Energiautomater, energifunktioner og Kleene-algebra

Uli Fahrenberg Kim G. Larsen

École polytechnique, France

Aalborg Universitet, Denmark

NIK 2018

Energy Automata, Energy Functions, Kleene Algebra

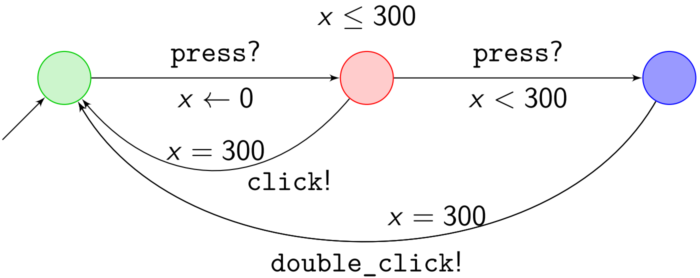
Uli Fahrenberg Kim G. Larsen

École polytechnique, France

Aalborg Universitet, Denmark

NIK 2018

Recall Timed Automata



Recall Timed Automata

Definition

The set $\Phi(C)$ of clock constraints ϕ over a finite set C is defined by the grammar

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

Definition

A timed automaton is a tuple $(L, \ell_0, C, \Sigma, I, E)$ consisting of a finite set L of locations, an initial location $\ell_0 \in L$, a finite set C of clocks, a finite set Σ of actions, a location invariants mapping $I : L \to \Phi(C)$, and a set $E \subseteq L \times \Phi(C) \times \Sigma \times 2^C \times L$ of edges.

Recall Timed Automata

- Useful for modeling synchronous real-time systems
- Reachability, emptiness, LTL model checking PSPACE-complete
- Universality undecidable
- Fast on-the-fly algorithms, using zones, for reachability, liveness, and Timed CTL model checking
- UppAal
- Extensions to weighted timed automata, real-time games, etc.

- This work: Energy problems in timed automata

Energy Constraints

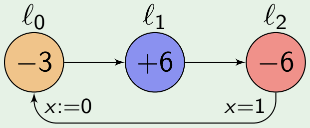
Energy is not only consumed, but can be regained.

$\leadsto$ "prices" can be negative;

$\leadsto$ the aim is to continuously satisfy cost constraints

$\leadsto$ in this paper, we focus on infinite runs.

Example



Energy Constraints

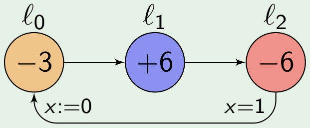
Energy is not only consumed, but can be regained.

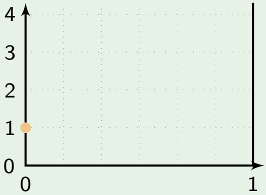
$\leadsto$ "prices" can be negative;

$\leadsto$ the aim is to continuously satisfy cost constraints

$\leadsto$ in this paper, we focus on infinite runs.

Example





lower-bound problem

Energy Constraints

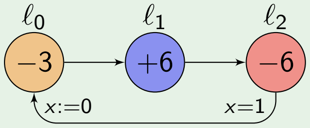
Energy is not only consumed, but can be regained.

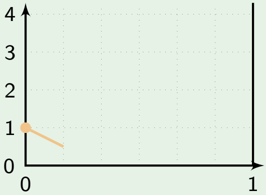
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-bound problem

Energy Constraints

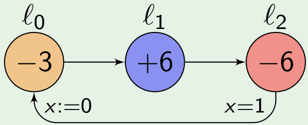
Energy is not only consumed, but can be regained.

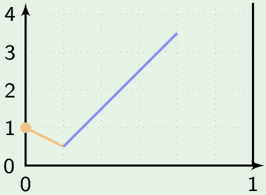
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-bound problem

Energy Constraints

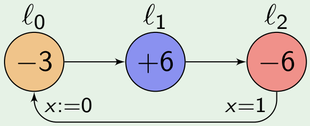
Energy is not only consumed, but can be regained.

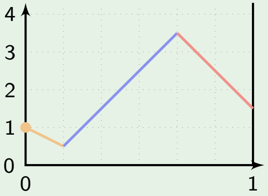
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-bound problem

Energy Constraints

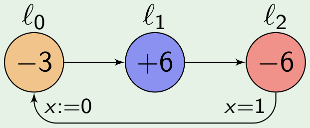
Energy is not only consumed, but can be regained.

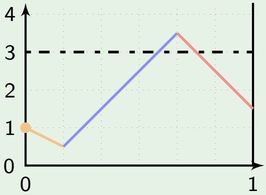
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





Energy Constraints

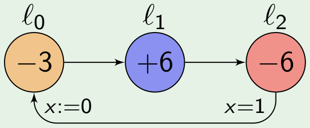
Energy is not only consumed, but can be regained.

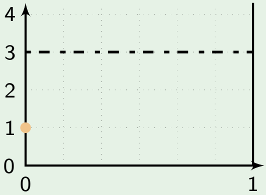
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

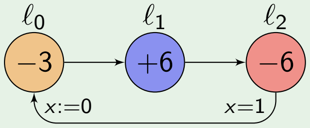
Energy is not only consumed, but can be regained.

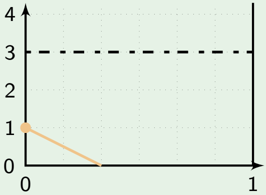
⇝ "prices" can be negative;

⇝ the aim is to continuously satisfy cost constraints

⇝ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

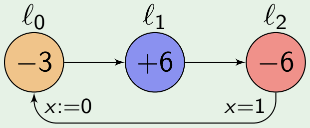
Energy is not only consumed, but can be regained.

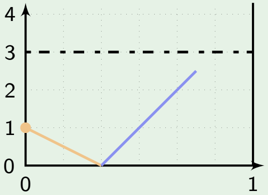
⇝ "prices" can be negative;

⇝ the aim is to continuously satisfy cost constraints

⇝ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

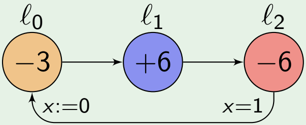
Energy is not only consumed, but can be regained.

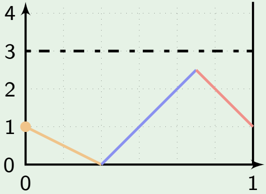
⇝ "prices" can be negative;

⇝ the aim is to continuously satisfy cost constraints

⇝ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

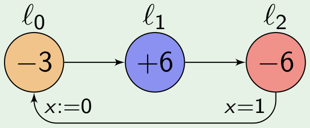
Energy is not only consumed, but can be regained.

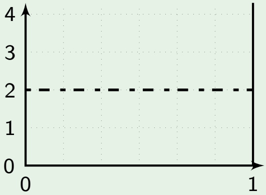
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

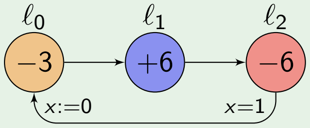
Energy is not only consumed, but can be regained.

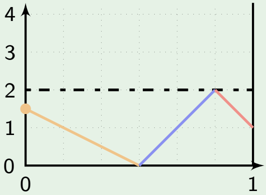
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

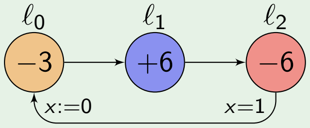
Energy Constraints

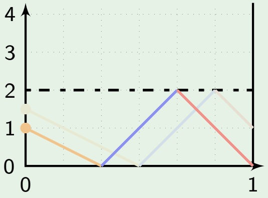
Energy is not only consumed, but can be regained.

⇝ “prices” can be negative;
⇝ the aim is to continuously satisfy cost constraints

⇝ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

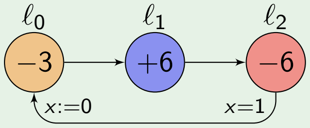
Energy is not only consumed, but can be regained.

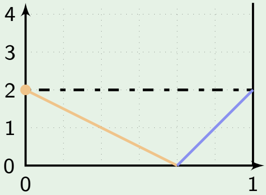
⇝ "prices" can be negative;

⇝ the aim is to continuously satisfy cost constraints

⇝ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

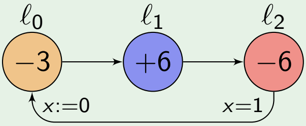
Energy is not only consumed, but can be regained.

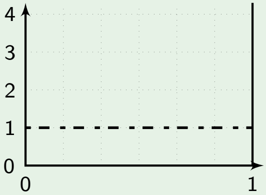
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Energy Constraints

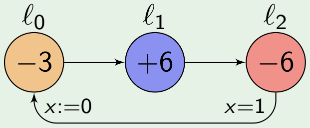
Energy is not only consumed, but can be regained.

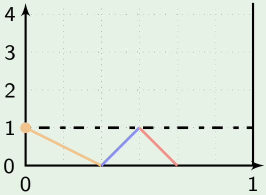
$\rightsquigarrow$ "prices" can be negative;

$\rightsquigarrow$ the aim is to continuously satisfy cost constraints

$\rightsquigarrow$ in this paper, we focus on infinite runs.

Example





lower-upper-bound problem

Problems

Definition:

- $\gamma = s_0 \xrightarrow{p_1} s_1 \xrightarrow{p_2} \cdots \xrightarrow{p_n} s_n$ a finite path in a weighted transition system
- $c \in \mathbb{R}_{\geq 0}$ initial credit
- $b \in \mathbb{R}_{\geq 0}$ (possible) upper bound
- accumulated cost of γ with initial credit c: $c + p_1 + p_2 + \dots$

Problems:

- lower bound: Find infinite run γ for which $c + p_1 + \dots + p_n \geq 0$ for all finite prefixes
- interval bound: Find infinite γ for which $c + p_1 + \dots + p_n \in [0, b]$ for all finite prefixes

Results

- First paper: FORMATS 2008 (170 citations for now)
- Lots of work since then, by lots of people
- Last paper for now: FM 2018 (Best Paper Award)
- My acknowledgements: Kim G. Larsen, Patricia Bouyer, Nicolas Markey[1], Jiří Srba, Zoltán Ésik†, David Cachera, Axel Legay, Pierre-Alain Reynier, Claus Thrane, Line Juhl, Giovanni Bacci

- lower-bound problem decidable for 1-clock WTA
  undecidable for 4-clock WTA
- interval problem undecidable for 2-clock WTA

- Applications in scheduling
  - of batch plants
  - of satellites

[1] also for some of the slides...

GOMSPACE: Scheduling of Nanosatellites Using UppAal

The Lower-Bound Problem for 1-Clock WTA

Theorem

For 1-clock WTA without weights on transitions, the lower-bound problem is solvable in polynomial time.

Proof Idea

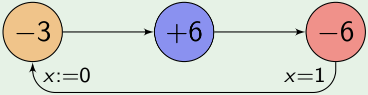
- can assume that delays within a region are elapsed in the most profitable location
- hence can use corner-point abstraction

Corner-point abstraction

Idea

Delays within a region are elapsed in the most profitable location.

Example

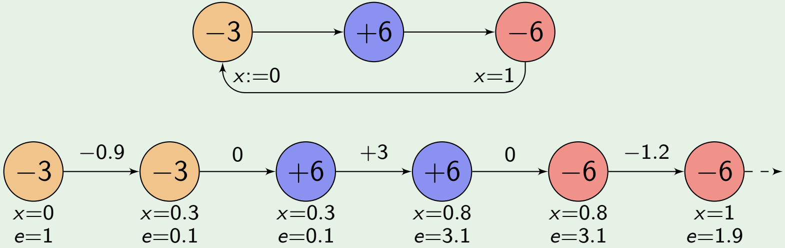


Corner-point abstraction

Idea

Delays within a region are elapsed in the most profitable location.

Example

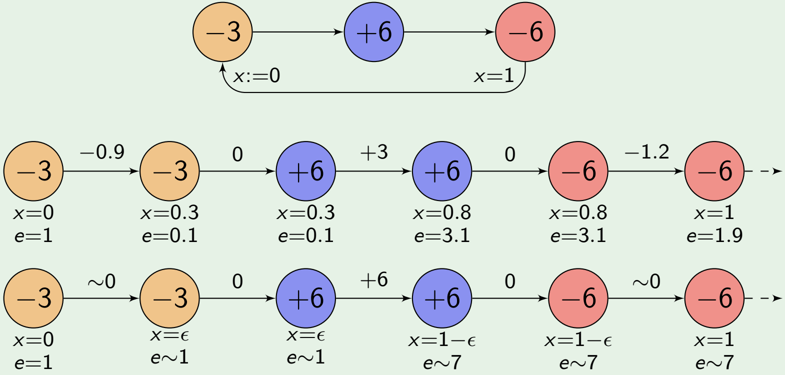


Corner-point abstraction

Idea

Delays within a region are elapsed in the most profitable location.

Example

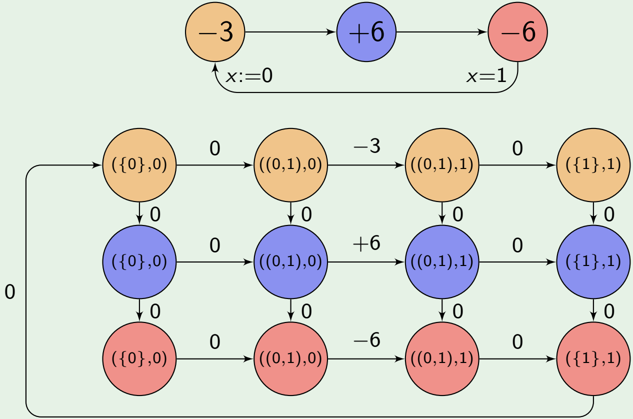


Corner-point abstraction

Idea

Delays within a region are elapsed in the most profitable location.

Example



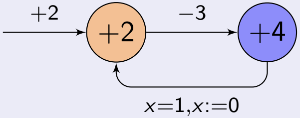
Corner-point abstraction

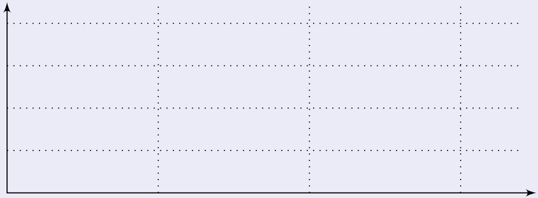
Idea

Delays within a region are elapsed in the most profitable location.

Remark

The corner-point abstraction is not correct if discrete transitions are weighted:





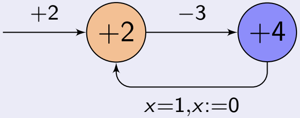
Corner-point abstraction

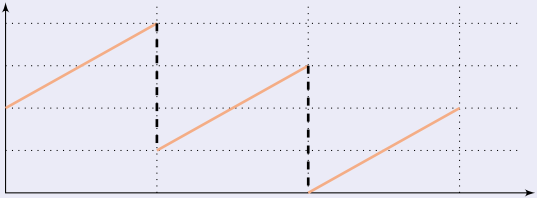
Idea

Delays within a region are elapsed in the most profitable location.

Remark

The corner-point abstraction is not correct if discrete transitions are weighted:





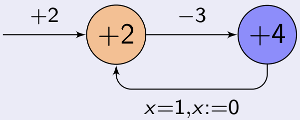
Corner-point abstraction

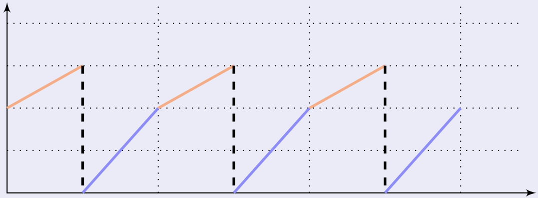
Idea

Delays within a region are elapsed in the most profitable location.

Remark

The corner-point abstraction is not correct if discrete transitions are weighted:





The Lower-Bound Problem for 1-Clock WTA

Theorem (FORMATS 2008)

For 1-clock WTA without weights on transitions, the lower-bound problem is solvable in polynomial time.

- corner-point abstraction

Theorem (HSCC 2010)

For 1-clock WTA with weights on transitions, the lower-bound problem is solvable in double-exponential time.

- completely different method, introducing energy functions

The Interval Problem for 1-Clock WTA
(Work in Progress)

Definition

An interval timed automaton $A = (L, E, I, F, r)$ consists of a finite set L of locations, a finite set $E \subseteq L \times \mathbb{Q}^3 \times L$ of transitions, subsets $I, F \subseteq L$ of initial and accepting locations, and weight rates $r : L \to \mathbb{Q}$.

- transitions $l \xrightarrow[{[a,b]}]{p} l'$: $[a, b]$ interval bound; p price
- spend some time in location l; take transition if $x \in [a, b]$; add p to x
- runs have initial energy and initial time budget
- can only spend time budget: no resets
  - almost a 1-clock WTA, but not quite

Interval Time Relations

- A basic interval timed automaton

$$l \xrightarrow[{[a,b]}]{p} l'$$

defines a relation

$$R = \{(x, t, x') \mid a \le x + r(l)\, t \le b, x' = x + r(l)\, t + p\}$$

- These can be composed:

$$l \xrightarrow[{[a,b]}]{p} l' \xrightarrow[{[c,d]}]{q} l''$$

corresponds to

$$R_1 \triangleright R_2 = \{(x_0, t_1 + t_2, x_2) \mid \exists x_1 : (x_0, t_1, x_1) \in R_1, (x_1, t_2, x_2) \in R_2\}\,.$$

Theorem

With operations $\cup$ and $\triangleright$, relations as above form an idempotent semiring.

The Algebraic Approach to Energy Problems, I

Let $\mathcal{Q} = \mathbb{Q}^\infty \times \mathbb{Q}^\infty_{\ge 0} \times \mathbb{Q}^\infty$: the set of interval timed relations

- together with operations $\cup$ (addition) and $\rhd$ (multiplication)

$\mathcal{Q}$ forms an idempotent semiring:

- $\cup$ is associative & commutative, with unit $\emptyset$
- $\rhd$ is associative, with unit $\mathrm{id}(x) = x$
- $\rhd$ distributes over $\cup$; $x \rhd \emptyset = \emptyset \rhd x = \emptyset$ for all x
- $x \cup x = x$ for all x

$\mathcal{Q}$ forms a continuous Kleene algebra:

- for all $Y \subseteq \mathcal{Q}$ and $x, z \in \mathcal{Q}$, $\bigcup Y$ exists and

$$x \rhd \left(\bigcup Y \right) \rhd z = \bigcup x \rhd Y \rhd z$$

The Algebraic Approach to Energy Problems, II

Let $n \geq 1$. $\mathcal{Q}^{n \times n}$: the semiring of $n \times n$ matrices over $\mathcal{Q}$

- with matrix addition $\cup$ and matrix multiplication $\rhd$

$\mathcal{Q}^{n \times n}$ is again a continuous Kleene algebra

- with $M^*_{i,j} = \bigcup_{m \geq 0} \bigcup_{1 \leq k_1, \dots, k_m \leq n} M_{i,k_1} M_{k_1,k_2} \cdots M_{k_m,j}$
- and for $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$,

$$M^* = \begin{bmatrix} (a \cup bd^*c)^* & (a \cup bd^*c)^* bd^* \\ (d \cup ca^*b)^* ca^* & (d \cup ca^*b)^* \end{bmatrix}$$

(recursively; "generalized Floyd-Warshal")

The Algebraic Approach to Energy Problems, III

$A = (\alpha, M, \kappa)$ an interval timed automaton

- $\alpha \in \{\emptyset, \mathrm{id}\}^n$ initial vector, $\kappa \in \{\emptyset, \mathrm{id}\}^n$ accepting vector, $M \in \mathcal{Q}^{n \times n}$ transition matrix
- finite path $s_i \xrightarrow{w_0} \cdots \xrightarrow{w_n} s_j$ accepting if $\alpha_i = \kappa_j = \mathrm{id}$
- finite behavior of A:

$$|A| = \bigvee \left\{ w_0 \cdots w_n \mid s_i \xrightarrow{w_0} \cdots \xrightarrow{w_n} s_j \text{ accepting finite path} \right\}$$

Theorem: $|A| = \alpha M^* \kappa$

The Algebraic Approach to Energy Problems, IV

Let $\mathcal{V} = \mathbb{Q}^\infty \times \mathbb{Q}_{\geq 0}^\infty$: interval timed relations without output

- for infinite runs
- with operation $\cup$ and unit $\emptyset$, $\mathcal{V}$ forms a commutative idempotent monoid

left $\mathcal{Q}$-action $\mathcal{Q} \times \mathcal{V} \to \mathcal{V}$: $(R, U) \mapsto R \triangleright U$

- $(\mathcal{Q}, \mathcal{V})$ semiring-semimodule pair

infinite product $\mathcal{Q}^\omega \to \mathcal{V}$: for $R_0, R_1, \ldots \in \mathcal{Q}$, define

$$\prod R_n = \Big\{(x, t) \;\Big|\; \exists x_0, x_1, \ldots \in \mathbb{Q}^\infty, t_1, t_2, \ldots \in \mathbb{Q}_{\geq 0}^\infty : \sum_{n=0}^{\infty} t_n = t, \forall n \geq 0 : (x_n, t_{n+1}, x_{n+1}) \in R_n\Big\}$$

- $(\mathcal{Q}, \mathcal{V})$ continuous Kleene ω-algebra

The Algebraic Approach to Energy Problems, V

$(\mathcal{Q}, \mathcal{V})$ continuous Kleene ω-algebra:

- $\mathcal{Q}$ continuous Kleene algebra; $\mathcal{V}$ complete lattice
- $\mathcal{Q}$-action on $\mathcal{V}$ preserves all suprema: $x \triangleright \left(\bigcup Y\right) \triangleright u = \bigcup x \triangleright Y \triangleright u$
- and three axioms for the infinite product:
  - For all $x_0, x_1, \ldots \in \mathcal{Q}$, $\prod x_n = x_0 \prod x_{n+1}$.
  - Let $x_0, x_1, \ldots \in \mathcal{Q}$ and $0 = n_0 \le n_1 \le \cdots$ a sequence which increases without a bound. Let $y_k = x_{n_k} \cdots x_{n_{k+1}-1}$ for all $k \ge 0$. Then $\prod x_n = \prod y_k$.
  - For all $X_0, X_1, \ldots \subseteq \mathcal{Q}$, $\prod(\bigvee X_n) = \bigvee\{\prod x_n \mid x_n \in X_n, n \ge 0\}$.

The Algebraic Approach to Energy Problems, VI

$(\mathcal{Q}^{n\times n}, \mathcal{V}^n)$ is again a continuous Kleene ω-algebra

- with $M_i^\omega = \bigcup_{1\le k_1,k_2,\dots\le n} M_{i,k_1} M_{k_1,k_2} \cdots$
- and for $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$,

$$M^\omega = \begin{bmatrix} (a \cup bd^* c)^\omega \cup (a \cup bd^* c)^* bd^\omega \\ (d \cup ca^* b)^\omega \cup (d \cup ca^* b)^* ca^\omega \end{bmatrix}$$

(recursively)

The Algebraic Approach to Energy Problems, VII

$A = (\alpha, M, \kappa)$ an interval timed automaton

- infinite path $s_i \xrightarrow{w_0}\xrightarrow{w_1} \cdots$ accepting if $\alpha_i = \text{id}$ and some s_j with $\kappa_j = \text{id}$ is visited infinitely often
- Büchi behavior of A:

$$\|A\| = \bigvee \{ \prod w_n \mid s_i \xrightarrow{w_0}\xrightarrow{w_1} \cdots \text{accepting infinite path} \}$$

- Re-order states so that $\kappa = (\text{id}, \dots, \text{id}, \emptyset, \dots, \emptyset)$
- i.e. the first $k \le n$ states are accepting

Theorem: with $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ for $a \in S^{k \times k}$, $\|A\| = \alpha \begin{bmatrix} (a + bd^* c)^\omega \\ d^* c(a + bd^* c)^\omega \end{bmatrix}$

Conclusion

- Formal methods for solving energy problems
- Applications in scheduling
- Continuous Kleene ω-algebras: obscure algebraic theory with real-world applications!

The work on the interval problem presented here is only half complete: we've found a nice algebraic setting; but we've said nothing about actual computations

- See our FM 2018 paper for actual computations in a restricted setting ("segmented energy timed automata")
- Rest is future work