

An Algebraic Approach to Energy Problems II The Algebra of Energy Functions*

Zoltán Ésik,^a Uli Fahrenberg,^b Axel Legay,^c and Karin Quaas^d

Abstract

Energy and resource management problems are important in areas such as embedded systems or autonomous systems. They are concerned with the question whether a given system admits infinite schedules during which certain tasks can be repeatedly accomplished and the system never runs out of energy (or other resources). In order to develop a general theory of energy problems, we introduce energy automata: finite automata whose transitions are labeled with energy functions which specify how energy values change from one system state to another.

We show that energy functions form a *-continuous Kleene ω -algebra, as an application of a general result that finitely additive, locally *-closed and \top -continuous functions on complete lattices form *-continuous Kleene ω -algebras. This permits to solve energy problems in energy automata in a generic, algebraic way. In order to put our work in context, we also review extensions of energy problems to higher dimensions and to games.

Keywords: Energy problem, Kleene algebra, *-continuity, *-continuous Kleene ω -algebra

1 Introduction

Energy and resource management problems are important in areas such as embedded systems or autonomous systems. They are concerned with the question whether a given system admits infinite schedules during which (1) certain tasks can be repeatedly accomplished and (2) the system never runs out of energy (or

*This research was supported by grant no. K 108448 from the National Foundation of Hungary for Scientific Research (OTKA), by ANR MALTHY, grant no. ANR-13-INSE-0003 from the French National Research Foundation, and by Deutsche Forschungsgemeinschaft (DFG), projects QU 316/1-1 and QU 316/1-2.

^aUniversity of Szeged, Hungary (deceased)

^bÉcole polytechnique, Palaiseau, France. Most of this work was carried out while this author was still employed at Inria Rennes.

^cInria Rennes, France

^dUniversität Leipzig, Germany

other specified resources). Starting with [11], formal modeling and analysis of such problems has recently attracted some attention [10, 12, 15, 19, 22, 33, 39, 46, 48].

As an example, Fig. 1 shows a simple model of an electric car, modeled as a weighted timed automaton [4, 5]. In the *working* state W , energy is consumed at a rate of 10 energy units per time unit; in the two *recharging* states R_1 and R_2 , the battery is charged at a rate of 20, respectively 10, energy units per time unit. As the clock c is reset ($c \leftarrow 0$) when entering state W and has guard $c \geq 1$ on outgoing transitions, we ensure that the car always has to be in state W for at least one time unit. Similarly, the system can only transition back from states R_1, R_2 to W if it has spent at most one time unit in these states.

Passing from state W to R_2 (and back) requires 2 energy units, whereas passing from W to R_1 requires 6 energy units and passing back from R_1 to W requires 4 energy units. Passing from R_2 to R_1 requires 5 energy units, and passing from R_1 to R_2 requires 1 energy unit.

Altogether, this is intended to model the fact that there are two recharge stations available, one close to work but less powerful, and a more powerful one further away and located uphill, so that moving upwards costs more energy than moving downwards. Now assume that the initial state W is entered with a given *initial energy* x_0 , then the energy problem of this model is as follows: Does there exist an infinite trace which (1) visits W infinitely often and (2) never has an energy level below 0?

In order to develop a general theory which can be applied to the above and other types of energy problems, we have in [27, 36] introduced *energy automata*. These are finite automata whose transitions are labeled with *energy functions* which specify how energy values change from one system state to another. Using the theory of semiring-weighted automata [24], we have shown in [27] that energy problems in such automata can be solved in a simple static way which only involves manipulations of energy functions.

In order to put the work of [27] on a more solid theoretical footing and with an eye to future generalizations, we have recently introduced a new algebraic structure of **-continuous Kleene ω -algebras* [25, 26].

In this paper, we are concerned with conditions under which functions on complete lattices form *-continuous Kleene ω -algebras. We show that sets of functions which are *finitely additive*, *locally *-closed* and \top -*continuous*, all natural conditions which we will introduce later, form *-continuous Kleene ω -algebras. We then show that energy functions are an example of such functions.

Using general results concerning coverability and Büchi acceptance in automata with transition weights in *-continuous Kleene ω -algebras, we are then able to solve

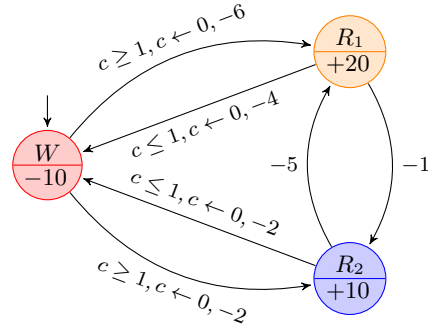


Figure 1: Simple model of an electric car as a weighted timed automaton.

energy problems in energy automata in a generic, algebraic way.

In order to put our work in context, we also review extensions of energy problems to higher dimensions and to games. We show that even though our algebraic setting does not apply here, coverability for multi-dimensional energy automata is decidable. Energy games, on the other hand, are shown to be undecidable from dimension two.

Structure of the Paper This is the second in a series of two papers which are concerned with energy problems and their algebraic foundation. In the first paper of the series [28], we have introduced continuous and $*$ -continuous Kleene ω -algebras and exposed some of their algebraic properties. We have shown that every $*$ -continuous Kleene ω -algebra is an iteration semiring-semimodule pair.

In this paper, we continue our work by showing how to compute Büchi acceptance in Section 3. Note that our two papers can be read independently, as we have taken care to recall the relevant results obtained in [28].

We then turn our attention to $*$ -continuous Kleene ω -algebras of functions. In Section 4 we introduce the properties of finite additivity, $*$ -closedness and \top -continuity and show that any set S of finitely additive, $*$ -closed and \top -continuous functions on a complete lattice L form $*$ -continuous Kleene algebras.

In Section 5 we extend this result and show that if (S, V) is such that S is a $*$ -continuous Kleene algebra of functions $L \rightarrow L$, V consists of finitely additive and \top -continuous functions $L \rightarrow \mathbf{2}$, where $\mathbf{2}$ denotes the Boolean lattice, then (S, V) forms a $*$ -continuous Kleene ω -algebra. We then apply this result to energy automata in Section 6.

In Section 7 we take a more detailed look on two important subclasses of (computable) energy functions and obtain some complexity results. Section 8 reviews a reduction from energy problems on weighted timed automata to our energy automata, in order to further motivate our notions of energy function and energy automaton.

The final Section 9 is concerned with extensions of energy problems to higher dimensions and to games. Using an extension of the Rackoff technique for affine Petri nets, we show that coverability for multi-dimensional energy automata is decidable in exponential time. On the other hand, for a slightly relaxed version of energy function, coverability becomes undecidable from dimension four. Likewise, reachability games on two-dimensional energy automata and on one-dimensional relaxed energy automata are undecidable.

Related Work A simple class of energy automata is the one of *integer-weighted automata*, where all energy functions are updates of the form $x \mapsto x + k$ for some (positive or negative) integer k . Energy problems on these automata, and their extensions to multiple weights (also called *vector addition systems with states* (VASS)) and to games, have been considered for example in [11, 14, 17–20, 33]. The exact complexity of the reachability problem for VASS is one of the most challenging open problems in theoretical computer science; plenty of very recent results aim

to get more insight into this problem [7, 38, 42, 43]. Our energy automata may be considered as a generalization of one-dimensional VASS to arbitrary updates; in the final section of this paper we will also be concerned with multi-dimensional energy automata and games.

Energy problems on *timed automata* [2] have been considered in [10–12, 46]. Here timed automata are enriched with integer weights in locations and on transitions (the *weighted timed automata* of [4, 5], cf. Fig. 1), with the semantics that the weight of a delay in a location is computed by multiplying the length of the delay by the location weight. In [11] it is shown that energy problems for one-clock weighted timed automata without updates on transitions (hence only with weights in locations) can be reduced to energy problems on integer-weighted automata with additive updates.

For one-clock weighted timed automata *with* transition updates, energy problems are shown decidable in [10], using a reduction to energy automata as we use them here. Intuitively, each path in the timed automaton in which the clock is not reset is converted to an edge in an energy automaton, labeled with a *piecewise affine* energy function. We review the reduction from [10] in Section 8 of the present paper. In a recent paper [16], this class of real-time energy problems is treated directly, in the setting of *-continuous Kleene ω -algebras, without a reduction to the untimed setting.

Also another class of energy problems on weighted timed automata is considered in [10], in which weights during delays are increasing *exponentially* rather than linearly. These are shown decidable using a reduction to energy automata with *piecewise polynomial* energy functions; again our present framework applies.

Acknowledgment We are deeply indebted to our colleague and friend Zoltán Ésik with whom we started this research and who has led us along the way. Unfortunately Zoltán could not see this work completed, so any errors are the responsibility of the last three authors.

The second author also acknowledges interesting discussions with Patricia Bouyer, Kim G. Larsen and Nicolas Markey which led to [10] and eventually to Section 8 of this paper.

2 Energy Automata

We recall the energy automata introduced in [28] and the decision problems we are interested in. Let $[0, \infty]_{\perp} = \{\perp\} \cup [0, \infty]$ denote the complete lattice of non-negative real numbers together with extra elements \perp and ∞ , with the standard order on $\mathbb{R}_{\geq 0}$ extended by $\perp < x < \infty$ for all $x \in \mathbb{R}_{\geq 0}$. Also, $\perp + x = \perp - x = \perp$ for all $x \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ and $\infty + x = \infty - x = \infty$ for all $x \in \mathbb{R}_{\geq 0}$.

Definition 1. An (*extended*) energy function is a mapping $f : [0, \infty]_{\perp} \rightarrow [0, \infty]_{\perp}$, for which $\perp f = \perp$ and

$$yf \geq xf + y - x \tag{*}$$

for all $x \leq y$. Moreover, $\infty f = \infty$, unless $xf = \perp$ for all $x \in [0, \infty]_{\perp}$. The class of all extended energy functions is denoted \mathcal{E} .

We write function composition and application in diagrammatical order, from left to right. Hence we write $f;g$, or simply fg , for the composition $g \circ f$ and $x;f$ or xf for function application $f(x)$. This is because we will be concerned with *algebras* of functions, in which function composition is multiplication, and where it is customary to write multiplication in diagrammatical order.

We define a partial order on \mathcal{E} , by $f \leq g$ iff $xf \leq xg$ for all $x \in [0, \infty]_{\perp}$. We will need three special energy functions, \perp , id and \top ; these are given by $x\perp = \perp$, $x;\text{id} = x$ for $x \in [0, \infty]_{\perp}$, and $\perp\top = \perp$, $x\top = \infty$ for $x \in [0, \infty]$.

Lemma 1 ([28]). *With the ordering \leq , \mathcal{E} is a complete lattice with bottom element \perp and top element \top . The supremum on \mathcal{E} is pointwise, i.e., $x(\sup_{i \in I} f_i) = \sup_{i \in I} xf_i$ for any set I , all $f_i \in \mathcal{E}$ and $x \in [0, \infty]_{\perp}$. Also, $h(\sup_{i \in I} f_i) = \sup_{i \in I}(hf_i)$ for all $h \in \mathcal{E}$.*

We denote binary suprema using the symbol \vee ; hence $f \vee g$, for $f, g \in \mathcal{E}$, is the function $x(f \vee g) = \max(xf, xg)$. For a subset $\mathcal{E}' \subseteq \mathcal{E}$, we write $\langle \mathcal{E}' \rangle$ for the set of all finite suprema $a_1 \vee \dots \vee a_m$ with $a_i \in \mathcal{E}'$ for each $i = 1, \dots, m$.

Definition 2. *Let $\mathcal{E}' \subseteq \mathcal{E}$ and $n \geq 1$. An \mathcal{E}' -automaton of dimension n is a structure (α, M, k) , where $\alpha \in \{\perp, \text{id}\}^n$ is the initial vector, $M \in \langle \mathcal{E}' \rangle^{n \times n}$ is the transition matrix, and k is an integer $0 \leq k \leq n$.*

Combinatorially, this may be represented as a transition system whose set of states is $\{1, \dots, n\}$. For any pair of states i, j , the transitions from i to j are determined by the entry $M_{i,j}$ of the transition matrix: if $M_{i,j} = f_1 \vee \dots \vee f_m$, then there are m transitions from i to j , respectively labeled f_1, \dots, f_m . The states i with $\alpha_i = \text{id}$ are *initial*, and the states $\{1, \dots, k\}$ are *accepting*.

Recall that an *idempotent semiring* [6,37] $S = (S, \vee, \cdot, \perp, 1)$ consists of a commutative idempotent monoid (S, \vee, \perp) and a monoid $(S, \cdot, 1)$ such that the distributive laws

$$\begin{aligned} x(y \vee z) &= xy \vee xz \\ (y \vee z)x &= yx \vee zx \end{aligned}$$

and the zero laws

$$\perp \cdot x = \perp = x \cdot \perp$$

hold for all $x, y, z \in S$. It follows that the product operation distributes over all finite sums.

Each idempotent semiring S is partially ordered by its *natural order* relation $x \leq y$ iff $x \vee y = y$, and then sum and product preserve the partial order and \perp is the least element. Moreover, for all $x, y \in S$, $x \vee y$ is the least upper bound of the set $\{x, y\}$.

Lemma 2 ([28]). *$(\mathcal{E}, \vee, \circ, \perp, \text{id})$ is an idempotent semiring with natural order \leq .*

An energy automaton is hence a weighted automaton over the semiring \mathcal{E} , in the sense of [24]. We recall the decision problems which we are interested in. As the input to a decision problem must be in some way finitely representable, we will state them for subsets $\mathcal{E}' \subseteq \mathcal{E}$ of *computable* energy functions. Note that we give no technical meaning to the term “computable” here; we simply need to take care that the input can be finitely represented.

Problem 1 (State reachability). Given an \mathcal{E}' -automaton (α, M, k) of dimension $n \geq 1$ and a computable initial energy $x_0 \in \mathbb{R}_{\geq 0}$: does there exist a finite sequence (k_0, \dots, k_m) of indices $1 \leq k_i \leq n$ such that $\alpha_{k_0} = \text{id}$, $k_m \leq k$, and $x_0 M_{k_0, k_1} \cdots M_{k_{m-1}, k_m} \neq \perp$?

Using the representation of $A = (\alpha, M, k)$ as a transition system, we see that the above problem amounts to asking whether there exists a finite path in A , with transition labels $M_{k_0, k_1}, \dots, M_{k_{m-1}, k_m}$, such that the path starts in an initial state, ends in an accepting state, and $x_0 M_{k_0, k_1} \cdots M_{k_{m-1}, k_m} \neq \perp$.

Problem 2 (Coverability). Given an \mathcal{E}' -automaton (α, M, k) of dimension $n \geq 1$, a computable initial energy $x_0 \in \mathbb{R}_{\geq 0}$ and a computable function $z : \{1, \dots, k\} \rightarrow \mathbb{R}_{\geq 0}$: does there exist a sequence (k_0, \dots, k_m) of indices $1 \leq k_i \leq n$ such that $\alpha_{k_0} = \text{id}$, $k_m \leq k$, and $x_0 M_{k_0, k_1} \cdots M_{k_{m-1}, k_m} \geq k_m z$?

In the transition system representation of $A = (\alpha, M, k)$, this amounts to asking whether there exists a finite path in A as above, starting in an initial state and ending in an accepting state k_m , and such that $x_0 M_{k_0, k_1} \cdots M_{k_{m-1}, k_m} \geq k_m z$. Using the function z with $iz = 0$ for all $i = 1, \dots, k$, coverability reduces to state reachability.

Problem 3 (Büchi acceptance). Given an \mathcal{E}' -automaton (α, M, k) of dimension $n \geq 1$ and a computable initial energy $x_0 \in \mathbb{R}_{\geq 0}$: does there exist an infinite sequence (k_0, k_1, \dots) of indices $1 \leq k_i \leq n$ such that $\alpha_{k_0} = \text{id}$, $k_i \leq k$ for infinitely many indices i , and $x_0 M_{k_0, k_1} \cdots M_{k_{m-1}, k_m} \neq \perp$ for all $m \geq 1$?

Again using the representation of $A = (\alpha, M, k)$ as a transition system, we see that this last problem amounts to asking whether there exists an infinite path in A , with transition labels $M_{k_0, k_1}, M_{k_1, k_2}, \dots$, such that the path starts in an initial state, visits an accepting state infinitely often, and all finite prefixes $x_0 M_{k_0, k_1} \cdots M_{k_{m-1}, k_m} \neq \perp$.

We let $\text{Reach}_{\mathcal{E}'}$ denote the function which maps an \mathcal{E}' -automaton together with an initial energy to the Boolean values **ff** or **tt** depending on whether the answer to the concrete state reachability problem is negative or positive. $\text{Cover}_{\mathcal{E}'}$ and $\text{Büchi}_{\mathcal{E}'}$ denote the similar mappings for the coverability and Büchi acceptance problems.

3 Büchi Automata in *-Continuous Kleene ω -Algebras

We recall the notion of **-continuous Kleene ω -algebra* introduced in [28]. First, a **-continuous Kleene algebra* is an idempotent semiring $(S, \vee, \cdot, \perp, 1)$ in which the

infinite suprema $\bigvee\{x^n \mid n \geq 0\}$ exist for all $x \in S$ and product preserves such suprema:

$$y\left(\bigvee_{n \geq 0} x^n\right) = \bigvee_{n \geq 0} yx^n \quad \text{and} \quad \left(\bigvee_{n \geq 0} x^n\right)y = \bigvee_{n \geq 0} x^n y$$

for all $x, y \in S$. One then defines $x^* = \bigvee\{x^n \mid n \geq 0\}$ for every $x \in S$.

A *-continuous Kleene algebra is *continuous* if all suprema $\bigvee X, X \subseteq S$, exist and are preserved by products. *-continuous Kleene algebras are hence a generalization of continuous Kleene algebras. There are interesting Kleene algebras which are *-continuous but not continuous, for example the Kleene algebra of all regular languages over some alphabet, see [28].

Recall that an *idempotent semiring-semimodule pair* [8, 31] (S, V) consists of an idempotent semiring $S = (S, \vee, \cdot, \perp, 1)$ and a commutative idempotent monoid $V = (V, \vee, \perp)$ which is equipped with a left S -action $S \times V \rightarrow V, (x, v) \mapsto xv$, satisfying

$$\begin{aligned} (x \vee x')v &= xv \vee x'v & x(v \vee v') &= xv \vee xv' \\ (xx')v &= x(x'v) & \perp v &= \perp \\ x\perp &= \perp & 1v &= v \end{aligned}$$

for all $x, x' \in S$ and $v \in V$. In that case, we also call V a (*left*) S -semimodule.

A *generalized *-continuous Kleene algebra* [28] is a semiring-semimodule pair (S, V) where $S = (S, \vee, \cdot, *, \perp, 1)$ is a *-continuous Kleene algebra such that

$$xy^*v = \bigvee_{n \geq 0} xy^n v$$

for all $x, y \in S$ and $v \in V$.

A **-continuous Kleene ω -algebra* [28] consists of a generalized *-continuous Kleene algebra (S, V) together with an infinite product operation $S^\omega \rightarrow V$ which maps every infinite sequence x_0, x_1, \dots in S to an element $\prod_{n \geq 0} x_n$ of V . The infinite product is subject to the following conditions:

Ax1: For all $x_0, x_1, \dots \in S, \prod_{n \geq 0} x_n = x_0 \prod_{n \geq 0} x_{n+1}$.

Ax2: Let $x_0, x_1, \dots \in S$ and $0 = n_0 \leq n_1 \dots$ be a sequence which increases without a bound. Let $y_k = x_{n_k} \dots x_{n_{k+1}-1}$ for all $k \geq 0$. Then $\prod_{n \geq 0} x_n = \prod_{k \geq 0} y_k$.

Ax3: For all x_0, x_1, \dots and y, z in $S, \prod_{n \geq 0} (x_n(y \vee z)) = \bigvee_{x'_n \in \{y, z\}} \prod_{n \geq 0} x_n x'_n$.

Ax4: For all $x, y_0, y_1, \dots \in S, \prod_{n \geq 0} x^* y_n = \bigvee_{k_n \geq 0} \prod_{n \geq 0} x^{k_n} y_n$.

A *continuous Kleene ω -algebra* [31] is a semiring-semimodule pair (S, V) in which S is a continuous Kleene algebra, V is a complete lattice, and the S -action on V preserves all suprema in either argument, together with an infinite product as above which satisfies conditions Ax1 and Ax2 above and preserves all suprema. *-continuous Kleene ω -algebras are hence a generalization of continuous Kleene ω -algebras.

For any idempotent semiring S and $n \geq 1$, we can form the matrix semiring $S^{n \times n}$ whose elements are $n \times n$ -matrices of elements of S and whose sum and product are given as the usual matrix sum and product. It is known [41] that when S is a $*$ -continuous Kleene algebra, then $S^{n \times n}$ is also a $*$ -continuous Kleene algebra, with the $*$ -operation defined by

$$M_{i,j}^* = \bigvee_{m \geq 0} \bigvee_{1 \leq k_1, \dots, k_m \leq n} M_{i,k_1} M_{k_1,k_2} \cdots M_{k_m,j}$$

for all $M \in S^{n \times n}$ and $1 \leq i, j \leq n$. The above infinite supremum exists, as it is taken over a regular set, see [28]. Also, if $n \geq 2$ and $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, where a and d are square matrices of dimension less than n , then

$$M^* = \begin{pmatrix} (a \vee bd^*c)^* & (a \vee bd^*c)^*bd^* \\ (d \vee ca^*b)^*ca^* & (d \vee ca^*b)^* \end{pmatrix}. \tag{1}$$

For any semiring-semimodule pair (S, V) and $n \geq 1$, we can form the matrix semiring-semimodule pair $(S^{n \times n}, V^n)$ whose elements are $n \times n$ -matrices of elements of S and n -dimensional (column) vectors of elements of V , with the action of $S^{n \times n}$ on V^n given by the usual matrix-vector product.

When (S, V) is a $*$ -continuous Kleene ω -algebra, then $(S^{n \times n}, V^n)$ is a generalized $*$ -continuous Kleene algebra. By [28, Lemma 14], there is an ω -operation on $S^{n \times n}$ defined by

$$M_i^\omega = \bigvee_{1 \leq k_1, k_2, \dots \leq n} M_{i,k_1} M_{k_1,k_2} \cdots$$

for all $M \in S^{n \times n}$ and $1 \leq i \leq n$. Also, if $n \geq 2$ and $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, where a and d are square matrices of dimension less than n , then

$$M^\omega = \begin{pmatrix} (a \vee bd^*c)^\omega \vee (a \vee bd^*c)^*bd^\omega \\ (d \vee ca^*b)^\omega \vee (d \vee ca^*b)^*ca^\omega \end{pmatrix}. \tag{2}$$

Note [28] that it is not generally the case that $(S^{n \times n}, V^n)$ is again a $*$ -continuous Kleene ω -algebra, as the infinite product may not exist.

Let (S, V) be a $*$ -continuous Kleene ω -algebra and $A \subseteq S$ a subset. We write $\langle A \rangle$ for the set of all finite suprema $a_1 \vee \cdots \vee a_m$ with $a_i \in A$ for each $i = 1, \dots, m$.

A *weighted automaton* [32] over A of dimension $n \geq 1$ is a tuple (α, M, k) , where $\alpha \in \{\perp, 1\}^n$ is the initial vector, $M \in \langle A \rangle^{n \times n}$ is the transition matrix, and k is an integer $0 \leq k \leq n$. Combinatorially, this may be represented as a transition system whose set of states is $\{1, \dots, n\}$. For any pair of states i, j , the transitions from i to j are determined by the entry $M_{i,j}$ of the transition matrix: if $M_{i,j} = a_1 \vee \cdots \vee a_m$, then there are m transitions from i to j , respectively labeled a_1, \dots, a_m . The states i with $\alpha_i = 1$ are *initial*, and the states $\{1, \dots, k\}$ are *accepting*.

The *finite behavior* of a weighted automaton $A = (\alpha, M, k)$ is defined to be

$$|A| = \alpha M^* \kappa,$$

where $\kappa \in \{\perp, 1\}^n$ is the vector given by $\kappa_i = 1$ for $i \leq k$ and $\kappa_i = \perp$ for $i > k$. (Note that α has to be used as a *row* vector for this multiplication to make sense.) It is clear by (1) that $|A|$ is the supremum of the products of the transition labels along all paths in A from any initial to any accepting state.

The *Büchi behavior* of a weighted automaton $A = (\alpha, M, k)$ is defined to be

$$\|A\| = \alpha \begin{pmatrix} (a \vee bd^*c)^\omega \\ d^*c(a \vee bd^*c)^\omega \end{pmatrix},$$

where $a \in \langle A \rangle^{k \times k}$, $b \in \langle A \rangle^{k \times (n-k)}$, $c \in \langle A \rangle^{(n-k) \times n}$ and $d \in \langle A \rangle^{(n-k) \times (n-k)}$ are such that $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. By (2), $\|A\|$ is the supremum of the products of the transition labels along all infinite paths in A from any initial state which infinitely often visit an accepting state.

For completeness we also mention a Kleene theorem for the Büchi automata introduced above, which is a direct consequence of the Kleene theorem for Conway semiring-semimodule pairs, cf. [29, 32].

Theorem 1. *An element of V is the Büchi behavior weighted automaton over A iff it is rational over A , i.e., when it can be generated from the elements of A by the semiring and semimodule operations, the action, and the star and omega operations.*

It is a routine matter to show that an element of V is rational over A iff it can be written as $\bigvee_{i=1}^n x_i y_i^\omega$, where each x_i and y_i can be generated from A by \vee , \cdot , and $*$.

4 Generalized *-Continuous Kleene Algebras of Functions

In the following two sections our aim is to establish properties which ensure that semiring-semimodule pairs of *functions* form *-continuous Kleene ω -algebras. We will use these properties in Section 6 to show that energy functions form a *-continuous Kleene ω -algebra.

Let L and L' be complete lattices with bottom and top elements \perp and \top . Then a function $f : L \rightarrow L'$ is said to be *finitely additive* if $\perp f = \perp$ and $(x \vee y)f = xf \vee yf$ for all $x, y \in L$. (Recall that we write function application and composition in the diagrammatic order, from left to right.) When $f : L \rightarrow L'$ is finitely additive, then $(\bigvee X)f = \bigvee Xf$ for all finite sets $X \subseteq L$.

Consider the collection $\text{FinAdd}_{L,L'}$ of all finitely additive functions $f : L \rightarrow L'$, ordered pointwise. Since the (pointwise) supremum of any set of finitely additive functions is finitely additive, $\text{FinAdd}_{L,L'}$ is also a complete lattice, in which the supremum of any set of functions can be constructed pointwise. The least and greatest elements are the functions \perp and \top given by $x\perp = \perp$ for $x \in L$, $\perp\top = \perp$, and $x\top = \top$ for $x \in \setminus\{\perp\}$.

Definition 3. A function $f \in \text{FinAdd}_{L,L'}$ is said to be \top -continuous if $f = \perp$ or for all $X \subseteq L$ with $\bigvee X = \top$, also $\bigvee Xf = \top$.

Note that if $f \neq \perp$ is \top -continuous, then $\top f = \top$. The functions id and \perp are \top -continuous. Also, the (pointwise) supremum of any set of \top -continuous functions is again \top -continuous.

We will first be concerned with functions in $\text{FinAdd}_{L,L}$, which we just denote FinAdd_L . Since the composition of finitely additive functions is finitely additive and the identity function id over L is finitely additive, and since composition of finitely additive functions distributes over finite suprema, FinAdd_L , equipped with the operation \vee (binary supremum), $;$ (composition), and the constant function \perp and the identity function id as 1, is an idempotent semiring. It follows that when f is finitely additive, then so is $f^* = \bigvee_{n \geq 0} f^n$. Moreover, $f \leq f^*$ and $f^* \leq g^*$ whenever $f \leq g$.

Lemma 3. Let S be any subsemiring of FinAdd_L closed under the $*$ -operation. Then S is a $*$ -continuous Kleene algebra iff for all $g, h \in S$, $g^*h = \bigvee_{n \geq 0} g^n h$.

Proof. Suppose that the precondition of the lemma holds. We need to show that $f(\bigvee_{n \geq 0} g^n)h = \bigvee_{n \geq 0} f g^n h$ for all $f, g, h \in S$. But $f(\bigvee_{n \geq 0} g^n)h = f(\bigvee_{n \geq 0} g^n h)$ by assumption, and we conclude that $f(\bigvee_{n \geq 0} g^n h) = \bigvee_{n \geq 0} f g^n h$ since the supremum is pointwise. \square

Compositions of \top -continuous functions in FinAdd_L are again \top -continuous, so that the collection of all \top -continuous functions in FinAdd_L is itself an idempotent semiring.

Definition 4. A function $f \in \text{FinAdd}_L$ is said to be locally $*$ -closed if for each $x \in L$, either $x f^* = \top$ or there exists $N \geq 0$ such that $x f^* = x \vee \dots \vee x f^N$.

The functions id and \perp are locally $*$ -closed. As the next example demonstrates, compositions of locally $*$ -closed (and \top -continuous) functions are not necessarily locally $*$ -closed.

Example 1. Let L be the following complete lattice (the linear sum of three infinite chains):

$$\perp < x_0 < x_1 < \dots < y_0 < y_1 < \dots < z_0 < z_1 < \dots < \top$$

Since L is a chain, a function $L \rightarrow L$ is finitely additive iff it is monotone and preserves \perp .

Let $f, g : L \rightarrow L$ be the following functions. First, $\perp f = \perp g = \perp$ and $\top f = \top g = \top$. Moreover, $x_i f = y_i$, $y_i f = z_i g = \top$ and $x_i g = \perp$, $y_i g = x_{i+1}$, and $z_i g = \top$ for all i . Then f, g are monotone, $u f^* = u \vee u f \vee u f^2$ and $u g^* = u \vee u g$ for all $u \in L$. Also, f and g are \top -continuous, since if $\bigvee X = \top$ then either $\top \in X$ or $X \cap \{z_0, z_1, \dots\}$ is infinite, but then $\bigvee X f = \bigvee X g = \top$. However, $f g$ is not locally $*$ -closed, since $x_0 (f g)^* = x_0 \vee x_0 (f g) \vee x_0 (f g)^2 \dots = x_0 \vee x_1 \vee \dots = y_0$.

Lemma 4. Let $f \in \text{FinAdd}_L$ be locally $*$ -closed. Then also f^* is locally $*$ -closed. If f is additionally \top -continuous, then so is f^* .

Proof. We prove that $xf^{**} = x \vee xf^* = xf^*$ for all $x \in L$. Indeed, this is clear when $xf^* = \top$, since $f^* \leq f^{**}$. Otherwise $xf^* = \bigvee_{k \leq n} xf^k$ for some $n \geq 0$.

By finite additivity, it follows that $xf^*f^* = \bigvee_{k \leq n} xf^k f^*$. But for each k , $xf^k f^* = xf^k \vee xf^{k+1} \vee \dots \leq xf^*$, thus $xf^* = xf^*f^*$ and $xf^* = xf^{**}$. It follows that f^* is locally $*$ -closed.

Suppose now that f is additionally \top -continuous. We need to show that f^* is also \top -continuous. To this end, let $X \subseteq L$ with $\bigvee X = \top$. Since $x \leq xf^*$ for all $x \in X$, it holds that $\bigvee X f^* \geq \bigvee X = \top$. Thus $\bigvee X f^* = \top$. \square

Proposition 1. *Let S be any subsemiring of FinAdd_L closed under the $*$ -operation. If each $f \in S$ is locally $*$ -closed and \top -continuous, then S is a $*$ -continuous Kleene algebra.*

Proof. Let $g, h \in S$. By Lemma 3, it suffices to show that $g^*h = \bigvee_{n \geq 0} g^n h$. Since this is clear when $h = \perp$, assume that $h \neq \perp$. As $g^n h \leq g^*h$ for all $n \geq 0$, it holds that $\bigvee_{n \geq 0} g^n h \leq g^*h$. To prove the opposite inequality, suppose that $x \in L$. If $xg^* = \top$, then $\bigvee_{n \geq 0} xg^n = \top$, so $\bigvee_{n \geq 0} xg^n h = \top$ by \top -continuity. Thus, $xg^*h = \top = \bigvee_{n \geq 0} xg^n h$.

Suppose that $xg^* \neq \top$. Then there is $m \geq 0$ with

$$xg^*h = (x \vee \dots \vee xg^m)h = xh \vee \dots \vee xg^m h \leq \bigvee_{n \geq 0} xg^n h = x(\bigvee_{n \geq 0} g^n h).$$

The proof is complete. \square

Now define a left action of FinAdd_L on $\text{FinAdd}_{L,L'}$ by $fv = f;v$, for all $f \in \text{FinAdd}_L$ and $v \in \text{FinAdd}_{L,L'}$. It is a routine matter to check that $\text{FinAdd}_{L,L'}$, equipped with the above action, the binary supremum operation \vee and the constant \perp is an (idempotent) left FinAdd_L -semimodule, that is, $(\text{FinAdd}_L, \text{FinAdd}_{L,L'})$ is a semiring-semimodule pair.

Lemma 5. *Let $S \subseteq \text{FinAdd}_L$ be a $*$ -continuous Kleene algebra and $V \subseteq \text{FinAdd}_{L,L'}$ an S -semimodule. Then (S, V) is a generalized $*$ -continuous Kleene algebra iff for all $f \in S$ and $v \in V$, $f^*v = \bigvee_{n \geq 0} f^n v$.*

Proof. Similar to the proof of Lemma 3. \square

Proposition 2. *Let $S \subseteq \text{FinAdd}_L$ be a $*$ -continuous Kleene algebra and $V \subseteq \text{FinAdd}_{L,L'}$ an S -semimodule. If each $f \in S$ is locally $*$ -closed and \top -continuous and each $v \in V$ is \top -continuous, then (S, V) is a generalized $*$ -continuous Kleene algebra.*

Proof. Similar to the proof of Proposition 1. \square

5 *-Continuous Kleene ω -Algebras of Functions

In this section, let L be an arbitrary complete lattice and $L' = \mathbf{2}$, the two-element lattice $\{\perp, \top\}$. We define an infinite product $\text{FinAdd}_L^\omega \rightarrow \text{FinAdd}_{L, \mathbf{2}}$. Let $f_0, f_1, \dots \in \text{FinAdd}_L$ be an infinite sequence and define $v = \prod_{n \geq 0} f_n : L \rightarrow \mathbf{2}$ by

$$xv = \begin{cases} \perp & \text{if there is } n \geq 0 \text{ such that } xf_0 \cdots f_n = \perp, \\ \top & \text{otherwise} \end{cases}$$

for all $x \in L$. We will write $\prod_{n \geq k} f_n$, for $k \geq 0$, as a shorthand for $\prod_{n \geq 0} f_{n+k}$.

It is easy to see that $\prod_{n \geq 0} f_n$ is finitely additive. Indeed, $\perp \prod_{n \geq 0} f_n = \perp$ clearly holds, and for all $x \leq y \in L$, $x \prod_{n \geq 0} f_n \leq y \prod_{n \geq 0} f_n$. Thus, to prove that $(x \vee y) \prod_{n \geq 0} f_n = x \prod_{n \geq 0} f_n \vee y \prod_{n \geq 0} f_n$ for all $x, y \in L$, it suffices to show that if $x \prod_{n \geq 0} f_n = y \prod_{n \geq 0} f_n = \perp$, then $(x \vee y) \prod_{n \geq 0} f_n = \perp$. But if $x \prod_{n \geq 0} f_n = y \prod_{n \geq 0} f_n = \perp$, then there exist $m, k \geq 0$ such that $xf_0 \cdots f_m = yf_0 \cdots f_k = \perp$. Let $n = \max\{m, k\}$. We have $(x \vee y)f_0 \cdots f_n = xf_0 \cdots f_n \vee yf_0 \cdots f_n = \perp$, and thus $(x \vee y) \prod_{n \geq 0} f_n = \perp$.

It is clear that this infinite product satisfies conditions Ax1 and Ax2 in the definition of *-continuous Kleene ω -algebra. Below we show that also Ax3 and Ax4 hold.

Lemma 6. *For all $f_0, f_1, \dots, g_0, g_1, \dots \in \text{FinAdd}_L$,*

$$\prod_{n \geq 0} (f_n \vee g_n) = \bigvee_{h_n \in \{f_n, g_n\}} \prod_{n \geq 0} h_n.$$

Note that this implies Ax3.

Proof. Since infinite product is monotone, the term on the right-hand side of the equation is less than or equal to the term on the left-hand side. To prove that equality holds, let $x \in L$ and suppose that $x \prod_{n \geq 0} (f_n \vee g_n) = \top$. It suffices to show that there is a choice of the functions $h_n \in \{f_n, g_n\}$ such that $x \prod_{n \geq 0} h_n = \top$.

Consider the infinite ordered binary tree where each node at level $n \geq 0$ is the source of an edge labeled f_n and an edge labeled g_n , ordered as indicated. We can assign to each node u the composition h_u of the functions that occur as the labels of the edges along the unique path from the root to that node.

Let us mark a node u if $xh_u \neq \perp$. As $x \prod_{n \geq 0} (f_n \vee g_n) = \top$, each level contains a marked node. Moreover, whenever a node is marked and has a predecessor, its predecessor is also marked. By König's lemma [40] there is an infinite path going through marked nodes. This infinite path gives rise to the sequence h_0, h_1, \dots with $x \prod_{n \geq 0} h_n = \top$. \square

Lemma 7. *Let $f \in \text{FinAdd}_L$ and $v \in \text{FinAdd}_{L, \mathbf{2}}$ such that f is locally *-closed and v is \top -continuous. If $xf^*v = \top$, then there exists $k \geq 0$ such that $xf^k v = \top$.*

Proof. If $xf^* = \bigvee_{n=0}^N xf^n$ for some $N \geq 0$, then $xf^*v = \bigvee_{n=0}^N xf^n v = \top$ implies the claim of the lemma. If $xf^* = \top$, then \top -continuity of v implies that $\bigvee_{n \geq 0} xf^n v = \top$, which again implies the claim. \square

Lemma 8. *Let $f, g_0, g_1, \dots \in \text{FinAdd}_L$ be locally $*$ -closed and \top -continuous such that for each $m \geq 0$, $g_m \prod_{n \geq m+1} f^* g_n \in \text{FinAdd}_{L,2}$ is \top -continuous. Then*

$$\prod_{n \geq 0} f^* g_n = \bigvee_{k_0, k_1, \dots \geq 0} \prod_{n \geq 0} f^{k_n} g_n.$$

Proof. As infinite product is monotone, the term on the right-hand side of the equation is less than or equal to the term on the left-hand side. To prove that equality holds, let $x \in L$ and suppose that $x \prod_{n \geq 0} f^* g_n = \top$. We want to show that there exist integers $k_0, k_1, \dots \geq 0$ such that $x \prod_{n \geq 0} f^{k_n} g_n = \top$.

Let $x_0 = x$. By Lemma 7, $x \prod_{n \geq 0} f^* g_n = x_0 f^* g_0 \prod_{n \geq 1} f^* g_n = \top$ implies that there is $k_0 \geq 0$ for which $x_0 f^{k_0} g_0 \prod_{n \geq 1} f^* g_n = \top$. We finish the proof by induction.

Assume that we have $k_0, \dots, k_m \geq 0$ such that $x f^{k_0} g_0 \dots f^{k_m} g_m \prod_{n \geq m+1} f^* g_n = \top$ and let $x_{m+1} = x f^{k_0} g_0 \dots f^{k_m} g_m$. Then $x_{m+1} f^* g_{m+1} \prod_{n \geq m+2} f^* g_n = \top$ implies, using Lemma 7, that there exists an exponent $k_{m+1} \geq 0$ for which $x_{m+1} f^{k_{m+1}} g_{m+1} \prod_{n \geq m+2} f^* g_n = \top$. \square

Proposition 3. *Let $S \subseteq \text{FinAdd}_L$ and $V \subseteq \text{FinAdd}_{L,2}$ such that (S, V) is a generalized $*$ -continuous Kleene algebra of locally $*$ -closed and \top -continuous functions $L \rightarrow L$ and \top -continuous functions $L \rightarrow \mathbf{2}$. If $\prod_{n \geq 0} f_n \in V$ for all sequences f_0, f_1, \dots of functions in S , then (S, V) is a $*$ -continuous Kleene ω -algebra.*

Proof. This is clear from Lemmas 6 and 8. \square

We finish the section by a lemma which exhibits a condition on the lattice L which ensures that infinite products of locally $*$ -closed and \top -continuous functions are again \top -continuous.

Lemma 9. *Assume that L has the property that whenever $\bigvee X = \top$ for some $X \subseteq L$, then for all $x < \top$ in L there is $y \in X$ with $x \leq y$. If $f_0, f_1, \dots \in \text{FinAdd}_L$ is a sequence of locally $*$ -closed and \top -continuous functions, then $\prod_{n \geq 0} f_n \in \text{FinAdd}_{L,2}$ is \top -continuous.*

Proof. Let $v = \prod_{n \geq 0} f_n$. We already know that v is finitely additive. We need to show that v is \top -continuous. But if $v \neq \perp$, then there is some $x < \top$ with $xv = \top$, i.e., such that $xf_0 \dots f_n > \perp$ for all n . By assumption, there is some $y \in X$ with $x \leq y$. It follows that $yf_0 \dots f_n \geq xf_0 \dots f_n > \perp$ for all n and thus $\bigvee Xv = \top$. \square

6 State Reachability, Coverability and Büchi Acceptance in Energy Automata

We now show how the setting developed in the last sections can be applied to solve the energy problems of Section 2. Recall that $L = [0, \infty]_{\perp}$ denotes the complete lattice of nonnegative real numbers together with ∞ and an extra bottom element \perp , and that \mathcal{E} denotes the idempotent semiring of energy functions $L \rightarrow L$. Note that L satisfies the precondition of Lemma 9.

Lemma 10. *Energy functions are finitely additive and \top -continuous, hence $\mathcal{E} \subseteq \text{FinAdd}_L$.*

Proof. Finite additivity follows from monotonicity. For \top -continuity, let $X \subseteq L$ such that $\bigvee X = \infty$ and $f \in \mathcal{E}$, $f \neq \perp$. By $\bigvee X = \infty$, we know that for every $n \in \mathbb{N}$, there exists $x_n \in X$ with $x_n \geq n$. Choose such a sequence (x_n) and let $y_n = x_n f$ for all n .

If $y_n = \perp$ for all $n \geq 0$, then also $n f = \perp$ for all $n \geq 0$ (as $x_n \geq n$), hence $f = \perp$. We must thus have an index N for which $y_N > \perp$. But then $y_{N+k} \geq y_N + k \geq k$ for all $k \geq 0$, hence $\bigvee X f = \infty$. \square

Lemma 11. *For $f \in \mathcal{E}$, f^* is given by $x f^* = x$ if $x f \leq x$ and $x f^* = \infty$ if $x f > x$. Hence f is locally $*$ -closed and $f^* \in \mathcal{E}$.*

Proof. We have $\perp f^* = \perp$ and $\infty f^* = \infty$. Let $x \neq \perp, \infty$. If $x f \leq x$, then $x f^n \leq x$ for all $n \geq 0$, so that $x \leq \bigvee_{n \geq 0} x f^n \leq x$, whence $x f^* = x$. If $x f > x$, then let $a = x f - x > 0$. We have $x f \geq x + a$, hence by $(*)$, $x f^n \geq x + n a$ for all $n \geq 0$, so that $x f^* = \bigvee_{n \geq 0} x f^n = \infty$. \square

Not all locally $*$ -closed functions $f : L \rightarrow L$ are energy functions: the function f defined by $x f = 1$ for $x < 1$ and $x f = x$ for $x \geq 1$ is locally $*$ -closed, but $f \notin \mathcal{E}$.

Corollary 1. *\mathcal{E} is a $*$ -continuous Kleene algebra.*

Proof. This is clear by Proposition 1. \square

Remark 1. It is *not* true that \mathcal{E} is a *continuous* Kleene algebra: Let $f_n, g \in \mathcal{E}$ be defined by $x f_n = x + 1 - \frac{1}{n+1}$ for $x \geq 0$, $n \geq 0$ and $x g = x$ for $x \geq 1$, $x g = \perp$ for $x < 1$. Then $0(\bigvee_{n \geq 0} f_n)g = (\bigvee_{n \geq 0} 0 f_n)g = 1g = 1$, whereas $0 \bigvee_{n \geq 0} (f_n g) = \bigvee_{n \geq 0} (0 f_n g) = \bigvee_{n \geq 0} ((1 - \frac{1}{n+1})g) = \perp$.

Lemma 12. *For any $g \in \mathcal{E}$, there exists $f \in \mathcal{E}$ such that $g = f^*$ iff there is $k \in [0, \infty]_{\perp}$ such that $x g = x$ for all $x < k$, $x g = \infty$ for all $x > k$, and $k g = k$ or $k g = \infty$.*

Proof. We first note that if $g \in \mathcal{E}$ is such that there is k for which $x g = x$ for $x < k$ and $x g = \infty$ for $x > k$, then $x g^* = x g$ for $x \neq k$, and if $k g = k$ or $k g = \infty$, then also $k g^* = k g$.

Now let $g \in \mathcal{E}$. If there is $f \in \mathcal{E}$ with $g = f^*$, then we set $k = \sup\{x \mid x f \leq x\}$. Then $x f > x$ and hence $x g = \infty$ for all $x > k$, and whenever $x < k$, then there is y with $x \leq y \leq k$ and $y f \leq y$, hence by $(*)$, $x f \leq x$, so that $x g = x$. If $k f \leq k$, then $k g = k$, otherwise $k g = \infty$ as claimed. \square

Let \mathcal{V} denote the \mathcal{E} -semimodule of all \top -continuous functions $L \rightarrow \mathbf{2}$. For $f_0, f_1, \dots \in \mathcal{E}$, define the infinite product $f = \prod_{n \geq 0} f_n : L \rightarrow \mathbf{2}$ by $x f = \perp$ if there is an index n for which $x f_0 \cdots f_n = \perp$ and $x f = \top$ otherwise, like in Section 5. By Lemma 9, $\prod_{n \geq 0} f_n$ is \top -continuous, i.e., $\prod_{n \geq 0} f_n \in \mathcal{V}$.

By Proposition 2, $(\mathcal{E}, \mathcal{V})$ is a generalized $*$ -continuous Kleene algebra.

Corollary 2. $(\mathcal{E}, \mathcal{V})$ is a $*$ -continuous Kleene ω -algebra.

Proof. This is clear by Proposition 3. □

Remark 2. As \mathcal{E} is not a continuous Kleene algebra, it also holds that $(\mathcal{E}, \mathcal{V})$ is not a continuous Kleene ω -algebra; in fact it is clear that there is no \mathcal{E} -semimodule \mathcal{V}' for which $(\mathcal{E}, \mathcal{V}')$ would be a continuous Kleene ω -algebra. The initial motivation for the work in [25, 26, 28] and the present paper was to generalize the theory of continuous Kleene ω -algebras so that it would be applicable to energy functions.

Lemma 13. For $f \in \mathcal{E}$, f^ω is given by $\perp f^\omega = \perp$, and for $x \neq \perp$, $xf^\omega = \perp$ if $xf < x$ and $xf^\omega = \top$ if $xf \geq x$.

Proof. The claim that $\perp f^\omega = \perp$ is clear, and so is the lemma for $f = \perp$. For $f \neq \perp$ and $x = \infty$, $xf^n = \infty$ for all $n \geq 0$, hence $\infty f^\omega = \top$. Now let $x \neq \perp, \infty$. If $xf \geq x$, then $xf^n \geq x$ for all $n \geq 0$, hence $xf^\omega = \top$. If $xf < x$, then let $a = x - xf > 0$. We have $xf \leq x - a$, hence by (*), $xf^n \leq x - na$ for all $n \geq 0$, so that there is $N \geq 0$ for which $xf^N = \perp$, whence $xf^\omega = \perp$. □

We can now solve the state reachability, coverability, and Büchi problems for energy automata. We say that $\mathcal{E}' \subseteq \mathcal{E}$ is *fixed-point decidable* if it is decidable, for any $f \in \mathcal{E}'$ and $x \in L$, whether $xf < x$, $xf = x$ or $xf > x$.

Theorem 2. Let $A = (\alpha, M, k)$ be an energy automaton of dimension $n \geq 1$, $x_0 \in \mathbb{R}_{\geq 0}$, and $z : \{1, \dots, k\} \rightarrow \mathbb{R}_{\geq 0}$. Then

- $\text{Reach}_{\mathcal{E}'}(A, x_0) = \mathbf{tt}$ iff $x_0|A| \neq \perp$;
- $\text{Cover}_{\mathcal{E}'}(A, x_0, z) = \mathbf{tt}$ iff there exists $i \leq k$ such that $(x_0\alpha M^*)_i \geq iz$;
- $\text{Büchi}_{\mathcal{E}'}(A, x_0) = \mathbf{tt}$ iff $x_0\|A\| = \top$.

Proof. For state reachability and Büchi acceptance the claims are clear. For coverability, we note that

$$(x_0\alpha M^*)_i = \bigvee_{m \geq 0} \bigvee_{1 \leq k_1, \dots, k_m \leq n} x_0\alpha_{k_1} M_{k_1, k_2} \cdots M_{k_m, i},$$

and the claim follows. □

Corollary 3. For fixed-point decidable subalgebras $\mathcal{E}' \subseteq \mathcal{E}$, Problems 1, 2, and 3 are decidable. For an energy automaton of dimension n , the decision procedures use $O(n^3)$, $O(n^3)$, respectively $O(n^4)$, algebra operations.

Proof. If \mathcal{E}' is fixed-point decidable, then Lemmas 11 and 13 imply that the $*$ and ω operations are computable in \mathcal{E}' , and the matrix operations in Theorem 2 can be reduced to compositions, binary suprema, and these two operations. The complexity results follow from the fact that computation of M^* uses $O(n^3)$ operations and computation of M^ω uses $O(n^4)$ operations, cf. [30]. □

7 Some Complexity Results

We proceed to identify two important subclasses of computable energy functions, which cover most of the related work mentioned in the introduction, and to give complexity results on their reachability and Büchi acceptance problems.

The *integer update functions* in \mathcal{E} are the functions f_k , for $k \in \mathbb{Z}$, given by

$$xf_k = \begin{cases} x + k & \text{if } x \geq \max(0, -k), \\ \perp & \text{otherwise,} \end{cases}$$

together with $f_{-\infty} := \perp$ and $f_{\infty} := \top$. These are the update functions usually considered in integer-weighted automata and VASS [11, 14, 17–20, 33]. We have $f_k f_\ell = f_{k+\ell}$, $f_k \vee f_\ell = f_{\max(k,\ell)}$, and

$$f_k^* = \begin{cases} f_0 & \text{for } k \leq 0, \\ f_{\infty} & \text{for } k > 0, \end{cases} \quad f_k^\omega = \begin{cases} f_{-\infty} & \text{for } k < 0, \\ f_{\infty} & \text{for } k \geq 0, \end{cases}$$

whence the class \mathcal{E}_{int} of integer update functions forms a subalgebra of \mathcal{E} . A function $f_k \in \mathcal{E}_{\text{int}}$ can be represented by the (extended) integer k , and algebra operations can then be performed in constant time. Also, \mathcal{E}_{int} is trivially fixed-point decidable, so that Corollary 3 implies the following result.

Theorem 3. *For \mathcal{E}_{int} -automata, Problems 1, 2 and 3 are decidable in PTIME.*

Remark 3. This means that state reachability, coverability and Büchi acceptance for one-dimensional VASS are decidable in PTIME, which seems not to have been noted before. (But see the recent [38], where reachability for one-dimensional *branching VASS* is shown decidable in PTIME. In [7] it is claimed that coverability for one-dimensional VASS is NP-complete.)

Next we turn our attention to piecewise affine functions.

Definition 5. *A function $f \in \mathcal{E}$ is said to be (rational) piecewise affine if there exist $0 \leq x_0 < x_1 < \dots < x_k \in \mathbb{Q} \cup \{\infty\}$ such that*

- $xf = \perp$ for $x < x_0$ and $xf = \infty$ for $x > x_k$,
- $x_j f \in \mathbb{Q} \cup \{\perp, \infty\}$ for all j , and
- all restrictions $f|_{]x_j, x_{j+1}[}$ are affine functions $x \mapsto a_j x + b_j$ with $a_j, b_j \in \mathbb{Q}$, $a_j \geq 1$.

Let $\mathcal{E}_{\text{pw}} \subseteq \mathcal{E}$ denote the class of piecewise affine energy functions. The notion of *integer piecewise affine functions*, \mathcal{E}_{pwi} , is defined similarly, with all occurrences of \mathbb{Q} above replaced by \mathbb{Z} . Clearly $\mathcal{E}_{\text{int}} \subseteq \mathcal{E}_{\text{pwi}} \subseteq \mathcal{E}_{\text{pw}}$.

Note that the definition does not make any assertion about continuity at the x_j , but (*) implies that $\lim_{x \nearrow x_j} xf \leq x_j f \leq \lim_{x \searrow x_j} xf$. A piecewise affine function as above can be represented by its break points x_0, \dots, x_k , the values $x_0 f, \dots, x_k f$,

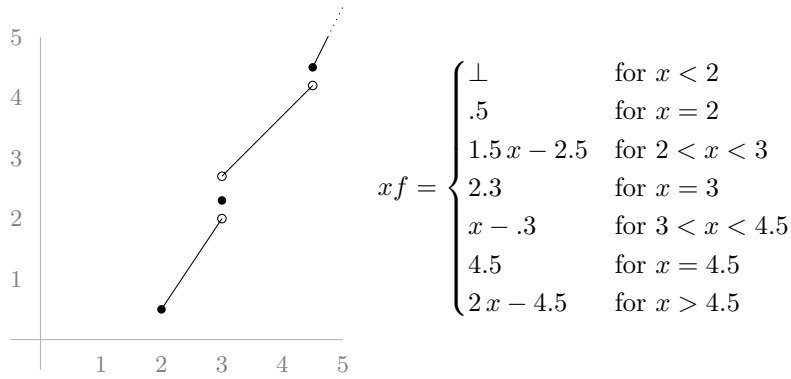


Figure 2: A piecewise affine energy function

and the numbers $a_0, b_0, \dots, a_k, b_k$. These functions arise in the reduction used in [10] to show decidability of energy problems for one-clock timed automata with transition updates, see Section 8. Fig. 2 shows an example of a piecewise affine energy function.

The class of piecewise affine energy functions forms a subsemiring of \mathcal{E} : if $f, g \in \mathcal{E}_{pw}$ with break points x_0, \dots, x_k and y_0, \dots, y_ℓ , respectively, then $f \vee g$ is piecewise affine with break points obtained from the break points of f and g together with intersection points of lines (which are rational), and fg is piecewise affine with break points a subset of $\{x_0, \dots, x_k, y_0f^{-1}, \dots, y_\ell f^{-1}\}$ (which are all rational). Hence maxima and compositions of piecewise affine energy functions are computable, but may increase the size of their representation.

Now let, for any $p \in \mathbb{Q}$ with $p \geq 0$, $g_p^-, g_p^+ \in \mathcal{E}_{pw}$ be the functions defined by

$$xg_p^- = \begin{cases} x & \text{for } x < p, \\ \infty & \text{for } x \geq p, \end{cases} \quad xg_p^+ = \begin{cases} x & \text{for } x \leq p, \\ \infty & \text{for } x > p. \end{cases}$$

Proposition 4. \mathcal{E}_{pw} is a $*$ -continuous Kleene algebra.

Proof. In lieu of Proposition 1, we need to show that \mathcal{E}_{pw} is closed under the $*$ -operation. Let $f \in \mathcal{E}_{pw}$, then by Lemma 12, there is a $p \in \mathbb{Q}$ such that $f^* = g_p^-$ or $f^* = g_p^+$. \square

Remark that, unlike \mathcal{E}_{pw} , the class \mathcal{E}_{pwi} of integer piecewise affine functions does not form a subsemiring of \mathcal{E} , as composites of \mathcal{E}_{pwi} -functions are not necessarily integer piecewise affine. As an example, for the functions $f, g \in \mathcal{E}_{pwi}$ given by

$$xf = 2x, \quad xg = \begin{cases} x + 1 & \text{for } x < 3, \\ x + 2 & \text{for } x \geq 3, \end{cases}$$

we have

$$xfg = \begin{cases} 2x + 1 & \text{for } x < 1.5, \\ 2x + 2 & \text{for } x \geq 1.5, \end{cases}$$

which is not integer piecewise affine. The semiring generated by \mathcal{E}_{pwi} is the sub-semiring of \mathcal{E}_{pw} of functions with *rational* break points x_0, \dots, x_k , but *integer* values $a_0, b_0, \dots, a_k, b_k$.

Similarly, the class of rational *affine* functions $x \mapsto ax + b$, $a, b \in \mathbb{Q}$, $a \geq 1$ (without break points) is not closed under maximum, and it can be seen that \mathcal{E}_{pw} is the semiring generated by rational affine functions.

Lemma 14. \mathcal{E}_{pw} is fixed-point decidable.

Proof. Let $f \in \mathcal{E}_{\text{pw}}$, with representation $(x_0, \dots, x_k, x_0f, \dots, x_kf, a_0, \dots, a_k, b_0, \dots, b_k)$. Let $x \in \mathbb{R}_{\geq 0}$ be computable; we need to decide whether $xf < x$, $xf = x$ or $xf > x$. If $x < x_0$, then $xf = \perp$. If $x = x_j$ for some j , we can simply compare x_j with x_jf .

Assume now that $x \in]x_j, x_{j+1}[$ for some j . If $a_jx_j + b_j < x_j$ and $a_jx_{j+1} + b_j \leq x_{j+1}$, then $xf < x$ by (*). If $a_jx_j + b_j = x_j$ and $a_jx_{j+1} + b_j = x_{j+1}$, then also $xf = x$, and if $a_jx_j + b_j \geq x_j$ and $a_jx_{j+1} + b_j > x_{j+1}$, then $xf > x$. The cases $a_jx_j + b_j > x_j$, $a_jx_{j+1} + b_j \leq x_{j+1}$ and $a_jx_j + b_j \geq x_j$, $a_jx_{j+1} + b_j < x_{j+1}$ cannot occur because of (*).

The last case to consider is $a_jx_j + b_j < x_j$ and $a_jx_{j+1} + b_j > x_{j+1}$. Then we must have $a_j > 1$, and then $xf < x$ if $x < \frac{b_j}{1-a_j}$, $xf = x$ if $x = \frac{b_j}{1-a_j}$, and $xf > x$ if $x > \frac{b_j}{1-a_j}$. \square

Theorem 4. For \mathcal{E}_{pw} -automata, Problems 1, 2 and 3 are decidable in EXPTIME.

Proof. Decidability follows from Corollary 3 and Lemma 14. For the complexity claim, we note that all algebra operations in \mathcal{E}_{pw} can be performed in time linear in the size of the representations of the involved functions. However, the maximum and composition operations may triple the size of the representations, hence our procedure may take time $O(3^{n^3}p)$ for state reachability and coverability, and $O(3^{n^4}p)$ for Büchi acceptance, for an \mathcal{E}_{pw} -automaton of dimension n and energy functions of representation length at most p . \square

We believe that the above complexity bound of EXPTIME can be considerably sharpened, but we leave this for future work.

8 Reduction from Weighted Timed Automata

To further motivate the introduction of our notion of energy automata, we review here how the treatment of lower-bound energy problems for one-clock weighted timed automata in [10, 11] naturally leads to our energy functions and energy automata. In this section, and this section only, function application and composition will be written in the standard right-to-left order.

A *weighted timed automaton* $A = (L, l_0, C, I, E, r)$ consists of a finite set of *locations* L with initial location l_0 , a finite set of *clocks* C , location *invariants*

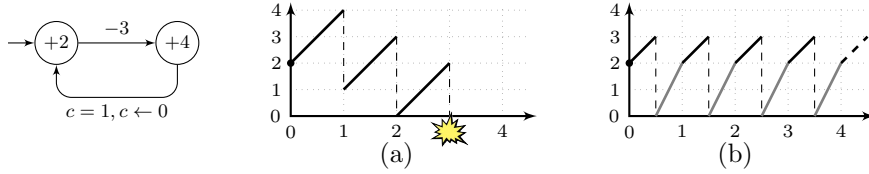


Figure 3: One-clock weighted timed automaton with discrete updates. Any region-stable scheduler (*i.e.*, with switches at *integer* times) is doomed (a), but there exists a feasible schedule with switches at half-integer times (b).

$I : L \rightarrow \Phi(C)$, weighted edges $E \subseteq L \times \Phi(C) \times 2^C \times \mathbb{Z} \times L$ and location weight rates $r : L \rightarrow \mathbb{Z}$. Here the set $\Phi(C)$ of clock constraints ϕ is defined by the grammar

$$\phi ::= c \bowtie k \mid \phi_1 \wedge \phi_2 \quad (c \in C, k \in \mathbb{Z}, \bowtie \in \{\leq, <, \geq, >, =\}).$$

A *clock valuation* is a mapping $C \rightarrow \mathbb{R}_{\geq 0}$. For a clock valuation $v : C \rightarrow \mathbb{R}_{\geq 0}$ and a clock constraint $\phi \in \Phi(C)$, we write $v \models \phi$ to indicate that v satisfies ϕ . We denote by $v_0 : C \rightarrow \mathbb{R}_{\geq 0}$ the clock valuation given by $v_0(c) = 0$ for all $c \in C$. For a clock valuation $v : C \rightarrow \mathbb{R}_{\geq 0}$, $d \in \mathbb{R}_{\geq 0}$, and $R \subseteq C$, we denote by $v + d$ and $v[R \leftarrow 0]$ the clock valuations given by $(v + d)(c) = v(c) + d$ for all $c \in C$ and $v[R \leftarrow 0](c) = 0$ for $c \in R$, $v[R \leftarrow 0](c) = v(c)$ for $c \notin R$.

The *semantics* of a weighted timed automaton $A = (L, l_0, C, I, E, r)$ is given by an infinite weighted automaton $\llbracket A \rrbracket = (S_A, s_0, T_A)$ with states $S_A = \{(l, v) \mid v \models I(l)\} \subseteq L \times \mathbb{R}_{\geq 0}^C$, initial state $s_0 = (l_0, v_0)$, and transitions $T_A \subseteq S_A \times \mathbb{R} \times S_A$ of two types:

- *delay transitions* $(l, v) \xrightarrow{r(l)d} (l, v + d)$ for all $d \in \mathbb{R}_{\geq 0}$ such that $v + d' \models I(l)$ for all $d' \in [0, d]$;
- *switch transitions* $(l, v) \xrightarrow{p} (l', v')$, where $e = (l, \phi, R, p, l') \in E$ is a transition of A , $v \models \phi$ and $v' = v[R \leftarrow 0]$.

We refer to [34] for a thorough survey on timed automata and weighted timed automata.

The *lower-bound energy problem* for a weighted timed automaton A as above is, given an initial energy $x_0 \in \mathbb{R}_{\geq 0}$, to decide whether there exist an infinite path

$$(l_0, v_0) \xrightarrow{p_0} (l_1, v_1) \xrightarrow{p_1} (l_2, v_2) \xrightarrow{p_2} \dots$$

of delay and switch transitions in $\llbracket A \rrbracket$ for which $x_0 + \sum_{i=0}^n p_i \geq 0$ for all $n \in \mathbb{N}$. We hence want to decide whether there is a run in A where the *accumulated energy* $x_0 + \sum_{i=0}^n p_i$ never drops below zero. We shall say that such a run is *feasible*. Figure 1 in the introduction shows an example of such an energy problem.

For *one-clock* weighted timed automata *without discrete updates*, *i.e.*, with $C = \{c\}$ a singleton and $p = 0$ for all $(l, \phi, R, p, l') \in E$, it was shown in [11] that this problem can be decided via a simple reduction to a refinement of the region

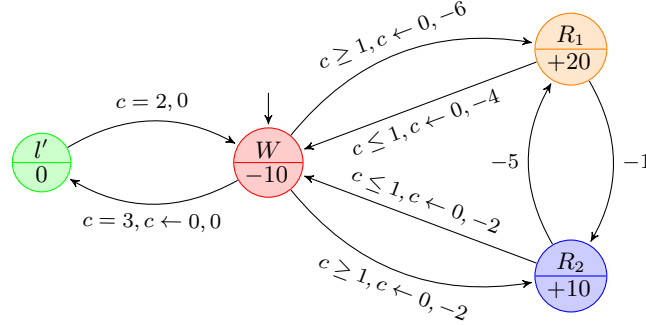


Figure 4: Conversion of the weighted timed automaton of Fig. 1 to a 3-bounded weighted timed automaton. To simplify the example, we assume that the invariant of R_1 and R_2 in the automaton of Fig. 1 (and thus here) is $c \leq 1$. The invariant of location W is $c \leq 3$, and the invariant of the new location l' is $c \leq 2$.

graph [2] of A . Figure 3, taken from [10], however, shows that a similar region-stable reduction is not available for one-clock weighted timed automata *with* discrete updates. In the rest of this section we review the substantially more complicated reduction from [10]. For simplicity of presentation we assume the input timed automaton to be *closed*, *i.e.*, only using closed clock constraints $c \leq k$, $c \geq k$, $c = k$ and their conjunctions.

Let $A = (L, l_0, \{c\}, I, E, r)$ be a closed one-clock weighted timed automaton. First, we make sure that A is *bounded*, *i.e.*, that the value of c never exceeds a constant M during any run of A . The construction, based on [5], works essentially by resetting the clock whenever it reaches value M . Formally, let m be the greatest integer constant which appears in any invariant $I(l)$, $l \in L$ and any constraint ϕ in $(l, \phi, R, p, l') \in E$ and set $M = m + 2$. Now for each location $l \in L$, add an invariant $c \leq M$ to $I(l)$ and a new location l' to L , with $I(l') = (c \leq M - 1)$ and $r(l') = 0$, and edges $(l, (c = M), \{c\}, 0, l')$, $(l', (c = M - 1), \emptyset, 0, l)$. Figure 4 shows an example of the construction for the weighted timed automaton in Fig. 1 for $M = 3$. It can be shown [5, 10] that the so-constructed bounded automaton admits the same infinite lower-bounded runs for the same initial energies as the old one.

Second, we make sure that the bound $M = 1$. This is done, like in [13], by splitting A into stages, one for every integer $k \in \{0, \dots, M - 1\}$; the intuition is that a state $((l, k), \nu)$ at stage k corresponds to a state (l, ν') with old clock value $\nu' = k + \nu$. Figure 5 shows an example of the construction. Third, we also eliminate edges $(l, (c' \leq 1), \{c'\}, p, l')$ by introducing new locations l'' with $r(l'') = 0$, $I(l'') = c' \leq 1$, and edges $(l, (c' \leq 1), \emptyset, 0, l'')$, $(l'', (c' = 1), \{c'\}, p, l')$, see Figure 6. It can again be shown [10] that this construction does not affect energy properties.

Next, noticing that A now only has three types of edges: reset-free edges with constraint $c' \leq 1$ and resetting edges with constraint $c' = 0$ or $c' = 1$, we split the

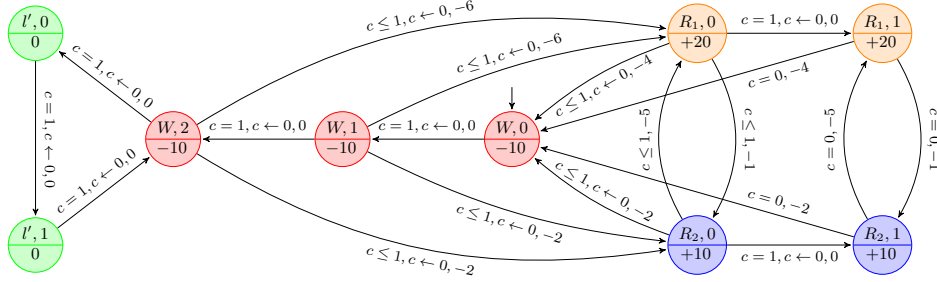


Figure 5: Conversion of the 3-bounded weighted timed automaton of Fig. 4 to a 1-bounded weighted timed automaton.

locations of A so that each location l either has only incoming reset-free edges or only incoming resetting edges. Possibly adding a new initial location, we also make sure that l_0 has no incoming edges. Fig. 7 shows the complete conversion of the weighted timed automaton of Fig. 1.

Let S be the set of locations without incoming reset-free edges, then $l_0 \in S$. For each pair $l, l' \in S$, let $P(l, l')$ be the (finite) set of paths in A from l to l' which

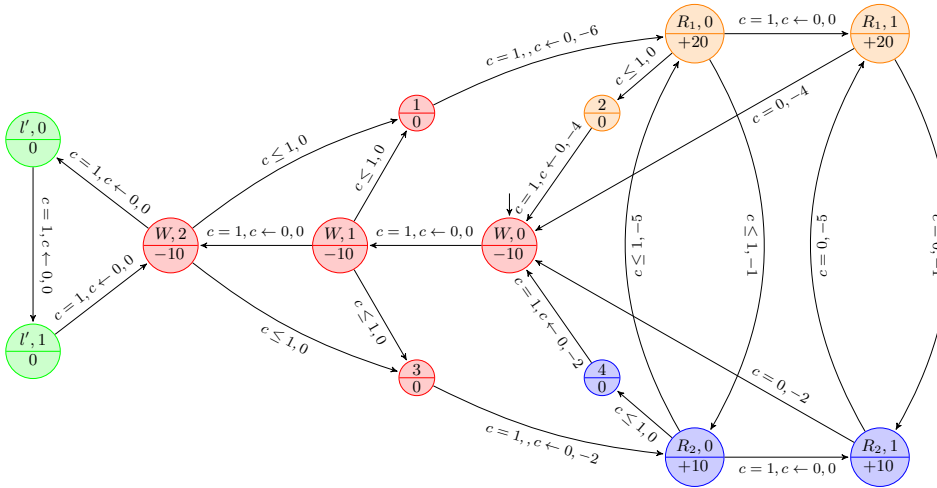


Figure 6: The 1-bounded weighted timed automaton obtained by eliminating resetting edges with clock constraint $c \leq 1$ from the 1-bounded weighted timed automaton in Fig. 5. We eliminate, for instance, the edge $(\langle W, 2 \rangle, (c \leq 1), \{c\}, -6, \langle R_1, 0 \rangle)$ by introducing an auxiliary location 1 with invariant $c \leq 1$ and rate 0, and edges $(\langle W, 2 \rangle, (c \leq 1), \emptyset, 0, 1)$ and $(1, c = 1, \{c\}, -6, \langle R_1, 0 \rangle)$. Note that in the resulting automaton all edges with constraint $c \leq 1$ are reset-free, and all resetting edges have constraint $c = 1$ or $c = 0$.

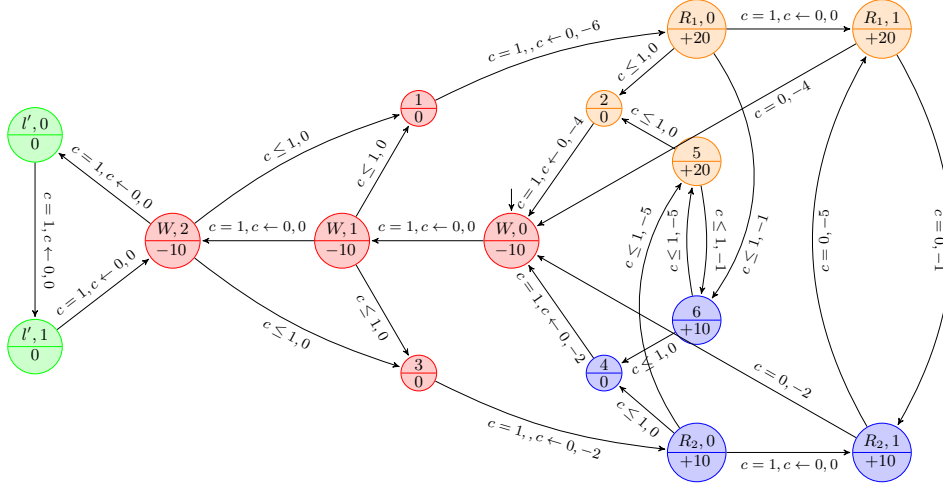


Figure 7: Conversion of the weighted timed automaton in Fig. 6 to one with locations partitioned into locations with only reset-free incoming edges with constraint $c \leq 1$ and locations with only resetting edges and constraint $c = 1$ or $c = 0$ (we omit resets at edges with constraint $c = 0$). Note that in Fig. 6 locations $(R_1, 0)$ and $(R_2, 0)$ have both resetting and reset-free incoming edges. We introduce two auxiliary locations 5 and 6 and redirect edges accordingly.

only go through locations in $L \setminus S$, and which contain at most two copies of any simple cycle. (It is shown in [10] that paths which contain more than two copies of a simple cycle can be reduced to paths in $P(l, l')$ by collecting all delays in the first two copies.) For each such path $\pi \in P(l, l')$, let $f_\pi : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ be the function mapping input energy in l to maximal achievable output energy in l' and define $T = \{(l, f_\pi, l') \mid \pi \in P(l, l')\}$. Let $B = (S, l_0, T, S)$; we will show below that B is an energy automaton.

It is clear that any infinite run in A which traverses resetting edges infinitely often will translate to an infinite path in B . We need, however, to take special care of infinite runs in A which are *eventually reset-free*. To do so, it is shown in [10] that we can compute, in EXPTIME, a mapping $z : L \rightarrow \mathbb{R}_{\geq 0}$ such that there is a *feasible* infinite reset-free run from a state (l, v) in $\llbracket A \rrbracket$ iff $v \geq z(l)$. For the proof of this, one observes that any such run can be converted, by eliminating all delays in locations with nonpositive rates and collecting all delays in locations with positive rates in their first occurrences, into one where, after a finite prefix, no more time elapses. Also, the length of these prefixes is at most $|L|(|L| + 1)$, so $z(l)$ can be computed in finite time.

It can now be shown [10, Lemma 17] that, for any $x_0 \in \mathbb{R}_{\geq 0}$, there is a feasible infinite run from (l_0, x_0) in A iff (1) there is an accepting infinite run in B with initial energy x_0 , or (2) a state $l \in S$ is reachable in B , with initial energy x_0 , such that the energy in l is at least $z(l)$. Hence the lower-bound energy problem for A

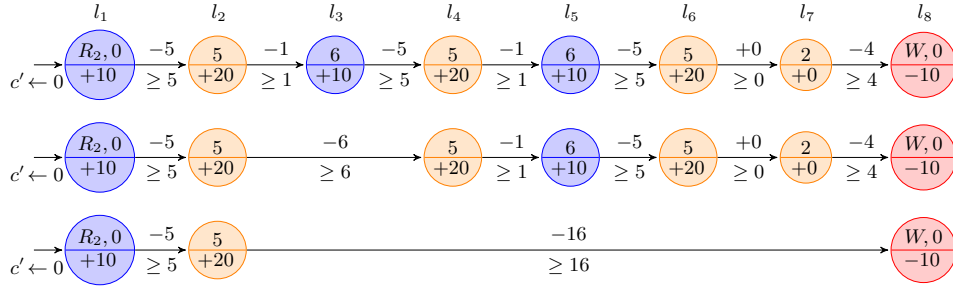


Figure 8: Reset-free path with annotations (top) and after repeated application of the first normalization operation.

from x_0 reduces to $\text{Büchi}(B)(x_0)$ and $\text{Cover}(B)(x_0, z)$.

We miss to show that $B = (S, l_0, T, S)$ is an energy automaton. Let $l, l' \in S$ and $\pi = (l = l_1, e_1, l_2, \dots, e_n, l_{n+1} = l') \in P(l, l')$; note that e_n is the only resetting edge. If the constraint on e_n is $c' = 0$, then no time elapses during π , and, letting p_i denote the weight of $e_i = (l_i, \phi, \emptyset, p_i, l_{i+1})$ (where $\phi = (c' \leq 1)$), we have

$$f_\pi(x) = \begin{cases} \text{undefined} & \text{if } x + \sum_{i=1}^k p_i < 0 \text{ for some } k \in \{1, \dots, n\}, \\ x + \sum_{i=1}^n p_i & \text{otherwise.} \end{cases}$$

If, on the other hand, the last constraint in π is $\phi_n = (c' = 1)$, then we face the task of distributing one time unit of delay optimally through the locations along π . In order to do so, we first annotate the edges along π with *lower-bound constraints*. Hence each e_i is now of the form $e_i = (l_i, \phi, \emptyset, p_i, b_i, l_{i+1})$, with $b_i = -p_i$ initially and the semantics that the edge e_i is enabled for input energy $x \geq b_i$.

We modify π by removing locations in which an optimal path (*i.e.*, with maximal energy output) will not delay. To ease the presentation, we assume that the maximal rate along π is positive, *i.e.*, $\max\{r(l_i) \mid i = 1, \dots, n\} > 0$. The constructions are similar in the other case; see [10] for details. Figure 8 shows as an example a path from $(R_2, 0)$ to $(W, 0)$ in the weighted timed automaton of Fig. 7.

First we note that if $r(l_i) \geq r(l_{i+1})$ for some $i \in \{1, \dots, n-1\}$, then any delay spent in l_{i+1} could just as well (or better) have been spent in l_i . (This is the case for $i = 2$ (location 6 with rate +10) in the example.) Hence we can remove l_{i+1} and update π with an edge $(l_i, \phi, \emptyset, p_i + p_{i+1}, \max(b_i, b_{i+1} - p_i), l_{i+2})$. The new lower-bound constraint $\max(b_i, b_{i+1} - p_i)$ is chosen so that the new edge can be taken precisely when the sequence of the old edges could be taken without intermediate delay.

This modified path π has the property that $r(l_i) < r(l_j)$ for all $1 \leq i < j \leq n$. Next, we see that if there is $i \in \{1, \dots, n-1\}$ for which $b_i + p_i \geq b_{i+1}$, then there is no need to spend any delay in l_{i+1} , as we can go directly to l_{i+2} which has a higher rate. (In our example, this case does not occur.) Hence we modify π once again, removing l_{i+1} and adding a new edge $(l_i, \phi, \emptyset, p_i + p_{i+1}, b_i, l_{i+2})$. The result

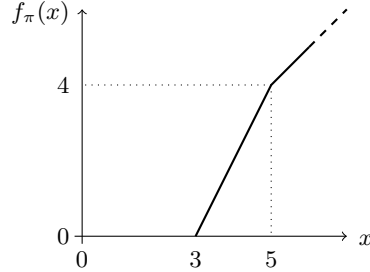


Figure 9: The energy function associated with the path of Fig. 8.

of these two kinds of modifications is a reset-free path in so-called *normal form*, see again Fig. 8.

As the last step, we show by example how to compute the energy function of a path in normal form; we refer to [10] for the general algorithm. Let π be the path at the bottom of Fig. 8; we want to compute the partial function $f_\pi : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ which maps the input energy x , entering the first location of π , to the maximum available output energy $f_\pi(x)$ when leaving π .

First we notice that if $x < 3$, then we need to spend a delay of $d_1 \geq \frac{5-x}{10} > \frac{1}{5}$ in l_1 to meet its output constraint of $x \geq 5$. The energy value when entering l_2 is then $x + 10d_1 - 5$, which is equal to 0, so that after l_2 , the value is $20d_2 \leq 20(1 - d_1) = 20 - 20d_1 < 16$. Hence we cannot match the output constraint $x \geq 16$ on l_2 , so that $f_\pi(x)$ is undefined for $x < 3$. On the other hand, when $x = 3$ at the start of π , then we can delay $\frac{1}{5}$ time units in l_1 , then delay $\frac{4}{5}$ time units in l_2 , and finally achieve $f_\pi(3) = 0$.

It can be shown [10] that the general strategy for maximizing the output energy, given a path in normal form such as the one in the bottom of Fig. 8, is to delay in every location precisely the time necessary for meeting its output constraint, and then to spend any remaining time in the last location. Hence when x is between 3 and 5, we can let $d_1 = \frac{5-x}{10}$ and then $d_2 = 1 - d_1$. This gives a value of $f_\pi(4) = 4 + \frac{1}{10} \cdot 10 - 5 + \frac{9}{10} \cdot 20 - 16 = 2$. For $x \geq 5$, we need not delay in l_1 at all, so in this case, $f_\pi(x) = x - 1$. See Fig. 9 for a graph of the function thus obtained.

Given this general strategy for maximizing output energy, it can be shown that the energy function f_π associated with a path in normal form, or indeed with a general reset-free path π , is a *continuous piecewise affine* function which satisfies (*), cf. Definition 5. Now when the input timed automaton is not closed, then the definition interval of f_π ($[1, \infty[$ in the example) may be left-open, and taking maxima of such functions may introduce discontinuities, so that in the end, $B = (S, l_0, T, S)$ is an automaton which has transition weights from the general class \mathcal{E}_{pw} of piecewise-affine energy functions.

The results of Section 7 thus apply and allow us to compute $\text{Büchi}(B)(x_0)$ and $\text{Cover}(B)(x_0, z)$ in exponential time. As the reduction from A to B shown here may incur an exponential blow-up, our overall procedure for solving lower-bound

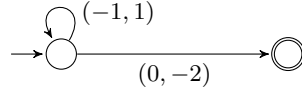


Figure 10: A simple two-dimensional VASS

energy problems in weighted timed automata has double exponential complexity.

9 Multi-dimensional Energy Automata and Games

We turn our attention to several variants of energy automata. We will first be concerned with multi-dimensional energy automata and show that their coverability problem is EXPSPACE-complete. Then we will show that this does not apply to *flat* energy functions, which are not required to satisfy (*); for such functions, coverability is undecidable from dimension four. Finally, we show that reachability games on two-dimensional energy automata and on one-dimensional flat energy automata are undecidable.

An *n*-dimensional energy automaton, or \mathcal{E}^n -automaton for short, (S, T) , for $n \geq 1$, consists of finite sets S of states and $T \subseteq S \times \mathcal{E}^n \times S$ of transitions. By restricting transition labelings, we can define subclasses of \mathcal{E}^n -automata, \mathcal{E}_{pw}^n -automata, and \mathcal{E}_{int}^n -automata.

A *global state* in such an automaton is a pair $(s, \mathbf{x}) \in S \times \mathbb{R}_{\geq 0}^n$, and *transitions* are of the form $(s, \mathbf{x}) \xrightarrow{\mathbf{f}} (s', \mathbf{x}')$ such that $(s, \mathbf{f}, s') \in T$ and $\mathbf{x}'(i) = \mathbf{x}(i) + \mathbf{f}(i)$ for each $i \in \{1, \dots, n\}$. Here, $\mathbf{u}(i)$ denotes the *i*th element of the vector \mathbf{u} .

A *run* of an \mathcal{E}^n -automaton (S, T) from (s, \mathbf{x}) to (s', \mathbf{x}') is a finite sequence $(s_0, \mathbf{x}_0), \dots, (s_k, \mathbf{x}_k)$ of global states such that $(s_0, \mathbf{x}_0) = (s, \mathbf{x})$, $(s_k, \mathbf{x}_k) = (s', \mathbf{x}')$, and for all $i \in \{1, \dots, k\}$ there exists $\mathbf{f}_i \in \mathcal{E}^n$ such that $(s_{i-1}, \mathbf{x}_{i-1}) \xrightarrow{\mathbf{f}_i} (s_i, \mathbf{x}_i)$. We say that such a run has length k .

We define an ordering \leq on global states by $(s, \mathbf{x}) \leq (s', \mathbf{x}')$ if, and only if, $s = s'$ and $\mathbf{x}(i) \leq \mathbf{x}'(i)$ for each $i = 1, \dots, n$. We restate the *coverability problem* and the *state reachability problem*: Given an \mathcal{E}^n -automaton, an initial global state (s, \mathbf{x}) , where $\mathbf{x} \in \mathbb{R}_{\geq 0}^n$ is a computable initial energy, and some global state (s', \mathbf{x}') , the coverability problem is to decide whether there exists a run from (s, \mathbf{x}) to (s', \mathbf{x}'') such that $\mathbf{x}' \leq \mathbf{x}''$. The state reachability problem is a special case of the coverability problem for $\mathbf{x}' = \mathbf{0}$.

For reachability in \mathcal{E}^n -automata with $n \geq 2$, our algebraic results do not apply. To see this, we refer to the state reachability problem in Fig. 10: with initial energy $(1, 1)$, the loop needs to be taken precisely once, but with initial energy $(2, 0)$, one needs to loop twice. To make this argument precise, let f denote the function corresponding to the $(-1, 1)$ loop and g the function on the $(0, -2)$ edge. Then we should have $(1, 1)f^*g = (1, 1)fg$ and $(2, 0)f^*g = (2, 0)f^2g$. Hence our framework of computing with energy *functions* will not apply.

9.1 $\mathcal{E}_{\text{pw}}^n$ -automata

Recall that n -dimensional VASS form a subclass of our $\mathcal{E}_{\text{pw}}^n$ -automata. The coverability problem for VASS is EXPSPACE-complete [44, 47]. In this subsection we aim to show the same complexity for the coverability and the state reachability problem for $\mathcal{E}_{\text{pw}}^n$ -automata. For EXPSPACE-membership, we extend Rackoff's proof for VASS [47].

The proof is inspired by a proof for EXPSPACE-completeness for the class of *strongly increasing affine nets* [9]. Affine nets are extensions of classical Petri nets. Recall that in Petri nets the current placement of tokens in the places (called *marking*) changes according to the transition rules, which simply add or subtract tokens from a place. In affine nets, the transition rules are affine functions of the form $\mathbf{A}M + \mathbf{B}$, where $\mathbf{A} \in \mathbb{N}^n \times \mathbb{N}^n$, $\mathbf{B} \in \mathbb{N}^n$, and M is a marking of the n -dimensional net. Such a function is *strongly increasing* if \mathbf{A} is greater than or equal to the identity matrix, cf. condition (*) for energy functions. Note that strongly increasing affine nets operate on vectors over \mathbb{N} , while $\mathcal{E}_{\text{pw}}^n$ -automata operate on vectors over $\mathbb{R}_{\geq 0}$. However, we remark that even $\mathcal{E}_{\text{pwi}}^n$ -automata and strongly increasing affine nets are incomparable in expressiveness: affine nets do not allow for *piecewise* affine functions, and in $\mathcal{E}_{\text{pwi}}^n$ -automata the value of an energy variables cannot influence the value of another energy variable as it is the case in affine nets. That is, in $\mathcal{E}_{\text{pwi}}^n$ -automata all \mathbf{A} matrices are *diagonal*.

Before we prove EXPSPACE-completeness of the coverability problem for $\mathcal{E}_{\text{pw}}^n$ -automata, we introduce some helpful notions and prove some lemmas.

Recall that every 1-dimensional integer piecewise affine energy function $f \in \mathcal{E}_{\text{pw}}$ can be represented by its breakpoints $x_0, \dots, x_m \in \mathbb{Q}$, the values $x_0f, \dots, x_mf \in \mathbb{Q}$, and the numbers $a_0, b_0, a_1, b_1, \dots, a_m, b_m \in \mathbb{Z}$, where $a_j \geq 1$ for all $0 \leq j \leq m$. We use $\text{xmin}_f = x_0$ to denote the *minimal break point* of f . For simplicity we assume that xf is defined iff $x \geq \text{xmin}_f$, but our arguments also apply to the case where the definition interval of f is open. For n -dimensional integer piecewise affine energy functions $\mathbf{f} \in \mathcal{E}_{\text{pw}}^n$, we define $\text{xmin}_{\mathbf{f}} \in \mathbb{Q}^n$ to be the n -dimensional vector defined by $\text{xmin}_{\mathbf{f}}(i) = \text{xmin}_{\mathbf{f}(i)}$ for all $i \in \{1, \dots, n\}$.

Fix some $\mathcal{E}_{\text{pw}}^n$ -automaton (S, T) . We use bmax to denote the maximum of the absolute values of all negative constants smaller than or equal to -1 and occurring in the representation of any energy function in (S, T) , or 1 if there is no such constant. We further use xminmax to denote the maximum of 1 and the minimal break points of all energy functions occurring in (S, T) . (Hence $\text{xminmax} \geq 1$.)

The next lemma states the easy fact that the decrease in the value of an energy variable during a run is bounded.

Lemma 15. *For every run from (s, \mathbf{x}) to (s', \mathbf{x}') of length k , $\mathbf{x}'(i) \geq \mathbf{x}(i) - k \text{bmax}$.*

On the other hand, the value of an energy variable can grow quickly even in very short runs. However, for deciding the coverability problem, already for VASS, it is not necessary to store the exact concrete value of an energy variable once it has exceeded a certain high value. Instead, we will represent high values symbolically by ω . The energy variables in the global states of our algorithm will hence take

values in $\mathbb{R}_\omega := \mathbb{R}_{\geq 0} \cup \{\omega\}$. Define for every $y \in \mathbb{R}_\omega$, $\omega + y = \omega - y = \omega$, $y \cdot \omega = \omega$, and $y \leq \omega$. Using this, we can extend the definition of $f \in \mathcal{E}_{pw}$ to a function $f : \mathbb{R}_\omega \rightarrow \mathbb{R}_\omega$ in a natural way. We further extend these definitions to the n -dimensional case. In the following, we will use \mathbf{x} to denote vectors in \mathbb{R}^n , and \mathbf{y} to denote vectors in $(\mathbb{R}_\omega)^n$.

Let $\mathbf{y} \in (\mathbb{R}_\omega)^n$ and assume $\mathbf{y}(i) \in \mathbb{R}_{\geq 0}$. We explain when to replace the concrete value $\mathbf{y}(i)$ by ω . The crucial point here is that there is not a single threshold value $t \in \mathbb{R}_{\geq 0}$ such that $\mathbf{y}(i)$ is replaced by ω whenever $\mathbf{y}(i) \geq t$. Instead, $\mathbf{y}(i)$ is replaced by ω whenever $\mathbf{y}(i) \geq t(r)$, where r is the number of indices j for which $\mathbf{y}(j) \in \mathbb{R}_{\geq 0}$, and t is a mapping $t : \{0, \dots, n\} \rightarrow \mathbb{R}_{\geq 0}$. To make this formal, we define

$$\begin{aligned} \text{omega}(\mathbf{y}) &= \{i \in \{1, \dots, n\} \mid \mathbf{y}(i) = \omega\}, \\ \text{real}(\mathbf{y}) &= \{i \in \{1, \dots, n\} \mid \mathbf{y}(i) \in \mathbb{R}_{\geq 0}\}. \end{aligned}$$

For a finite set λ we use $|\lambda|$ to denote its cardinality. Let $t : \{0, 1, \dots, n\} \rightarrow \mathbb{R}_{\geq 0}$ be a mapping. We define the vector $\mathbf{y}_t \in (\mathbb{R}_\omega)^n$ by

$$\mathbf{y}_t(i) = \begin{cases} \mathbf{y}(i) & \text{if } \mathbf{y}(i) < t(|\text{real}(\mathbf{y})|), \\ \omega & \text{if } \mathbf{y}(i) \geq t(|\text{real}(\mathbf{y})|). \end{cases}$$

Thus, in \mathbf{y}_t all entries which are greater than or equal to the value $t(|\text{real}(\mathbf{y})|)$ are replaced by ω ; other entries do not change.

Next, we define the *abstract t -semantics* of (S, T) . For this, let $t : \{0, 1, \dots, n\} \rightarrow \mathbb{R}_{\geq 0}$ be a mapping. A *global t -state* of (S, T) is a pair $(s, \mathbf{y}) \in S \times (\mathbb{R}_\omega)^n$. We define a t -transition relation over the set of global t -states by $(s, \mathbf{y}) \xrightarrow{f}_t (s', \mathbf{y}')$ iff $(s, \mathbf{f}, s') \in T$, $\mathbf{y}\mathbf{f}$ is defined, and $\mathbf{y}' = (\mathbf{y}\mathbf{f})_t$. A t -run of (S, T) from (s, \mathbf{y}) to (s', \mathbf{y}') is a finite sequence $(s_0, \mathbf{y}_0), (s_1, \mathbf{y}_1), \dots, (s_k, \mathbf{y}_k)$ of global t -states such that $(s_0, \mathbf{y}_0) = (s, \mathbf{y})$, $(s_k, \mathbf{y}_k) = (s', \mathbf{y}')$, and $(s_{i-1}, \mathbf{y}_{i-1}) \xrightarrow{f_i}_t (s_i, \mathbf{y}_i)$ for some $f_i \in \mathcal{E}_{pw}^n$, for all $i \in \{1, \dots, k\}$. We say that such a t -run has length k .

The following observation can be easily proved.

Lemma 16. *If $\mathbf{y}(i) = \omega$ and $(s, \mathbf{y}) \xrightarrow{f}_t (s', \mathbf{y}')$, then $\mathbf{y}'(i) = \omega$, hence $\text{omega}(\mathbf{y}) \subseteq \text{omega}(\mathbf{y}')$.*

The following lemma will be needed to prove the completeness of our algorithm. The proof is simple and left to the reader.

Lemma 17. *If there is a run from (s_1, \mathbf{x}_1) to (s_2, \mathbf{x}_2) of length k , then there is for every $\mathbf{x}'_1 \geq \mathbf{x}_1$ a t -run from (s_1, \mathbf{x}'_1) to (s_2, \mathbf{y}_2) of length k and such that $\mathbf{y}_2 \geq \mathbf{x}_2$.*

For proving the soundness of our algorithm, we need to be able to reverse the process of abstracting concrete data values. Given $\mathbf{y} \in (\mathbb{R}_\omega)^n$ with $\mathbf{y}(i) = \omega$ for some $i \in \{1, \dots, n\}$, we define the vector $\mathbf{y}_{t^{\text{rev}}} \in \mathbb{R}_{\geq 0}^n$ by

$$\mathbf{y}_{t^{\text{rev}}}(i) = \begin{cases} \mathbf{y}(i) & \text{if } \mathbf{y}(i) \in \mathbb{R}_{\geq 0}, \\ t(|\text{real}(\mathbf{y})| + 1) & \text{if } \mathbf{y}(i) = \omega. \end{cases} \quad (3)$$

The following lemma will be crucial for proving the soundness of our algorithm.

Lemma 18. *For every $P \in \mathbb{R}_{\geq 0}$, if there is a t -run from (s, \mathbf{y}) to (s', \mathbf{y}') of length k , $|\text{real}(\mathbf{y})| = |\text{real}(\mathbf{y}')|$, and $t(|\text{real}(\mathbf{y})| + 1) \geq k \text{bmax} + \text{xminmax} + P$, then there exists a run from $(s, \mathbf{y}_{t^{\text{rev}}})$ to some global state (s', \mathbf{x}') of length at most k , where $\mathbf{x}'(i) = \mathbf{y}'(i)$ for all $i \in \text{real}(\mathbf{y}')$ and $\mathbf{x}'(i) \geq \text{xminmax} + P$ for all $i \in \text{omega}(\mathbf{y}')$.*

Proof. Note that $|\text{real}(\mathbf{y})| = |\text{real}(\mathbf{y}')|$ implies that in the t -run from (s, \mathbf{y}) to (s', \mathbf{y}') no ω -abstraction took place. We prove the lemma by induction on k .

For the induction base, let $k = 0$. Hence $(s', \mathbf{y}') = (s, \mathbf{y})$. Then we have $\mathbf{y}_{t^{\text{rev}}}(i) = t(|\text{real}(\mathbf{y})| + 1) \geq \text{xminmax} + P$ for all $i \in \text{omega}(\mathbf{y})$, where the first equality holds by (3), and the inequality holds by assumption.

For the induction step, assume $k > 0$. Hence there exists (s'', \mathbf{y}'') such that $(s, \mathbf{y}) \xrightarrow{f}_t (s'', \mathbf{y}'')$, and there exists a t -run from (s'', \mathbf{y}'') to (s', \mathbf{y}') of length $k - 1$.

Note that by assumption $|\text{omega}(\mathbf{y})| = |\text{omega}(\mathbf{y}')|$, and by Lemma 16 the global t -states occurring in the t -run have ω -entries in the same dimensions.

We first argue that we can execute the transition labeled with f on the global state $(s, \mathbf{y}_{t^{\text{rev}}})$, *i.e.*, $\mathbf{y}_{t^{\text{rev}}}(i)f(i)$ is defined for all $i \in \{1, \dots, n\}$. For $i \in \text{real}(\mathbf{y})$ this is clear. So let $i \in \text{omega}(\mathbf{y})$. By assumption,

$$\mathbf{y}_{t^{\text{rev}}}(i) = t(|\text{real}(\mathbf{y})| + 1) \geq k \text{bmax} + \text{xminmax} + P. \quad (4)$$

It follows by $\text{xminmax} \geq \text{xmin}_{f(i)}$ that $\mathbf{y}_{t^{\text{rev}}}(i)f(i)$ is defined.

Let (s'', \mathbf{x}'') be the global state that results from applying f to $(s, \mathbf{y}_{t^{\text{rev}}})$, *i.e.*,

$$(s, \mathbf{y}_{t^{\text{rev}}}) \xrightarrow{f} (s'', \mathbf{x}'').$$

By (4) and Lemma 15, we have for all $i \in \text{omega}(\mathbf{y}'')$, and $\mathbf{x}''(i) = \mathbf{y}''(i)$ for all $i \in \text{real}(\mathbf{y}'')$.

Define the mapping $t_1 : \{0, 1, \dots, n\} \rightarrow \mathbb{R}_{\geq 0}$ by

$$t_1(j) = \begin{cases} t(j) & \text{if } j \neq |\text{real}(\mathbf{y}'')| + 1, \\ t(|\text{real}(\mathbf{y}'')| + 1) - \text{bmax} & \text{otherwise.} \end{cases}$$

Note that

$$\begin{aligned} t_1(|\text{real}(\mathbf{y}'')| + 1) &= t(|\text{real}(\mathbf{y}'')| + 1) - \text{bmax} \\ &= t(|\text{real}(\mathbf{y})| + 1) - \text{bmax} \\ &\geq (k - 1)\text{bmax} + \text{xminmax} + P, \end{aligned}$$

where the first equation holds by definition of t_1 , the second equation holds by $|\text{real}(\mathbf{y}'')| = |\text{real}(\mathbf{y})|$, and the inequality holds by assumption. We can thus apply the induction hypothesis on the t_1 -run from (s'', \mathbf{y}'') to (s', \mathbf{y}') of length $k - 1$. Hence there exists a run from $(s'', \mathbf{y}'_{t_1^{\text{rev}}})$ to (s', \mathbf{x}') of length at most $k - 1$ and such that $\mathbf{x}'(i) = \mathbf{y}'(i)$ for all $i \in \text{real}(\mathbf{y}')$, and $\mathbf{x}'(i) \geq \text{xminmax} + P$ for all $i \in \text{omega}(\mathbf{y}')$.

Observe that $\mathbf{x}'' \geq \mathbf{y}'_{t_1^{\text{rev}}}$: $\mathbf{x}''(i) = \mathbf{y}'_{t_1^{\text{rev}}}(i)$ for all $i \in \text{real}(\mathbf{y}'')$, and $\mathbf{x}''(i) \geq t(|\text{real}(\mathbf{y})| + 1) - \text{bmax}$ and $\mathbf{y}'_{t_1^{\text{rev}}}(i) = t(|\text{real}(\mathbf{y})| + 1) - \text{bmax}$ for all $i \in \text{omega}(\mathbf{y}'')$. We can thus conclude that there is a run from (s'', \mathbf{x}'') to some global state (s', \mathbf{x}'_1) of length at most $k - 1$ and with $\mathbf{x}'_1 \geq \mathbf{x}'$, which yields the statement of the lemma. \square

We are finally ready to state the main result of this subsection.

Theorem 5. *State reachability and coverability are EXPSPACE-complete for $\mathcal{E}_{\text{pw}}^n$ -automata for $n \geq 3$.*

Proof. The lower bound follows from EXPSPACE-hardness for VASS [44]. For the upper bound, let (S, T) be an $\mathcal{E}_{\text{pw}}^n$ -automaton, let (s, \mathbf{x}) and (s', \mathbf{x}') , respectively, be the initial global state and the global state to be covered, respectively. We use $\text{cmax} = \max\{\text{bmax}, \max_i \mathbf{x}'(i)\}$ to denote the maximum of bmax and the maximum entry in \mathbf{x}' ; note that $\text{cmax} \geq 1$. Define $\text{thd} : \{0, 1, \dots, n\} \rightarrow \mathbb{Q}_{\geq 0}$ by

$$\text{thd}(0) = 0, \quad \text{thd}(i) = \text{bmax} \cdot \text{len}(i-1) + \text{xminmax} + \text{cmax}$$

for every $i \in \{1, \dots, n\}$, where $\text{len} : \{0, \dots, n\} \rightarrow \mathbb{Q}_{\geq 0}$ is defined inductively by

$$\text{len}(0) = |S|, \quad \text{len}(i) = (\text{thd}(i))^i |S| + \text{len}(i-1)$$

for every $i \in \{1, \dots, n\}$.

The correctness of our algorithm is based on the following two claims.

Soundness Claim. If there exists a thd -run from (s, \mathbf{y}) to (s', \mathbf{y}') of length k and such that $\mathbf{y}' \geq \mathbf{x}'$, then there exists a run from $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$ to (s', \mathbf{x}'') of length at most k and such that $\mathbf{x}'' \geq \mathbf{x}'$.

Completeness Claim. If there exists a run from (s_0, \mathbf{x}_0) to (s', \mathbf{x}'') for some $\mathbf{x}'' \geq \mathbf{x}'$, then there is also a thd -run from (s_0, \mathbf{x}_0) to (s', \mathbf{y}') of length $k \leq \text{len}(n)$ and such that $\mathbf{y}' \geq \mathbf{x}'$.

By these two claims, there is a run from (s, \mathbf{x}) to (s', \mathbf{x}'') for some $\mathbf{x}'' \geq \mathbf{x}'$ if, and only if, there is a thd -run from (s, \mathbf{x}) to (s', \mathbf{y}') such that $\mathbf{y}' \geq \mathbf{x}'$ and of length bounded by $\text{len}(n)$. The existence of such a thd -run can thus be verified by a non-deterministic Turing machine that keeps in memory one global thd -state as well as one counter counting up to $\text{len}(n)$. Every entry in $(\mathbb{R}_\omega)^n$ occurring in the analysis is either ω or less than $\text{thd}(n) = \text{bmax} \cdot \text{len}(n) + \text{xminmax} + \text{cmax}$. By the Length Claim below, the memory space needed by the algorithm is $\mathcal{O}((n+1)!(\log(\text{bmax}) + \log(\text{cmax}) + \log(\text{xminmax})) + \log(|S|)) = \mathcal{O}(2^{cn \log n}(\log(\text{cmax}) + \log(\text{xminmax}) + \log(|S|)))$, which is in NEXPSpace.

Length Claim. For all $i \in \mathbb{N}$, $\text{len}(i) \leq (6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{(i+1)!} |S|$.

In the remainder of this subsection, we prove the three claims stated above.

Proof of the Soundness Claim. The proof is by induction on $|\text{real}(\mathbf{y})|$.

For the induction base, assume $|\text{real}(\mathbf{y})| = 0$, *i.e.*, we have $\mathbf{y}(i) = \omega$ for all $i \in \{1, \dots, n\}$. By definition, for every $i \in \{1, \dots, n\}$ we have

$$\begin{aligned} \mathbf{y}_{\text{thd}^{\text{rev}}}(i) &= \text{thd}(|\text{real}(\mathbf{y})| + 1) = \text{thd}(1) \\ &= \text{bmax} \cdot \text{len}(0) + \text{xminmax} + \text{cmax} \\ &= \text{bmax} |S| + \text{xminmax} + \text{cmax}. \end{aligned} \tag{5}$$

For defining the sequence of transitions that leads $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$ to some (s', \mathbf{x}'') in at most k transition steps and such that $\mathbf{x}'' \geq \mathbf{x}'$, we distinguish two cases. First, assume that the length k of the thd -run from (s, \mathbf{y}) to (s', \mathbf{y}') satisfies $k > |S|$. There must exist a (syntactical) cycle-free path from s to s' in (S, T) of length bounded by $|S|$. By (5) and Lemma 15, there is a run corresponding to this path from $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$ to (s', \mathbf{x}'') , where \mathbf{x}'' satisfies $\mathbf{x}''(i) \geq \text{xminmax} + \text{cmax}$ for all $i \in \{1, \dots, n\}$. By definition of cmax , we can conclude $\mathbf{x}'' \geq \mathbf{x}'$.

Second, assume that the length k of the thd -run from (s, \mathbf{y}) to (s', \mathbf{y}') satisfies $k \leq |S|$. By (5) and Lemma 15, we can argue that starting from $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$ we can use the same sequence of transitions as in the thd -run, yielding a run of length k , and reaching a global state (s', \mathbf{x}'') for some $\mathbf{x}'' \geq \mathbf{x}'$. This finishes the induction base.

For the induction step assume $|\text{real}(\mathbf{y})| = i + 1$. We consider two cases.

(i) First assume $|\text{real}(\mathbf{y})| = |\text{real}(\mathbf{y}')|$. By definition of the thd -semantics, every global thd -state (s_1, \mathbf{y}_1) that occurs in the thd -run ρ from (s, \mathbf{y}) to (s', \mathbf{y}') satisfies

- $|\text{real}(\mathbf{y}_1)| = i + 1$,
- $\mathbf{y}_1(j) < \text{thd}(|\text{real}(\mathbf{y}_1)|) = \text{thd}(i + 1)$ for all $j \in \text{real}(\mathbf{y}_1)$.

Let ρ' be the run from (s, \mathbf{y}) to (s', \mathbf{y}') that is obtained from ρ by removing all cycles between identical global thd -states. The length k' of ρ' is bounded by

$$k' \leq (\text{thd}(i + 1))^{i+1} |S|.$$

Next we prove that there is a corresponding run from $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$. For this, note that

$$\begin{aligned} \text{thd}(|\text{real}(\mathbf{y})| + 1) &= \text{thd}((i + 1) + 1) \\ &= \text{bmax} \cdot \text{len}(i + 1) + \text{xminmax} + \text{cmax} \\ &= \text{bmax}((\text{thd}(i + 1))^{i+1} |S| + \text{len}(i)) + \text{xminmax} + \text{cmax} \\ &\geq \text{bmax}((\text{thd}(i + 1))^{i+1} |S|) + \text{xminmax} + \text{cmax} \\ &\geq \text{bmax} \cdot k' + \text{xminmax} + \text{cmax}. \end{aligned}$$

By Lemma 18 (for $P = \text{cmax}$), there exists a run from $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$ to (s', \mathbf{x}'') of length at most k' and such that $\mathbf{x}''(j) = \mathbf{y}'(j)$ for all $j \in \text{real}(\mathbf{y}')$, and $\mathbf{x}''(j) \geq \text{xminmax} + \text{cmax}$ for all $j \in \text{omega}(\mathbf{y}')$. Then $\mathbf{x}'' \geq \mathbf{x}'$ follows by $\text{cmax} \geq \mathbf{x}'(j)$ for all $j \in \{1, \dots, n\}$.

(ii) Now assume $|\text{real}(\mathbf{y})| > |\text{real}(\mathbf{y}')|$. In the thd -run ρ from (s, \mathbf{y}) to (s', \mathbf{y}') , let (s_1, \mathbf{y}_1) be the last global thd -state with $|\text{real}(\mathbf{y}_1)| = i + 1$. Partition ρ into three parts: A thd -run ρ_1 of length k_1 from (s, \mathbf{y}) to (s_1, \mathbf{y}_1) , a thd -transition $(s_1, \mathbf{y}_1) \xrightarrow{f}_{\text{thd}} (s_2, \mathbf{y}_2)$ (where $\text{real}(\mathbf{y}_2) \leq i$), and a thd -run ρ_2 of length k_2 from (s_2, \mathbf{y}_2) to (s', \mathbf{y}') . Hence $k = k_1 + k_2 + 1$. Note that ρ_1 may be empty, in case $(s_1, \mathbf{y}_1) = (s, \mathbf{y})$; in that case, $k_1 = 0$ and $k = k_2 + 1$.

As in case (i), we can show that there is a run ρ'_1 from (s, \mathbf{y}) to (s_1, \mathbf{y}_1) of length $k'_1 \leq (\text{thd}(i + 1))^{i+1} |S|$. Using similar arguments as above, we obtain

$$\text{thd}(|\text{real}(\mathbf{y})| + 1) \geq \text{bmax} \cdot k'_1 + \text{bmax} \cdot \text{len}(i) + \text{xminmax} + \text{cmax}.$$

By Lemma 18 (with $P = \text{bmax} \cdot \text{len}(i) + \text{cmax}$) there exists a run from $(s, \mathbf{y}_{\text{thd}^{\text{rev}}})$ to (s_1, \mathbf{x}_1) of length at most k'_1 and such that $\mathbf{x}_1(j) = \mathbf{y}_1(j)$ for all $j \in \text{real}(\mathbf{y}_1)$, and $\mathbf{x}_1(j) \geq \text{bmax} \cdot \text{len}(i) + \text{xminmax} + \text{cmax}$ for all $j \in \text{omega}(\mathbf{y}_1)$. Note that $\mathbf{x}_1 \geq \mathbf{y}_{1\text{thd}^{\text{rev}}}$. Hence we also have $(s_1, \mathbf{x}_1) \xrightarrow{f} (s_2, \mathbf{x}_2)$ for some $\mathbf{x}_2 \geq \mathbf{y}_{2\text{thd}^{\text{rev}}}$. By induction hypothesis, there exists a run from $(s_2, \mathbf{y}_{2\text{thd}^{\text{rev}}})$ to (s', \mathbf{x}'') for some $\mathbf{x}'' \geq \mathbf{x}'$ and of length at most k_2 . But by $\mathbf{x}_2 \geq \mathbf{y}_{2\text{thd}^{\text{rev}}}$ there also exists such a run from (s, \mathbf{x}_2) . This completes the induction step and thus the proof of the soundness claim.

Proof of the Completeness Claim. Assume that there exists a run from (s, \mathbf{x}) to (s', \mathbf{x}'') for some $\mathbf{x}'' \geq \mathbf{x}'$. By Lemma 17 there exists a thd-run ρ from (s, \mathbf{x}) to (s', \mathbf{y}) for some $\mathbf{y} \geq \mathbf{x}'$. Let ρ' be the thd-run from (s, \mathbf{x}) to (s', \mathbf{y}) that is obtained from ρ by removing all cycles between identical global thd-states. We prove that $|\rho'| \leq \text{len}(n)$. For all $i \in \{|\text{real}(\mathbf{y})|, \dots, n\}$, let (s_i, \mathbf{y}_i) be the first global thd-state occurring in ρ' such that $|\text{real}(\mathbf{y}_i)| \leq i$, and let ρ'_i be the suffix of ρ' that starts in (s_i, \mathbf{y}_i) . We prove by induction on i that $|\rho'_i| \leq \text{len}(i)$.

For the base case, let $i = |\text{real}(\mathbf{y})|$. By definition of the thd-semantics, every global thd-state (s', \mathbf{y}') occurring in ρ'_i satisfies $\mathbf{y}'(j) < \text{thd}(i)$ for every $j \in \text{real}(\mathbf{y}')$. Since there are exactly i entries that take values in $\{0, \dots, \text{thd}(i) - 1\}$, the length $|\rho'_i|$ of ρ'_i is bounded by $\text{thd}(i)^i \cdot |S|$. This and the definition of len yields $|\rho'_i| \leq \text{len}(i)$.

For the induction step, let ρ_{i+1} be the prefix of ρ'_{i+1} that starts in $(s_{i+1}, \mathbf{y}_{i+1})$ and ends in (s_i, \mathbf{y}_i) . Every global thd-state (s', \mathbf{y}') that occurs in ρ_{i+1} (except for (s_i, \mathbf{y}_i)) satisfies $\mathbf{y}'(j) < \text{thd}(i+1)$ for all $j \in \text{real}(\mathbf{y}') = \text{real}(\mathbf{y}_{i+1})$. Hence $|\rho_{i+1}| \leq (\text{thd}(i+1))^{i+1} |S|$. By induction hypothesis, $|\rho'_i| \leq \text{len}(i)$. Altogether, we obtain

$$|\rho'_{i+1}| = |\rho_{i+1}| + |\rho'_i| \leq (\text{thd}(i+1))^{i+1} |S| + \text{len}(i) \leq \text{len}(i+1).$$

This completes the induction step and the proof of the completeness claim.

Proof of the Length Claim. The claim is proved by induction on i . The base case, $i = 0$, trivially holds. So let us assume that the claim holds for i . Then

$$\begin{aligned} \text{len}(i+1) &= (\text{bmax} \cdot \text{len}(i) + \text{xminmax} + \text{cmax})^{i+1} |S| + \text{len}(i) \\ &\leq (2 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax} \cdot \max(\text{len}(i), 1))^{i+1} |S| + \text{len}(i) \\ &\leq 3(2 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax} \cdot \max(\text{len}(i), 1))^{i+1} |S| \\ &\leq 3^{i+1} (2 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{i+1} (\max(\text{len}(i), 1))^{i+1} |S| \\ &\leq (6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{i+1} ((6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{(i+1)!})^{i+1} |S| \\ &\leq (6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{(i+1)!} (6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{(i+1)(i+1)!} |S| \\ &= (6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{(i+2)(i+1)!} |S| \\ &= (6 \text{bmax} \cdot \text{cmax} \cdot \text{xminmax})^{(i+2)!} |S|. \end{aligned}$$

With the proves of these three lemmas, the proof of Theorem 5 is complete. \square

9.2 Flat $\mathcal{E}_{\text{pw}}^n$ -automata

Next we show that if the requirement $(*)$ on energy functions, that $yf \geq xf + y - x$ for each $x \leq y$, is lifted, then reachability becomes undecidable from dimension 4. We call such functions *flat* energy functions; remark that we still require them to be strictly increasing, but the derivative, where it exists, may be less than 1. The class of all flat energy functions is denoted $\bar{\mathcal{E}}$ and its restrictions by $\bar{\mathcal{E}}_{\text{pw}}$ and $\bar{\mathcal{E}}_{\text{pwi}}$.

Theorem 6. *State reachability, coverability, and Büchi acceptance are undecidable for $\bar{\mathcal{E}}_{\text{pw}}^4$ -automata.*

Proof. The proof is a reduction from the halting problem (respectively, the recurrent state problem) for two-counter machines. A *two-counter machine* \mathcal{M} is a finite sequence $(\mathcal{I}_j)_{j=1}^n$ of instructions operating on two counters denoted by C_1 and C_2 , where \mathcal{I}_j is one of the following instructions (with $i \in \{1, 2\}$ and $j, k, m \in \{1, \dots, n\}$):

increment	$\mathcal{I}_j: C_i := C_i + 1; \text{ go to } \mathcal{I}_k$
zero test/dec	$\mathcal{I}_j: \text{if } C_i = 0 \text{ then goto } \mathcal{I}_k \text{ else } C_i := C_i - 1; \text{ goto } \mathcal{I}_m$
halt	$\mathcal{I}_j: \text{halt}$

A *configuration* of such a two-counter machine \mathcal{M} is a triple $\gamma = (J, c, d) \in \{\mathcal{I}_1, \dots, \mathcal{I}_n\} \times \mathbb{N} \times \mathbb{N}$, where J indicates the current instruction, and c and d are the current values of the counters C_1 and C_2 , respectively. A *computation* of \mathcal{M} is a finite or infinite sequence $(\gamma_i)_{i \geq 0}$ of configurations, such that $\gamma_0 = (\mathcal{I}_1, 0, 0)$ and γ_{i+1} is the result of executing the instruction \mathcal{I}_i on γ_i for each $i \geq 0$. Without loss of generality, we assume that \mathcal{I}_n is the only instruction of the form **halt**. The *halting problem* for two-counter machines asks, given a two-counter machine \mathcal{M} , whether the (unique) computation of \mathcal{M} reaches a configuration with instruction \mathcal{I}_n , *i.e.*, the halting instruction. This problem is Σ_1^0 -complete [45]. The *recurrent state problem* for two-counter machines asks, given a two-counter machine \mathcal{M} , whether the (unique) computation of \mathcal{M} visits instruction \mathcal{I}_1 infinitely often. This problem is Σ_1^1 -complete [3].

Given a two-counter machine \mathcal{M} , we construct an $\bar{\mathcal{E}}_{\text{pw}}^4$ -automaton $\mathcal{A}_{\mathcal{M}}$ with state set $S \subseteq \{\mathcal{I}_1, \dots, \mathcal{I}_n\}$ such that \mathcal{M} halts (visits instruction \mathcal{I}_1 infinitely often, respectively) if, and only if, $\mathcal{A}_{\mathcal{M}}$ reaches state \mathcal{I}_n (visits \mathcal{I}_1 infinitely often, respectively).

We use two variables x and y to store the values of the counters C_1 and C_2 by requiring that $x = 1/(2^{c_1}3^{c_2})$ and $y = 2^{c_1}3^{c_2}$. Two other variables z and z' are used for storing temporary information needed for encoding zero tests and decrementation operations of the 2-counter machine. The initial value of all energy variables is 1.

For encoding the instructions of the 2-counter machine, we first define the following flat energy functions: dec_2 multiplies the current values of x and z' with 3, and divides the current values of y and z by 3; inc_2 turns these operations back, *i.e.*, it divides the current values of x and z' by 3 and multiplies the values of y and z with 3. Functions dec_1 and inc_1 are defined analogously by replacing 3 by 2. Finally, function dec'_1 behaves like dec_1 but does not change the values of z

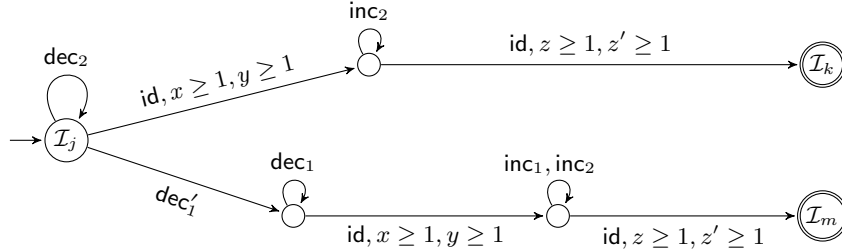


Figure 11: Module for encoding zero test/decrement instructions of a 2-counter machine

and z' ; likewise for inc'_1 . An increment instruction for counter C_1 is then encoded by a simple transition from state \mathcal{I}_j to state \mathcal{I}_k labeled by inc'_1 ; analogously for an increment of counter C_2 . The encoding zero test/decrement instructions for counter C_1 can be done as shown in Fig. 11. Here, the additional inequalities for the definition intervals of the energy functions are crucial for the correctness of the construction. \square

9.3 Reachability games on \mathcal{E}_{pw}^n -automata

Next we extend our energy automata formalism to (turn based) *reachability games*. Let (S, T) be an n -dimensional energy automaton such that $S = S_A \cup S_B$ forms a partition of S and $T \subseteq (S_A \times \mathcal{E}_{pw}^n \times S_B) \cup (S_B \times \mathcal{E}_{pw}^n \times S_A)$. Then (S, S_A, S_B, T) induces an n -dimensional *energy game* G . The intuition of the reachability game is that the two players A and B take turns to move along the game graph (S, T) , updating energy values at each turn. The goal of player A is to reach a state in F , the goal of player B is to prevent this from happening.

The reachability game is a game on a well-structured transition system as in [1]. In general, the reachability game on well-structured transition systems is undecidable; in particular, the game on 2-dimensional vector addition systems with states is undecidable [1]. It is hence clear that it is undecidable whether player A wins the reachability game in 2-dimensional \mathcal{E}_{int} -automata.

Theorem 7. *Whether player A wins the reachability game in \mathcal{E}_{int}^2 -automata is undecidable.*

As a corollary, we can show that for *flat* energy functions, already one-dimensional reachability games are undecidable.

Theorem 8. *It is undecidable for $\bar{\mathcal{E}}_{pw}$ -automata whether player A wins the reachability game.*

Proof. We show a reduction from reachability games on 2-dimensional \mathcal{E}_{int} -automata to reachability games on 1-dimensional $\bar{\mathcal{E}}_{pw}$ -automata. Let (S, T) be a 2-dimensional

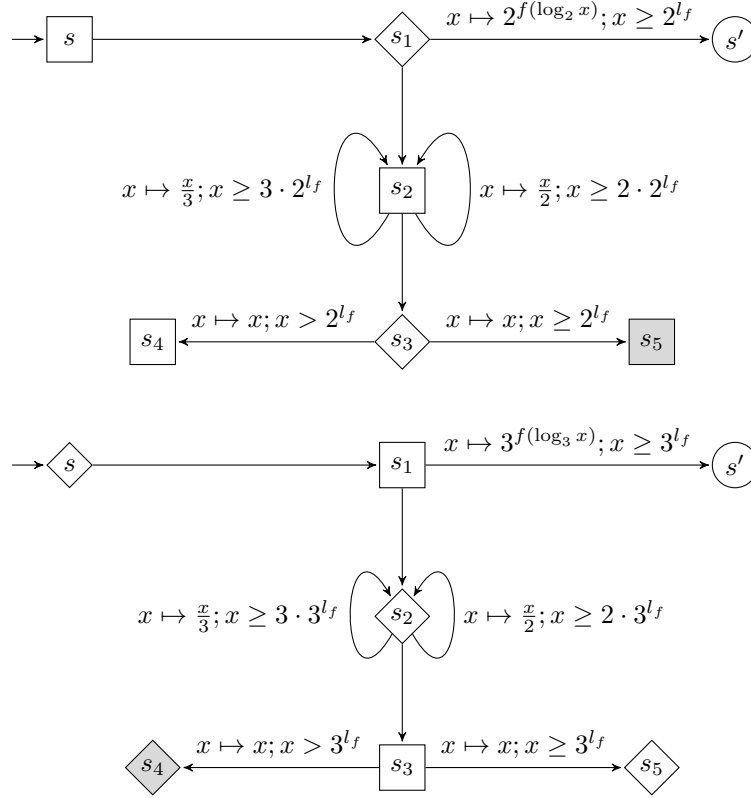


Figure 12: Conversion of two types of edges in (S, T) . Top: an edge $(s, (f, \text{id}), s')$ from a player-A state s ; bottom: an edge $(s, (\text{id}, f), s')$ from a player-B state s . Player-A states are depicted using squares, player-B states are diamonds. Accepting states have a gray background color. The ownership of state s' is unchanged.

\mathcal{E}_{int} -automaton. By inserting extra states (and transitions) if necessary, we can assume that for any $(s, (f, g), s') \in T$, either $f = \text{id}$ with $l_f = 0$, or $g = \text{id}$ with $l_g = 0$. We build an energy automaton (S', T') .

Let $(s, (f, \text{id}), s') \in T$ be a player-A transition (*i.e.*, $s \in S_A$) in (S, T) (with lower bound l_f as usual), then we model this in (S', T') using s, s' and the following new states and transitions; see Figure 12 for a pictorial description.

- player-A states: s_2, s_4, s_5 (accepting); player-B states: s_1, s_3
- transitions:
 - $(s, [x \mapsto x; x \geq 0], s_1); (s_1, [x \mapsto 2^{f(\log_2 x)}; x \geq 2^{l_f}], s')$
 - $(s_1, [x \mapsto x; x \geq 0], s_2)$

- $(s_2, [x \mapsto \frac{x}{3}; x \geq 3 \cdot 2^{l_f}], s_2); (s_2, [x \mapsto \frac{x}{2}; x \geq 2 \cdot 2^{l_f}], s_2)$
- $(s_2, [x \mapsto x; x \geq 0], s_3)$
- $(s_3, [x \mapsto x; x > 2^{l_f}], s_4); (s_3, [x \mapsto x; x \geq 2^{l_f}], s_5)$

Note that s_4 is a deadlock state, hence player A loses the reachability game if s_4 is reached. Similarly, she wins if s_5 is reached.

The intuition is that the new energy variable x encodes the two old ones as $x = 2^{x_1} 3^{x_2}$. If player A wants to bring (S', T') from s to s' , and commits to this by taking the transition $s \rightarrow s_1$, she may be interrupted by player B taking the $s_1 \rightarrow s_2$ transition. Here player A has to prove that x_1 was really $\geq l_f$, by using the loops at s_2 to bring x to the precise value 2^{l_f} . If she manages this, then player B has only the $s_3 \rightarrow s_5$ transition available in s_3 , hence player A wins. Otherwise, player B wins.

The conversions of other types of transitions are similar. One can easily see that player A can reach a state in F in the original energy automaton (S, T) if, and only if, she can reach a state in F , or one of the new accepting states, in the new automaton (S', T') .

We miss to argue that all energy functions in (S', T') are piecewise affine. Looking at the defined modules, we see that this is the case except perhaps for the functions defined as $g_2(x) = 2^{f(\log_2 x)}$ and $g_3(x) = 3^{f(\log_3 x)}$. However, f is an integer update function, so that $f(x) = x + k$ for some $k \in \mathbb{Z}$; hence $g_2(x) = 2^k x$ and $g_3(x) = 3^k x$, which are indeed piecewise affine. \square

10 Conclusion

We have in this paper introduced a functional framework for modeling and analyzing energy problems, and we have seen that our framework encompasses most existing formal approaches to energy problems. In the first paper of this series [28], we have developed a theory of *-continuous Kleene ω -algebras in order to analyze energy problems algebraically.

We have seen here that the algebraic setting of *-continuous Kleene ω -algebras applies to energy functions and that it allows to solve reachability and Büchi acceptance problems in energy automata in a generic way. For the important class of piecewise affine energy functions, we have shown that reachability and Büchi acceptance are decidable in EXPTIME.

In the last part of this paper, we have seen that one quickly comes into trouble with undecidability if the class of energy functions is extended or if two-player games are considered. This may be remedied by considering *approximate* solutions instead, using notions of distances for energy automata akin to the ones in [35] to provide quantitative measures for similar energy behavior; this is future work.

In the two papers of this series, we have seen that *-continuous Kleene ω -algebras provide a natural generalization of continuous Kleene ω -algebras, much in the same way in which *-continuous Kleene algebras are a natural generalization of continuous Kleene algebras. We have left open a few algebraic problems, in particular a

characterization of the free (non-finitary) $*$ -continuous Kleene ω -algebras. We have seen that $*$ -continuous Kleene ω -algebras find a natural application in energy functions and energy problems, but we are confident that they will find numerous other applications. In honor of the late Zoltán Ésik, we propose to rename $*$ -continuous Kleene ω -algebras to “*Ésik algebras*”.

References

- [1] Parosh Aziz Abdulla, Ahmed Bouajjani, and Julien d’Orso. Monotonic and downward closed games. *J. Log. Comput.*, 18(1):153–169, 2008.
- [2] Rajeev Alur and David L. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2):183–235, 1994.
- [3] Rajeev Alur and Thomas A. Henzinger. A really temporal logic. *J. ACM*, 41(1):181–204, 1994.
- [4] Rajeev Alur, Salvatore La Torre, and George J. Pappas. Optimal paths in weighted timed automata. In Di Benedetto and Sangiovanni-Vincentelli [23], pages 49–62.
- [5] Gerd Behrmann, Ansgar Fehnker, Thomas Hune, Kim G. Larsen, Paul Pettersson, Judi Romijn, and Frits W. Vaandrager. Minimum-cost reachability for priced timed automata. In Di Benedetto and Sangiovanni-Vincentelli [23], pages 147–161.
- [6] Jean Berstel and Christophe Reutenauer. *Noncommutative Rational Series With Applications*. Cambridge Univ. Press, 2010.
- [7] Michael Blondin, Alain Finkel, Stefan Göller, Christoph Haase, and Pierre McKenzie. Reachability in two-dimensional vector addition systems with states is PSPACE-complete. In *LICS*, pages 32–43. IEEE, 2015.
- [8] Stephen L. Bloom and Zoltán Ésik. *Iteration Theories: The Equational Logic of Iterative Processes*. EATCS monographs on theoretical computer science. Springer-Verlag, 1993.
- [9] Rémi Bonnet, Alain Finkel, and M. Praveen. Extending the Rackoff technique to affine nets. In Deepak D’Souza, Telikepalli Kavitha, and Jaikumar Radhakrishnan, editors, *FSTTCS*, volume 18 of *Leibniz Int. Proc. Inf.*, pages 301–312. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.
- [10] Patricia Bouyer, Uli Fahrenberg, Kim G. Larsen, and Nicolas Markey. Timed automata with observers under energy constraints. In Karl Henrik Johansson and Wang Yi, editors, *HSCC*, pages 61–70. ACM, 2010.

- [11] Patricia Bouyer, Uli Fahrenberg, Kim G. Larsen, Nicolas Markey, and Jiří Srba. Infinite runs in weighted timed automata with energy constraints. In Franck Cassez and Claude Jard, editors, *FORMATS*, volume 5215 of *Lect. Notes Comput. Sci.*, pages 33–47. Springer-Verlag, 2008.
- [12] Patricia Bouyer, Kim G. Larsen, and Nicolas Markey. Lower-bound-constrained runs in weighted timed automata. *Perform. Eval.*, 73:91–109, 2014.
- [13] Patricia Bouyer, Kim G. Larsen, Nicolas Markey, and Jacob Illum Rasmussen. Almost optimal strategies in one clock priced timed games. In S. Arun-Kumar and Naveen Garg, editors, *FSTTCS*, volume 4337 of *Lect. Notes Comput. Sci.*, pages 345–356. Springer-Verlag, 2006.
- [14] Tomáš Brázdil, Petr Jančar, and Antonín Kučera. Reachability games on extended vector addition systems with states. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide, and Paul G. Spirakis, editors, *ICALP*, volume 6199 of *Lect. Notes Comput. Sci.*, pages 478–489. Springer-Verlag, 2010.
- [15] Romain Brenguier, Franck Cassez, and Jean-François Raskin. Energy and mean-payoff timed games. In Martin Fränzle and John Lygeros, editors, *HSCC*, pages 283–292. ACM, 2014.
- [16] David Cachera, Uli Fahrenberg, and Axel Legay. An omega-algebra for real-time energy problems. In Prahladh Harsha and G. Ramalingam, editors, *FSTTCS*, volume 45 of *Leibniz Int. Proc. Inf.*, pages 394–407. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.
- [17] Jakub Chaloupka. Z-reachability problem for games on 2-dimensional vector addition systems with states is in P. In Antonín Kučera and Igor Potapov, editors, *RP*, volume 6227 of *Lect. Notes Comput. Sci.*, pages 104–119. Springer-Verlag, 2010.
- [18] Tat-hung Chan. The boundedness problem for three-dimensional vector addition systems with states. *Inf. Proc. Letters*, 26(6):287–289, 1988.
- [19] Krishnendu Chatterjee and Laurent Doyen. Energy parity games. *Theor. Comput. Sci.*, 458:49–60, 2012.
- [20] Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, and Jean-François Raskin. Generalized mean-payoff and energy games. In Kamal Lodaya and Meena Mahajan, editors, *FSTTCS*, volume 8 of *Leibniz Int. Proc. Inf.*, pages 505–516, 2010.
- [21] Adrian Horia Dediu, Shunsuke Inenaga, and Carlos Martín-Vide, editors. *Language and Automata Theory and Applications - 5th International Conference, LATA 2011, Tarragona, Spain, May 26-31, 2011. Proceedings*, volume 6638 of *Lect. Notes Comput. Sci.* Springer-Verlag, 2011.

- [22] Aldric Degorre, Laurent Doyen, Raffaella Gentilini, Jean-François Raskin, and Szymon Toruńczyk. Energy and mean-payoff games with imperfect information. In Anuj Dawar and Helmut Veith, editors, *CSL*, volume 6247 of *Lect. Notes Comput. Sci.*, pages 260–274. Springer-Verlag, 2010.
- [23] Maria Domenica Di Benedetto and Alberto L. Sangiovanni-Vincentelli, editors. *Hybrid Systems: Computation and Control, 4th International Workshop, HSCC 2001, Rome, Italy, March 28-30, 2001, Proceedings*, volume 2034 of *Lect. Notes Comput. Sci.* Springer-Verlag, 2001.
- [24] Manfred Droste, Werner Kuich, and Heiko Vogler. *Handbook of Weighted Automata*. EATCS Monographs in Theoretical Computer Science. Springer-Verlag, 2009.
- [25] Zoltán Ésik, Uli Fahrenberg, and Axel Legay. $*$ -continuous Kleene ω -algebras. In Igor Potapov, editor, *DLT*, volume 9168 of *Lect. Notes Comput. Sci.*, pages 240–251. Springer-Verlag, 2015.
- [26] Zoltán Ésik, Uli Fahrenberg, and Axel Legay. $*$ -continuous Kleene ω -algebras for energy problems. In Ralph Matthes and Matteo Mio, editors, *FICS*, volume 191 of *Electr. Proc. Theor. Comput. Sci.*, pages 48–59, 2015.
- [27] Zoltán Ésik, Uli Fahrenberg, Axel Legay, and Karin Quaas. Kleene algebras and semimodules for energy problems. In Dang Van Hung and Mizuhito Ogawa, editors, *ATVA*, volume 8172 of *Lect. Notes Comput. Sci.*, pages 102–117. Springer-Verlag, 2013.
- [28] Zoltán Ésik, Uli Fahrenberg, Axel Legay, and Karin Quaas. An algebraic approach to energy problems I: $*$ -continuous Kleene ω -algebras. *Acta Cyb.*, 2017. In this issue.
- [29] Zoltán Ésik and Werner Kuich. A semiring-semimodule generalization of ω -regular languages, Parts 1 and 2. *J. Aut. Lang. Comb.*, 10:203–264, 2005.
- [30] Zoltán Ésik and Werner Kuich. *Modern Automata Theory*. 2007. <http://dmg.tuwien.ac.at/kuich/mat.pdf>.
- [31] Zoltán Ésik and Werner Kuich. On iteration semiring-semimodule pairs. *Semigroup Forum*, 75:129–159, 2007.
- [32] Zoltán Ésik and Werner Kuich. Finite automata. In *Handbook of Weighted Automata* [24], pages 69–104.
- [33] Uli Fahrenberg, Line Juhl, Kim G. Larsen, and Jiří Srba. Energy games in multiweighted automata. In Antonio Cerone and Pekka Pihlajasaari, editors, *ICTAC*, volume 6916 of *Lect. Notes Comput. Sci.*, pages 95–115. Springer-Verlag, 2011.

- [34] Uli Fahrenberg, Kim G. Larsen, and Axel Legay. Model-based verification, optimization, synthesis and performance evaluation of real-time systems. In Zhiming Liu, Jim Woodcock, and Huibiao Zhu, editors, *ICTAC Training School on Software Engineering*, volume 8050 of *Lect. Notes Comput. Sci.*, pages 67–108. Springer-Verlag, 2013.
- [35] Uli Fahrenberg and Axel Legay. The quantitative linear-time–branching-time spectrum. *Theor. Comput. Sci.*, 538:54–69, 2014.
- [36] Uli Fahrenberg, Axel Legay, and Karin Quaas. Büchi conditions for generalized energy automata. In Manfred Droste and Heiko Vogler, editors, *WATA*, page 47, 2012.
- [37] Jonathan S. Golan. *Semirings and their Applications*. Springer-Verlag, 1999.
- [38] Stefan Göller, Christoph Haase, Ranko Lazić, and Patrick Totzke. A polynomial-time algorithm for reachability in branching VASS in dimension one. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *ICALP*, volume 55 of *Leibniz Int. Proc. Inf.*, pages 105:1–105:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.
- [39] Line Juhl, Kim G. Larsen, and Jean-François Raskin. Optimal bounds for multiweighted and parametrised energy games. In Zhiming Liu, Jim Woodcock, and Huibiao Zhu, editors, *Theories of Programming and Formal Methods*, volume 8051 of *Lect. Notes Comput. Sci.*, pages 244–255. Springer-Verlag, 2013.
- [40] Dénes König. Über eine Schlussweise aus dem Endlichen ins Unendliche. *Acta Sci. Math. (Szeged)*, 3(2-3):121–130, 1927.
- [41] Dexter Kozen. On Kleene algebras and closed semirings. In Branislav Rován, editor, *MFCS*, volume 452 of *Lect. Notes Comput. Sci.*, pages 26–47. Springer-Verlag, 1990.
- [42] Jérôme Leroux. The general vector addition system reachability problem by Presburger inductive invariants. *Logical Meth. Comput. Sci.*, 6(3), 2010.
- [43] Jérôme Leroux. Vector addition system reachability problem: A short self-contained proof. In Dediu et al. [21], pages 41–64.
- [44] Richard J. Lipton. The reachability problem requires exponential space. Technical report, Department of Computer Science, Yale University, 1976.
- [45] Marvin L. Minsky. Recursive unsolvability of Post’s problem of “Tag” and other topics in theory of Turing machines. *Annals Math.*, 74(3):437–455, 1961.
- [46] Karin Quaas. On the interval-bound problem for weighted timed automata. In Dediu et al. [21], pages 452–464.
- [47] Charles Rackoff. The covering and boundedness problems for vector addition systems. *Theor. Comput. Sci.*, 6:223–231, 1978.

- [48] Yaron Velner, Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, Alexander Moshe Rabinovich, and Jean-François Raskin. The complexity of multi-mean-payoff and multi-energy games. *Inf. Comput.*, 241:177–196, 2015.