Quantitative Analysis of Weighted Transition Systems

Claus Thrane, Uli Fahrenberg, Kim G. Larsen

Dept. of Computer Science, Aalborg University, Denmark

Abstract

We present a general framework for the analysis of quantitative and qualitative properties of reactive systems, based on a notion of weighted transition systems. We introduce and analyze three different types of distances on weighted transition systems, both in a linear and a branching version. Our quantitative notions appear to be reasonable extensions of the standard qualitative concepts, and the three different types introduced are shown to measure inequivalent properties.

When applied to the formalism of weighted timed automata, we show that some standard decidability and undecidability results for timed automata extend to our quantitative setting.

Key words: Quantitative analysis, weighted transition systems, weighted timed automata, simulation, trace inclusion, hemimetrics

1. Introduction

The research presented in this work is motivated by the *Challenge on embedded systems design*, posed by Henzinger and Sifakis in [7]. Henzinger and Sifakis express the need for a coherent theory of embedded systems design, where concern for physical constraints is supported by the computational models used to model software, thus achieving a more heterogeneous approach to design. Highly distilled, Henzinger and Sifakis call for a new mathematical basis for systems modeling which facilitates modeling of behavioural properties as well as environmental constraints.

Analysis and verification of concurrent and reactive systems [1] is a well-established research field, a branch of which is referred to as *implementation verification*: verification of systems design based on *behavioural equivalence checking*. This approach requires a model of the system and specification, as well as a procedure for checking whether the two are related with respect to some equivalence. The choice of this equivalence relation reflects what one wants to observe and how. Classical examples of such relations include *trace inclusion* and various types of *simulation*, see *e.g.* the survey provided in [12]. Correspondingly, the models which are analyzed must encompass all the relevant information to facilitate the analysis. Specifically, the formalism used to model the system must be rich enough to express the characteristics of the system, in order for the analysis to prove or refute the proposed equivalence.

In a quantitative setting, equivalences are replaced by real-valued *distances*; intuitively the problem is lifted from a decision problem to a search problem, *i.e.* from deciding on $\{true, false\}$ to computing a distance $\varepsilon \in \mathbb{R}_{\geq 0}$. A distance of 0 (zero) is given to instances which are accepted by the binary decision procedure, and the meaning of values $\varepsilon > 0$ is that the instance is not equal to the specification, yet related up to some error margin given by the distance ε.

1.1. Motivation

Although the standard approach to implementation verification may be adequate when analyzing qualitative properties such as behaviour of systems, it is arguably insufficient for reasoning about their *quantitative* aspects. Indeed, it can be argued that in a setting where system models and properties include both discrete and continuous, *i.e.* quantitative, information, *e.g.* real-time or probabilistic systems, a quantitative approach to implementation verification is necessary.

As an example of how quantitative models and analysis may be applied in an industrial setting, consider the design of a *hybrid vehicle* using two or more power sources, *e.g.* electricity and petrol. Not only would we like to be able to verify the behaviour of the vehicle: steering, breaks etc., but also quantities such as performance, *e.g.* in terms of horse power and the ratio of energy consumption from the different sources. Hence given the option to configure the fuel management system or suspension, a quantitative analysis should reveal not only the qualitative property, *i.e.* whether the alternative component will supply fuel or not, or whether the suspension will hold, but also the impact on fuel consumption.

Generalizing the above example, quantitative methods are also increasingly used for modeling *optimal scheduling and control* problems for hybrid systems. In this setting, quantitative approaches to implementation verification, and to controller generation, are essential. When generating controllers for hybrid systems for example, *implementability* and *robustness* are important issues, and both need a quantitative approach to verification.

1.2. Contribution

We present a general framework for the analysis of quantitative and qualitative properties of reactive systems, based on a notion of *weighted transition systems*. Weighted transition systems can be used for specifying the semantics of systems with quantitative and qualitative properties, such as weighted timed automata for example, which feature both weights and time.

We introduce and analyze three different types of distances on weighted transition systems, but note that other interesting types may be treated in a similar manner. The three types are

- *point-wise* distance, which measures the largest individual difference between systems,
- *accumulated* distance, which measures the sum of (absolute) differences accumulated during executions of the systems, and

- *maximum-lead* distance, which measures the largest distance between accumulated differences occurring during executions of the systems.

We find that there are subtle equivalences and differences between these distances.

All three kinds of distances are defined and analyzed both in a linear setting, *i.e.* extending the standard notion of *trace inclusion*, and in a branching version, generalizing the notion of *simulation*. We find that the usual relation between simulation and trace inclusion generalizes to our quantitative setting.

We apply our quantitative framework to implementation verification for *weighted timed automata*, and we collect evidence that the standard result on undecidability of timed language inclusion for timed automata can be lifted to our quantitative setting, and that on the other hand (and again generalizing standard results), simulation distances are computable for weighted timed automata.

2. Hemi-metrics

We need to recall a few basic facts about asymmetric distances before we can begin our journey into the quantification of trace inclusion and simulation between weighted transition systems. For this section, X denotes a general set. Also, $\mathbb{R}_{\geq 0}$ is the set of non-negative real numbers, $\mathbb{R}_+$ the set of positive real numbers, and 2^X denotes the power set of X.

Recall first the definition of hemimetric, and note that for us, a hemimetric can assume the value ∞:

Definition 1. *Let $d : X \times X \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ be a function for which $d(x, x) = 0$ for all $x \in X$, and which satisfies the triangle inequality*

$$d(x, y) + d(y, z) \geq d(x, z) \qquad \text{for all } x, y, z \in X$$

Then d is called a hemimetric.

We will need two different notions of equivalence of hemimetrics later; for metrics, these are standard and can be found in any textbook:

Definition 2. *Hemimetrics d_1, d_2 on X are said to be*

- topologically equivalent *provided that for all $x \in X$ and all $\varepsilon \in \mathbb{R}_+$, there exists $\delta \in \mathbb{R}_+$ such that $d_1(x, y) < \delta$ implies $d_2(x, y) < \varepsilon$ and $d_2(x, y) < \delta$ implies $d_1(x, y) < \varepsilon$ for all $y \in X$,*

- Lipschitz equivalent *if there exist $m, M \in \mathbb{R}_+$ such that $m d_1(x, y) \leq d_2(x, y) \leq M d_1(x, y)$ for all $x, y \in X$.*

Recall also that topological equivalence is the same as asking the identity function $(X, d_1) \to (X, d_2)$ to be continuous, and Lipschitz equivalence is the same as requiring it to be a Lipschitz function; hence Lipschitz equivalence is stronger than topological equivalence. We shall later see that topological equivalence is useful for transferring *negative* results from one hemimetric to another, whereas Lipschitz equivalence is used for positive results.

We shall use the following standard construction to lift hemimetrics from a set to its set of subsets:

Definition 3. *The* Hausdorff hemimetric *d on 2^X associated with a hemimetric d on X is given by*

$$d(A,B) = \sup_{x \in A} \inf_{y \in B} d(x,y)$$

Note the following alternative formulation, which follows straight from the definition:

Proposition 4. *For a hemimetric d on X, $A, B \subseteq X$, and $\varepsilon \in \mathbb{R}_+$, we have $d(A,B) \leq \varepsilon$ if and only if for any $x \in A$ there exists $y \in B$ for which $d(x,y) \leq \varepsilon$.*

For distance 0, we have the following useful fact:

Lemma 5. *For a hemimetric d on X and $A, B \subseteq X$, we have $d(A,B) = 0$ if and only if $\bar{A} \subseteq \bar{B}$, where $\bar{A}$, $\bar{B}$ denote the closures of A, respectively B, in the topology induced on X by d.*

3. Weighted transition systems and weighted timed automata

We now define our notion of *weighted transition system* (WTS), essentially an extension of the standard concept of (labeled) transition system [11], which have been used to introduce operational semantics for a wide range of systems. The intention of WTS is to describe a system's behaviour as well as quantitative properties in terms of *weights* and *lengths*. Recall that a transition system is a quadruple (S, s_0, Γ, R) consisting of a set S of states with initial state $s_0 \in S$, a finite set Γ of labels, and a set of transitions $R \subseteq S \times \Gamma \times S$.

Definition 6. *A* weighted transition system *is a triple $(\mathcal{S}, w, lg)$, where*

- *$\mathcal{S} = (S, s_0, \Gamma, R)$ is a transition system,*
- *$w : R \to \mathbb{R}_{\geq 0}$ assigns weights to transitions, and*
- *$lg : \Gamma \to \mathbb{R}_{\geq 0}$ assigns lengths to labels.*

We write $s \xrightarrow{\alpha, w} s'$ whenever $(s, \alpha, s') \in R$ and $w(s, \alpha, s') = w$, and $s \not\to$ if there is no transition (s, α, s') in R for any α and s'. The presence of both weights of transitions and lengths of labels is to some degree redundant, but we will see later that it is indeed merited.

We lift the standard notions of path and trace to WTS:

Definition 7. *Let $\mathcal{S} = ((S, s_0, \Gamma, R), w, lg)$ be a WTS and $s \in S$. A* path from *s in $\mathcal{S}$ is a (possibly infinite) sequence $((s_0, \alpha_0, s_1), (s_1\alpha_1, s_2), \dots)$ of transitions $(s_i, \alpha_i, s_{i+1}) \in R$ with $s_0 = s$. A* (weighted) trace from *s is a sequence $((\alpha_0, w_0), (\alpha_1, w_1), \dots)$ of pairs $(\alpha_i, w_i) \in \Gamma \times \mathbb{R}_{\geq 0}$ for which there exists a path $((s_0, \alpha_0, s_1), (s_1\alpha_1, s_2), \dots)$ from s for which $w_i = w(s_i, \alpha_i, s_{i+1})$.*

The set of traces from a state s is denoted $\mathsf{Tr}(s)$. Given a trace σ, we denote by $U(\sigma) \in \Gamma^\omega$ its label sequence (*i.e.* the associated unweighted trace), by $lg(\sigma) \in \mathbb{N} \cup \{\infty\}$ its length, and by σ_i its i'th label-weight pair.

As finite models of weighted transition systems we use *weighted timed automata*. Recall [2, 4] that a timed automaton is essentially a finite automaton augmented with a set $\mathcal{C}$ of clocks, which are used for imposing invariants on locations and guards on transitions and hence controlling when these are enabled. These invariants and guards are given as clock constraints, where the set $\Psi(\mathcal{C})$ of clock constraints is generated by the following grammar:

$$\psi ::= x \bowtie k \mid x - y \bowtie k \mid \psi_1 \wedge \psi_2 \qquad x, y \in \mathcal{C}, k \in \mathbb{Z}, \bowtie \in \{\leq, <, =, >, \geq\}$$

Weighted timed automata (WTA), introduced in [3, 5], are an extension of timed automata with weights:

Definition 8. *A weighted timed automaton is a tuple $(L, \ell_0, \mathcal{C}, I, E, r)$ where:*

- *L is a finite set of locations, with ℓ_0 as the initial location,*
- *$\mathcal{C}$ is a finite set of clocks,*
- *$I : L \to \Psi(\mathcal{C})$ assigns invariants to locations,*
- *$E \in L \times \Psi(\mathcal{C}) \times 2^{\mathcal{C}} \times \mathbb{N} \times L$ is a set of weighted edges, and*
- *$r : L \to \mathbb{N}$ is a location weight-rate function.*

We write $\ell \xrightarrow[p]{\psi, C} \ell'$ instead of $(\ell, \psi, C, p, \ell') \in E$.

An example of a WTA, taken from [5], is depicted in Figure 1. It represents a simple production plant with three different levels of productivity *Low*, *Medium*, and *High* and rates modeling the cost of operation at each level. The plant will automatically decrease in production level (action d) if unattended for 3 time units.

The operational semantics of a timed automaton is given as a *timed transition system*, *i.e.* an infinite transition system with both discrete (*switch*) and continuous (*delay*) transitions. Similarly, the semantics of a WTA is usually defined by a timed transition system with weights on transitions. Here we use a slightly different approach, translating a WTA into a WTS with lengths:

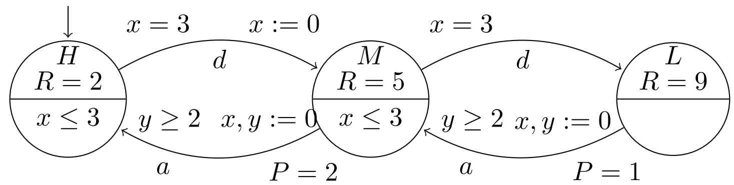


Figure 1: Simple production system example

Definition 9. *The semantics of a weighted timed automaton A is given by the weighted transition system $[\![A]\!] = ((S, \{\star\} \cup \mathbb{R}_{\geq 0}, T), w, lg)$ with*

$$S = \big\{(\ell, v) \in L \times \mathbb{R}_{\geq 0}^C \bigm| v \models I(\ell)\big\}$$
$$T = \big\{(\ell, v) \xrightarrow{\star, p} (\ell', v') \bigm| \exists\, \ell \xrightarrow[p]{\psi, C} \ell' \in E : v \models \psi, v' = v[C \leftarrow 0]\big\}$$
$$\cup \big\{(\ell, v) \xrightarrow{\delta, r} (\ell, v + \delta) \bigm| \forall\, \delta' \in [0, \delta] : v + \delta' \models I(\ell), r(\ell) = r\big\}$$
$$lg(\star) = 1 \qquad lg(\delta) = \delta \text{ for } \delta \in \mathbb{R}_{\geq 0}$$

4. Quantitative Analysis

In this section we introduce our quantitative analysis of WTS, both in a linear and branching setting. For ease of exposition we concentrate on *trace inclusion* and *strong simulation* here and defer treatment of both trace equivalence and bisimulation, and of weak relations, to other work. We shall introduce three different quantitative notions of trace inclusion and of strong simulation, all filling in the gap between the *unweighted* and the *weighted* relations, which we recall below:

Definition 10. *Let $((S, s_0, \Gamma, R), w, lg)$ be a WTS. A relation $R \subseteq S \times S$ is*

- *an* unweighted simulation *provided that for all $(s, t) \in R$ and $s \xrightarrow{\alpha, c} s'$, also $t \xrightarrow{\alpha, d} t'$ for some $d \in \mathbb{R}_{\geq 0}$ and $(s', t') \in R$,*
- *a* (weighted) simulation *provided that for all $(s, t) \in R$ and $s \xrightarrow{\alpha, c} s'$, also $t \xrightarrow{\alpha, c} t'$ for some $(s', t') \in R$.*

We write

- *$s \preccurlyeq^u t$ if $(s, t) \in R$ for some unweighted simulation R,*
- *$s \preccurlyeq t$ if $(s, t) \in R$ for some weighted simulation R.*

Also, we write

- *$s \leq^u t$ if $U(\mathsf{Tr}(s)) \subseteq U(\mathsf{Tr}(t))$,*
- *$s \leq t$ if $\mathsf{Tr}(s) \subseteq \mathsf{Tr}(t)$.*

We shall fill in the gap between unweighted and weighted relations using (asymmetric) *distance functions* $R : S \times S \to \mathbb{R}_{\geq 0} \cup \{\infty\}$. Any of the distances defined below will obey the properties given in the following definition; note that we require them to be hemimetrics:

Definition 11. *A hemimetric $R : S \times S \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ defined on the states of a WTS $((S, s_0, \Gamma, R), w, lg)$ is called*

- *a* trace distance *if $s \leq t$ implies $R(s,t) = 0$, and $s \not\leq^u t$ implies $R(s,t) = \infty$,*
- *a* simulation distance *if $s \preccurlyeq t$ implies $R(s,t) = 0$, and $s \not\preccurlyeq^u t$ implies $R(s,t) = \infty$.*

As usual, we can generalize distances between states of a single WTS to distances between two different WTS by taking their disjoint union.

Our distance functions are essentially based on three different metrics on the set of sequences of real numbers. Throughout the paper, these are referred to as *point-wise* (1), *accumulated* (2), and *maximum-lead* (3) distances, respectively. For sequences $a = (a_i)$, $b = (b_i)$ these are defined as follows:

$$d_{\bullet}(a,b) = \sup_i \big\{ |a_i - b_i| \big\} \tag{1}$$

$$d_{+}(a,b) = \sum_i |a_i - b_i| \tag{2}$$

$$d_{\pm}(a,b) = \sup_i \Big\{ \Big| \sum_{j=0}^{i} a_j - \sum_{j=0}^{i} b_j \Big| \Big\} \tag{3}$$

The intuition behind these metrics is that $d_{\bullet}$ measures the largest individual difference of sequence entries, d_+ measures the accumulated sum of (the absolute values of) the entries' differences, and $d_\pm$ measures the largest *lead* of one sequence over the other, *i.e.* the maximum difference in accumulated values. Hence the maximum-lead distance of two sequences is the same as the point-wise distance of their partial sum sequences.

Besides the above three, other metrics on sequences of reals are also of interest, and we expect that linear and branching distances of WTS based on these other metrics can be developed similarly to the ones we introduce in this paper.

In the following we will consider *discounted* distances, where the contribution of each step is decreased exponentially over time. To this end, we fix a discounting factor $\lambda \in [0,1]$; as extreme cases, $\lambda = 1$ means that the future is undiscounted, and $\lambda = 0$ means that only the present is considered.

Also, we fix a WTS $(\mathcal{S}, w, lg)$ with $\mathcal{S} = (S, s_0, \Gamma, R)$.

4.1. Linear distances

We will now introduce our quantitative trace distances. In the following we denote by $s_i(\sigma) = \sum_{j=0}^{i} lg(\sigma_j)$ the accumulated lengths of labels up to the i'th step; recall that $U(\sigma)$ denotes the label sequence of a trace σ. The *cost* of a label-weight pair $\sigma_i = (\alpha_i, w_i)$ is given by $\mathsf{c}(\sigma_i) = w_i \cdot lg(\alpha_i)$.

Definition 12. *For traces σ, σ', the point-wise, accumulating, and maximum-lead* distances *are given by* $|\sigma,\sigma'|_\bullet = |\sigma,\sigma'|_+ = |\sigma,\sigma'|_\pm = \infty$ *if* $U(\sigma) \neq U(\sigma')$, *and for* $U(\sigma) = U(\sigma')$,

$$|\sigma,\sigma'|_\bullet = \sup_i \left\{ \lambda^{s_i(\sigma)} |\mathsf{c}(\sigma_i) - \mathsf{c}(\sigma'_i)| \right\}$$

$$|\sigma,\sigma'|_+ = \sum_i \lambda^{s_i(\sigma)} |\mathsf{c}(\sigma_i) - \mathsf{c}(\sigma'_i)|$$

$$|\sigma,\sigma'|_\pm = \sup_i \left\{ \lambda^{s_i(\sigma)} \Big| \sum_{j=0}^{i} \mathsf{c}(\sigma_j) - \sum_{j=0}^{i} \mathsf{c}(\sigma'_j) \Big| \right\}$$

Observe that the above distances on traces are symmetric; they are indeed metrics on the set of traces. This is not the case when lifted to states:

Definition 13. *For states $s, t \in S$, the point-wise, accumulating and maximum-lead* trace distances *are given by*

$$|s,t|_\bullet = \sup_{\sigma \in \mathsf{Tr}(s)} \inf_{\sigma' \in \mathsf{Tr}(t)} |\sigma,\sigma'|_\bullet \qquad |s,t|_+ = \sup_{\sigma \in \mathsf{Tr}(s)} \inf_{\sigma' \in \mathsf{Tr}(t)} |\sigma,\sigma'|_+$$

$$|s,t|_\pm = \sup_{\sigma \in \mathsf{Tr}(s)} \inf_{\sigma' \in \mathsf{Tr}(t)} |\sigma,\sigma'|_\pm$$

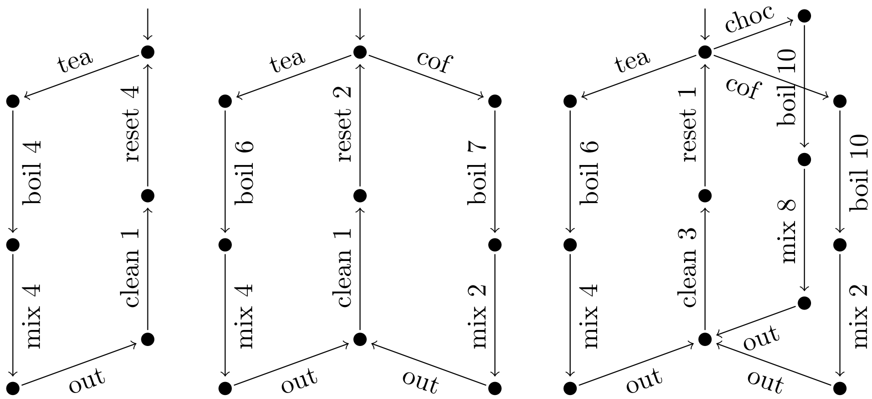
Note that this is precisely the Hausdorff-hemimetric construction from Definition 3, hence it can be generalized to other distances between traces. Also, it is quite natural, *cf.* Proposition 4. It can easily be shown that the distances defined above are indeed trace distances in the sense of Definition 11.

Example 1. *To illustrate differences between the three trace distances introduced above, consider the three WTS models of beverage machines depicted in Figure 2; a Tea maker $\mathsf{M_T}$, a Tea and Coffee maker $\mathsf{M_{TC}}$ and a Tea, Coffee and Chocolate maker $\mathsf{M_{TCC}}$. In the figure, all edges have length 1, and edges without specified weight have weight 0.*

The production of a beverage consists of six operations: Selecting the drink, boiling the water, mixing the beverage, outputting the finished product, self cleaning, and resetting. Each operation consumes a certain amount of power depending on its implementation by electrical components. Weights thus model power consumption, and are given in such a way as that in more powerful machines, some operations, as e.g. boiling, require more power, whereas some other, as e.g. resetting, require less.

By design of the beverage machines, there are unweighted trace inclusions $\mathsf{M_T} \le^u \mathsf{M_{TC}} \le^u \mathsf{M_{TCC}}$; any behaviour of a "lesser" machine can be emulated qualitatively by a "better" one. What is less obvious is how they compare in power consumption.

Noting that any infinite behaviour in the beverage machines is cyclic in loops of width 6, we can introduce some ad-hoc notation to simplify calculations. Let $|\mathsf{M_T}, \mathsf{M_{TC}}|^6_\bullet$ denote point-wise distance from $\mathsf{M_T}$ to $\mathsf{M_{TC}}$ when only traces of



(a) M_T (b) M_TC (c) M_TCC

Figure 2: Three beverage machines

length at most 6 *are considered, and similarly for the other machines and distances. For a (realistic) discount factor of* $\lambda = .90$, *the point-wise distances can be computed as follows:*

$$
\begin{aligned}
|\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_\bullet &= \sup_i \left\{ |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_\bullet^6 \cdot \lambda^{6i} \right\} = |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_\bullet^6 &&= 1.80 \\
|\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_\bullet &= \sup_i \left\{ |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 \cdot \lambda^{6i} \right\} = |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 &&= 1.80 \\
|\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_\bullet &= \sup_i \left\{ |\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 \cdot \lambda^{6i} \right\} = |\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 &&= 2.70
\end{aligned}
$$

For the accumulating distances,

$$
\begin{aligned}
|\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_+ &= \sum_i |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_\bullet^6 \cdot \lambda^{6i} = |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_\bullet^6 \frac{1}{1-\lambda^6} &&\approx 2.52 \\
|\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_+ &= \sum_i |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 \cdot \lambda^{6i} = |\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 \frac{1}{1-\lambda^6} &&\approx 8.80 \\
|\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_+ &= \sum_i |\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 \cdot \lambda^{6i} = |\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_\bullet^6 \frac{1}{1-\lambda^6} &&\approx 7.41
\end{aligned}
$$

Similarly, the maximum-lead distances can be computed as follows:

$$
\begin{aligned}
|\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TC}|_\pm &\approx 1.62 \\
|\mathsf{M}_\mathsf{T}, \mathsf{M}_\mathsf{TCC}|_\pm &\approx 2.62 \\
|\mathsf{M}_\mathsf{TC}, \mathsf{M}_\mathsf{TCC}|_\pm &\approx 3.34
\end{aligned}
$$

The following lemma provides recursive bounds on trace distances and will be useful as motivation for the definition of branching distance below.

Lemma 14. *For states $s, t \in S$,*

$$|s,t|_{\bullet} \le \sup_{s \xrightarrow{\alpha,c} s'} \inf_{t \xrightarrow{\alpha,d} t'} \max\big(|c-d| lg(\alpha), \lambda^{lg(\alpha)} \cdot |s',t'|_{\bullet}\big)$$
$$|s,t|_{+} \le \sup_{s \xrightarrow{\alpha,c} s'} \inf_{t \xrightarrow{\alpha,d} t'} |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_{+}$$
$$|s,t|_{\pm}^{\delta} \le \sup_{s \xrightarrow{\alpha,c} s'} \inf_{t \xrightarrow{\alpha,d} t'} \max\Big(|\delta|, \lambda^{lg(\alpha)} \cdot |s',t'|_{\pm}^{\frac{\delta+(c-d)lg(\alpha)}{\lambda^{lg(\alpha)}}}\Big)$$

Proof. We only show the proof for accumulated distance; the others are similar. If $\mathsf{Tr}(s) = \emptyset$, then $|s,t|_+ = 0$ and we are done. Otherwise, let $\sigma \in \mathsf{Tr}(s)$; we need to show that

$$\inf_{\sigma' \in \mathsf{Tr}(t)} |\sigma,\sigma'|_+ \le \sup_{s \xrightarrow{\alpha,c} s'} \inf_{t \xrightarrow{\alpha,d} t'} |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_+$$

Let π be a path from s which realizes σ, write $\pi = s \xrightarrow{\alpha,c} s_1 \to \dots$, and let σ_1 be the trace generated by the suffix of π starting in s_1. If $t \overset{\alpha}{\not\to}$, then the infimum on the right hand side of the equation is ∞, and we are done. Assume that the infimum is finite, then $t \xrightarrow{\alpha,d} t'$ for some d, t'.

Let $\varepsilon \in \mathbb{R}_+$ and $t \xrightarrow{\alpha,d} t'$ be such that

$$|c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_+ \le \inf_{t \xrightarrow{\alpha,d} t'} |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_+ + \tfrac{\varepsilon}{2}$$

and let $\sigma_1' \in \mathsf{Tr}(t')$ be such that $|\sigma_1,\sigma_1'|_+ \le |s',t'|_+ + \frac{\varepsilon}{2lg(\alpha)}$. Then

$$\begin{aligned} |\sigma,\sigma'|_+ &= |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |\sigma_1,\sigma_1'|_+ \\ &\le |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_+ + \tfrac{\varepsilon}{2} \\ &\le \inf_{t \xrightarrow{\alpha,d} t'} |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_+ + \varepsilon \end{aligned}$$

The above holds for all $\varepsilon \in \mathbb{R}_+$, hence

$$\inf_{\sigma' \in \mathsf{Tr}(t)} |\sigma,\sigma'|_+ \le \inf_{t \xrightarrow{\alpha,d} t'} |c-d| lg(\alpha) + \lambda^{lg(\alpha)} \cdot |s',t'|_+$$

and the claim follows.

4.2. Simulation distances

As usual in implementation verification, the above linear approach may not yield a sufficient correctness criterion for certain systems; moreover, there are some uncomputability issues with trace inclusion, see Section 6. Thus we now introduce quantitative extensions of simulation.

In the following we use parameterized families $\{\mathcal{R}_\varepsilon \subseteq S \times S\}$ and $\{\mathcal{R}_{\varepsilon,\delta} \subseteq S \times S\}$, *i.e.* functions $\mathbb{R}_{\ge 0} \to 2^{S \times S}$ and $\mathbb{R}_{\ge 0} \times \mathbb{R}_{\ge 0} \to 2^{S \times S}$, respectively; we shall show how these give rise to distances in Section 4.3.

Definition 15. *A family of relations* $\mathbf{R} = \{\mathcal{R}_\varepsilon \subseteq S \times S \mid \varepsilon \geq 0\}$ *is*

- *a* point-wise simulation family *provided that for all* $(s,t) \in \mathcal{R}_\varepsilon \in \mathbf{R}$ *and* $s \xrightarrow{\alpha,c} s'$*, also* $t \xrightarrow{\alpha,d} t'$ *with* $|c-d|lg(\alpha) \leq \varepsilon$ *for some* $d \in \mathbb{R}_{\geq 0}$ *and* $(s',t') \in \mathcal{R}_{\varepsilon'} \in \mathbf{R}$ *with* $\varepsilon' \leq \frac{\varepsilon}{\lambda^{lg(\alpha)}}$*,*
- *an* accumulating simulation family *provided that for all* $(s,t) \in \mathcal{R}_\varepsilon \in \mathbf{R}$ *and* $s \xrightarrow{\alpha,c} s'$*, also* $t \xrightarrow{\alpha,d} t'$ *with* $|c-d|lg(\alpha) \leq \varepsilon$ *for some* $d \in \mathbb{R}_{\geq 0}$ *and* $(s',t') \in \mathcal{R}_{\varepsilon'} \in \mathbf{R}$ *with* $\varepsilon' \leq \frac{\varepsilon - |c-d|lg(\alpha)}{\lambda^{lg(\alpha)}}$*.*

A family of relations $\mathbf{R} = \{\mathcal{R}_{\varepsilon,\delta} \subseteq S \times S \mid \varepsilon \geq 0, -\varepsilon \leq \delta \leq \varepsilon\}$ *is*

- *a* maximum-lead simulation family *provided that for all* $(s,t) \in \mathcal{R}_{\varepsilon,\delta} \in \mathbf{R}$ *and* $s \xrightarrow{\alpha,c} s'$*, also* $t \xrightarrow{\alpha,d} t'$ *with* $|\delta + (c-d)lg(\alpha)| \leq \varepsilon$ *for some* $d \in \mathbb{R}_{\geq 0}$ *and* $(s',t') \in \mathcal{R}_{\varepsilon',\delta'} \in \mathbf{R}$ *with* $\varepsilon' \leq \frac{\varepsilon}{\lambda^{lg(\alpha)}}$ *and* $\delta' \leq \frac{\delta + (c-d)lg(\alpha)}{\lambda^{lg(\alpha)}}$*.*

We write

- $s \preccurlyeq^{\bullet}_{\varepsilon} t$ *if* $(s,t) \in \mathcal{R}_\varepsilon \in \mathbf{R}$ *for some point-wise simulation family* $\mathbf{R}$*,*
- $s \preccurlyeq^{+}_{\varepsilon} t$ *if* $(s,t) \in \mathcal{R}_\varepsilon \in \mathbf{R}$ *for some accumulating simulation family* $\mathbf{R}$*,*
- $s \preccurlyeq^{\pm}_{\varepsilon} t$ *if* $(s,t) \in \mathcal{R}_{\varepsilon,0} \in \mathbf{R}$ *for some maximum-lead simulation family* $\mathbf{R}$*.*

Note that the relations defined in the last part above again can be collected into families $\preccurlyeq^{\bullet} = \{\preccurlyeq^{\bullet}_{\varepsilon} \mid \varepsilon \geq 0\}$, $\preccurlyeq^{+} = \{\preccurlyeq^{+}_{\varepsilon} \mid \varepsilon \geq 0\}$, and $\preccurlyeq^{\pm} = \{\preccurlyeq^{\pm}_{\varepsilon} \mid \varepsilon \geq 0\}$.

Some explanatory remarks regarding these definitions will be in order. For point-wise simulation, $(s,t) \in \mathcal{R}_\varepsilon$ is to mean that any computation from s can be matched by one from t with the same labels and a point-wise cost difference of at most ε. Hence the requirement that $s \xrightarrow{\alpha,c} s'$ imply $t \xrightarrow{\alpha,d} t'$ with cost difference $|c-d|lg(\alpha) \leq \varepsilon$, and that computations from the target states s', t' be matched with (inverse) discounted point-wise distance $\varepsilon' = \frac{\varepsilon}{\lambda^{lg(\alpha)}}$.

For accumulated simulation, $(s,t) \in \mathcal{R}_\varepsilon$ is interpreted so that any computation from s can be matched by one from t with the same labels and accumulated absolute-value cost difference at most ε. Hence we again require that $|c-d|lg(\alpha) \leq \varepsilon$, but now computations from the target states have to be matched by what is left of ε after $|c-d|lg(\alpha)$ has been used (and inverse discounting applied).

Maximum-lead simulation is slightly more complicated, because we need to keep track of the lead δ which one computation has accomplished over the other. Hence $(s,t) \in \mathcal{R}_{\varepsilon,\delta}$ is to mean that any computation from s which starts with a lead of δ, can be matched by a computation from t with accumulated cost difference at most ε. Thus we require that lead plus cost difference, $\delta + (c-d)lg(\alpha)$, be in-between $-\varepsilon$ and ε, and the new lead for computations from the target states is set to that value (again with inverse discounting applied).

For later use we collect the following easy facts about the above simulations:

Lemma 16.

1. *The families* $\preccurlyeq^{\bullet}$, $\preccurlyeq^{+}$ *and* $\preccurlyeq^{\pm}$ *are the largest respective simulation families.*

2. *For* $\varepsilon \le \varepsilon'$ *and* $\mathcal{R}_\varepsilon, \mathcal{R}_{\varepsilon'} \in \mathbf{R}$ *a point-wise or accumulating simulation family,* $\mathcal{R}_\varepsilon \subseteq \mathcal{R}_{\varepsilon'}$*. For* $\varepsilon \le \varepsilon'$*,* $-\varepsilon \le \delta \le \varepsilon$ *and* $\mathcal{R}_{\varepsilon,\delta}, \mathcal{R}_{\varepsilon',\delta} \in \mathbf{R}$ *a maximum-lead simulation family,* $\mathcal{R}_{\varepsilon,\delta} \subseteq \mathcal{R}_{\varepsilon',\delta}$*.*

3. *For states* $s, t \in S$ *and* $\varepsilon \le \varepsilon'$*,* $s \preccurlyeq^{\bullet}_{\varepsilon} t$ *implies* $s \preccurlyeq^{\bullet}_{\varepsilon'} t$*,* $s \preccurlyeq^{+}_{\varepsilon} t$ *implies* $s \preccurlyeq^{+}_{\varepsilon'} t$*, and* $s \preccurlyeq^{\pm}_{\varepsilon} t$ *implies* $s \preccurlyeq^{\pm}_{\varepsilon'} t$*.*

4. *For states* $s, t \in S$*,* $s \preccurlyeq t$ *implies* $s \preccurlyeq^{\bullet}_{0} t$*,* $s \preccurlyeq^{+}_{0} t$*, and* $s \preccurlyeq^{\pm}_{0} t$*.*

5. *For states* $s, t \in S$*,* $s \not\preccurlyeq^{u} t$ *implies* $s \not\preccurlyeq^{\bullet}_{\varepsilon} t$*,* $s \not\preccurlyeq^{+}_{\varepsilon} t$*, and* $s \not\preccurlyeq^{\pm}_{\varepsilon} t$ *for any* ε*.*

4.3. Branching distances

We present an alternative characterization of the above simulation relations in form of recursive equations; note that these closely resemble the inequalities of Lemma 14:

Definition 17. *For states* $s, t \in S$*, the* point-wise*,* accumulated*, and* maximum-lead branching distances *are the respective minimal fixed points to the following recursive equations:*

$$
\begin{aligned}
\wr s, t \wr_{\bullet} &= \sup_{s \xrightarrow{\alpha, c} s'} \inf_{t \xrightarrow{\alpha, d} t'} \max\big(|c-d|\,lg(\alpha), \lambda^{lg(\alpha)} \cdot \wr s', t' \wr_{\bullet}\big) \\
\wr s, t \wr_{+} &= \sup_{s \xrightarrow{\alpha, c} s'} \inf_{t \xrightarrow{\alpha, d} t'} |c-d|\,lg(\alpha) + \lambda^{lg(\alpha)} \cdot \wr s', t' \wr_{+} \\
\wr s, t \wr_{\pm} &= \wr s, t \wr_{\pm}^{0} \\
&\qquad \textit{with } \wr s, t \wr_{\pm}^{\delta} = \sup_{s \xrightarrow{\alpha, c} s'} \inf_{t \xrightarrow{\alpha, d} t'} \max\Big(|\delta|, \lambda^{lg(\alpha)} \cdot \wr s', t' \wr_{\pm}^{\frac{\delta + (c-d) lg(\alpha)}{\lambda^{lg(\alpha)}}}\Big)
\end{aligned}
$$

Again, some remarks regarding these definitions will be in order. First note that sup and inf are taken over the complete lattice $\mathbb{R}_{\ge 0} \cup \{\infty\}$ here, whence $\inf \emptyset = \infty$ and $\sup \emptyset = 0$. Thus $\wr s, t \wr_{\bullet} = 0$ in case $s \not\to$ and $\wr s, t \wr_{\bullet} = \infty$ in case $s \xrightarrow{\alpha, c}$ but $t \overset{\alpha}{\not\to}$ for some α, and similarly for the other distances.

The functionals defined by the first two equations above are endofunctions on the complete lattice of functions $S \times S \to \mathbb{R}_{\ge 0} \cup \{\infty\}$; they are easily shown to be monotone, hence minimal fixed points exist. For the last equation, the functional is an endofunction on the complete lattice $\mathbb{R} \to \big(S \times S \to \mathbb{R}_{\ge 0} \cup \{\infty\}\big)$, mapping each lead $\delta \in \mathbb{R}$ to a function $\wr \cdot, \cdot \wr_{\pm}^{\delta}$. Also this functional can be shown to be monotone and hence to have a minimal fixed point.

It is not difficult to see that the branching distances defined above are simulation distances in the sense of Definition 11. Below we show that they are closely related to the simulations of Definition 15:

Proposition 18. *For states* $s, t \in S$ *and* $\varepsilon \in \mathbb{R}_{\ge 0}$*, we have*

- $s \preccurlyeq^{\bullet}_{\varepsilon} t$ *if and only if* $\wr s, t \wr_{\bullet} \leq \varepsilon$,
- $s \preccurlyeq^{+}_{\varepsilon} t$ *if and only if* $\wr s, t \wr_{+} \leq \varepsilon$,
- $s \preccurlyeq^{\pm}_{\varepsilon} t$ *if and only if* $\wr s, t \wr_{\pm} \leq \varepsilon$.

Proof. Each of the six implications involved can be shown using standard structural-induction arguments.

Example 2. *We show a computation of the six different distances between states s_1 and t_1 in the (unlabeled) weighted transition system in Fig 3. All edges have length* 1*, edges without specified weight have weight* 0*, and the discount factor is $\lambda = .90$.*

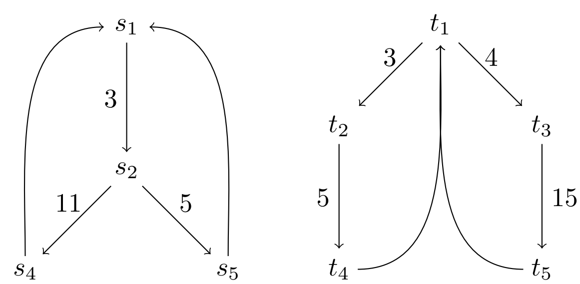


Figure 3: Example WTS

We compute trace distances first. It is easy to see that supremum trace distance is obtained for the path from s_1 which always turns left at s_2, i.e. takes the transition $s_2 \xrightarrow{11} s_4$, and then for the point-wise and accumulating trace distances, that the matching trace from t_1 giving infimum trace distance in turn is obtained for the path which always takes the transition $t_1 \xrightarrow{4} t_3$. Hence we can compute

$$|s_1, t_1|_{\bullet} = \sup_i \big\{ \max(1, 4\lambda) \cdot \lambda^{3i} \big\} = 3.60$$

$$|s_1, t_1|_{+} = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

For maximum-lead trace distance the situation is more involved. It can be shown that for this distance, an infimum trace σ' from t_1 follows the path which takes $t_1 \xrightarrow{4} t_3$, followed by $t_1 \xrightarrow{3} t_2$ three times, and then repeats $t_1 \xrightarrow{4} t_3$ indefinitely. Using this trace, we obtain

$$|s_1, t_1|_{\pm} = 4.60$$

For the branching distances, repeated application of the definition yields the *following fixed-point equations:*

$$
\begin{aligned}
\wr s_1, t_1 \wr_{\bullet} &= \inf \big\{ \max \big(6\lambda, \lambda^3 \wr s_1, t_1 \wr_{\bullet}\big), \max \big(10\lambda, \lambda^3 \wr s_1, t_1 \wr_{\bullet}\big) \big\} \\
\wr s_1, t_1 \wr_{+} &= 1 + 10\lambda + \lambda^3 \wr s_1, t_1 \wr_{+} \\
\wr s_1, t_t \wr_{\pm}^{\delta} &= \inf \Big\{ \max \Big(|\delta|, |\delta + 6\lambda|, \lambda^3 \wr s_1, t_t \wr_{\pm}^{\delta\lambda^{-3}+6\lambda^{-2}}, \lambda^3 \wr s_1, t_t \wr_{\pm}^{\delta\lambda^{-3}} \Big), \\
&\qquad \max \Big(|\delta|, |\delta - 1|, |\delta - 1 - 4\lambda|, |\delta - 1 - 10\lambda|, \\
&\qquad\quad \lambda^3 \wr s_1, t_t \wr_{\pm}^{(\delta-1)\lambda^{-3}-4\lambda^{-2}}, \lambda^3 \wr s_1, t_t \wr_{\pm}^{(\delta-1)\lambda^{-3}-10\lambda^{-2}} \Big) \Big\}
\end{aligned}
$$

Solving these, one arrives at $\wr s_1, t_1 \wr_{\bullet} = 5.40$*,* $\wr s_1, t_1 \wr_{+} \approx 36.9$*, and also* $\wr s_1, t_1 \wr_{\pm} = 5.40$.

5. Properties of distances

In this section we present a number of properties of the six distances introduced above.

5.1. Simulation versus trace distance

For the qualitative relations, simulation implies trace inclusion, *i.e.* $s \preccurlyeq^u t$ implies $s \le^u t$, and $s \preccurlyeq t$ implies $s \le t$. Below we show a natural generalization of this to our quantitative setting, where implications translate to inequalities; note that an equivalent statement of the theorem is that for any ε, $\wr s, t \wr \le \varepsilon$ implies $|s, t| \le \varepsilon$ for all three distances considered.

Theorem 19. *For all states $s, t \in S$, we have*

$$
|s, t|_{\bullet} \le \wr s, t \wr_{\bullet} \qquad |s, t|_{+} \le \wr s, t \wr_{+} \qquad |s, t|_{\pm} \le \wr s, t \wr_{\pm}
$$

Proof. This follows from Lemma 14 by an easy structural-induction argument.

Note that Example 2 shows that indeed, all distances in the equations above can be finite. Other, standard examples show however that WTS exist for which $s \not\preccurlyeq t$ and yet $s \le t$, hence $\wr s, t \wr = \infty$ and $|s, t| = 0$ for all three distances, showing the following theorem:

Theorem 20. *The distances $|\cdot, \cdot|_{\bullet}$ and $\wr \cdot, \cdot \wr_{\bullet}$ are topologically inequivalent. Similarly, $|\cdot, \cdot|_{+}$ and $\wr \cdot, \cdot \wr_{+}$, and also $|\cdot, \cdot|_{\pm}$ and $\wr \cdot, \cdot \wr_{\pm}$, are topologically inequivalent.*

5.2. Relationship between distances

The theorems below sum up the relationship between our three trace distances; note that the results depend heavily on whether or not discounting is applied. The following lemma is useful and easily shown:

Lemma 21. *For states $s, t \in S$, we have*

$$\begin{array}{lll} |s,t|_{\bullet} \le |s,t|_{+} & |s,t|_{\pm} \le |s,t|_{+} & |s,t|_{\bullet} \le 2|s,t|_{\pm} \\ \wr s,t\wr_{\bullet} \le \wr s,t\wr_{+} & \wr s,t\wr_{\pm} \le \wr s,t\wr_{+} & \wr s,t\wr_{\bullet} \le 2\wr s,t\wr_{\pm} \end{array}$$

The restrictions on traces mentioned below are understood to be applied to the sets $\mathsf{Tr}(s)$, $\mathsf{Tr}(t)$ in Definition 13.

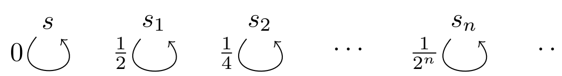
Theorem 22. *Assume the discounting factor $\lambda = 1$.*

1. *When restricted to traces of bounded length, the three trace distances $|\cdot,\cdot|_{\bullet}$, $|\cdot,\cdot|_{+}$, $|\cdot,\cdot|_{\pm}$ on S are Lipschitz equivalent.*

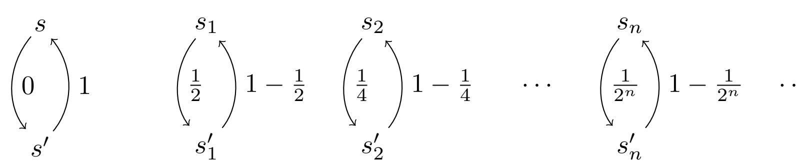
2. *For traces of unbounded length, the trace distances are mutually topologically inequivalent.*

Proof. If the length of traces is bounded above by $N \in \mathbb{N}$, then $|s,t|_{+} \le N|s,t|_{\bullet}$ for all $s, t \in S$, and the result follows with Lemma 21.

For traces of unbounded length, topological inequivalence of $|\cdot,\cdot|_{\bullet}$ and $|\cdot,\cdot|_{+}$, and of $|\cdot,\cdot|_{\bullet}$ and $|\cdot,\cdot|_{\pm}$, can be shown by the following infinite WTS:



Here we have $|s,s_n|_{+} = |s,s_n|_{\pm} = \infty$ for all n, but for any $\delta \in \mathbb{R}_{+}$ there is an n for which $|s,s_n|_{\bullet} < \delta$. Similarly, topological inequivalence of $|\cdot,\cdot|_{+}$ and $|\cdot,\cdot|_{\pm}$ is shown by the infinite WTS below:



Theorem 23. *For discounting factor $\lambda < 1$, the three trace distances $|\cdot,\cdot|_{\bullet}$, $|\cdot,\cdot|_{+}$, $|\cdot,\cdot|_{\pm}$ on S are Lipschitz equivalent.*

Proof. This is similar to the first claim of the previous theorem: For all states $s, t \in S$, we have $|s,t|_{+} \le \frac{1}{1-\lambda}|s,t|_{\bullet}$, and the result follows with Lemma 21.

Theorem 24. *For discounting factor $\lambda = 1$, the three simulation distances $\wr\cdot,\cdot\wr_{\bullet}$, $\wr\cdot,\cdot\wr_{+}$, $\wr\cdot,\cdot\wr_{\pm}$ on S are mutually topologically inequivalent. For $\lambda < 1$, they are Lipschitz equivalent.*

Proof. The first claim can be shown using the same example WTS as for the second part of the proof of Theorem 22, and for the second claim we have $\wr s,t\wr_{+} \le \frac{1}{1-\lambda}\wr s,t\wr_{\bullet}$ and can apply Lemma 21.

6. Computability

This section present our results on computability of distances on the subset of WTS generated by the set $\mathcal{A}$ of WTA.

First we provide the following easy result regarding upper bounds on distances. Recall that a timed automaton is said to be bounded if there is an upper bound M on all its reachable clock valuations, *i.e.* if every reachable state (ℓ, v) has $v(c) \leq M$ for all clocks c, and that any WTA is weighted bisimilar to a bounded WTA.

Proposition 25. *Assuming the discounting factor $\lambda < 1$, upper bounds on point-wise, accumulating, and maximum-lead trace and branching distances are computable for bounded weighted timed automata.*

Proof. If P denote the maximum edge weight, R the maximum location rate, and M the clock bound of the WTA A under investigation, then the weight of any transition in $[\![A]\!]$ is bounded by $k = \max(P, M \cdot R)$. Hence the point-wise distances, if finite, are bounded by k, and accumulating and maximum-lead distances, if finite, are bounded by $\frac{k}{1-\lambda}$.

For standard (unweighted) timed automata, it is well-known that trace inclusion is undecidable, but similarity is decidable. The following theorems provide a partial generalization to our quantitative setting:

Theorem 26. *For discounting factor $\lambda < 1$ and $|\cdot,\cdot|$ any of the three trace distances, it is undecidable whether $|s,t| = 0$ for weighted timed automata.*

Proof. By Theorem 23, we have $|s,t|_{\bullet} = 0$ if and only if $|s,t|_{+} = 0$, if and only if $|s,t|_{\pm} = 0$, hence it suffices to consider point-wise trace distance. By Lemma 5, $|s,t|_{\bullet} = 0$ if and only if $\overline{\mathsf{Tr}(s)} \subseteq \overline{\mathsf{Tr}(t)}$, where $\overline{\mathsf{Tr}(s)}$ denotes closure in the topology generated on $\mathsf{Tr}(s)$ by the point-wise distance.

Let $\bar{A}$ denote the closure of the WTA A under investigation as defined in [9], then it is easy to see that $\overline{\mathsf{Tr}(s)} = \mathsf{Tr}(s)_{\bar{A}}$, the set of traces from s in $\bar{A}$. Hence $|s,t|_{\bullet} = 0$ if and only if $\mathsf{Tr}(s)_{\bar{A}} \subseteq \mathsf{Tr}(t)_{\bar{A}}$, but by [10], language inclusion for closed timed automata, and hence also for closed WTA, is undecidable.

Theorem 27. *For discounting factor $\lambda < 1$, accumulating branching distance from deterministic to non-deterministic weighted timed automata is computable.*

The proof of this result will occupy the rest of this section and will proceed along the following lines: We will show that from deterministic WTS to (non-deterministic) WTS, calculating accumulating branching distance reduces from a sup-inf computation to an inf computation, *i.e.* a minimization problem. For WTA, we are then able to reduce this minimization problem to one of minimizing accumulated (discounted) weight of infinite paths in a corresponding product WTA, which is shown computable by results in [8].

Definition 28. *The* independent product $U \otimes V = (S, s_0, \Gamma, R, w, lg)$ *of WTS* $U = (S^U, s_0^U, \Gamma^U, R^U, w^U, lg^U)$*,* $V = (S^V, s_0^V, \Gamma^V, R^V, w^V, lg^V)$ *is defined by*

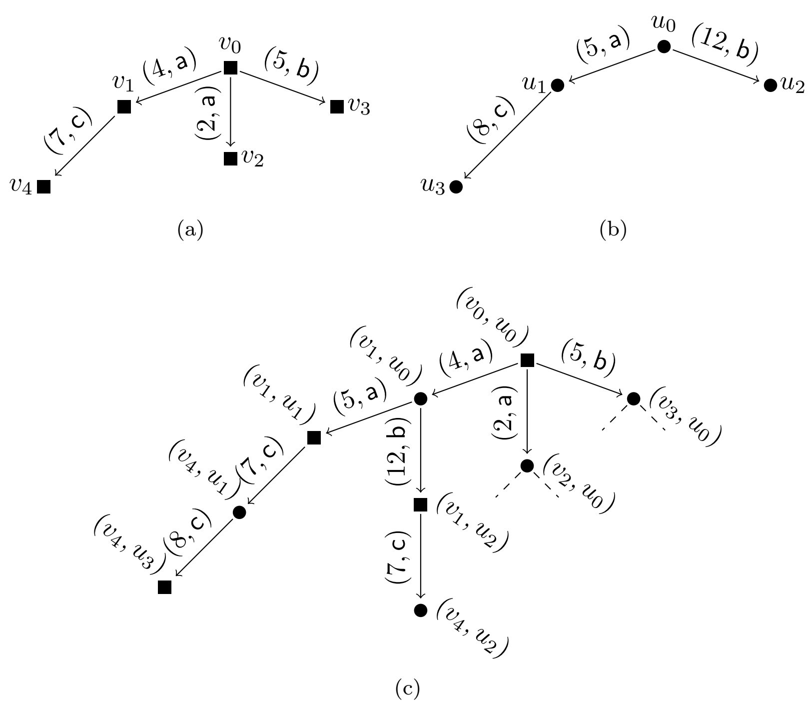


Figure 4: Independent product (c) of WTS (a) and (b).

- *$S \subseteq S^U \times S^V$ partitioned by $S_{VU} \uplus S_{UV}$ with $s_0 = (s_0^U, s_0^V) \in S_{UV}$.*
- *$\Gamma = \Gamma^U \uplus \Gamma^V$ and $R = R_{UV} \cup R_{VU}$ is constructed such that:*

  $R_{UV} = \{(u, v) \xrightarrow{\alpha} (u', v) \mid u \xrightarrow{\alpha} u' \in R^U\} \subseteq S_{UV} \times \Gamma \times S_{VU},$

  $R_{VU} = \{(u, v) \xrightarrow{\alpha} (u, v') \mid v \xrightarrow{\alpha} v' \in R^V\} \subseteq S_{VU} \times \Gamma \times S_{UV}$
- $w(t) \begin{cases} w^U(u \xrightarrow{\alpha} u') \text{ if } t = (u, v) \xrightarrow{\alpha} (u', v) \in R_{UV} \\ w^V(v \xrightarrow{\alpha} v') \text{ if } t = (u, v) \xrightarrow{\alpha} (u, v') \in R_{VU} \end{cases}$
- *$lg(\alpha) = lg^V(\alpha)$ if $\alpha \in \Gamma^V$ and $lg^U(\alpha)$ otherwise.*

Observe that this product construction, starting with a transition from U, ensures that transitions alternate. That is, whenever a transition from U is taken, it is followed by a transition from V and vice versa. Alternately, the product may be viewed as bipartite graph or a two-player game graph. Fig. 4 illustrates the construction.

Lemma 29. *For U, V WTS with U deterministic, we have $\wr U, V \wr_+ = \min_+(s_0)$, where s_0 is the initial state of $U \otimes V$ and $\min_+$ is given recursively by*

$$\min_+(s) = \inf\left\{|c - d| lg(\alpha) + \lambda^{lg(\alpha)} \min_+(s') \;\middle|\; s \xrightarrow{\alpha, c} t \xrightarrow{\alpha, d} s'\right\} \tag{4}$$

Proof. Compared to the definition of $\wr U, V \wr_+$, the initial sup part can be removed because of determinacy of U. The resulting inf computation can then be carried out in the independent product $U \otimes V$.

By introducing a similar independent-product construction on WTA, synchronized on time but not on actions, we obtain a finite product construction lifting the semantic product to the syntactic level:

Definition 30. *The* independent product $A \otimes B = (L, \ell_0, \mathcal{C} \uplus \{u\}, I, E, r)$ *of WTA* $A = (L^A, \ell_0^A, \mathcal{C}, I^A, E^A, r^A)$, $B = (L^B, \ell_0^B, \mathcal{C}, I^B, E^B, r^B)$ *is defined by*

- $L \subseteq L^A \times L^B$, *partitioned by* $L_A \uplus L_B$ *with* $\ell_0 = (\ell_0^A, \ell_0^B) \in L_A$
- $E = \rightarrow_A \cup \rightarrow_B \cup \rightarrow_{\mathsf{TL}}$ *(A-, B-, and time-locked edges) given by:*

  $\rightarrow_A = \big\{(p, q) \xrightarrow[p]{\psi, C \cup \{u\}} (p', q) \bigm| p \xrightarrow[p]{\psi, C} p' \in E(A)\big\} \subseteq L_A \times \Psi(\mathcal{C}) \times 2^{\mathcal{C}} \times \mathbb{N} \times L_B\big\}$

  $\rightarrow_B = \big\{(p, q) \xrightarrow[p]{\psi, C} (p, q') \bigm| q \xrightarrow[p]{\psi, C} q' \in E(B)\big\} \subseteq L_B \times \Psi(\mathcal{C}) \times 2^{\mathcal{C}} \times \mathbb{N} \times L_A\big\}$

  $\rightarrow_{\mathsf{TL}} = \big\{(p, q) \xrightarrow[\infty]{\mathbf{tt}, \emptyset} (p, q) \bigm| (p, q) \in L\big\}$

- $I(\ell, \ell') = \begin{cases} I^A(\ell) \wedge I^B(\ell') & \textit{for } (\ell, \ell') \in L_A \\ \{u = 0\} & \textit{otherwise} \end{cases}$
- $r(\ell, \ell') = |r^A(\ell) - r^B(\ell')|$

Note that an extra clock u is introduced in the product WTA, in order to make L_B locations *urgent*. The following is clear by construction:

Lemma 31. *For A, B WTA we have* $[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$.

We are now able to present the proof of Theorem 27:

Proof (of Theorem 27 (Sketch)). We need to compute $\min_+(s_0)$, where $s_0 = (\ell_0^A, \ell_0^B, v_0)$ is the initial state of $[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$. It can be shown that this amounts to solving the minimization problem for a corresponding discounted cost function on traces of $A \otimes B$, and as this cost function is *concave-regular* in the sense of [8], it is computable.

7. Conclusion

We have argued above that our proposed extension of the qualitative notion of trace inclusion and simulation to a quantitative setting is reasonable, and we have shown some evidence that for weighted timed automata, our generalization works as expected with respect to standard undecidability and decidability results. As a side note to this, we should mention that in [6] it is shown that a variant of our maximum-lead branching distance can be approximated with arbitrary precision for timed automata (without weights); whether a similar result can be obtained for our WTA maximum-lead branching distance is open.

We should also remark that our algorithm for computing accumulated branching distance, and also the algorithm for computing maximum-lead branching distance given in [6], are not computationally efficient. To devise feasible algorithms for these kinds of calculations remains future work.

For the three types of distances considered in this work, we have seen that trace distances can easily be introduced, whereas definition of simulation distances requires more work and involves fixed-point computations. Our Lemma 14 remedies some of these difficulties, and we expect this remedy to also be applicable for other interesting trace distances; hence a general procedure for obtaining simulation distances from trace distances should be available.

We have shown that all our three trace distances are topologically inequivalent to their corresponding simulation distance, thus measure inherently different properties. Still, and analogously to the qualitative setting, simulation distance can be used as an over-approximation of trace distance. Also, and perhaps more surprisingly, whether different trace, or simulation, distances are mutually equivalent depends on the usage of discounting. We expect all these results to also hold for other kinds of trace and simulation distances.

As a side remark to this, we should note that inequivalence of hemimetrics does not pass to subsets, hence for weighted timed automata some of the above inequivalences might turn into equivalences. This issue is potentially important for distance calculation algorithms and hence should be investigated.

We have mentioned earlier that in this work we concentrate on trace inclusion and simulation (asymmetric) distances, and of course similar treatment should be given to trace equivalence and bisimulation distances. Symmetric trace distances are easily defined as symmetrizations of the trace distances introduced here, but for the branching distances there are subtle differences between symmetrized simulation distances on the one hand and bisimulation distances on the other hand which should be analyzed in depth.

In regards to bisimulation distances, it would be appropriate to investigate whether quantitative logical characterizations of these distances can be developed. Early results indicate that this is indeed the case, but further research is needed. Quantitative logical characterizations are expected to be useful for model checking and compositional reasoning about systems with quantitative properties.

References

[1] Luca Aceto, Anna Ingólfsdóttir, Kim G. Larsen, and Jiři Srba. *Reactive Systems: Modelling, Specification and Verification.* Cambridge University Press, 2007.

[2] R. Alur and D. Dill. Automata for modeling real-time systems. In *Proc. ICALP'90*, volume 443 of *Lecture Notes in Computer Science*, pages 322–335. Springer-Verlag, 1990.

[3] R. Alur, S. La Torre, and G. J. Pappas. Optimal paths in weighted timed automata. In *Proc. 4th Int. Workshop Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 49–62. Springer-Verlag, 2001.

[4] Rajeev Alur and David L. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2):183–235, 1994.

[5] G. Behrmann, A. Fehnker, T. Hune, K. G. Larsen, P. Pettersson, J. Romijn, and F. Vaandrager. Minimum-cost reachability for priced timed automata. In *Proc. 4th Int. Workshop on Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 147–161. Springer-Verlag, 2001.

[6] Thomas A. Henzinger, Rupak Majumdar, and Vinayak Prabhu. Quantifying similarities between timed systems. In *Proc. FORMATS'05*, volume 3829 of *Lecture Notes in Computer Science*, pages 226–241. Springer-Verlag, 2005.

[7] Thomas A. Henzinger and Joseph Sifakis. The embedded systems design challenge. In *14th International Symposium on Formal Methods (FM)*, Lecture Notes in Computer Science, pages 1–15. Springer-Verlag, September 2006.

[8] Marcin Jurdzinski and Ashutosh Trivedi. Concavely-priced timed automata. In Franck Cassez and Claude Jard, editors, *FORMATS*, volume 5215 of *Lecture Notes in Computer Science*, pages 48–62. Springer-Verlag, 2008.

[9] Joël Ouaknine and James Worrell. Revisiting digitization, robustness, and decidability for timed automata. In *LICS*, pages 198–207. IEEE Computer Society, 2003.

[10] Joël Ouaknine and James Worrell. Universality and language inclusion for open and closed timed automata. In Oded Maler and Amir Pnueli, editors, *HSCC*, volume 2623 of *Lecture Notes in Computer Science*, pages 375–388. Springer, 2003.

[11] G. D. Plotkin. A structural approach to operational semantics. Technical Report DAIMI FN-19, University of Aarhus, 1981.

[12] R. J. van Glabbeek. The linear time - branching time spectrum I. The semantics of concrete, sequential processes. In *Handbook of Process Algebra*, pages 3–99. Elsevier, 2001.