Checking Linear Duration Invariants by Linear Programming

Li Xuandong ⋆ and Dang Van Hung ⋆⋆

International Institute for Software Technology
The United Nations University, P.O.Box 3058, Macau
E-Mail: {lxd,dvh}@iist.unu.edu

Abstract. In this paper, the problem of verifying a timed automaton for a Duration Calculus formula in the form of linear duration invariants [2] is addressed. We show that by linear programming, a particular class of timed automata including the class of real-time automata as a proper subset, can be checked for linear duration invariants. The so-called real-time regular expressions is introduced to express the real-time behaviour of the timed automata in this class. Using real-time regular expressions, an algorithm based on linear programming is presented for checking an automaton in the class with respect to a linear duration invariant.

1 Introduction

A central task of computer science is to provide systems which work correctly with respect to their specifications. Typically, this task involves a specification language for specifying desirable properties of the system and an implementation language (for expressing how the system is built [4].

Duration Calculus (DC) [3] is a logic for specifying and reasoning about requirements for real-time systems. It is an extension of Interval Temporal Logic which can be used to reason about integrated constraints over time-dependent and Boolean value states without explicit mention of absolute time. In DC, states are modelled as functions from reals (representing continuous time) to $\{0, 1\}$, where 1 denotes state presence, and 0 denotes state absence. For a state S, interval variable $\int S$ of DC is a function from bounded and closed intervals to reals. $\int S$ stands for the integrated duration of state S over intervals. For a bounded interval $[a, b]$ $(b \geq a)$,

$$\int S[a, b] \mathrel{\hat{=}} \int_a^b S(t)dt\,,$$

and hence $\int 1[a, b] = (b - a)$, i.e. the length of [a,b].

Using Timed Automata [1], one can describe real-time systems at a low level of detail which is close to their implementation. In this paper, we use Duration Calculus as a specification language and Timed Automata as an implementation

⋆ On leave from the Department of Computer Science, Nanjing University, P.R.China
⋆⋆ On leave from the Institute of Information Technology, Hanoi, Vietnam

language for real-time systems, and consider the correctness of an implementation of a system with respect to a specification in the form of *linear duration invariants*:

$$T \geq \int 1 \geq t \Rightarrow \bigwedge_{j=1}^{k} (\sum_{i=1}^{n} c_{ij} \int S_i \leq M_j)\,,$$

where T, t, c_{ij}, M_j range over real numbers (T may be ∞).

For real-time automata, an algorithm [2] has been presented for checking the correctness with respect to a linear duration invariant, which reduces the correctness problem to a finite set of very simple linear programming problems. On the other hand, it has been shown in [5] that the satisfaction problem of a linear duration invariant for a timed automaton can be solved by mixed integer/linear programming. Since integer programming is more complex than linear programming, it would be interesting to extend the techniques in [2] to the class of timed automata.

In this paper, we show that by linear programming, a particular class of timed automata with the class of real-time automata as a proper subset, can be checked for linear duration invariants.

The paper is organised as follows. In the next section, after recalling the notion of timed automata and linear duration invariants, we introduce the notion of real-time regular expressions for expressing the behaviour of timed automata in the class. Our algorithm is presented in Section 3. The last section is the conclusion of our paper.

2 Timed automata and linear duration invariants

2.1 Untimed automata

An (untimed) automaton consists of a finite set $\mathcal{S}$ of states, a finite set $\mathcal{E}(\mathcal{E} \subseteq \mathcal{S} \times \mathcal{S})$ of state transitions, and a set $\mathcal{S}_0 \subseteq \mathcal{S}$ of initial states. For a transition $e \in \mathcal{E}$, $e = (S, S')$, S (S') are called *pre-state* (*post-state*) of e, and denoted by $\overleftarrow{e}$ ($\overrightarrow{e}$).

The behaviour of an automaton can be represented by finite sequences of its state transitions. Let ϵ denote the empty transition sequence. ϵ represents the behaviour of the automaton before it starts. A non-empty transition sequence

$$e_1\hat{}e_2\hat{}\ldots\hat{}e_m \quad (m > 0)$$

represents a behaviour of the automaton if and only if $\overleftarrow{e_1} \in \mathcal{S}_0$ and for $i = 1, 2, \ldots, m-1$, $\overrightarrow{e_i} = \overleftarrow{e}_{i+1}$.

It is well known that the set of transition sequences which characterises the finite behaviour of an automaton can be expressed by a regular expression over state transitions [6].

2.2 Timed automata

We adopt the main concepts of the Timed Automata from [1]. A timed automaton is a conventional (untimed) automaton extended with a finite number of clock variables. Its state transitions may be labelled with clock constraints such as $a \leq x \leq b$ and/or clock reset actions such as $x := 0$, where x is a clock variable, and a, b are real numbers. The automaton starts at one of its initial states with all its clock variables initialised to 0. As time advances the values of all clock variables change, reflecting the elapsed time. A transition e is enabled if the constraints on the clock variables is satisfied, i.e, if e is labelled with a clock constraint $a \leq x \leq b$, then the current value t of clock variable x satisfies $a \leq t \leq b$. If e is labelled with a clock reset action $y := 0$, then clock variable y is reset to 0 when e takes place. Transitions are instantaneous.

In order to represent the real-time behaviour of automata, we use sequences of *time-distance stamped transitions* [2]. A time-distance stamped transition is a pair of a transition e and a nonnegative real number t; (e, t) expresses an occurrence of e at t time units after the occurrence of the last transition (or after the initiation of the automaton if e is the first transition of a sequence).

Definition 1. A time-distance stamped transition sequence

$$(e_1, t_1)\hat{}(e_2, t_2)\hat{}\ldots\hat{}(e_m, t_m) \qquad (m > 0)$$

represents a real-time behaviour of a timed automaton, if and only if

- its projection $e_1\hat{}e_2\hat{}\ldots\hat{}e_m$ is a (untimed) behaviour of the automaton, and
- its time-distance stamps satisfy the constraints on the clock variables. That is, for every transition e_i $(1 \leq i \leq m)$, for every clock constraint $a \leq x \leq b$ on e_i, if there is j $(1 \leq j \leq i-1)$ such that e_j is the last transition to reset clock variable x, i.e.
  - e_j is labelled with a clock reset action $x := 0$, and
  - every transition e_k $(j < k < i)$ is not labelled with clock reset action $x := 0$,

  then $a \leq t_{j+1} + t_{j+2} + \ldots + t_i \leq b$; otherwise (clock variable x has not been reset) $a \leq t_1 + t_2 + \ldots + t_i \leq b$.

□

For example, $(a, 20.5)\hat{}(b, 1.5)\hat{}(c, 0.2)\hat{}(d, 0.3)$ represents a real-time behaviour of the timed automaton with two clock variables (x and y) in Fig.1.

For simplicity, ϵ is overloaded to be the *empty* time-distance stamped transition sequence also. For any time-distance stamped transition sequence

$$\sigma = (e_1, t_1)\hat{}(e_2, t_2)\hat{}\ldots\hat{}(e_m, t_m)\,,$$

we denote $t_1 + t_2 + \ldots + t_m$ by $\tau(\sigma)$. By convention, $\tau(\epsilon) = 0$.

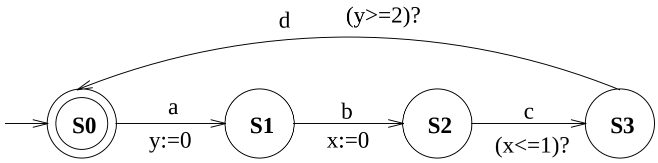

Fig. 1. A timed automaton

2.3 Real-time regular expressions

In this section, we extend regular expressions into *real-time regular expressions* to express the set of time-distance stamped transition sequences which characterises the real-time behaviour of a timed automaton. Let R^+ denote the set of nonnegative real numbers.

Definition 2. *Real-time regular expressions (RRE for short) $\mathcal{R}$ and the sets $\mathcal{L}(\mathcal{R})$ of time-distance stamped transition sequences expressed by $\mathcal{R}$ are defined recursively as:*

- ϵ *is a real-time regular expression, and* $\mathcal{L}(\epsilon) = \{\epsilon\}$.
- *If* e *is a transition, then* e *is a real-time regular expression, and*

$$\mathcal{L}(e) = \{(e, t) \mid 0 \leq t < \infty\} .$$

- *If* $\mathcal{R}_1$ *and* $\mathcal{R}_2$ *are real-time regular expressions, then* $\mathcal{R}_1 \hat{}\mathcal{R}_2$ *is a real-time regular expression, and*

$$\mathcal{L}(\mathcal{R}_1 \hat{}\mathcal{R}_2) = \{\sigma_1 \hat{}\sigma_2 \mid \sigma_1 \in \mathcal{L}(\mathcal{R}_1) \wedge \sigma_2 \in \mathcal{L}(\mathcal{R}_2)\} .$$

- *If* $\mathcal{R}_1$ *and* $\mathcal{R}_2$ *are real-time regular expressions, then* $\mathcal{R}_1 \oplus \mathcal{R}_2$ *is a real-time regular expression, and*

$$\mathcal{L}(\mathcal{R}_1 \oplus \mathcal{R}_2) = \{\sigma \mid \sigma \in \mathcal{L}(\mathcal{R}_1) \vee \sigma \in \mathcal{L}(\mathcal{R}_2)\} .$$

- *If* $\mathcal{R}$ *is a real-time regular expression, then* $\mathcal{R}^*$ *is a real-time regular expression, and*

$$\mathcal{L}(\mathcal{R}^*) = \{\sigma_1 \hat{}\ldots \hat{}\sigma_m \mid (m \geq 0) \bigwedge_{i=1}^{m} (\sigma_i \in \mathcal{L}(\mathcal{R}))\} \quad ,$$

  where $\sigma_1 \hat{}\ldots \hat{}\sigma_m \hat{=} \epsilon$ *when* $m = 0$.
- *If* $\mathcal{R}$ *is a real-time regular expression,* $a \in R^+, b \in R^+ \cup \{\infty\}, a \leq b$*, and* $b > 0$*, then* $(\mathcal{R}, [a, b])$ *is a real-time regular expression (when* $b = \infty$*,* $(\mathcal{R}, [a, b])$ *is taken to be* $(\mathcal{R}, [a, \infty))$*), and*

$$\mathcal{L}((\mathcal{R}, [a, b])) = \{\sigma \mid \sigma \in \mathcal{L}(\mathcal{R}) \wedge a \leq \tau(\sigma) \leq b\} .$$

□

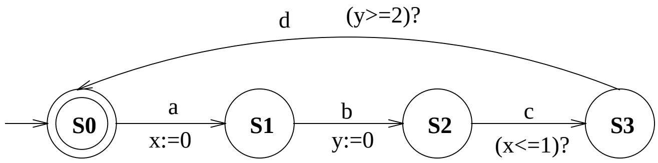

Fig. 2. A timed automaton with the behaviour unrepresentable by RREs

For example, $(a\,\hat{}\,(b\,\hat{}\,(c,[0,1])\,\hat{}\,d,[2,\infty)))^*$ is an RRE. It expresses the set of the time-distance stamped transition sequences which characterises the real-time behaviour of the timed automaton in Fig.1.

It is clear that not every timed automaton has its real-time behaviour expressed by real-time a regular expression. For example, the real-time behaviour of the timed automaton of Fig.2 can not be expressed by an RRE.

It should be noted that the class of those timed automata whose real-time behaviour can be expressed by RREs includes the class of real-time automata [2] as a proper subset, since a real-time automaton can be taken to be a timed automaton which has only one clock variable and whose every transition resets the clock variable. In this paper, we are concerning the satisfactory problem of the timed automata in this class for linear duration invariants. Before formulating this problem, we need to introduce some concepts about RREs.

If there is an RRE $\mathcal{R}_1$ in a real-time regular expression $\mathcal{R}$, then we say that $\mathcal{R}_1$ is a *sub-expression* of $\mathcal{R}$. For example, let

$$\mathcal{R} = \ ((e_1\,\hat{}\,e_2)^* \oplus ((e_3,[1,5])^*,[2,4]),[2,3])\,.$$

Then e_1, $(e_3,[1,5])$, and $((e_3,[1,5])^*,[2,4])$ are sub-expressions of $\mathcal{R}$.

For an RRE $\mathcal{R}$, if $\mathcal{L}(\mathcal{R}) = \phi$ then $\mathcal{R}$ is said to be *empty*. For example, $(((e_1,[3,4])\,\hat{}\,(e_2,[4.5,5])),[4,7])$ is an empty RRE. For an empty RRE $\mathcal{R}_1$ and for an RRE $\mathcal{R}$, it follows from the definition of RREs that

$$\begin{array}{l}\mathcal{L}(\mathcal{R}_1\,\hat{}\,\mathcal{R}) = \mathcal{L}(\mathcal{R}\,\hat{}\,\mathcal{R}_1) = \phi,\\ \mathcal{L}(\mathcal{R}_1 \oplus \mathcal{R}) = \mathcal{L}(\mathcal{R} \oplus \mathcal{R}_1) = \mathcal{L}(\mathcal{R}),\\ \mathcal{L}(\mathcal{R}_1^*) = \{\varepsilon\},\ \text{and}\ \mathcal{L}(\mathcal{R}_1,[a,b]) = \phi\,.\end{array}$$

Furthermore, for an RRE $\mathcal{R}$, it is not difficult to give an efficient algorithm for checking the emptiness of $\mathcal{R}$. Therefore, if $\mathcal{R}$ is not an empty RRE, we can find out an RRE $\mathcal{R}'$ efficiently such that there is no empty sub-expression in $\mathcal{R}'$ and that $\mathcal{L}(\mathcal{R}) = \mathcal{L}(\mathcal{R}')$. For the simplicity, from now on, unless otherwise stated, we assume that all RREs under consideration are not empty and do not have any empty sub-expression.

A *simple* RRE is a real-time regular expression having no occurrence of combinators $*$ (repetition) and $\oplus$ (union) . For example,

$$e_1\,\hat{}\,e_2\,\hat{}\,(e_3,[5,6])\,\hat{}\,(e_4\,\hat{}\,e_5,[6,9])$$

is a simple RRE.

By a *normal form* we mean an RRE of the form

$$\mathcal{A}_1 \oplus \mathcal{A}_2 \oplus \ldots \oplus \mathcal{A}_m \quad ,$$

where $\mathcal{A}_j$s are simple RREs.

2.4 Linear duration invariants

Linear duration invariants [2] form a class of DC formulae that are used to specify real-time system requirements. A linear duration invariant $\mathcal{D}$ is of the form

$$T \geq \int 1 \geq t \Rightarrow \bigwedge_{j=1}^{k} (\sum_{i=1}^{n} c_{ij} \int S_i \leq M_j) \,,$$

where T, t, c_{ij}, M_j are real numbers (T may be ∞). The premise of $\mathcal{D}$ specifies a concerned range of the observation time interval of the system, and the conclusion specifies a set of system invariants in terms of linear inequalities of state durations. A system satisfies a linear duration invariant, if and only if the integrated durations of its states satisfy the conclusion whenever the system is observed for a time period which satisfies the premise. For example, the safety-critical requirement for a gas burner [3]:

"The proportion of time when gas is leak is not more than one twentieth of elapsed time, if the system is observed for more than one minute",

can be specified by a linear duration invariant:

$$\int 1 \geq 60\text{sec} \Rightarrow 20 \int \mathit{Leak} \leq \int 1$$

where *Leak* is a Boolean function to represent the leak state of the gas burner.

Let $\mathcal{D}$ be a linear duration invariant. Let $\sigma = (e_1, t_1)\hat{}(e_2, t_2)\hat{} \ldots \hat{}(e_m, t_m)$ be a timed-distance stamped transition sequence representing a real-time behaviour of a timed automaton. For given timed behaviour, the integrated duration of state S_i, i.e. the value of $\int S_i$, can be calculated as

$$\int S_i = \sum_{u \in \alpha_i} t_u \,,$$

where $\alpha_i \hat{=} \{u \mid (1 \leq u \leq m) \wedge (\overleftarrow{e_u} \Rightarrow S_i)\}$. It follows that $\int 1 = \sum_{u=1}^{m} t_u$, as $\{u \mid (1 \leq u \leq m) \wedge (\overleftarrow{e_u} \Rightarrow 1)\} = \{1, 2, \ldots, m\}$.

Definition 3. Time-distance stamped transition sequence σ satisfies linear duration invariant $\mathcal{D}$, if

$$\bigwedge_{j=1}^{k} (\sum_{i=1}^{n} c_{ij} (\sum_{u \in \alpha_i} t_u) \leq M_j) \qquad \text{(denoted by } I\text{)}$$

when

$$T \geq \sum_{u=1}^{m} t_u \geq t \qquad \text{(denoted by } C2\text{)}\,.$$

□

Definition 4. A real-time regular expression $\mathcal{R}$ satisfies a linear duration invariant $\mathcal{D}$, denoted by $\mathcal{R} \models \mathcal{D}$, if and only if every time-distance stamped transition sequence $\sigma \in \mathcal{L}(\mathcal{R})$ satisfies $\mathcal{D}$. □

Let $\mathcal{A}$ be a simple RRE such that $\mathcal{L}(\mathcal{A}) \neq \{\epsilon\}$. By Definition 2, there are state transitions $e_1, e_2, \ldots, e_m$ $(m \geq 1)$ such that any $\sigma \in \mathcal{L}(\mathcal{A})$ has the form $(e_1, t_1)\hat{}(e_2, t_2)\hat{}\ldots(e_m, t_m)$, and there are $[a_i, b_i]$ $(a_i \in R^+, b_i \in R^+ \cup \{\infty\}$, $a_i \leq b_i, b_i > 0, i = 1, 2, \ldots, w)$ such that

$$(e_1, t_1)\hat{}(e_2, t_2)\hat{}\ldots(e_m, t_m) \in \mathcal{L}(\mathcal{A})$$

if and only if

$$\begin{array}{ll} a_i \leq t_{p_i} + t_{p_i+1} + \ldots + t_{p_i+q_i} \leq b_i & \\ i = 1, \ldots, w & \text{(denoted by } C1) \end{array}$$

where $1 \leq p_i \leq m$ and $1 \leq p_i + q_i \leq m$. Hence, the satisfaction problem of linear duration invariant $\mathcal{D}$ for simple RRE $\mathcal{A}$ is equivalent to some linear programming problems, which has constraints $C1$ and $C2$, and requests that subject to $C1$ and $C2$ the maximum value of the linear objective function

$$\sum_{i=1}^{n} c_{ij}(\sum_{u \in \alpha_i} t_u)$$

is not greater than M_j for $j = 1, 2, \ldots, k$, i.e. that, under $C1$ and $C2$, I holds.

Let $\mathcal{N}$ be a normal form, $\mathcal{N} = \mathcal{A}_1 \oplus \mathcal{A}_2 \oplus \ldots \oplus \mathcal{A}_m$, where $\mathcal{A}_j$s are simple RREs, and let $\mathcal{D}$ be a linear duration invariant. Since

$$\mathcal{N} \models D \Leftrightarrow \bigwedge_{i=1}^{m} \mathcal{A}_i \models D\,,$$

the problem of checking $\mathcal{N}$ for $\mathcal{D}$ can be solved by linear programming.

Therefore, for a general RRE $\mathcal{R}$, for a linear duration invariant $\mathcal{D}$, if we can effectively find a normal form $\mathcal{N}$ such that $\mathcal{R} \models \mathcal{D}$ if and only if $\mathcal{N} \models \mathcal{D}$, then we can check $\mathcal{R} \models \mathcal{D}$ effectively.

2.5 Contexts

Let $\mathcal{R}$ be an RRE and $\mathcal{R}_1$ be a sub-expression of $\mathcal{R}$. Replacing an occurrence of $\mathcal{R}_1$ in $\mathcal{R}$ with a letter X, we obtain a *context* of X. Any context $\mathcal{C}(X)$ of X, is associated with two real numbers $\varphi(\mathcal{C}(X))$ and $\omega(\mathcal{C}(X))$. Contexts and their associated real numbers are defined as follows.

Definition 5. A context $\mathcal{C}(X)$ of X, $\varphi(\mathcal{C}(X))$ and $\omega(\mathcal{C}(X))$ are defined recursively as:

- X is a context of X, and $\varphi(X) = 0$; $\omega(X) = \infty$.

– If $\mathcal{C}_1(X)$ is a context of X and $\mathcal{R}$ is a RRE, then $\mathcal{C}_1(X)\hat{}\mathcal{R}$ and $\mathcal{R}\hat{}\mathcal{C}_1(X)$ is a context of X, and

$$\varphi(\mathcal{C}_1(X)\hat{}\mathcal{R}) = \varphi(\mathcal{R}\hat{}\mathcal{C}_1(X)) = \varphi(\mathcal{C}_1(X)),$$
$$\omega(\mathcal{C}_1(X)\hat{}\mathcal{R}) = \omega(\mathcal{R}\hat{}\mathcal{C}_1(X)) = \omega(\mathcal{C}_1(X)).$$

– If $\mathcal{C}_1(X)$ is a context of X and $\mathcal{R}$ is a RRE, then $\mathcal{C}_1(X) \oplus \mathcal{R}$ and $\mathcal{R} \oplus \mathcal{C}_1(X)$ are contexts of X, and

$$\varphi(\mathcal{C}_1(X) \oplus \mathcal{R}) = \varphi(\mathcal{R} \oplus \mathcal{C}_1(X)) = \varphi(\mathcal{C}_1(X)),$$
$$\omega(\mathcal{C}_1(X) \oplus \mathcal{R}) = \omega(\mathcal{R} \oplus \mathcal{C}_1(X)) = \omega(\mathcal{C}_1(X)).$$

– If $\mathcal{C}_1(X)$ is a context of X, then $\mathcal{C}_1(X)^*$ is a context of X, and

$$\varphi(\mathcal{C}_1(X)^*) = \varphi(\mathcal{C}_1(X)), \quad \omega(\mathcal{C}_1(X)^*) = \omega(\mathcal{C}_1(X)).$$

– If $\mathcal{C}_1(X)$ is a context of X, $a \in R^+, b \in R^+ \cup \{\infty\}, b > 0,$ and $a \leq b$, then $(\mathcal{C}_1(X), [a, b])$ is a context of X, and

$$\varphi((\mathcal{C}_1(X), [a, b])) = max(\varphi(\mathcal{C}_1(X)), a),$$
$$\omega((\mathcal{C}_1(X), [a, b])) = min(\omega(\mathcal{C}_1(X)), b).$$

□

For example, let $\mathcal{C}(X) = ((s_1\hat{}s_2)^* \oplus ((X, [1, 5])^*, [2, 4]), [2, 3])$. It follows that $\varphi(\mathcal{C}(X)) = 2$ and $\omega(\mathcal{C}(X)) = 3$.

For any context $\mathcal{C}(X)$, replacing X in $\mathcal{C}(X)$ with any RRE, say $\mathcal{R}$, we can obtain an RRE, denoted by $\mathcal{C}(\mathcal{R})$. For example, replacing X in $X^* \oplus e_1\hat{}(e_2, [a, b])$ with an RRE $e_1\hat{}e_2$, we get a RRE $(e_1\hat{}e_2)^* \oplus e_1\hat{}(e_2, [a, b])$.

3 Checking RREs for linear duration invariants

In this section, we give an algorithm to check a real-time regular expression $\mathcal{R}$ for a linear duration invariant $\mathcal{D}$. Without loss of generality, throughout this section, let $\mathcal{D}$ be

$$t \leq \textstyle\int 1 \leq T \Rightarrow \displaystyle\sum_{i=1}^{n} c_i \textstyle\int S_i \leq M,$$

and for any $\sigma = (e_1, t_1)\hat{}(e_2, t_2)\hat{}\ldots\hat{}(e_m, t_m) \in \mathcal{L}(\mathcal{R})$, let

$$\theta(\sigma, \mathcal{D}) = \sum_{i=1}^{n} c_i(\sum_{u \in \alpha_i} t_u),$$

where $\alpha_i = \{u \mid (1 \leq u \leq m) \wedge (\overleftarrow{e_u} \Rightarrow S_i)\})$, and for every e_u $(1 \leq u \leq m)$, let $\gamma(e_u, \mathcal{D}) = c_i$, where $1 \leq i \leq n$ and $\overleftarrow{e_u} \Rightarrow S_i$.

For any simple RRE $\mathcal{R}$, let $M_\tau(\mathcal{R})$ $(m_\tau(\mathcal{R}))$ denote the supremum (infimum) of the set of $\{\tau(\sigma) | \sigma \in \mathcal{L}(\mathcal{R})\}$. $M_\tau(\mathcal{R})$ $(m_\tau(\mathcal{R}))$ can be calculated by finding the maximal (minimal) value of linear objective function $t_1 + t_2 + \ldots + t_m$ subject to a linear constraint similar to $C1$ in section 2.4, which is a classical linear programming problem. If $m_\tau(\mathcal{R}) = 0$, $\mathcal{R}$ is said to be a *zero-simple* RRE; otherwise $\mathcal{R}$ is said to be a *nonzero-simple* RRE.

For any nonzero-simple RRE $\mathcal{R}$, let $M_\theta(\mathcal{R})$ denote the supremum of the set

$$\{\theta(\sigma, \mathcal{D}) | \sigma \in \mathcal{L}(\mathcal{R})\} \ .$$

Similarly to $M_\tau(\mathcal{R})$, $M_\theta(\mathcal{R})$ can be calculated effectively by finding the maximal value of the linear objective function $\sum_{i=1}^{n} c_i(\sum_{u \in \alpha_i} t_u)$ subject to the group of linear inequalities $C1$.

Let $\lfloor x \rfloor$ denote the floor of a real number x. Let $\mathcal{A}^j$ denote the j-repetition of $\mathcal{A}$

$$\mathcal{A}^j = \underbrace{\mathcal{A}\hat{}\mathcal{A}\hat{}\ldots\hat{}\mathcal{A}}_{j}, \qquad \mathcal{A}^0 = \epsilon \, .$$

3.1 Some preliminary results

To derive some fundamental theorems for our model-checking algorithm, the following lemmas are useful. Their detailed proofs are presented in the appendix.

Let $\mathcal{C}(X)$ be a context.

Lemma 1. *(1.)* Let $\mathcal{R}$ and $\mathcal{R}'$ be RREs. If for any $\sigma \in \mathcal{L}(\mathcal{R})$, there is $\sigma' \in \mathcal{L}(\mathcal{R}')$ such that $\tau(\sigma) = \tau(\sigma')$ and $\theta(\sigma, \mathcal{D}) \le \theta(\sigma', \mathcal{D})$, then $\mathcal{R}' \models \mathcal{D}$ implies $\mathcal{R} \models \mathcal{D}$. □

Lemma 2. *(2.)* Let $\mathcal{R}$ and $\mathcal{R}'$ be real-time regular expressions. If for any $\sigma \in \mathcal{L}(\mathcal{R})$, there is $\sigma' \in \mathcal{L}(\mathcal{R}')$ such that $\tau(\sigma) = \tau(\sigma')$ and $\theta(\sigma, \mathcal{D}) \le \theta(\sigma', \mathcal{D})$, then $\mathcal{C}(\mathcal{R}') \models \mathcal{D}$ implies $\mathcal{C}(\mathcal{R}) \models \mathcal{D}$. □

Lemma 3. *(3.)*Let $\mathcal{R}$ and $\mathcal{R}'$ be RREs. If $\mathcal{L}(\mathcal{R}') \supseteq \mathcal{L}(\mathcal{R})$, then $\mathcal{C}(\mathcal{R}') \models \mathcal{D}$ implies $\mathcal{C}(\mathcal{R}) \models \mathcal{D}$. □

Lemma 4. *(4.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression such that $M_\theta(\mathcal{A}) \le 0$, and $\omega(\mathcal{C}(X)) = \infty$. Then for any real number N_t and for any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ such that $\tau(\sigma) \ge N_t$, there is $\sigma' \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)$ such that $\tau(\sigma') \ge N_t$ and $\theta(\sigma, \mathcal{D}) \le \theta(\sigma', \mathcal{D})$, where $p = (\lfloor h/m_\tau(\mathcal{A}) \rfloor + 1)$, and $h = \max(\varphi(\mathcal{C}(X), N_t)$. □

Lemma 5. *(5.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression such that $M_\theta(\mathcal{A}) > 0$, and $\omega(\mathcal{C}(X)) = \infty$. For any nonnegative real numbers N_t and M_r , there is $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ such that $\tau(\sigma) > N_t$ and $\theta(\sigma, \mathcal{D}) > M_r$. □

Lemma 6. *(6.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression, and $T \neq \infty$. Then for any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, $\tau(\sigma) \le T$ implies $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, where $p = \lfloor T/m_\tau(\mathcal{A}) \rfloor + 1$. □

Lemma 7. *(7.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression, $a \in R^+$,
$b \in R^+ \cup \{\infty\}$, $a \le b$, and $b > 0$. Then $\mathcal{L}((\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j), [a, b])) \supseteq \mathcal{L}((\mathcal{C}(\mathcal{A}^*), [a, b]))$, where $p = \lfloor b/m_\tau(\mathcal{A}) \rfloor + 1$. □

Lemma 8. *(8.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression. Then $\mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)) \supseteq \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, where $p = \lfloor \omega(\mathcal{C}(X))/m_\tau(\mathcal{A}) \rfloor + 1$. □

The following theorems constitute the foundation of our algorithm.

Theorem 1. *(1.)* Let $\mathcal{R}$ and $\mathcal{R}'$ be real-time regular expressions. If $\mathcal{L}(\mathcal{R}') = \mathcal{L}(\mathcal{R})$, then $\mathcal{C}(\mathcal{R}) \models \mathcal{D}$ if and only if $\mathcal{C}(\mathcal{R}') \models \mathcal{D}$.

Proof. By Lemma 3, the result follows trivially. □

Theorem 2. *(2.)* Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be real-time regular expressions. Then $\mathcal{C}((\mathcal{R}_1 \oplus \mathcal{R}_2)^*) \models \mathcal{D}$ if and only if $\mathcal{C}((\mathcal{R}_1{}^*)\hat{\ }(\mathcal{R}_2{}^*)) \models \mathcal{D}$.

Proof. By Definition 2, $\mathcal{L}((\mathcal{R}_1{}^*)\hat{\ }(\mathcal{R}_2{}^*)) \subseteq \mathcal{L}((\mathcal{R}_1 \oplus \mathcal{R}_2)^*)$. By Lemma 3, we can prove the half of the claim, i.e. $\mathcal{C}((\mathcal{R}_1 \oplus \mathcal{R}_2)^*) \models \mathcal{D}$ implies $\mathcal{C}((\mathcal{R}_1{}^*)\hat{\ }(\mathcal{R}_2{}^*)) \models \mathcal{D}$. The other half can be proved as follows. For any time-distance stamped state sequences σ_1 and σ_2, since $\tau(\sigma_1\hat{\ }\sigma_2) = \tau(\sigma_1) + \tau(\sigma_2)$ and $\theta(\sigma_1\hat{\ }\sigma_2, \mathcal{D}) = \theta(\sigma_1, \mathcal{D}) + \theta(\sigma_2, \mathcal{D})$, we have $\tau(\sigma_1\hat{\ }\sigma_2) = \tau(\sigma_2\hat{\ }\sigma_1)$ and $\theta(\sigma_1\hat{\ }\sigma_2, \mathcal{D}) = \theta(\sigma_2\hat{\ }\sigma_1, \mathcal{D})$. Since any $\sigma \in \mathcal{L}(\mathcal{C}((\mathcal{R}_1 \oplus \mathcal{R}_2)^*)$ can be permuted to $\sigma' \in \mathcal{L}(\mathcal{C}((\mathcal{R}_1{}^*)\hat{\ }(\mathcal{R}_2{}^*)))$, by Lemma 2, the result follows. □

Theorem 3. *(3.)* Let e be a transition. Then $\mathcal{C}(e^*) \models \mathcal{D}$ if and only if $\mathcal{C}(e) \models \mathcal{D}$.

Proof. Since, by Definition 2, $\mathcal{L}(e^*) \supseteq \mathcal{L}(e)$, by Lemma 3, we can prove the half of the claim, i.e. $\mathcal{C}(e^*) \models \mathcal{D}$ implies $\mathcal{C}(e) \models \mathcal{D}$. The other half can be proved as follows. For any $\sigma = (e, t_1)\hat{\ }(e, t_2)\hat{\ } \ldots \hat{\ }(e, t_m) \in \mathcal{L}(e^*)$, let $\sigma' = (e, t_1 + t_2 + \ldots + t_m)$. It follows that $\sigma' \in \mathcal{L}(e)$. Since $\theta(\sigma, \mathcal{D}) = \theta(\sigma', \mathcal{D})$ and $\tau(\sigma) = \tau(\sigma')$, by Lemma 2, the result follows. □

Theorem 4. *(4.)* Let $\mathcal{A}$ be a zero-simple real-time regular expression such that any $\sigma \in \mathcal{L}(\mathcal{A})$ has the form $(e_1, t_1)\hat{\ }(e_2, t_2)\hat{\ } \ldots \hat{\ }(e_m, t_m)$ $(m \geq 1)$. Suppose $\gamma(e_j, \mathcal{D}) = \max(\gamma(e_1, \mathcal{D}), \gamma(e_2, \mathcal{D}), \ldots, \gamma(e_m, \mathcal{D}))$ $(1 \leq j \leq m)$. Then $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ if and only if $\mathcal{C}(e_j{}^*) \models \mathcal{D}$.

Proof. Since $\mathcal{A}$ is a zero-simple real-time regular expression such that any $\sigma \in \mathcal{L}(\mathcal{A})$ has the form $(e_1, t_1)\hat{\ }(e_2, t_2)\hat{\ } \ldots \hat{\ }(e_m, t_m)$ $(m \geq 1)$, by Definition 2, there is b $(b \in R^+, b > 0)$ such that for any t $(0 \leq t \leq b)$,

$$(e_1, 0)\hat{\ }(e_2, 0)\hat{\ } \ldots \hat{\ }(e_{j-1}, 0)\hat{\ }(e_j, t)\hat{\ }(e_{j+1}, 0)\hat{\ } \ldots \hat{\ }(e_m, 0) \in \mathcal{L}(\mathcal{A}) \quad .$$

For any $\sigma = (e_j, t_1)\hat{\ }(e_j, t_2)\hat{\ } \ldots \hat{\ }(e_j, t_n) \in \mathcal{L}(e_j{}^*)$, let $t_i = k_i b + d_i$ (k_i is a natural number, $0 \leq d_i < b$, $i = 1, 2, \ldots, n$). For $i = 1, 2, \ldots, n$, let

$$\sigma_i = \underbrace{\alpha\hat{\ }\alpha\hat{\ } \ldots \hat{\ }\alpha}_{k_i}\hat{\ }\beta_i \quad ,$$

where

$$\alpha = (e_1, 0)\hat{\ }(e_2, 0)\hat{\ } \ldots \hat{\ }(e_{j-1}, 0)\hat{\ }(e_j, b)\hat{\ }(e_{j+1}, 0)\hat{\ } \ldots \hat{\ }(e_m, 0), \text{ and}$$

$$\beta_i \;=\; (e_1,0)\hat{}(e_2,0)\hat{}\ldots\hat{}(e_{j-1},0)\hat{}(e_j,d_i)\hat{}(e_{j+1},0)\hat{}\ldots\hat{}(e_m,0) \quad .$$

Let $\sigma' = \sigma_1\hat{}\sigma_2\hat{}\ldots\hat{}\sigma_n$. It follows that $\sigma' \in \mathcal{L}(\mathcal{A}^*)$. Since $\theta(\sigma,\mathcal{D}) = \theta(\sigma',\mathcal{D})$ and $\tau(\sigma) = \tau(\sigma')$, by Lemma 2, we can prove the half of the claim, i.e. $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ implies $\mathcal{C}({e_j}^*) \models \mathcal{D}$.

The proof of the second half can be explained below. By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{A}^*)$ has the form $\sigma_1\hat{}\sigma_2\hat{}\ldots\hat{}\sigma_n$, where

$$\sigma_i = (e_1,t_{i1})\hat{}(e_2,t_{i2})\hat{}\ldots\hat{}(e_m,t_{im}) \;\in\; \mathcal{L}(\mathcal{A}) \quad (i = 1,2,\ldots,n) \quad .$$

Let $\sigma'_i = (e_j,t_{i1})\hat{}(e_j,t_{i2})\hat{}\ldots\hat{}(e_j,t_{im})$ $(1 \le i \le n)$. Since, by assumption, $\gamma(e_j,\mathcal{D}) = max(\gamma(e_1,\mathcal{D}),\gamma(e_2,\mathcal{D}),\ldots,\gamma(e_m,\mathcal{D}))$ $(1 \le j \le m)$, it follows that $\theta(\sigma_i,\mathcal{D}) \le \theta({\sigma'}_i,\mathcal{D})$. Let $\sigma' = \sigma'_1\hat{}\sigma'_2\hat{}\ldots\hat{}\sigma'_n$. Consequently, $\sigma' \in \mathcal{L}({e_j}^*)$ and $\theta(\sigma,\mathcal{D}) \le \theta(\sigma',\mathcal{D})$. Since $\tau(\sigma) = \tau(\sigma')$, by Lemma 2, the result follows. □

Theorem 5. *(5.)* Let $\mathcal{A}$ be a zero-simple real-time regular expression such that any $\sigma \in \mathcal{L}(\mathcal{A})$ have the form $(e_1,t_1)\hat{}(e_2,t_2)\hat{}\ldots\hat{}(e_m,t_m)$ $(m \ge 1)$. Suppose $\gamma(e_j,\mathcal{D}) = \max(\gamma(e_1,\mathcal{D}),\gamma(e_2,\mathcal{D}),\ldots,\gamma(e_m,\mathcal{D}))$ $(1 \le j \le m)$. Then $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ if and only if $\mathcal{C}(e_j) \models \mathcal{D}$.

Proof. The theorem follows immediately by Theorem 3 and Theorem 4. □

Theorem 6. *(6.)* Let $\mathcal{A}$ be a nonzero-simple real-time regular expression such that $M_\theta(\mathcal{A}) > 0$, $\omega(\mathcal{C}(X)) = \infty$, and $T = \infty$. Then $\mathcal{C}(\mathcal{A}^*) \not\models \mathcal{D}$.

Proof. Since $T = \infty$, the theorem follows immediately by Lemma 5. □

Theorem 7. *(7.)* Let $\mathcal{A}$ be a nonzero-simple real-time regular expression such that $M_\theta(\mathcal{A}) \le 0$, $\omega(\mathcal{C}(X)) = \infty$, and $T = \infty$. Then $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ if and only if $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) \models \mathcal{D}$, where $p = (\lfloor h/m_\tau(\mathcal{A})\rfloor + 1)$, $h = \max(\varphi(\mathcal{C}(X),t)$.

Proof. Since, by Definition 2, $\mathcal{L}(\mathcal{A}^*) \supseteq \mathcal{L}(\oplus_{j=0}^{p}\mathcal{A}^j)$, by Lemma 3, we can prove a half of the claim, i.e. $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ implies $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) \models \mathcal{D}$. By Lemma 4, the other half can be proved trivially. □

Theorem 8. *(8.)* Let $\mathcal{A}$ be a nonzero-simple real-time regular expression. Suppose $\omega(\mathcal{C}(X)) \ne \infty$ or $T \ne \infty$. Then $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ if and only if $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) \models \mathcal{D}$, where $p = (\lfloor h/m_\tau(\mathcal{A})\rfloor + 1)$, $h = \min(\omega(\mathcal{C}(X),T)$.

Proof. Since, by Definition 2, $\mathcal{L}(\mathcal{A}^*) \supseteq \mathcal{L}(\oplus_{j=0}^{p}\mathcal{A}^j)$, by Lemma 2, we can prove a half of the claim, i.e. $\mathcal{C}(\mathcal{A}^*) \models \mathcal{D}$ implies $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) \models \mathcal{D}$. By Lemma 6 and 8, the other half can be proved trivially. □

3.2 Model-checking algorithm

Based on the theorems given in section 3.1, we now give an algorithm to check an RRE $\mathcal{R}$ for a linear duration invariant $\mathcal{D}$. Here we only describe the essential parts of the algorithm, i.e. how to find a normal form $\mathcal{N}$ such that $\mathcal{R} \models \mathcal{D}$ if and only if $\mathcal{N} \models \mathcal{D}$.

A sub-expression of $\mathcal{R}$ is said to be *reducible* if it is not a normal form and has one of the following forms:

$$(1)\ \mathcal{N}_1\hat{}\mathcal{N}_2\hat{}\ldots\hat{}\mathcal{N}_m\,, \qquad (2)\ \mathcal{N}^*\,, \qquad (3)\ (\mathcal{N},[a,b])\,,$$

where $\mathcal{N}_1,\mathcal{N}_2\ldots,\mathcal{N}_m$, and $\mathcal{N}$ are normal forms. By Definition 2, if $\mathcal{R}$ has no reducible sub-expression, it is a normal form.

The basic part of the algorithm is a transformation process described as follows. First, we find out a reducible sub-expression $\mathcal{R}_1$ in $\mathcal{R}$. Then, replacing an occurrence of $\mathcal{R}_1$ in $\mathcal{R}$ with a letter X, we get a context $\mathcal{R}(X)$ such that $\mathcal{R}=\mathcal{R}(\mathcal{R}_1)$. By Algorithm $\mathcal{B}$ (see below), we can either discover $\mathcal{R}\not\models\mathcal{D}$, or find out $\mathcal{R}_2$, which is a normal form, such that $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2)\models\mathcal{D}$. Let $\mathcal{R}'=\mathcal{R}(\mathcal{R}_2)$. We say that $\mathcal{R}$ is transformed into $\mathcal{R}'$. If $\mathcal{R}'$ is a normal form, then the problem is solved; otherwise the transformation process is repeated until we reach an RRE $\mathcal{Q}$ in which no reducible sub-expression can be found , i.e. $\mathcal{Q}$ is a normal form.

Algorithm $\mathcal{B}$ can be explained as follows:

Step 1. If a reducible sub-expression $\mathcal{R}_1$ of $\mathcal{R}$ is of the form $(\mathcal{N},[a,b])$, where $\mathcal{N}$ is a normal form, we can find a normal form $\mathcal{R}_2$ such that $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2)\models\mathcal{D}$ as follows. Suppose $\mathcal{N}=\mathcal{A}_1\oplus\mathcal{A}_2\oplus\ldots\oplus\mathcal{A}_q$, where $\mathcal{A}_j$s are simple RREs. Let

$$\mathcal{R}_2=(\mathcal{A}_1,[a,b])\oplus(\mathcal{A}_2,[a,b])\oplus\ldots\oplus(\mathcal{A}_q,[a,b])\,.$$

It is clear that $\mathcal{R}_2$ is a normal form. Since, by Definition 2, $\mathcal{L}(\mathcal{R}_1)=\mathcal{L}(\mathcal{R}_2)$, by Theorem 1, $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2)\models\mathcal{D}$.

Step 2. If a reducible sub-expression $\mathcal{R}_1$ of $\mathcal{R}$ is of the form $\mathcal{N}_1\hat{}\mathcal{N}_2\hat{}\ldots\hat{}\mathcal{N}_m$, where $\mathcal{N}_1,\mathcal{N}_2,\ldots,\mathcal{N}_m$ are normal forms, we can find a normal form $\mathcal{R}_2$ such that $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2)\models\mathcal{D}$ as follows. For $i=1,2,\ldots,m$, suppose

$$\mathcal{N}_i=\mathcal{A}_{i1}\oplus\mathcal{A}_{i2}\oplus\ldots\oplus\mathcal{A}_{iq_i}\,,$$

where $\mathcal{A}_{ij}$s are simple RREs. Let

$$\mathcal{R}_2=\mathcal{A}_{11}\hat{}\mathcal{A}_{21}\hat{}\ldots\hat{}\mathcal{A}_{m1}\oplus\ldots\oplus\mathcal{A}_{1q_1}\hat{}\mathcal{A}_{2q_2}\hat{}\ldots\hat{}\mathcal{A}_{mq_m}\quad.$$

Since every $\mathcal{A}_{ij}$ $(1\le i\le m, 1\le j\le q_i)$ is a simple real-time regular expression, $\mathcal{R}_2$ is a normal form. Since, by Definition 2, $\mathcal{L}(\mathcal{R}_1)=\mathcal{L}(\mathcal{R}_2)$, by Theorem 1, it follows that $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ if and only if $\mathcal{R}(\mathcal{R}_2)\models\mathcal{D}$.

Step 3. If a reducible sub-expression $\mathcal{R}_1$ of $\mathcal{R}$ is of the form $\mathcal{N}^*$ where $\mathcal{N}$ is a normal form, we can either discover $\mathcal{R}\not\models\mathcal{D}$, or find $\mathcal{R}_2$, which is a normal form, such that $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ if and only if $\mathcal{R}(\mathcal{R}_2)\models\mathcal{D}$ as follows. Suppose

$$\mathcal{N}=\mathcal{A}_1\oplus\mathcal{A}_2\oplus\ldots\oplus\mathcal{A}_q\,,$$

where $\mathcal{A}_j$s are simple RREs. Let

$$\mathcal{R}_1'=(\mathcal{A}_1)^*\hat{}(\mathcal{A}_2)^*\hat{}\ldots\hat{}(\mathcal{A}_q)^*\,.$$

By Theorem 2, it follows that $\mathcal{R}(\mathcal{R}_1)\models\mathcal{D}$ if and only if $\mathcal{R}(\mathcal{R}_1')\models\mathcal{D}$.

For any context $\mathcal{C}(X)$, replacing X in context $\mathcal{R}(X)$ with $\mathcal{C}(X)$, we get a context $\mathcal{R}(\mathcal{C}(X))$. Note that for any ${\mathcal{A}_i}^*$ $(1 \leq i \leq q)$, if $\mathcal{A}_i$ is a zero-simple RRE, then by Theorem 5, we can find out a normal form $\mathcal{A}'_i$ such that

$$\mathcal{R}(\mathcal{C}({\mathcal{A}_i}^*)) \models \mathcal{D} \text{ iff } \mathcal{R}(\mathcal{C}(\mathcal{A}'_i)) \models \mathcal{D};$$

otherwise we can either discover $\mathcal{R}(\mathcal{C}({\mathcal{A}_i}^*)) \not\models \mathcal{D}$ by Theorem 6, or find out p by Theorems 7 and 8 such that

$$\mathcal{R}(\mathcal{C}({\mathcal{A}_i}^*)) \models \mathcal{D} \text{ iff } \mathcal{R}(\mathcal{C}(\mathcal{A}'_i)) \models \mathcal{D},$$

where $\mathcal{A}'_i = \oplus_{j=0}^{p}{\mathcal{A}_i}^j$. Thus, we can either discover $\mathcal{R} \not\models \mathcal{D}$, or find an RRE $\mathcal{R}'_2$ of the form $\mathcal{R}'_2 = \mathcal{N}_1\hat{}\mathcal{N}_2\hat{}\ldots\hat{}\mathcal{N}_q$, where $\mathcal{N}_i$s are normal forms, such that $\mathcal{R}(\mathcal{R}'_1) \models \mathcal{D}$ if and only if $\mathcal{R}(\mathcal{R}'_2) \models \mathcal{D}$ as follows:

(a). First, let $\mathcal{C}_1 = \mathcal{R}'_1$. Replacing the first occurrence of ${\mathcal{A}_1}^*$ in $\mathcal{C}_1$ with a letter X, we get a context $\mathcal{C}_1(X)$. Replacing X in context $\mathcal{R}(X)$ with $\mathcal{C}_1(X)$, we get a context $\mathcal{R}(\mathcal{C}_1(X))$. Since $\mathcal{R}(\mathcal{R}'_1) = \mathcal{R}(\mathcal{C}_1) = \mathcal{R}(\mathcal{C}_1({\mathcal{A}_1}^*))$, we can either discover $\mathcal{R} \not\models \mathcal{D}$, or find out a normal form $\mathcal{N}_1$ such that

$$\mathcal{R}(\mathcal{C}_1({\mathcal{A}_1}^*)) \models \mathcal{D} \text{ iff } \mathcal{R}(\mathcal{C}_1(\mathcal{N}_1)) \models \mathcal{D}.$$

(b) Then, if we can not discover $\mathcal{R} \not\models \mathcal{D}$, let $\mathcal{C}_2 = \mathcal{C}_1(\mathcal{N}_1)$. It follows that

$$\mathcal{C}_2 = \mathcal{N}_1\hat{}{\mathcal{A}_2}^*\hat{}\ldots\hat{}{\mathcal{A}_q}^*.$$

Replacing the first occurrence of ${\mathcal{A}_2}^*$ in $\mathcal{C}_2$ with a letter X, we get a context $\mathcal{C}_2(X)$. Replacing X in context $\mathcal{R}(X)$ with $\mathcal{C}_2(X)$, we get a context $\mathcal{R}(\mathcal{C}_2(X))$. Since $\mathcal{R}(\mathcal{C}_1(\mathcal{N}_1)) = \mathcal{R}(\mathcal{C}_2) = \mathcal{R}(\mathcal{C}_2({\mathcal{A}_2}^*))$, we can either discover $\mathcal{R} \not\models \mathcal{D}$, or find out a normal form $\mathcal{N}_2$ such that $\mathcal{R}(\mathcal{C}_2({\mathcal{A}_2}^*)) \models \mathcal{D}$ iff $\mathcal{R}(\mathcal{C}_2(\mathcal{N}_2)) \models \mathcal{D}$.

(c). In the same way, at last we can either discover $\mathcal{R} \not\models \mathcal{D}$, or find out an RRE $\mathcal{R}'_2$ of the form $\mathcal{R}'_2 = \mathcal{N}_1\hat{}\mathcal{N}_2\hat{}\ldots\hat{}\mathcal{N}_q$, where $\mathcal{N}_i$s are normal forms, such that $\mathcal{R}(\mathcal{R}'_1) \models \mathcal{D}$ iff $\mathcal{R}(\mathcal{R}'_2) \models \mathcal{D}$. If we can discover $\mathcal{R} \not\models \mathcal{D}$, then we are done, otherwise by Step 2, we can find a normal form $\mathcal{R}_2$ such that $\mathcal{R}(\mathcal{R}'_2) \models \mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2) \models \mathcal{D}$. It follows that $\mathcal{R}(\mathcal{R}_1) \models \mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2) \models \mathcal{D}$.

Example 1. Let $\mathcal{D}$ be $0 \leq \int 1 < \infty \Rightarrow \int S_1 + 2\int S_2 + 4\int S_3 - 3\int S_4 \leq 20$ and $\mathcal{R} = ((e_1\hat{}e_2, [0,1])^*\hat{}(e_3, [0,4]), [2,5]) \oplus (e_3, [0,4])\hat{}(e_4, [1,2])^*$, where $\overleftarrow{e_1} \Rightarrow S_1$, $\overleftarrow{e_2} \Rightarrow S_2$, $\overleftarrow{e_3} \Rightarrow S_3$, and $\overleftarrow{e_4} \Rightarrow S_4$. Using our model-checking algorithm, checking $\mathcal{R} \models \mathcal{D}$ can be done as follows.

Let $\mathcal{R}_1 = (e_1\hat{}e_2, [0,1])^*$. It follows that $\mathcal{R}_1$ is reducible sub-expression of $\mathcal{R}$. Replacing $\mathcal{R}_1$ in $\mathcal{R}$ with a letter X, we get a context

$$\mathcal{R}(X) = (X\hat{}(e_3, [0,4]), [2,5]) \oplus (e_3, [0,4])\hat{}(e_4, [1,2])^*$$

such that $\mathcal{R} = \mathcal{R}(\mathcal{R}_1)$. Now, by Algorithm $\mathcal{B}$, we have to find a normal $\mathcal{R}_2$ such that $\mathcal{R}(\mathcal{R}_1) \models \mathcal{D}$ iff $\mathcal{R}(\mathcal{R}_2) \models \mathcal{D}$. Let $\mathcal{A}_1 = (e_1\hat{}e_2, [0,1])$. Then, $\mathcal{A}_1$ is a zero-simple RRE. By Theorem 10, $\mathcal{R}({\mathcal{A}_1}^*) \models \mathcal{D}$ iff $\mathcal{R}(e_2) \models \mathcal{D}$. Let $\mathcal{R}' = \mathcal{R}(e_2)$. It follows that

$$\mathcal{R}' = (e_2\hat{}(e_3, [0,4]), [2,5]) \oplus (e_3, [0,4])\hat{}(e_4, [1,2])^*.$$

Since $\mathcal{R}'$ is not a normal form, we have to repeat the process.

Let $\mathcal{R}_1 = (e_4, [1, 2])^*$. Then, $\mathcal{R}_1$ is a reducible sub-expression of $\mathcal{R}'$. Replacing $\mathcal{R}_1$ in $\mathcal{R}'$ with a letter X, we get the context

$$\mathcal{R}'(X) = (e_2\hat{}(e_3, [0, 4]), [2, 5]) \oplus (e_3, [0, 4])\hat{}X$$

such that $\mathcal{R}' = \mathcal{R}'(\mathcal{R}_1)$. Now, by Algorithm $\mathcal{B}$, we have to find a normal form $\mathcal{R}_2$ such that $\mathcal{R}'(\mathcal{R}_1) \models \mathcal{D}$ iff $\mathcal{R}'(\mathcal{R}_2) \models \mathcal{D}$. Let $\mathcal{A}_1 = (e_4, [1, 2])$. It follows that $\mathcal{A}_1$ is a nonzero-simple RRE. By Theorem 12, $\mathcal{R}'({\mathcal{A}_1}^*) \models \mathcal{D}$ iff $\mathcal{R}'(\epsilon \oplus \mathcal{A}_1) \models \mathcal{D}$. Let $\mathcal{Q} = \mathcal{R}'(\epsilon \oplus \mathcal{A}_1)$. It follows that

$$\mathcal{Q} = (e_2\hat{}(e_3, [0, 4]), [2, 5]) \oplus (e_3, [0, 4]) \oplus (e_3, [0, 4])\hat{}(e_4, [1, 2])\,,$$

and $\mathcal{Q}$ is a normal form.

Since $\mathcal{R} \models \mathcal{D}$ iff $\mathcal{Q} \models \mathcal{D}$, the model-checking problem is equivalent to the following three problems that can be solved by linear programming:

1. Find the maximum value of the linear objective function $2t_2 + 4t_3$ subject to constraints $2 \leq t_2 + t_3 \leq 5$ and $0 \leq t_3 \leq 4$, and check whether it is not greater than 20.
2. Find the maximum value of the linear objective function $4t_3$ subject to constraint $0 \leq t_3 \leq 4$, and check whether it is not greater than 20.
3. Find the maximum value of the linear objective function $4t_3 - 3t_4$ subject to constraints $0 \leq t_3 \leq 4$ and $1 \leq t_4 \leq 2$, and check whether it is not greater than 20. □

4 Conclusion

The work in [2] has been a major source of inspiration since it gives an algorithm for checking real-time automata with respect to linear duration invariants by linear programming. We have generalised these techniques for checking a class of timed automata for an improvement of execution efficiency for model-checking of this class of real-time systems.

We plan to incorporate the results described here in a Duration Calculus proof assistant described in [7]. In the future, we would like to combine this technique with the compositional model-checking techniques for checking a network of timed automata runing in parallel.

References

1. Rajeev Alur, David L. Dill. A theory of timed automata. In *Theoretical Computer Science*, 126(1994), pp.183-235.
2. Zhou Chaochen, Zhang Jingzhong, Yang Lu and Li Xiaoshan. Linear Duration Invariants. In *Formal Techniques in Real-Time and Fault-Tolerant Systems, LNCS 863*, pp.88-109.
3. Zhou Chaochen, C.A.R. Hoare, A.P. Ravn. A Calculus of Durations. In *Information Processing Letter*, 40, 5, 1991, pp.269-276.

4. Michael R. Hansen. Model-Checking Discrete Duration Calculus. In *Formal Aspects of Computing* (1994) 6A, pp.826-845.
5. Y. Kesten, A. Pnueli, J. Sifakis, S. Yovine. Integration Graphs: A Class of Decidable Hybrid Systems. In *Hybrid System, LNCS 736*, pp.179-208.
6. S.C. Kleene. Representation of Events in Nerve Nets and Finite Automata. In *Automata Studies*, C.Shannon and J. McCarthy (eds.), Princeton Univ. Press, Princeton, NJ, 1956, pp.3-41.
7. J.U. Skakkebæk and N. Shankar. Towards a Duration Calculus proof assistant in PVS. In *Formal Techniques in Real-Time and Fault-Tolerant Systems, LNCS 863*, pp.660-697.

A Proofs of Lemmas

In this appendix, we present the proofs of Lemmas 1-8. Most of these lemmas will be proved by induction on the structure of context. Since the proof for the structure

$$\mathcal{C}_1(X) \oplus \mathcal{R} \quad (\mathcal{C}_1(X)\hat{\ }\mathcal{R})$$

is similar to the one for the structure

$$\mathcal{R} \oplus \mathcal{C}_1(X) \quad (\mathcal{R}\hat{\ }\mathcal{C}_1(X)) \quad ,$$

we just give the proof for the structure $\mathcal{C}_1(X) \oplus \mathcal{R}$ $(\mathcal{C}_1(X)\hat{\ }\mathcal{R})$.

Lemma 9. *(1.)* Let $\mathcal{R}$ and $\mathcal{R}'$ be RREs. If for any $\sigma \in \mathcal{L}(\mathcal{R})$, there is $\sigma' \in \mathcal{L}(\mathcal{R}')$ such that $\tau(\sigma) = \tau(\sigma')$ and $\theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})$, then $\mathcal{R}' \models \mathcal{D}$ implies $\mathcal{R} \models \mathcal{D}$.

Proof. By Definition 3 and 4, the result follows trivially. □

Lemma 10. *(2.)* Let $\mathcal{R}$ and $\mathcal{R}'$ be real-time regular expressions. If for any $\sigma \in \mathcal{L}(\mathcal{R})$, there is $\sigma' \in \mathcal{L}(\mathcal{R}')$ such that $\tau(\sigma) = \tau(\sigma')$ and $\theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})$, then $\mathcal{C}(\mathcal{R}') \models \mathcal{D}$ implies $\mathcal{C}(\mathcal{R}) \models \mathcal{D}$.

Proof. If we can prove following claim:

– if for any $\sigma_1 \in \mathcal{L}(\mathcal{R})$, there is $\sigma_1' \in \mathcal{L}(\mathcal{R}')$ such that

$$\tau(\sigma_1) = \tau(\sigma_1') \quad \text{and} \quad \theta(\sigma_1, \mathcal{D}) \leq \theta(\sigma_1', \mathcal{D}) \,,$$

then for any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{R}))$, there is $\sigma' \in \mathcal{L}(\mathcal{C}(\mathcal{R}'))$ such that

$$\tau(\sigma) = \tau(\sigma') \quad \text{and} \quad \theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D}) \,,$$

the result follows trivially. We prove the claim by induction on the structure of context.

– **Basic Case:** Let $\mathcal{C}(X) = X$. Then $\mathcal{C}(\mathcal{R}) = \mathcal{R}$ and $\mathcal{C}(\mathcal{R}') = \mathcal{R}'$. By Lemma 1, the basic case holds.
– **Induction Step:** Assume that the claim holds for a context $\mathcal{C}_1(X)$, and let $\mathcal{C}(X)$ be defined from $\mathcal{C}_1(X)$.

1. If $\mathcal{C}(X) = \mathcal{C}_1(X) \oplus \mathcal{R}_1$ where $\mathcal{R}_1$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{R}) = \mathcal{C}_1(\mathcal{R}) \oplus \mathcal{R}_1 \;\text{ and }\; \mathcal{C}(\mathcal{R}') = \mathcal{C}_1(\mathcal{R}') \oplus \mathcal{R}_1 .$$

   Since, by Definition 2,

$$\mathcal{L}(\mathcal{C}(\mathcal{R})) = \mathcal{L}(\mathcal{C}_1(\mathcal{R})) \cup \mathcal{L}(\mathcal{R}_1) \;\text{ and }\; \mathcal{L}(\mathcal{C}(\mathcal{R}')) = \mathcal{L}(\mathcal{C}_1(\mathcal{R}')) \cup \mathcal{L}(\mathcal{R}_1) ,$$

   by the inductive hypothesis, the claim holds.
2. If $\mathcal{C}(X) = \mathcal{C}_1(X)\hat{\ }\mathcal{R}_1$ where $\mathcal{R}_1$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{R}) = \mathcal{C}_1(\mathcal{R})\hat{\ }\mathcal{R}_1 \;\text{ and }\; \mathcal{C}(\mathcal{R}') = \mathcal{C}_1(\mathcal{R}')\hat{\ }\mathcal{R}_1 .$$

   For any

$$\sigma = \sigma_1\hat{\ }\sigma_{\mathcal{R}_1} \in \mathcal{L}(\mathcal{C}(\mathcal{R}))$$

   where $\sigma_1 \in \mathcal{L}(\mathcal{C}_1(\mathcal{R}))$ and $\sigma_{\mathcal{R}1} \in \mathcal{L}(\mathcal{R}_1)$, since, by the inductive hypothesis, there is $\sigma_1' \in \mathcal{L}(\mathcal{C}_1(\mathcal{R}'))$ such that

$$\tau(\sigma_1) = \tau({\sigma'}_1) \;\text{ and }\; \theta(\sigma_1, \mathcal{D}) \leq \theta({\sigma'}_1, \mathcal{D}) ,$$

   it follows that

$$\sigma' = \sigma_1'\hat{\ }\sigma_{\mathcal{R}_1} \in \mathcal{L}(\mathcal{C}(\mathcal{R}')), \;\; \tau(\sigma) = \tau(\sigma'), \;\text{ and }\; \theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D}) ,$$

   i.e. the claim follows.
3. If $\mathcal{C}(X) = \mathcal{C}_1(X)^*$, then

$$\mathcal{C}(\mathcal{R}) = \mathcal{C}_1(\mathcal{R})^* \;\text{ and }\; \mathcal{C}(\mathcal{R}') = \mathcal{C}_1(\mathcal{R}')^* .$$

   For any

$$\sigma = \sigma_1\hat{\ }\sigma_2\hat{\ } \ldots \hat{\ }\sigma_m \in \mathcal{L}(\mathcal{C}(\mathcal{R}))$$

   where $\sigma_i \in \mathcal{L}(\mathcal{C}_1(\mathcal{R}))$ $(1 \leq i \leq m)$, since, by the inductive hypothesis, there is $\sigma_i' \in \mathcal{L}(\mathcal{C}_1(\mathcal{R}'))$ such that

$$\tau(\sigma_i) = \tau({\sigma'}_i) \;\text{ and }\; \theta(\sigma_i, \mathcal{D}) \leq \theta({\sigma'}_i, \mathcal{D}) ,$$

   it follows that

$$\sigma' = \sigma_1'\hat{\ }\sigma_2'\hat{\ } \ldots \hat{\ }\sigma_m' \in \mathcal{L}(\mathcal{C}(\mathcal{R}')), \; \tau(\sigma) = \tau(\sigma'), \text{ and } \theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D}) ,$$

   i.e. the claim holds.
4. If $\mathcal{C}(X) = (\mathcal{C}_1(X), [a, b])$ where $a \in R^+, b \in R^+, b \geq 0,$ and $a \leq b$, then

$$\mathcal{C}(\mathcal{R}) = (\mathcal{C}_1(\mathcal{R}), [a, b]) \;\text{ and }\; \mathcal{C}(\mathcal{R}') = (\mathcal{C}_1(\mathcal{R}'), [a, b]) .$$

   For any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{R}))$, since, by Definition 2, $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{R}))$, by the inductive hypothesis, there is $\sigma' \in \mathcal{L}(\mathcal{C}_1(\mathcal{R}'))$ such that

$$\tau(\sigma) = \tau(\sigma') \;\text{ and }\; \theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D}) .$$

   By Definition 2, it follows that $\sigma' \in \mathcal{L}(\mathcal{C}(\mathcal{R}'))$, i.e. the claim holds. □

Lemma 11. *(3.)*Let $\mathcal{R}$ and $\mathcal{R}'$ be RREs. If $\mathcal{L}(\mathcal{R}') \supseteq \mathcal{L}(\mathcal{R})$, then $\mathcal{C}(\mathcal{R}') \models \mathcal{D}$ implies $\mathcal{C}(\mathcal{R}) \models \mathcal{D}$.

Proof. By Lemma 2, the result follows trivially. □

Lemma 12. *(4.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression such that $M_\theta(\mathcal{A}) \leq 0$, and $\omega(\mathcal{C}(X)) = \infty$. Then for any real number N_t and for any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ such that $\tau(\sigma) \geq N_t$, there is $\sigma' \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)$ such that $\tau(\sigma') \geq N_t$ and $\theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})$, where $p = (\lfloor h/m_\tau(\mathcal{A})\rfloor + 1)$, and $h = \max(\varphi(\mathcal{C}(X), N_t)$.

Proof. We prove the claim by induction on the structure of context.

- **Basic Case:** Let $\mathcal{C}(X) = X$. Then $\mathcal{C}(\mathcal{A}^*) = \mathcal{A}^*$ and $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \oplus_{j=0}^{p}\mathcal{A}^j$. By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{A}^*)$ $(\tau(\sigma) \geq t)$ is of the form:

  $$\sigma_1\hat{}\sigma_2\hat{}\ldots\hat{}\sigma_m \ (\sigma_i \in \mathcal{L}(\mathcal{A}), 1 \leq i \leq m)\,.$$

  If $m \leq p$, then $\sigma \in \mathcal{L}(\oplus_{j=0}^{p}\mathcal{A}^j)$, i.e. the basis holds. If $m > p$, let

  $$\sigma' = \sigma_1\hat{}\sigma_2\hat{}\ldots\hat{}\sigma_p\,.$$

  It follows that $\sigma' \in \mathcal{L}(\oplus_{j=0}^{p}\mathcal{A}^j)$. Since, by assumption,

  $$p = (\lfloor h/m_\tau(\mathcal{A})\rfloor + 1) \ \text{ and } \ h = \max(\varphi(\mathcal{C}(X), N_t)\,,$$

  it follows that $\tau(\sigma') \geq N_t$. Since $M_\theta(\mathcal{A}) \leq 0$, it follows that

  $$\theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})\,,$$

  i.e. the basic case holds.
- **Induction Step:** Assume that the claim holds for a context $\mathcal{C}_1(X)$, and let $\mathcal{C}(X)$ be defined from $\mathcal{C}_1(X)$.
  1. If $\mathcal{C}(X) = \mathcal{C}_1(X)\hat{}\mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

     $$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*)\hat{}\mathcal{R} \ \text{ and } \ \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)\hat{}\mathcal{R}\,.$$

     By Definition 5, it follows that $\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X)) = \infty$. For any

     $$\sigma = \sigma_1\hat{}\sigma_{\mathcal{R}} \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*)) \ \ (\sigma_1 \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)),\ \sigma_{\mathcal{R}} \in \mathcal{L}(\mathcal{R}))\,,$$

     since, by assumption, $\tau(\sigma) = \tau(\sigma_1) + \tau(\sigma_{\mathcal{R}}) \geq N_t$, it follows that

     $$\tau(\sigma_1) \geq N_t - \tau(\sigma_{\mathcal{R}})\,.$$

     By the inductive hypothesis, it follows that there is $\sigma_1' \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p_1}\mathcal{A}^j))$ such that $\tau(\sigma_1') \geq N_t - \tau(\sigma_{\mathcal{R}})$ and $\theta(\sigma_1, \mathcal{D}) \leq \theta(\sigma_1', \mathcal{D})$ where

     $$p_1 = (\lfloor h/m_\tau(\mathcal{A})\rfloor + 1) \ \text{ and } \ h = \max(\varphi(\mathcal{C}_1(X), N_t - \tau(\sigma_{\mathcal{R}}))\,.$$

Since, by Definition 5, $\varphi(\mathcal{C}(X)) = \varphi(\mathcal{C}_1(X))$, it follows that $p \geq p_1$. By Definition 2,

$$\sigma_1' \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))\,.$$

Let $\sigma' = \sigma_1' \hat{} \sigma_{\mathcal{R}}$. By Definition 2, it follows that $\sigma' \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$. Since

$$\tau(\sigma') \geq N_t \;\text{ and }\; \theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})\,,$$

the claim follows.

2. If $\mathcal{C}(X) = \mathcal{C}_1(X) \oplus \mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*) \oplus \mathcal{R} \;\text{ and }\; \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j) \oplus \mathcal{R}\,.$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))\,.$$

Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $\omega(\mathcal{C}_1(X)) = \infty$. By Definition 2, it follows that

$$\mathcal{L}(\mathcal{C}(\mathcal{A}^*)) = \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)) \cup \mathcal{L}(\mathcal{R}), \text{ and}$$
$$\mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)) = \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)) \cup \mathcal{L}(\mathcal{R})\,.$$

By the inductive hypothesis, the result follows.

3. If $\mathcal{C}(X) = \mathcal{C}_1(X)^*$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*))^* \;\text{ and }\; \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = (\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))^*\,.$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))\,.$$

Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $\omega(\mathcal{C}_1(X)) = \infty$. By Definition 2, it follows that any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ has the following form:

$$\sigma_1 \hat{} \sigma_2 \hat{} \ldots \hat{} \sigma_m \;\; (\sigma_i \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)), 1 \leq i \leq m)\,.$$

Since, by assumption,

$$\tau(\sigma) = \tau(\sigma_1) + \tau(\sigma_2) + \ldots + \tau(\sigma_m) \geq N_t\,,$$

we can let $N_t = N_{t1} + N_{t2} + \ldots + N_{tm}$ such that $\tau(\sigma_i) \geq N_{ti}$ $(1 \leq i \leq m)$. By the inductive hypothesis, it follows that there is $\sigma_i' \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p_i}\mathcal{A}^j))$ such that

$$\tau(\sigma_i') \geq N_{ti} \;\text{ and }\; \theta(\sigma_i, \mathcal{D}) \leq \theta(\sigma_i', \mathcal{D})$$

where $p_i = (\lfloor h/m_\tau(\mathcal{A}) \rfloor + 1), h = \max(\varphi(\mathcal{C}_1(X), N_{ti})$. Since, by Definition 5, $\varphi(\mathcal{C}(X)) = \varphi(\mathcal{C}_1(X))$, it follows that $p \geq p_i$. Therefore, $\sigma_i' \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))$. Let $\sigma' = \sigma_1' \hat{} \sigma_2' \hat{} \ldots \hat{} \sigma_m'$. It follows that

$$\tau(\sigma') \geq N_t \;\text{ and }\; \theta(\sigma, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})\,.$$

Since, by Definition 2, $\sigma' \in \mathcal{L}((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))^*)$, the result follows.

4. If $\mathcal{C}(X) = (\mathcal{C}_1(X), [a, b])$ where $a \in R^+, b \in R^+, b > 0$, and $a \leq b$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*), [a, b]) \text{ and } \mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j) = (\mathcal{C}_1(\oplus_{j=0}^{p} \mathcal{A}^j), [a, b]) .$$

By Definition 5, it follows that $\omega(\mathcal{C}(X)) = \min(\omega(\mathcal{C}_1(X)), b)$. Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $\omega(\mathcal{C}_1(X)) = \infty$ and $b = \infty$. For any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, by Definition 2, it follows that $\tau(\sigma) \geq a$. Since, by Definition 2, $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$, by the inductive hypothesis, there is $\sigma' \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p_1} \mathcal{A}^j))$ such that

$$\tau(\sigma') \geq \max(N_t, a) \text{ and } \theta(\sigma_i, \mathcal{D}) \leq \theta(\sigma', \mathcal{D})$$

where $p_1 = (\lfloor h/m_\tau(\mathcal{A}) \rfloor + 1), h = \max(\varphi(\mathcal{C}_1(X), \max(N_t, a))$. Since, by Definition 5, $\varphi(\mathcal{C}(X)) = \max(\varphi(\mathcal{C}_1(X)), a)$, it follows that $p = p_1$. Since $\tau(\sigma') \geq a$, by definition 2, it follows that $\sigma' \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j))$, i.e. the claim holds. □

Lemma 13. *(5.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression such that $M_\theta(\mathcal{A}) > 0$, and $\omega(\mathcal{C}(X)) = \infty$. For any nonnegative real numbers N_t and M_r , there is $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ such that $\tau(\sigma) > N_t$ and $\theta(\sigma, \mathcal{D}) > M_r$.

Proof. We prove the claim by induction on the structure of context.

- **Basic Case:** Let $\mathcal{C}(X) = X$. Then $\mathcal{C}(\mathcal{A}^*) = \mathcal{A}^*$. Since $M_\theta(\mathcal{A}) > 0$, we can choose

$$\sigma' \in \mathcal{L}(\mathcal{A}) \text{ such that } \theta(\sigma', \mathcal{D}) = M_\theta(\mathcal{A}) ,$$

  and let

$$\sigma = \underbrace{\sigma'\hat{\ }\sigma'\hat{\ }\ldots\hat{\ }\sigma'}_{k}$$

  where $k = \max(\lfloor M_r/M_\theta(\mathcal{A}) \rfloor + 1, \lfloor N_t/m_\tau(\mathcal{A}) \rfloor + 1)$. It follows that $\sigma \in \mathcal{L}(\mathcal{A}^*)$. Since

$$\tau(\sigma) > N_t \text{ and } \theta(\sigma, \mathcal{D}) > M_r ,$$

  the basic case holds.
- **Induction Step:** Assume that claim holds for a context $\mathcal{C}_1(X)$, and let $\mathcal{C}(X)$ be defined form $\mathcal{C}_1(X)$.

1. If $\mathcal{C}(X) = \mathcal{C}_1(X)\hat{\ }\mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*)\hat{\ }\mathcal{R} .$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X)) .$$

Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $\omega(\mathcal{C}_1(X)) = \infty$. Let $\sigma_\mathcal{R} \in \mathcal{L}(\mathcal{R})$. By the inductive hypothesis, it follows that there is $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$ such that

$$\tau(\sigma) > N_t \text{ and } \theta(\sigma, \mathcal{D}) > M_r + \mid \theta(\sigma_\mathcal{R}, \mathcal{D}) \mid .$$

Let $\sigma' = \sigma\hat{\ }\sigma_{\mathcal{R}}$. By Definition 2, $\sigma' \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$. Since

$$\tau(\sigma') > N_t \text{ and } \theta(\sigma', \mathcal{D}) > M_r\,,$$

the result follows.

2. If $\mathcal{C}(X) = \mathcal{C}_1(X) \oplus \mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*) \oplus \mathcal{R}\,.$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))\,.$$

Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $\omega(\mathcal{C}_1(X)) = \infty$. By the inductive hypothesis, it follows that there is $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$ such that

$$\tau(\sigma) > N_t \text{ and } \theta(\sigma, \mathcal{D}) > M_r\,.$$

Since, by Definition 2, $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, the result follows.

3. If $\mathcal{C}(X) = \mathcal{C}_1(X)^*$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*))^*\,.$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))\,.$$

Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $\omega(\mathcal{C}_1(X)) = \infty$. By the inductive hypothesis, it follows that there is $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$ such that

$$\tau(\sigma) > N_t \text{ and } \theta(\sigma, \mathcal{D}) > M_r\,.$$

Since, by Definition 2, $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, the result follows.

4. If $\mathcal{C}(X) = (\mathcal{C}_1(X), [a, b])$ where $a \in R^+, b \in R^+, b > 0,$ and $a \leq b$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*), [a, b])\,.$$

By Definition 5, it follows that $\omega(\mathcal{C}(X)) = min(\omega(\mathcal{C}_1(X)), b)$. Since, by assumption, $\omega(\mathcal{C}(X)) = \infty$, it follows that $b = \infty$ and $\omega(\mathcal{C}_1(X)) = \infty$. By the inductive hypothesis, there is $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$ such that

$$\tau(\sigma) > max(N_t, a) \text{ and } \theta(\sigma, \mathcal{D}) > M_r\,.$$

Since, by Definition 2, $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, the claim follows. □

Lemma 14. *(6.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression, and $T \neq \infty$. Then for any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, $\tau(\sigma) \leq T$ implies $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, where $p = \lfloor T/m_\tau(\mathcal{A}) \rfloor + 1$.

Proof. We prove the claim by induction on the structure of context.

- **Basic Case:** Let $\mathcal{C}(X) = X$. Then $\mathcal{C}(\mathcal{A}^*) = \mathcal{A}^*$ and $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \oplus_{j=0}^{p}\mathcal{A}^j$. By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{A}^*)$ is of the following form:

$$\sigma_1 \hat{} \sigma_2 \hat{} \ldots \hat{} \sigma_m \quad (\sigma_i \in \mathcal{L}(\mathcal{A}), 1 \leq i \leq m)\,.$$

  Since $\tau(\sigma) = \tau(\sigma_1) + \tau(\sigma_2) + \ldots + \tau(\sigma_m) \leq T$, it follows that $m \leq p$. By Definition 2, it follows that $\sigma \in \mathcal{L}(\oplus_{j=0}^{p}\mathcal{A}^j)$, i.e. the basic case holds.
- **Induction Step:** Assume that the claim holds for a context $\mathcal{C}_1(X)$, and let $\mathcal{C}(X)$ be defined from $\mathcal{C}_1(X)$.
  1. If $\mathcal{C}(X) = \mathcal{C}_1(X)\hat{}\mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*)\hat{}\mathcal{R} \ \text{ and } \ \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)\hat{}\mathcal{R}\,.$$

     By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ is of the following form:

$$\sigma_1 \hat{} \sigma_{\mathcal{R}} \quad (\sigma_1 \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)), \sigma_{\mathcal{R}} \in \mathcal{L}(\mathcal{R}))\,.$$

     Since $\tau(\sigma) \leq T$, it follows that $\tau(\sigma_1) \leq T$. By inductive hypothesis, we have

$$\sigma_1 \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))\,.$$

     By Definition 2, it follows that $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, i.e. the claim holds.
  2. If $\mathcal{C}(X) = \mathcal{C}_1(X) \oplus \mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*) \oplus \mathcal{R} \ \text{ and } \ \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j) \oplus \mathcal{R}\,.$$

     By Definition 2, it follows that

$$\mathcal{L}(\mathcal{C}(\mathcal{A}^*)) = \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)) \cup \mathcal{L}(\mathcal{R}), \text{ and}$$
$$\mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)) = \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)) \cup \mathcal{L}(\mathcal{R})\,.$$

     For any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, if $\sigma \in \mathcal{L}(\mathcal{R})$ then $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, i.e. the claim holds; if $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$ then by the inductive hypothesis, the claim holds.
  3. If $\mathcal{C}(X) = \mathcal{C}_1(X)^*$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*))^* \ \text{ and } \ \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = (\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))^*\,.$$

     By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{A}^*)$ is of the following form:

$$\sigma_1 \hat{} \sigma_2 \hat{} \ldots \hat{} \sigma_m \quad (\sigma_i \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)), 1 \leq i \leq m)\,.$$

     Since $\tau(\sigma) \leq T$, it follows that $\tau(\sigma_i) \leq T$ $(1 \leq i \leq m)$. By the inductive hypothesis, it follows that

$$\sigma_i \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)) \quad (1 \leq i \leq m)\,.$$

     By Definition 2, it follows that $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, i.e. the claim holds.

4. If $\mathcal{C}(X) = (\mathcal{C}_1(X), [a,b])$ where $a \in R^+, b \in R^+ \cup \{\infty\}, b > 0,$ and $a \leq b$, then

$$\mathcal{C}(\mathcal{A}^*) = ((\mathcal{C}_1(\mathcal{A}^*)), [a,b]) \text{ and } \mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j) = ((\mathcal{C}_1(\oplus_{j=0}^{p} \mathcal{A}^j)), [a,b]) .$$

For any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, by Definition 2, it follows that $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$. By assumption, it follows that $\sigma \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b])$. Since

$$a \leq \tau(\sigma) \leq b ,$$

by Definiton 2, it follows that $\sigma \in \mathcal{L}(((\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j)), [a,b]))$, i.e. the claim holds. □

Lemma 15. *(7.)*Let $\mathcal{A}$ be a nonzero-simple real-time regular expression, $a \in R^+$,
$b \in R^+ \cup \{\infty\}$, $a \leq b$, and $b > 0$. Then $\mathcal{L}((\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b])) \supseteq \mathcal{L}((\mathcal{C}(\mathcal{A}^*), [a,b]))$, where $p = \lfloor b/m_\tau(\mathcal{A}) \rfloor + 1$.

Proof. We prove the claim by induction on the structure of context.

- **Basic Case:** Let $\mathcal{C}(X) = X$. Then

$$(\mathcal{C}(\mathcal{A}^*), [a,b]) = (\mathcal{A}^*, [a,b]) \text{ and } (\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b]) = (\oplus_{j=0}^{p} \mathcal{A}^j, [a,b]) .$$

By Definition 2, any $\sigma \in \mathcal{L}((\mathcal{C}(\mathcal{A}^*), [a,b]))$ is of the following form:

$$\sigma_1 \hat{} \sigma_2 \hat{} \ldots \hat{} \sigma_m \quad (\sigma_i \in \mathcal{L}(\mathcal{A}), 1 \leq i \leq m) .$$

Since, by Definition 2, $\tau(\sigma) = \tau(\sigma_1) + \tau(\sigma_2) + \ldots + \tau(\sigma_m) \leq b$, it follows that $m \leq p$. By Definition 2.2, it follows that $\sigma \in \mathcal{L}((\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b]))$, i.e. the basic case holds.
- **Induction Step:** Assume that the claim holds for a context $\mathcal{C}_1(X)$, and let $\mathcal{C}(X)$ be defined from $\mathcal{C}_1(X)$
  1. If $\mathcal{C}(X) = \mathcal{C}_1(X) \hat{} \mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$(\mathcal{C}(\mathcal{A}^*), [a,b]) = (\mathcal{C}_1(\mathcal{A}^*) \hat{} \mathcal{R}, [a,b]), \text{ and}$$
$$(\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b]) = (\mathcal{C}_1(\oplus_{j=0}^{p} \mathcal{A}^j) \hat{} \mathcal{R}, [a,b]) .$$

By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ is of the following form:

$$\sigma_1 \hat{} \sigma_{\mathcal{R}} \quad (\sigma_1 \in \mathcal{L}((\mathcal{C}_1(\mathcal{A}^*), [a,b])),\ \sigma_{\mathcal{R}} \in \mathcal{L}(\mathcal{R})) .$$

By the inductive hypothesis, it follows that

$$\sigma_1 \in \mathcal{L}((\mathcal{C}_1(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b])) ,$$

and hence by Definition 2,

$$\sigma \in \mathcal{L}((\mathcal{C}(\oplus_{j=0}^{p} \mathcal{A}^j), [a,b])) ,$$

i.e. the claim holds.

2. If $\mathcal{C}(X) = \mathcal{C}_1(X) \oplus \mathcal{R}$ where $\mathcal{R}$ is a RRE, then

$$(\mathcal{C}(\mathcal{A}^*), [a,b]) = (\mathcal{C}_1(\mathcal{A}^*) \oplus \mathcal{R}, [a,b]), \text{ and}$$
$$(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b]) = (\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j) \oplus \mathcal{R}, [a,b]) .$$

By Definition 2, it follows that

$$\mathcal{L}((\mathcal{C}(\mathcal{A}^*), [a,b])) = \mathcal{L}((\mathcal{C}_1(\mathcal{A}^*), [a,b])) \cup \mathcal{L}(\mathcal{R}), \text{ and}$$
$$\mathcal{L}((\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b])) = \mathcal{L}((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b])) \cup \mathcal{L}(\mathcal{R}) .$$

By assumption, the claim holds.

3. If $\mathcal{C}(X) = \mathcal{C}_1(X)^*$, then

$$(\mathcal{C}(\mathcal{A}^*), [a,b]) = ((\mathcal{C}_1(\mathcal{A}^*))^*, [a,b]), \text{ and}$$
$$(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b]) = ((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))^*, [a,b]) .$$

By Definition 2, it follows that any $\sigma \in \mathcal{L}((\mathcal{C}(\mathcal{A}^*), [a,b]))$ has the following form:

$$\sigma_1\hat{\ }\sigma_2\hat{\ }\ldots\hat{\ }\sigma_m \quad (\sigma_i \in \mathcal{L}((\mathcal{C}_1(\mathcal{A}^*)), [a,b]), 1 \le i \le m) .$$

By the inductive hypothesis, it follows that

$$\sigma_i \in \mathcal{L}((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b])) \quad (1 \le i \le m) .$$

By Definition 2, it follows that $\sigma \in \mathcal{L}((\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b]))$, i.e. the claim holds.

4. If $\mathcal{C}(X) = (\mathcal{C}_1(X), [a_1, b_1])$ where $a_1 \in R^+, b_1 \in R^+, b_1 > 0,$ and $a_1 \le b_1$, then

$$(\mathcal{C}(\mathcal{A}^*), [a,b]) = ((\mathcal{C}_1(\mathcal{A}^*), [a_1,b_1]), [a,b]), \text{ and}$$
$$(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b]) = ((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j), [a_1,b_1]), [a,b]) .$$

For any $\sigma \in \mathcal{L}((\mathcal{C}(\mathcal{A}^*), [a,b]))$, by Definition 2, it follows that

$$\sigma \in \mathcal{L}((\mathcal{C}_1(\mathcal{A}^*), [a,b])) .$$

By the inductive hypothesis, it follows that

$$\sigma \in \mathcal{L}((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j), [a,b])) .$$

Hence, by Definition 2, $\sigma \in \mathcal{L}(((\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j), [a_1,b_1]), [a,b]))$, i.e. the claim holds. □

Lemma 16. *(8.)* Let $\mathcal{A}$ be a nonzero-simple real-time regular expression. Then $\mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)) \supseteq \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, where $p = \lfloor \omega(\mathcal{C}(X))/m_\tau(\mathcal{A}) \rfloor + 1$.

Proof. We prove the claim by induction on the structure of context.

- **Basic Case:** Let $\mathcal{C}(X) = X$. Then $\mathcal{C}(\mathcal{A}^*) = \mathcal{A}^*$ and $\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \oplus_{j=0}^{p}\mathcal{A}^j$. Since, by Definition 5, $\omega(\mathcal{C}(X)) = \infty$, it follows that

$$p = \infty \quad \text{and} \quad \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \oplus_{j=0}^{\infty}\mathcal{A}^j .$$

Since, by Definition 2, $\mathcal{L}(\mathcal{A}^*) = \mathcal{L}(\oplus_{j=0}^{\infty}\mathcal{A}^j)$, the basic case holds.

– **Induction Step:** Assume that the claim holds for a context $\mathcal{C}_1(X)$, and let $\mathcal{C}(X)$ be defined from $\mathcal{C}_1(X)$

1. If $\mathcal{C}(X) = \mathcal{C}_1(X)\hat{}\mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*)\hat{}\mathcal{R} \text{ and } \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)\hat{}\mathcal{R}\,.$$

By Definition 2, any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ is of the following form:

$$\sigma_1\hat{}\sigma_{\mathcal{R}} \quad (\sigma_1 \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)),\ \sigma_{\mathcal{R}} \in \mathcal{L}(\mathcal{R}))\,.$$

Since, by Definition 5, $\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))$, by the inductive hypothesis, it follows that $\sigma_1 \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))$. By Definition 2, it follows that $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, i.e. the claim holds.

2. If $\mathcal{C}(X) = \mathcal{C}_1(X) \oplus \mathcal{R}$ where $\mathcal{R}$ is a real-time regular expression, then

$$\mathcal{C}(\mathcal{A}^*) = \mathcal{C}_1(\mathcal{A}^*) \oplus \mathcal{R} \text{ and } \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = \mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j) \oplus \mathcal{R}\,.$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))\,.$$

By Definition 2, it follows that

$$\mathcal{L}(\mathcal{C}(\mathcal{A}^*)) = \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)) \cup \mathcal{L}(\mathcal{R}), \text{ and}$$
$$\mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j)) = \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)) \cup \mathcal{L}(\mathcal{R})\,.$$

For any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$, if $\sigma \in \mathcal{L}(\mathcal{R})$, then the claim holds;
if $\sigma \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*))$, then by the inductive hypothesis, the claim holds.

3. If $\mathcal{C}(X) = \mathcal{C}_1(X)^*$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*))^* \text{ and } \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = (\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j))^*\,.$$

By Definition 5, it follows that

$$\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))\,.$$

By Definition 2, it follows that any $\sigma \in \mathcal{L}(\mathcal{C}(\mathcal{A}^*))$ has following form:

$$\sigma_1\hat{}\sigma_2\hat{}\ldots\hat{}\sigma_m \quad (\sigma_i \in \mathcal{L}(\mathcal{C}_1(\mathcal{A}^*)), 1 \le i \le m)\,.$$

By the inductive hypothesis, it follows that

$$\sigma_i \in \mathcal{L}(\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j)) \quad (1 \le i \le m)\,.$$

By Definition 2, it follows that $\sigma \in \mathcal{L}(\mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j))$, i.e. the claim holds.

4. If $\mathcal{C}(X) = (\mathcal{C}_1(X), [a, b])$ where $a \in R^+, b \in R^+ \cup \{\infty\}, b > 0,$ and $a \le b$, then

$$\mathcal{C}(\mathcal{A}^*) = (\mathcal{C}_1(\mathcal{A}^*), [a, b]) \text{ and } \mathcal{C}(\oplus_{j=0}^{p}\mathcal{A}^j) = (\mathcal{C}_1(\oplus_{j=0}^{p}\mathcal{A}^j), [a, b])\,.$$

By Definition 5, it follows that $\omega(\mathcal{C}(X)) = min(\omega(\mathcal{C}_1(X)), b)$.
If $\omega(\mathcal{C}(X)) = b$, by Lemma 7, the claim holds. If $\omega(\mathcal{C}(X)) = \omega(\mathcal{C}_1(X))$, by the inductive hypothesis, the claim holds. □