $|\mathcal{V}'| \times |\mathcal{V}'|$-matrix P' defined, for all $^{\alpha}i,^{\beta}j \in \mathcal{V}'$, by $P'(^{\beta}j,^{\alpha}i) = P(^{\beta}j,^{\alpha}i)$, with the convention that $P'(^{\beta}j,^{\alpha}i) = 0$ if there is no macro-step between $^{\beta}j$ and $^{\alpha}i$. Let us note that the first column of P' is null (because the system never returns to 0) and the last line corresponding to the point $^{\gamma}End$ are also null (because $\mathcal{V}'$ does *not* include point ^{abs}End, where *abs* is the transition looping at *End*).

Henceforth, all the points $^{\alpha}i,^{\beta}j,^{\gamma}End, ...$ are implicitly assumed to be elements of $\mathcal{V}'$.

Example 4. *Let us consider the automaton Δ of Example 1 (see Sect. 2). Let us compute the set of iterated predecessors of End. The only macro-step arriving to End is of the form: $^{G}3 \Rightarrow End$ (The only transition arriving to End is B. $v = TD$ is the guard of B, and G resets v to 0). We have: $Pred(End) = \{^{G}3\}$, with $P(^{G}3, End) = q$ and $T(^{G}3, End) = TD$.*

Let us now compute $Pred(^{G}3)$. Any macro-step arriving to $^{G}3$ is of the form: $^{I}1 \Rightarrow^{G} 3$, or $^{F}1 \Rightarrow^{G} 3$, since $u = TD$ is the guard of G and I (resp.F) resets u to 0. We have: $Pred(^{G}3) = \{^{I}1,^{F}1\}$, with $P(^{I}1,^{G}3) = p$ and $T(^{I}1,^{G}3) = TD$ (resp. with $P(^{F}1,^{G}3) = p$ and $T(^{F}1,^{G}3) = TD$).

Let us now compute $Pred(^{F}1)$. As seen in example 3, there is a macro-step $^{I}1 \Rightarrow^{F} 1$ with $P(^{I}1,^{F}1) = 1 - pq$ and $T(^{I}1,^{F}1) = T1$. Likewise, there is a macro-step $^{F}1 \Rightarrow^{F} 1$ with $P(^{F}1,^{F}1) = 1 - pq$ and $T(^{F}1,^{F}1) = T1$. We have: $Pred(^{F}1) = \{^{I}1,^{F}1\}$,

Finally, a simple inspection of the graph of the DPTA shows the existence of a macro-step $0 \Rightarrow^{I} 1$. We have $Pred(^{I}1) = \{0\}$, with $P(0,^{I}1) = 1$ and $T(0,^{I}1) = 0$.

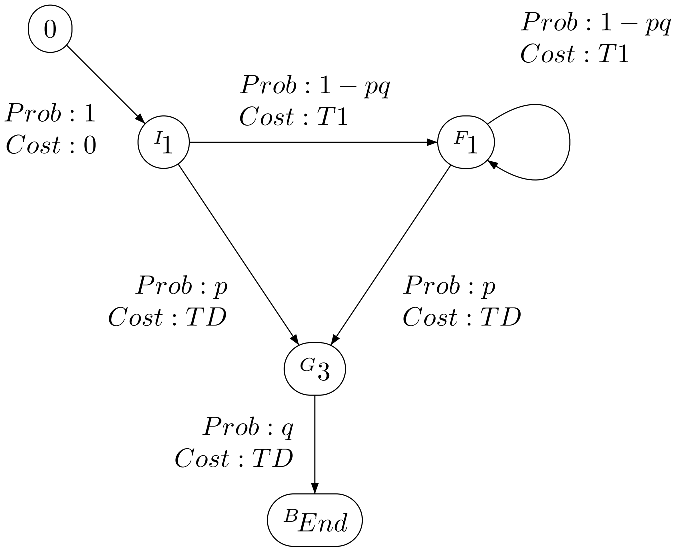

Figure 2. Graph of Macro-Steps for Example 1.

This ends the process of generating all the iterated predecessors of End. All these iterated predecessors are connected to 0*, therefore the set* $\mathcal{V}'$ *is* $(0,^I1,^F1,^G3,End)$*, and* $Pred'$ *coincides with* $Pred$*. The graph is depicted on Fig. 2. Accordingly, matrix* P' *is:*

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1-pq & p & 0 \\ 0 & 0 & 1-pq & p & 0 \\ 0 & 0 & 0 & 0 & q \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

4.3 Computation

Consider a macro-path τ:

$$^{\beta_0}j_0 \Rightarrow {}^{\beta_1}j_1 \Rightarrow {}^{\beta_2}j_2 \cdots \Rightarrow {}^{\beta_m}j_m,$$

where $^{\beta_0}j_0$ is 0. We say that the *length of* τ, denoted by $|\tau|$, is m. (The point 0 is considered as a macro-path of null length.) We define the *probability of* τ, denoted by $\pi(\tau)$, and the *duration of* τ, denoted by $\delta(\tau)$ as:

$$\pi(\tau) = \prod_{k=0}^{m-1} P(^{\beta_k}j_k, ^{\beta_{k+1}}j_{k+1})$$

$$\delta(\tau) = \sum_{k=1}^{m} a_{\beta_k}$$

By convention, the probability (resp. duration) of a macro-path of null length is equal to 1 (resp. 0).

For a point $^\alpha i \in \mathcal{V}'$, let:

$$C(^\alpha i, End) = \cup_{^\gamma End} C(^\alpha i, ^\gamma End)$$

and

$$T_{End}(^\alpha i) = \Sigma_{\tau \in \mathcal{C}(^\alpha i, End)} \delta(\tau)\pi(\tau).$$

Such a quantity is related to the notion of "average cost" on finite Markov chains with *costs* (or rewards) [3, 4]. Alternatively, it can be defined, for all $^\alpha i \in \mathcal{V}'$, by:

$$T_{End}(^\alpha i) = \lim_{m\to\infty} T^m_{End}(^\alpha i)$$

with:

$$T^m_{End}(^\alpha i) = 0 \quad \text{if } ^\alpha i = {}^\gamma End$$

$$T^m_{End}(^\alpha i) = \sum_{\tau \in \mathcal{C}^{\leq m}(^\alpha i, End)} \delta(\tau)\pi(\tau) \ \text{ if } ^\alpha i \in \mathcal{V}' \setminus \{End\},$$

where $\mathcal{C}^{\leq m}(^\alpha i, End)$ is the subset of $C(^\alpha i, End)$ of macro-paths of length no greater than m. The limit of T^m_{End} is well-defined because, for $^\alpha i = {}^\gamma End$, the sequence $T^m_{End}(^\alpha i)$ is constantly null, and, for $^\alpha i \in \mathcal{V}' \setminus \{^\gamma End\}$, the sequence $T^m_{End}(^\alpha i)$ is non-negative and non-decreasing.

The interest of computing T_{End} comes from the following result.

Proposition 4. *We have:*

$$\mathcal{A} = T_{End}(0).$$

The proof is given in Appendix 1.

Example 5. *Let us point out that Prop. 4 may allow us to compute $T_{End}(0)$ (hence $\mathcal{A}$) in a* parametric *manner, whenever the set of all paths of the macro-steps graph is easily described, as it is the case for DPTA depicted on Fig.1*[6]*:*

$$\begin{aligned}\mathcal{A} &= T_{End}(0)\\ &= 2TDpq + \sum_{n=0}^{\infty}((n+1)T1 + 2TD)(1-pq)^{n+1}pq\\ &= 2TD + \tfrac{1-pq}{pq}T1.\end{aligned}$$

Let us now give a method for computing $T_{End}(0)$ in the general case. In $\mathcal{V}'$, let us now distinguish the element End from the other ones. Let $\mathcal{V}'' = \mathcal{V}' \setminus \{End\}$, and P'' the restriction of P' on $\mathcal{V}'' \times \mathcal{V}''$ (obtained from P' by removing the last line and last column corresponding to the point End). Note that the first column of P'' is still null.

It is convenient to introduce for each ${}^{\alpha}i$ in $\mathcal{V}''$, a "correcting factor" or "weight", denoted by $w_{End}({}^{\alpha}i)$, defined by

$$w_{End}({}^{\alpha}i) = \lim_{m\to\infty} w^m_{End}({}^{\alpha}i)$$

with:

$$w^m_{End}({}^{\alpha}i) = \sum_{\tau\in\mathcal{C}^{\le m}({}^{\alpha}i,End)} \pi(\tau) \quad \text{if } {}^{\alpha}i \in \mathcal{V}'',$$

$\mathcal{C}^{\le m}({}^{\alpha}i, End)$ be the set of macro-paths from ${}^{\alpha}i$ to End of length m. The limit of $w^m_{End}({}^{\alpha}i)$ [7] is well-defined for the same reasons as for $T^m_{End}({}^{\alpha}i)$.

Proposition 5. *Let $X'' = (X_{{}^{\alpha}i})_{{}^{\alpha}i\in\mathcal{V}''}$ (resp. $W'' = (W_{{}^{\alpha}i})_{{}^{\alpha}i\in\mathcal{V}''}$) be two vectors with real components. Let consider the systems:*

$$X = P''X + AW + C \qquad (I),$$

$$W = P''W + B \qquad (II),$$

where A is the the $|\mathcal{V}''| \times |\mathcal{V}''|$-matrix defined, for all ${}^{\alpha}i, {}^{\beta}j \in \mathcal{V}''$, by $A({}^{\beta}j, {}^{\alpha}i) = a_{\alpha}P({}^{\beta}j, {}^{\alpha}i)$, with the convention that $P({}^{\beta}j, {}^{\alpha}i) = 0$ if there is no macro-step between ${}^{\beta}j$ and ${}^{\alpha}i$, B and C are $|\mathcal{V}''|$-dimensional vectors equal to $(P({}^{\alpha}i, End)_{{}^{\alpha}i\in\mathcal{V}''})$, and $(a_{\gamma}P({}^{\alpha}i, End)_{{}^{\alpha}i\in\mathcal{V}''})$ respectively. The solution of system (I) (resp. (II)) is unique. Futhermore the vector $(w_{End}({}^{\alpha}i))_{{}^{\alpha}i\in\mathcal{V}''}$ is the solution of (II) and the vector $(T_{End}({}^{\alpha}i))_{{}^{\alpha}i\in\mathcal{V}''}$ is the solution of (I).

The proof is given in Appendix 2.

Let us point out that Prop. 5 allows us to compute $T_{End}(0)$ (hence $\mathcal{A}$; see Prop. 4) in a *parametric* manner with the parameters of "cost" a_{α} appearing in matrix A and row vector C.[8]

Example 6. *Let us continue the previous example. Matrix P'' is obtained from P' by removing last line and last column. Matrix $\mathcal{I}'' - P''$ is:*

$$\begin{pmatrix} 1 & -1 & 0 & 0\\ 0 & 1 & -1+pq & -p\\ 0 & 0 & pq & -p\\ 0 & 0 & 0 & 1\end{pmatrix}$$

In this example, $\mathcal{V}'' = \{0, {}^{I}1, {}^{F}1, {}^{G}3\}$.
The vector $B : (B_0, B_{{}^{I}1}, B_{{}^{F}1}, B_{{}^{G}3})$ is given by:
$(P''(0, End), P''({}^{I}1, End), P''({}^{F}1, End), P''({}^{G}3, End))$.
We have: $B = (0, 0, 0, q)$.
The vector $W : (W_0, W_{{}^{I}1}, W_{{}^{F}1}, W_{{}^{G}3})$ is given by:
$(\mathcal{I}'' - P'')^{-1}B$.
We have: $W = (1, 1, 1, q)$.
The vector $C : (C_0, C_{{}^{I}1}, C_{{}^{F}1}, C_{{}^{G}3})$ is given by:
$(a_BP''(0, End), a_BP''({}^{I}1, End), a_BP''({}^{F}1, End), a_BP''({}^{G}3, End))$.
We have: $C = (0, 0, 0, TDq)$.
The $|\mathcal{V}''| \times |\mathcal{V}''|$-matrix A is equal to:

$$\begin{pmatrix} 0 & 0 & 0 & 0\\ 0 & 0 & T1(1-pq) & TDp\\ 0 & 0 & T1(1-pq) & TDp\\ 0 & 0 & 0 & 0\end{pmatrix}$$

The $|\mathcal{V}''|$-vector $X : (X_0, X_{{}^{I}1}, X_{{}^{F}1}, X_{{}^{G}3})$ is given by: $(\mathcal{I}'' - P'')^{-1}(AW + C)$.
We have:
$X = (\frac{1-pq}{pq}T1 + 2TD, \frac{1-pq}{pq}T1 + 2TD, \frac{1-pq}{pq}T1 + 2TD, qTD)$
Therefore:

$$\begin{aligned}\mathcal{A} &= X_0\\ &= \tfrac{1-pq}{pq}T1 + 2TD.\end{aligned}$$

4.4 Absorption Time for (More) General Automata

The method applies to the class of DPTA, which is very restrictive. In order to make our analysis useful for more general timed automata, let us sketch out how to transform a general timed automaton, say G, into a DPTA, say Δ, such that, *under certain constraints*:

[6] In example 1, in addition to parametric costs $T1$ and TD, there are parametric *probabilities* p and q, but this is specific to the example.

[7] $w^m_{End}({}^{\alpha}i)$ is the "probability"of the cone issued from ${}^{\alpha}i$.

[8] In example 1, in addition to parametric costs $T1$ and TD, there are parametric *probabilities* p and q, but this is specific to the example.

Δ "approximates" correctly G, in the sense that the absorption time in Δ, seen as a Markov chain, is an upper (resp. lower) bound of the maximum (resp. minimum) absorption time of G, seen as a Markov decision process.

We first observe that nondeterminism (in time) in G often occurs when it waits an undeterminate time before executing an enabled transition. The idea for removing this indecision, is to add a new timing *parameter*, and force the system to wait exactly for its corresponding value. This may be done by adding an outgoing condition of the form $x = a$, as well as the invariant $x \leq a$ in the relevant node.

Another source of nondeterminism occurs when there are two (or more) transitions outgoing from a same node. With additional information, the choice may disappear: for example, under appropriate constrained values of the parameters, only one transition may be enabled at a time. The system may then reduce to the sequential composition of two (more) deterministic subsystems. Such timing constraints may often be inferred from a separate analysis of the *non* probabilistic version of the model (where each distribution has been replaced by a set of nondeterministic transition). For example, in the non probabilistic version of BRP protocol [6], the authors show that, by adding timers and appropriate constraints on the timeouts, it is possible to preclude undesirable behaviors.[9] If additionally, one imposes that the channel component take always *exactly* (instead of 'at most') TD units of time to transmit a message, then one gets a DPTA model (after reintroduction of probabilities). Such a model captures the *worst* behavior of the original model, and allows to compute, e.g., the maximum expected sending time. More details will be given in the long version of this paper.

5 Example: CSMA/CD As a DPTA

We consider the protocol CSMA/CD, as modeled in [8, 11]. We take the case when there are two stations S_1 and S_2 trying to send data at the same time through a channel C. As in [11], we model the case when initially the stations collide. If there is no collision, then, after λ time units, the station finishes sending its data (event *end*). On the other hand, if there is a collision (event *cd*), the station attempts to retransmit the packet where the scheduling of the retransmission is determined by a *truncated binary exponential backoff* process. The number of slots (each, equal to 2σ time units) that the station waits after the nth transmission failure is chosen as a uniformly distributed random integer in $\{0, 1, ..., 2^{k+1} - 1\}$, where $k = \min(n, bcmax)$, $bcmax$ being a parameter taken here equal to 2 as in experiments of [11]. The overall model is given by the parallel composition of three probabilistic timed automata representing the senders S_1, S_2 and the channel C. These automata, borrowed from [11], are described in Fig. 5 and Fig. 6.

We are interested in computing $\mathcal{A}$, the expected time for one of the sender to finish the sending of its data. The time $\mathcal{A}$ corresponds to the expected time taken by the system from state $1 \equiv Collide_c.Transmit_1.Transmit_2$.
to state $4 \equiv Init_c.Done_1.Wait_2$ (or symmetrically: $4 \equiv Init_c.Done_2.Wait_1$).

We simplify the model by replacing the guard $y \leq \sigma$ of the transition between $Init_c$ and $Collide_c$ of the channel's automaton by the guard $y = \sigma$. Intuitively, this corresponds to focus on a "worst" scheduler.

Also, we assume that $\lambda \geq 2\sigma$, in order to avoid the Zeno behaviors.

Under these conditions, we claim that the system behaves as a DPTA. This DPTA is sketched out in Fig. 3. We distinguish two cases of collision detection cd_A and cd_B. The event cd_A corresponds to the case where the random numbers assigned to the backoffs of S_1 and S_2 are equal, which will lead to a subsequent new collision; The event cd_B corresponds to the case where the random numbers are distinct: the sender with the smaller backoff will then successfully send his message.

Additionally, we have to distinguish the states corresponding to the case where only one collision has occured $bc = 1$, from the cases where a 2nd collision has yet occurred $bc = 2$.

The multiplicity of states (and corresponding transitions) is just indicated by dashed lines on Fig.3, in a sketchy way. The DPTA can be easily transformed in a graph of macro-steps, whose (rough) description is given in Fig.4. We compute $\mathcal{A}$, using proposition 5: the macro-steps graph is, in that particular case, a Markov chain, therefore we know the vector W is the vector with all components equal to 1 (for all $^{\alpha}i$, the $^{\alpha}i$-component of W is the probability of the cone issued from $^{\alpha}i$ and thus is equal to 1). We use `Maple` to compute the inverse of matrix $I - P''$, which is a $(59, 59)$-matrix and the formal computations $X = (I - P'')^{-1}(AW + C)$ (see appendix 3). We get:
$\mathcal{A} = X_{End}(0) = (30/7)\sigma + \lambda$.

[9] For example, they add two clocks x, u satisfying invariants $x \leq T1$ and $u \leq TD$ respectively, and they introduce constraint $T1 > 2TD$, which makes impossible to have both a frame and an acknowledgement in transit at the same time.

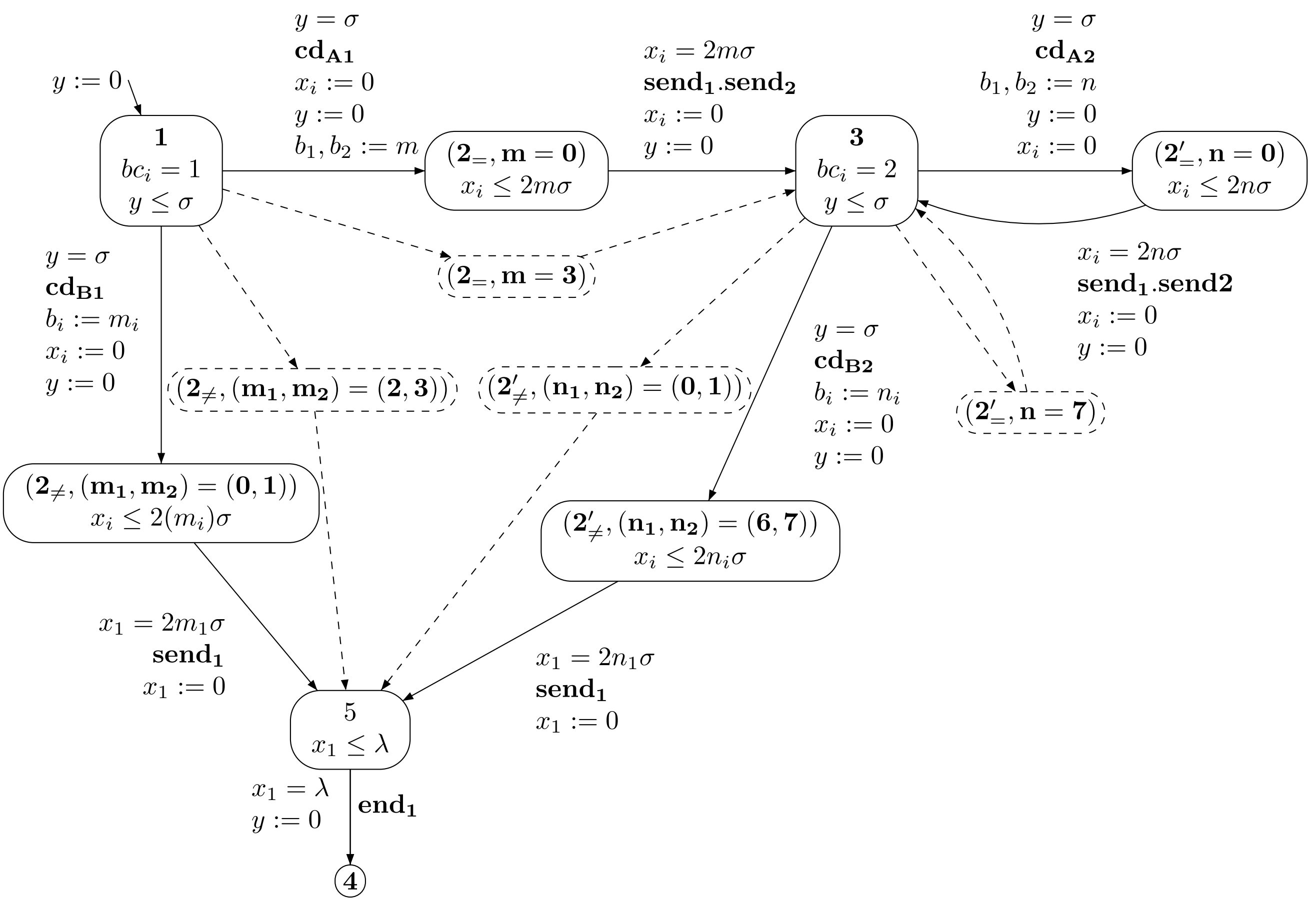

Figure 3. Product Timed Automaton for CSMA-CD

We have also checked the correctness of this formula by implementing the system in PRISM, and computing the relevant expected time for different values of λ and σ.

6 Final Remarks

We have explained how to compute the expected time of reaching an absorbing node, for a special class of probabilistic timed automata (DPTA). Under certain conditions, this method is parametric. The DPTA model is very restrictive, since it does not allow for nondeterminism. However, we claim that one can often transform a general probabilistic timed automata G into a DPTA, which computes (or overapproximates) the expected absorption time of G for its "worst" scheduler. Roughly speaking, this can be done by adding parametric delays, and assuming appropriate constraints on these delays. Such constraints can be inferred from the analysis of the non-probabilistic version of G. The method has been applied to examples of the literature (CSMA/CD, BRP protocols).

References

[1] R. Alur and D.L. Dill. A theory of timed automata. *TCS 126*, pages 183–235, 1994.

[2] C. Baier and M. Kwiatkowska. Model Checking for Probabilistic Branching Time Logic with Fairness. *Distributed Computing 11*, 1998.

[3] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, 1989.

[4] D.P. Bertsekas and J.N. Tsitsiklis. An Analysis of Stochastic Shortest Path Problems. *Mathematics of Operations Research 16:3*, pages 580–595, 1991.

[5] A. Bianco and L. de Alfaro. Model Checking of Probabilistic and Nondeterministic Systems. In *FST&TCS'95, LNCS 1026, Springer*, 1995.

[6] P. D'Argenio, J.-P. Katoen, T.C. Ruys, and J. Tretmans. The Bounded Retransmission Protocol Must Be on Time! In *TACAS, LNCS 1217, Springer*, pages 416–431, 1997.

[7] L. de Alfaro. Computing Minimum and Maximum Reachability Times in Probabilistic Systems. In *CONCUR 99, LNCS 1664, Springer*, pages 66–81, 1999.

[8] M. Kwiatkowska, G. Norman, D. Parker, and J. Sproston. Performance Analysis of Probabilistic Timed Automata using Digital Clocks. *Formal Methods in System Design 29*, pages 33–78, 2006.

[9] M. Kwiatkowska, G. Norman, R. Segala, and J. Sproston. Automatic Verification of Real-time Systems with Discrete Probability Distributions. *TCS 282*, pages 101–150, 2002.

[10] R. Segala and N. Lynch. Probabilistic Simulations for Probabilistic Processes. *Nordic Journal of Computing 2:2*, pages 250–273, 1995.

[11] Web site: PRISM. CSMA/CD Protocol. In `http://www.cs.bham.ac.uk/ dxp/prism/casestudies`, 2006.

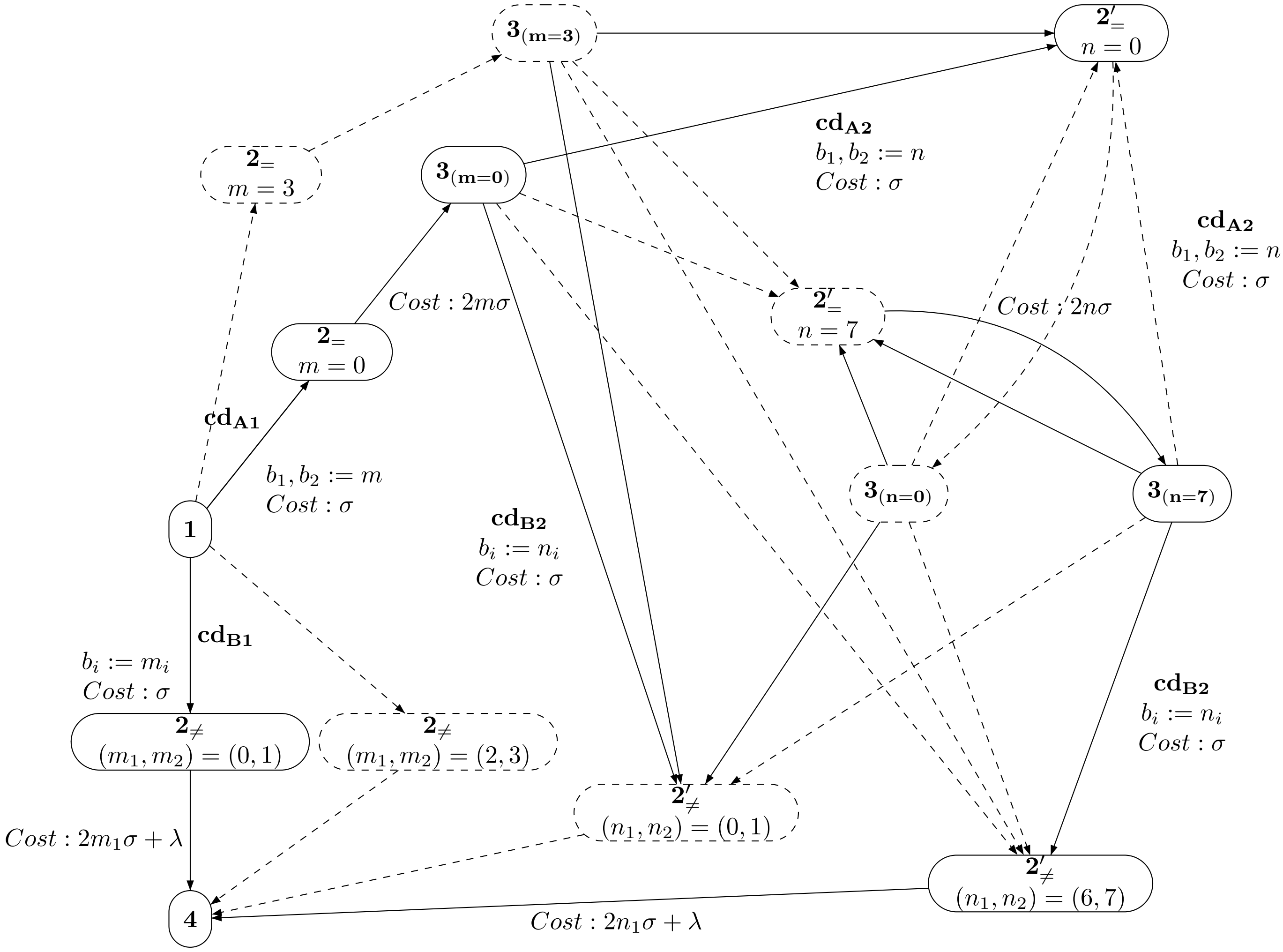

Figure 4. Graph of Macro-Steps for CSMA/CD.

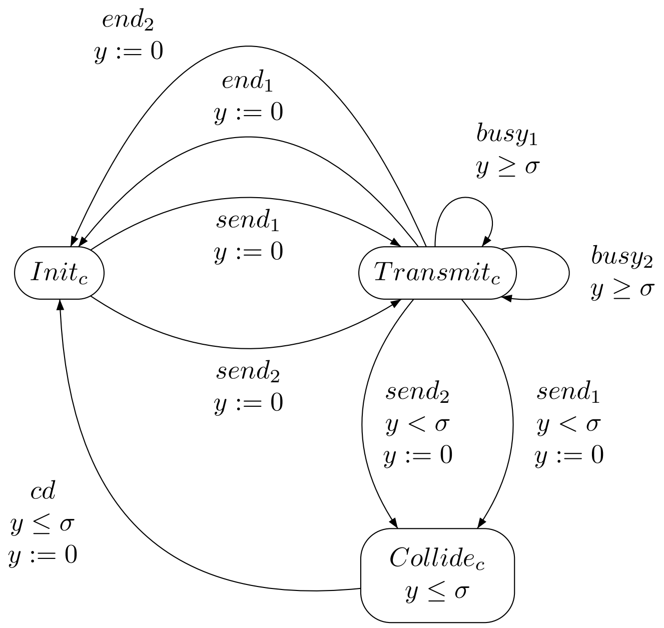

Figure 5. The Channel Automaton.

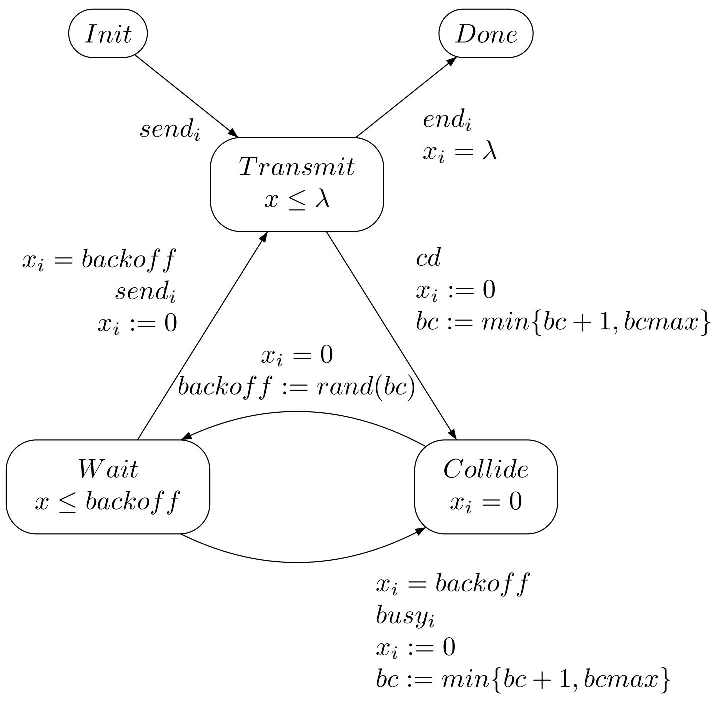

Figure 6. The Sender Automaton.

Appendix 1: Proof of Proposition 4

We are now going to show that $T_{End}(0)$ is equal to the expected time $\mathcal{A}$ taken by the DPTA to reach *End*. The equality will be established by mapping macro-paths to one-step paths.

Let ${}^{\beta}j, {}^{\alpha}i$ two points of $\mathcal{V}'$. Given a macro-path $\tau \in \mathcal{C}(0, {}^{\beta}j)$, and a macro-step: ${}^{\beta}j \Rightarrow {}^{\alpha}i$, let $\tau.({}^{\beta}j \Rightarrow {}^{\alpha}i)$ denote the macro-path of $\mathcal{C}(0, {}^{\alpha}i)$ corresponding to τ followed by the macro-step.

Likewise, given a path $\omega \in \Omega_{{}^{\beta}j}$ and a macro-step via σ: ${}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i$, let $\omega.({}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i)$ denote the path of $\Omega_{{}^{\alpha}i}$ corresponding to ω followed by the sequence σ to ${}^{\alpha}i$.

Definition 6. *Given a point ${}^{\alpha}i \in \mathcal{V}'$ and a macro-path $\tau \in \mathcal{C}(0, {}^{\alpha}i)$, the* associated set of paths*, denoted by $\mathcal{G}(\tau)$, is the subset of $\Omega_{{}^{\alpha}i}$ defined by:*
- $\mathcal{G}(\tau)$ is the path of null length, if $|\tau| = 0$, and
- $\mathcal{G}(\tau) = \{\omega.({}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i) \mid \omega \in \mathcal{G}(\tau'), \sigma \in S({}^{\beta}j, {}^{\alpha}i)\}$,
if $\tau = \tau'.({}^{\beta}j \Rightarrow {}^{\alpha}i)$, for some ${}^{\beta}j \in \mathcal{V}'$, $\tau' \in \mathcal{C}(0, {}^{\beta}j)$ and macro-step ${}^{\beta}j \Rightarrow {}^{\alpha}i$.

Lemma 1. *For all macro-path $\tau \in \mathcal{C}(0, {}^{\alpha}i)$, we have:*

$$\delta(\tau) = Cost(\omega), \quad \text{for all } \omega \in \mathcal{G}(\tau). \quad (III)$$

$$\pi(\tau) = \sum_{\omega \in \mathcal{G}(\tau)} Pr(\omega). \quad (IV)$$

Proof. Let us prove (IV) by induction on the length m of τ. The base case $m = 0$ is trivial (both sides being equal to 1). Consider a macro-path τ' of length $m+1$, of the form $\tau.({}^{\beta}j \Rightarrow {}^{\alpha}i)$, where τ is a macro-path of $\mathcal{C}(0, {}^{\beta}j)$ of length m. We have:
$\pi(\tau') = \pi(\tau)P({}^{\beta}j, {}^{\alpha}i)$.
On the other hand, we have:
$\Sigma_{\omega \in \mathcal{G}(\tau')} Pr(\omega)$
$= \Sigma_{\omega \in \mathcal{G}(\tau)} \Sigma_{\sigma \in S({}^{\beta}j, {}^{\alpha}i)} Pr(\omega.({}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i))$
$= \Sigma_{\omega \in \mathcal{G}(\tau)} \Sigma_{\sigma \in S({}^{\beta}j, {}^{\alpha}i)} (Pr(\omega) P_{\sigma}({}^{\beta}j, {}^{\alpha}i))$
(using Prop. 2)
$= \Sigma_{\omega \in \mathcal{G}(\tau)} Pr(\omega) \Sigma_{\sigma \in S({}^{\beta}j, {}^{\alpha}i)} P_{\sigma}({}^{\beta}j, {}^{\alpha}i)$
$= (\Sigma_{\omega \in \mathcal{G}(\tau)} Pr(\omega)) P({}^{\beta}j, {}^{\alpha}i)$
$= \pi(\tau) P({}^{\beta}j, {}^{\alpha}i)$ (using induction hypothesis).
Hence $\pi(\tau')$ and $\Sigma_{\omega \in \mathcal{G}(\tau')} Pr(\omega)$ are equal. This achieves the proof of the induction step for (IV), hence the proof of (IV).

Likewise, let us prove (III) by induction on the length m of τ. The base case $m = 0$ is trivial (both sides being equal to 0). Consider a macro-path τ' of length $m+1$, of the form $\tau.({}^{\beta}j \Rightarrow {}^{\alpha}i)$, where τ is a macro-path of $\mathcal{C}(0, {}^{\beta}j)$ of length m. We have:
$\delta(\tau') = \delta(\tau) + a_{\alpha}$.
On the other hand, we have, for all $\omega' \in \mathcal{G}(\tau')$, i.e., all ω' of the form $\omega.({}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i)$, for some $\omega \in \mathcal{G}(\tau)$ and some $\sigma \in S({}^{\beta}j, {}^{\alpha}i)$:
$Cost(\omega')$
$= Cost(\omega.({}^{\beta}j \Rightarrow_{\sigma} {}^{\alpha}i))$
$= Cost(\omega) + T_{\sigma}({}^{\beta}j, {}^{\alpha}i)$ (using Prop. 2)
$= Cost(\omega) + a_{\alpha}$
$= \delta(\tau) + a_{\alpha}$ (using induction hypothesis).
Hence $\delta(\tau')$ and $Cost(\omega')$ are equal. This achieves the proof of the induction step for (III), hence the proof of (III). □

Lemma 2. *For all ${}^{\alpha}i \in \mathcal{V}'$:*

$$\Omega_{{}^{\alpha}i} = \bigcup_{\tau \in \mathcal{C}(0, {}^{\alpha}i)} \mathcal{G}(\tau).$$

Proof. By definition, we have, for all point ${}^{\alpha}i$:

$$\Omega_{{}^{\alpha}i} \supset \bigcup_{\tau \in \mathcal{C}(0, {}^{\alpha}i)} \mathcal{G}(\tau).$$

It remains to show that, for all ${}^{\alpha}i \in \mathcal{V}'$, each $\omega \in \Omega_{{}^{\alpha}i}$ belongs to $\mathcal{G}(\tau)$, for some $\tau \in \mathcal{C}(0, {}^{\alpha}i)$. The proof is by induction on the length of ω. If ω is of null length, then $i = 0$, and $\omega \in \mathcal{G}(\tau)$ for the macro-path τ of null length. Otherwise, ω is of the form:
$0 \rightarrow_{\beta_1} j_1 \rightarrow_{\beta_2} j_2 \cdots \rightarrow_{\alpha} i$.
Let $x_{\alpha} = a_{\alpha}$ be the guard of transition α, and let k be the last index such that β_k resets x_{α} to 0. (Such an index exists because β_1 resets all the clocks to 0.) The path ω is of the form $\omega'.({}^{\beta_k}j_k \Rightarrow_{\sigma} {}^{\alpha}i)$, for some ${}^{\beta_k}j_k \in \mathcal{V}'$, $\omega' \in \Omega_{{}^{\beta_k}j_k}$ and $\sigma \in S({}^{\beta_k}j_k, {}^{\alpha}i)$. By induction hypothesis, $\omega' \in \mathcal{G}(\tau')$ for some $\tau' \in \mathcal{C}(0, {}^{\beta_k}j_k)$. Hence ω belongs to $\mathcal{G}(\tau'.({}^{\beta_k}j_k \Rightarrow {}^{\alpha}i))$. This achieves the proof of the induction step. □

Proposition 4. *We have:*

$$\mathcal{A} = T_{End}(0).$$

Proof. We have:
$\mathcal{A} = \Sigma_{\omega \in \Omega_{End}} Pr(\omega) Cost(\omega)$
$= \Sigma_{\tau \in \mathcal{C}(0, End)} \Sigma_{\omega \in \mathcal{G}(\tau)} Pr(\omega) Cost(\omega)$
(by Lemma 2)
$= \Sigma_{\tau \in \mathcal{C}(0, End)} \delta(\tau) \Sigma_{\omega \in \mathcal{G}(\tau)} Pr(\omega)$
(by (III) of Lemma 1)
$= \Sigma_{\tau \in \mathcal{C}(0, End)} \pi(\tau) \delta(\tau)$
(by (IV) of Lemma 1)
$= T_{End}(0)$. □

Appendix 2: Proof of Proposition 5

Lemma 3. *The matrix $\mathcal{I}'' - P''$ is invertible, where $\mathcal{I}''$ denotes the identity matrix on $\mathcal{V}'' \times \mathcal{V}''$.*

Proof. Let us first show $\lim_{m\to\infty} P^m({}^\beta j, {}^\alpha i) = 0$, for all ${}^\beta j, {}^\alpha i \in \mathcal{V}''$. Using an induction, as in Lemma 2, we have: $\lim_{m\to\infty} P^m({}^\beta j, {}^\alpha i)$ is the limit of the probability of the set of paths

$$ {}^\beta j \to_{\alpha_1} s_1 \to_{\alpha_2} s_2 \cdots \to_{\alpha_{l-1}} s_{l-1} \to_\alpha i, $$

for some nodes $s_1, ..., s_{l-1}$ and some transitions $\alpha_1, ..., \alpha_{l-1}$ distinct from α_{abs}) when l tends to infinity. As End is absorbing, this limit is 0.

From the nullity of the limit of $P^m({}^\beta j, {}^\alpha i)$ for all ${}^\beta j, {}^\alpha i \in \mathcal{V}''$, it follows that $\rho(P''^m) < 1$, for m large enough, where ρ is the spectral radius. The fact that $\rho(P''^m) < 1$ for some m, then implies $\rho(P'') < 1$, hence the invertibility of $\mathcal{I}'' - P''$ (see [3]). □

Proposition 5. Let $X = (X_{{}^\alpha i})_{{}^\alpha i \in \mathcal{V}''}$ (resp. $W = (W_{{}^\alpha i})_{{}^\alpha i \in \mathcal{V}''}$) be two vectors with real components. Let consider the systems:

$$ X = P''X + AW + C \quad (I), $$
$$ W = P''W + B \quad (II), $$

where A'' is the $|\mathcal{V}''| \times |\mathcal{V}''|$-matrix defined, for all ${}^\alpha i, {}^\beta j \in \mathcal{V}''$, by $A({}^\beta j, {}^\alpha i) = a_\alpha P({}^\beta j, {}^\alpha i)$, with the convention that $P({}^\beta j, {}^\alpha i) = 0$ if there is no macro-step between ${}^\beta j$ and ${}^\alpha i$, B and C are $|\mathcal{V}''|$-dimensional vectors equal to $(\sum_{{}^\gamma End} P({}^\alpha i, {}^\gamma End)_{{}^\alpha i \in \mathcal{V}''})$, and $(\sum_\gamma a_\gamma P({}^\alpha i, {}^\gamma End)_{{}^\alpha i \in \mathcal{V}''})$ respectively. The solution of system (I) (resp. (II)) is unique. Futhermore the vector $(w_{End}({}^\alpha i))_{{}^\alpha i \in \mathcal{V}''}$ is the solution of (II) and the vector $(T_{End}({}^\alpha i))_{{}^\alpha i \in \mathcal{V}''}$ is the solution of (I).

Proof. Let us show that the solution of system $(I) - (II)$ is *unique*. By Lemma 3, the matrix $\mathcal{I}'' - P''$ is invertible. Hence, the solution of (I) (resp. (II)) is *unique* and equal to $(\mathcal{I}'' - P'')^{-1}(A''W'' + C'')$ (resp. $(\mathcal{I}'' - P'')^{-1}B''$).

Let us show that $(w_{End}({}^\alpha i))_{{}^\alpha i}$ is the solution of equation (II). Let us suppose ${}^\alpha i \in \mathcal{V}''$. We have:

$$ \begin{aligned} w^{m+1}_{End}({}^\alpha i) &= \textstyle\sum_{\tau \in \mathcal{C}^{\le m+1}({}^\alpha i, End)} \pi(\tau) \\ &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) \sum_{\tau \in \mathcal{C}^{\le m}({}^\beta j, End)} \pi(\tau) \\ &\quad + \textstyle\sum_{{}^\gamma End} P({}^\alpha i, {}^\gamma End) \\ &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) w^m_{{}^\gamma End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\gamma End} P({}^\alpha i, {}^\gamma End) \end{aligned} $$

Therefore:

$$ w^{m+1}_{End}({}^\alpha i) = \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) w^m_{End}({}^\beta j) + \sum_{{}^\gamma End} P({}^\alpha i, {}^\gamma End). $$

By taking the limit of the two sides, when m tends to ∞, we have, for ${}^\alpha i \in \mathcal{V}''$:

$$ w_{End}({}^\alpha i) = \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) w_{End}({}^\beta j) + \sum_{{}^\gamma End} P({}^\alpha i, {}^\gamma End). $$

Hence, for all ${}^\alpha i \in \mathcal{V}''$, we have:

$$ w_{End}({}^\alpha i) = \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) w_{End}({}^\beta j) + B_{{}^\alpha i}, $$

with $B_{{}^\alpha i} = \sum_{\gamma End} P({}^\alpha i, {}^\gamma End)$. It follows that $(w_{End}({}^\alpha i))_{{}^\alpha i \in \mathcal{V}''}$ is the solution of (II).

Let us show that $(T_{End}({}^\alpha i))_{{}^\alpha i}$ is the solution of equations (I). Let us suppose ${}^\alpha i \in \mathcal{V}''$. We have:

$$ \begin{aligned} & T^{m+1}_{End}({}^\alpha i) = \textstyle\sum_{\tau \in \mathcal{C}^{\le m+1}({}^\alpha i, End)} \delta(\tau)\pi(\tau) = \\ & \textstyle\sum_{{}^\beta j \in \mathcal{V}''} \sum_{\tau \in \mathcal{C}^{\le m}({}^\beta j, End)} (a_\beta + \delta(\tau)) P({}^\alpha i, {}^\beta j) \pi(\tau) \\ &\quad + \textstyle\sum_{{}^\gamma End} a_\gamma P({}^\alpha i, {}^\gamma End) \\ &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} a_\beta P({}^\alpha i, {}^\beta j) \sum_{\tau \in \mathcal{C}^{\le m}({}^\beta j, End)} \pi(\tau) \\ &\quad + \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) \sum_{\tau \in \mathcal{C}^{\le m}({}^\beta j, {}^\gamma End)} \delta(\tau)\pi(\tau) \\ &\quad + \textstyle\sum_{{}^\gamma End} P({}^\alpha i, {}^\gamma End) \\ &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} a_\beta P({}^\alpha i, {}^\beta j) w^m_{End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) T^m_{End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\gamma End} a_\gamma P({}^\alpha i, {}^\gamma End) \end{aligned} $$

By taking the limit of the two sides, when m tends to ∞, we have, for ${}^\alpha i \in \mathcal{V}''$:

$$ \begin{aligned} T_{End}({}^\alpha i) &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} a_\beta P({}^\alpha i, {}^\beta j) w_{End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) T_{End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\gamma End} a_\gamma P({}^\alpha i, {}^\gamma End) \end{aligned} $$

Therefore:

$$ \begin{aligned} T_{End}({}^\alpha i) &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} a_\beta P({}^\alpha i, {}^\beta j) w_{End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P({}^\alpha i, {}^\beta j) T_{End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\gamma End} a_\gamma P({}^\alpha i, {}^\gamma End) \end{aligned} $$

Then we have:

$$ \begin{aligned} T_{End}({}^\alpha i) &= \textstyle\sum_{{}^\beta j \in \mathcal{V}''} A({}^\alpha i, {}^\beta j) w_{{}^\gamma End}({}^\beta j) \\ &\quad + \textstyle\sum_{{}^\beta j \in \mathcal{V}''} P'({}^\alpha i, {}^\beta j) T_{{}^\gamma End}({}^\beta j) \\ &\quad + C_{{}^\alpha i} \end{aligned} $$

with
$A({}^\alpha i, {}^\beta j) = a_\beta P({}^\alpha i, {}^\beta j)$ for all ${}^\beta j \in \mathcal{V}''$ and
$C_{{}^\alpha i} = \sum_{{}^\gamma End} a_\gamma P'({}^\alpha i, {}^\gamma End)$

It follows that $(T_{End}({}^\alpha i))_{{}^\alpha i \in \mathcal{V}''}$ is the solution of (I). □

Appendix 3: The computation of the time absorption of CSMA/CD

We describe the computation of the time absorption to reach the final state "End" on the CSMA/CD protocol.

```
> restart:
> with(linalg):
```

The matrix 60×60 - matrix P is defined such that for all ${}^{\beta}j, {}^{\alpha}i \in \mathcal{V}'$, we have $P'({}^{\beta}j, {}^{\alpha}i) = P({}^{\beta}j, {}^{\alpha}i)$. The set $\mathcal{V}'$ corresponds to the set of states of the macro-steps graph of CSMA/CD. We will assimilate an integer $i \in \{0, ..., 60\}$ to each state of the graph of macro-steps:

- state 1 is represented by 1.
- i from 2 to 5 represents respectively states $(2_=, m = j)$, $j \in \{0, ..., 3\}$.
- i from 6 to 9 represents respectively states $3_{(m=j)}$, $j \in \{0, ..., 3\}$.
- i from 10 to 17 represents respectively states $(2'_=, n = j)$, $j \in \{0, ..., 7\}$.
- i from 18 to 25 represents respectively states $3_{(n=j)}$, $j \in \{0, ..., 7\}$.
- i from 26 to 53 represents respectively states $(2'_{\neq}, (n_1, n_2) = (j, k)$, $j \in \{0, ..., 6\}$, $k \in \{0, ..., 7\}$ and $n_1 \leq n_2$.
- i from 54 to 59 represents respectively states $(2_{\neq}, (m_1, m_2) = (j, k)$, $j \in \{0, ..., 2\}$, $k \in \{0, ..., 3\}$ and $n_1 \leq n_2$.
- state $4 = End$ is represented by 60.

The code in Maple of the matrix P corresponds to the below description:

```
> P:=matrix(60,60,0):
> for i from 2 to 5 do P[1,i]:=1/16;
P[i,(4+i)]:=1; end do:
> for i from 6 to 9 do for j from 10 to
17 do P[i,j]:=1/64; end do; end do;
> for i from 10 to 17 do P[i,i+8]:=1; end
do:
> for i from 18 to 25 do for j from 10 to
17 do P[i,j]:=1/64; end do: end do:
> for i from 6 to 9 do for j from 26 to
53 do P[i,j]:=1/32; end do: end do:
> for i from 18 to 25 do for j from 26 to
53 do P[i,j]:=1/32; end do: end do:
> for i from 54 to 59 do P[1,i]:=1/8; end
do:
> for i from 26 to 59 do P[i,60]:=1: end
do:
> P[60,60]:=1:
> evalm(P):
```

The 59 × 59 - matrix P'' is obtained from P by removing the last line and last column corresponding to the point $4 = End$. We compute below the 59 × 59 - matrix $\mathcal{I}'' - P''$ denoted by Q. The code in Maple is:

```
> Q:=matrix(59,59,0):
> for i from 1 to 59 do for j from 1 to
59 do if (i=j) then Q[i,j]:=-P[i,j]+1;
else Q[i,j]:=-P[i,j]; end if; end do: end
do:
> evalm(Q):
```

The next code describe the computation of the inverse denoted by R of the matrix $\mathcal{I}'' - P''$ (i.e Q):

```
> R:=inverse(Q):
> evalm(R):
```

The 59-vector $C = (Cost(i, End)P(i, End))_{i \in \mathcal{V}''}$ with $\mathcal{V}'' = \mathcal{V}' \setminus End$ is described by the code below:

```
> C:=vector(59,0):
> for i from 26 to 32 do C[i]:=lambda
end do: for i from 33 to 38 do
C[i]:=2*sigma+lambda end do: for i
from 39 to 43 do C[i]:=4*sigma+lambda
end do: for i from 44 to 47 do
C[i]:=6*sigma+lambda end do: for i
from 48 to 50 do C[i]:=8*sigma+lambda
end do: for i from 51 to 52 do
C[i]:=10*sigma+lambda end do:
C[53]:=12*sigma+lambda: for i from 54
to 56 do C[i]:=lambda end do: for i from
57 to 58 do C[i]:=2*sigma+lambda end do:
C[59]:=4*sigma+lambda:
```

The 59 × 59 - matrix $A(j, i) = Cost(j, i)P(j, i)$ with $i, j \in \mathcal{V}''$ is:

```
> A:=matrix(59,59,0): for i from 2 to 5
do A[1,i]:=sigma/16; end do:
> for i from 6 to 9 do
A[i-4,i]:=(2*(i-6))*sigma end do:
> for i from 6 to 9 do for j from 26 to
53 do A[i,j]:=sigma/32: end do: end do:
```

```
>  for i from 18 to 25 do for j from 26 to
53 do A[i,j]:=sigma/32; end do:  end do:
```

```
>  for i from 6 to 9 do for j from 10 to
17 do A[i,j]:=sigma/64; end do:  end do:
```

```
>  for i from 10 to 17 do
A[i,i+8]:=(2*(i-10))*sigma; end do:
```

```
>  for i from 18 to 25 do for j from 10 to
17 do A[i,j]:=sigma/64; end do:  end do:
```

```
>  for i from 54 to 59 do A[1,i]:=sigma/8
end do:
```

Finally, we resolve the system of the Proposition 5 considering the fact that W is the unity vector 1:
$X = P''X + AW + C$ and $W = P''W + B$

```
>  evalm(A):
```

```
>  Z:=multiply(A,W):
```

```
>  H:=matadd(Z,C):
```

```
>  X:=multiply(R,H);
```

The solution of the system is the below 59 - vector X:

$$\mathrm{X} := \left[\frac{30}{7}\sigma+\lambda,\ 1,\ \frac{57}{7}\sigma+\lambda,\ \frac{71}{7}\sigma+\lambda,\ \frac{85}{7}\sigma+\lambda,\ 1,\ 1,\ 1,\ 1,\ 1,\ \frac{57}{7}\sigma+\lambda,\ \frac{71}{7}\sigma+\lambda,\ \frac{85}{7}\sigma+\lambda,\ \frac{99}{7}\sigma+\lambda,\ \frac{113}{7}\sigma+\lambda,\ \frac{127}{7}\sigma+\lambda,\ \frac{141}{7}\sigma+\lambda,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ \lambda,\ \lambda,\ \lambda,\ \lambda,\ \lambda,\ \lambda,\ \lambda,\ 2\sigma+\lambda,\ 2\sigma+\lambda,\ 2\sigma+\lambda,\ 2\sigma+\lambda,\ 2\sigma+\lambda,\ 2\sigma+\lambda,\ 4\sigma+\lambda,\ 4\sigma+\lambda,\ 4\sigma+\lambda,\ 4\sigma+\lambda,\ 4\sigma+\lambda,\ 6\sigma+\lambda,\ 6\sigma+\lambda,\ 6\sigma+\lambda,\ 6\sigma+\lambda,\ 8\sigma+\lambda,\ 8\sigma+\lambda,\ 8\sigma+\lambda,\ 10\sigma+\lambda,\ 10\sigma+\lambda,\ 12\sigma+\lambda,\ \lambda,\ \lambda,\ \lambda,\ 2\sigma+\lambda,\ 2\sigma+\lambda,\ 4\sigma+\lambda\right]$$

Therefore, the absorption time $T_{End}(0)$ to reach the state "End'' is given by $X[0] = 30/7\sigma + \lambda$.