Constraint Automata on Infinite Data Trees: From CTL($\mathbb{Z}$)/CTL*($\mathbb{Z}$) To Decision Procedures

Stéphane Demri

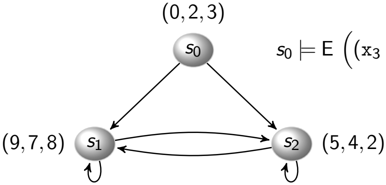
Joint work with Karin Quaas
(Leipzig University, Germany)

GT MCS & CDS, February 2024

Temporal Logics with Arithmetical Constraints

- Atomic formulae are constraints over the relational structure $(\mathbb{Z}, <, =, (=_{\mathfrak{d}})_{\mathfrak{d} \in \mathbb{Z}})$.

- Branching-time temporal logics CTL($\mathbb{Z}$) and CTL*($\mathbb{Z}$).



$s_0 \models \mathsf{E}\left((\mathtt{x}_3 < \mathsf{X}\mathtt{x}_3) \wedge (\mathsf{GF}(\mathtt{x}_1 < \mathsf{X}\mathtt{x}_2) \wedge \mathsf{GF}(\mathtt{x}_3 > \mathsf{X}\mathtt{x}_1))\right)$

$\mathtt{x}_1 < \mathsf{X}\mathtt{x}_2$: "current value of $\mathtt{x}_1$ is smaller than the value of $\mathtt{x}_2$ at the neXt position".

- We study the complexity of the satisfiability problem for CTL($\mathbb{Z}$) and CTL*($\mathbb{Z}$), respectively.

CTL($\mathbb{Z}$)/CTL*($\mathbb{Z}$)

- Terms and Boolean constraints.

$$\mathtt{t} := \mathtt{x} \mid \mathsf{X}\mathtt{t}, \qquad \Theta := \mathtt{t} < \mathtt{t}' \mid \mathtt{t} = \mathtt{t}' \mid \mathtt{t} = \mathfrak{d} \mid \neg\Theta \mid \Theta \wedge \Theta$$

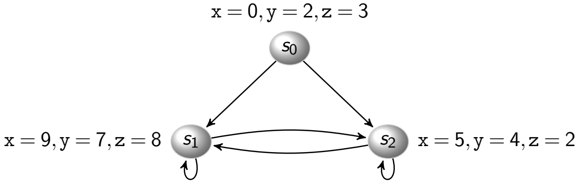
- CTL*($\mathbb{Z}$) state and path formulae

$$\phi := \neg\phi \mid \phi \wedge \phi \mid \mathsf{E}\Phi, \qquad \Phi := \phi \mid \Theta \mid \neg\Phi \mid \Phi \wedge \Phi \mid \mathsf{X}\Phi \mid \Phi\mathsf{U}\Phi$$

- CTL($\mathbb{Z}$) formulae

$$\phi := \mathsf{E}\ \Theta \mid \mathsf{A}\ \Theta \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathsf{EX}\phi \mid \mathsf{E}\phi\mathsf{U}\phi \mid \mathsf{AX}\phi \mid \mathsf{A}\phi\mathsf{U}\phi$$

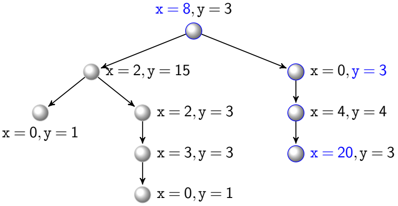
- Models



About Formulae

- Constraints Θ always in the scope of a path quantifier.

- $\mathsf{E}(\mathsf{X}y < x < \mathsf{XXX}x)$.



- Propositional variables can be encoded by $\mathsf{E}(x = 0)$.

- On all paths, infinitely often the value of the variable x is strictly smaller than the value of x at the next position.

$$\mathsf{AGF}(x < \mathsf{X}x)$$

- No finite model property, see e.g. $\mathsf{EG}(x < \mathsf{X}x)$.

Landmarks Results About CTL$^*(\mathbb{Z})$

- Satisfiability problem for CTL* is 2EXPTIME-complete.
  See e.g. [Vardi & Stockmeyer, STOC'85]

- Satisfiability problem for CTL$^*(\mathbb{Z})$ is decidable.
  See EHD-approach in [Carapelle & Kartzow & Lohrey, JCSS 2016]

- Model-checking problem for CTL$(\mathbb{Z})$ is undecidable.
  See e.g. [Mayr & Totzke, FI 2016]

- Satisfiability problem for existential/universal fragments of CTL* with gap-order constraints in PSPACE.
  [Bozzelli & Pinchinat, VMCAI'12]

- Satisfiability problem for LTL$(\mathbb{Z})$ is PSPACE-complete.
  [Demri & Gascon,TCS 2008]

Main Results

- The nonemptiness problem for tree constraint automata with Rabin acceptance is ExpTime-complete.

- Satisfiability problem for CTL($\mathbb{Z}$) is ExpTime-complete.

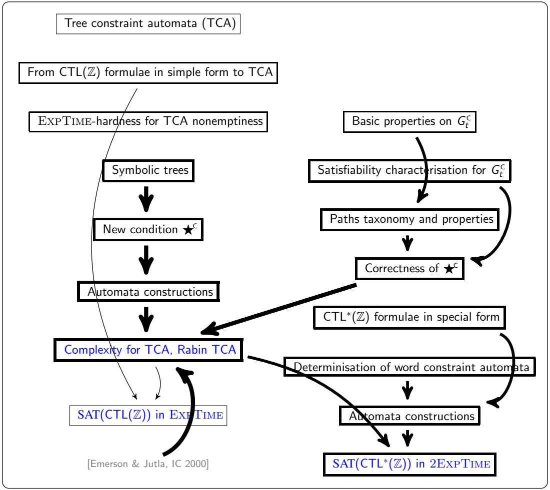
- Satisfiability problem for CTL*($\mathbb{Z}$) is 2ExpTime-complete.

- Intermediate results of more general interest:
  - Intersection construction for tree constraint automata with Rabin acceptance.
  - Determinisation of word constraint automata with Büchi acceptance extending standard Safra's construction.
  - Technical refinements from [Labai & Ortiz & Šimkus, KR'20].

Automata-Based Approach (see e.g., [Vardi & Wolper, IC 1994])

- Tree model property and formulae in simple form and special form, respectively.
- Translate the input formula ϕ into a tree constraint automaton $\mathbb{A}_\phi$ such that ϕ is satisfiable iff $\mathrm{L}(\mathbb{A}_\phi) \neq \emptyset$.
- Infer the complexity of the satisfiability problem from the complexity of deciding $\mathrm{L}(\mathbb{A}_\phi) \neq \emptyset$.

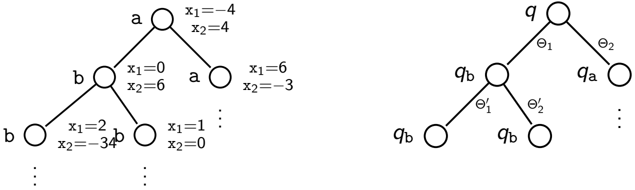
Organisation of the Results



Stéphane Demri, Karin Quaas:
Constraint Automata on Infinite Data Trees: From CTL(Z)/CTL*(Z) To Decision Procedures.
CoRR abs/2302.05327 (2023)

Tree Constraint Automata (TCA)

(or how to recognize infinite data trees)



- $\mathbb{A}$ accepts infinite trees $\mathbf{t} : [0, D-1]^* \to \Sigma \times \mathbb{Z}^\beta$.

- Transitions $(q, \mathtt{a}, (\Theta_1, q_1), \ldots, (\Theta_D, q_D))$ put constraints on values of current node and children nodes.

- Büchi and Rabin acceptance conditions.

- TCA also generalize word constraint automata.

Nonemptiness for TCA is ExpTime-complete

- Both for Büchi acceptance and Rabin acceptance conditions.

- ExpTime-hardness by reduction from the acceptance problem for alternating Turing machines running in polynomial-space.

- ExpTime-membership by reduction to nonemptiness of Rabin tree automata (over finite alphabet).

- Nonemptiness of Rabin tree automata is
  - polynomial in the number of transitions,
  - exponential in the number of Rabin acceptance pairs.

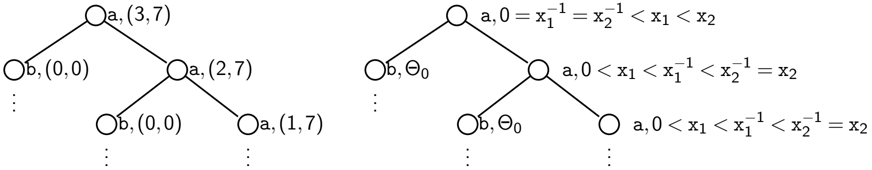
[Emerson & Jutla, IC 2000]

- In the reduction from TCA to Rabin tree automata we need to take care of these parameters.

Symbolic Trees

- Tuples $(z_1, \ldots, z_\beta) \in \mathbb{Z}^\beta$ represented by types (maximal consistent sets of constraints).

- Set of types for $\mathtt{x}_1, \ldots, \mathtt{x}_\beta, \mathtt{x}_1^{-1}, \ldots, \mathtt{x}_\beta^{-1}$ and constants in $[\mathfrak{d}_1, \mathfrak{d}_\alpha]$ is a finite alphabet.

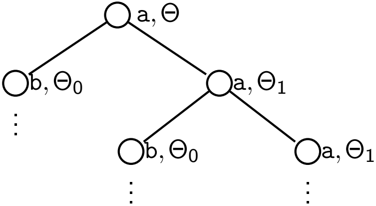


- Symbolic tree $t : [0, D-1]^* \rightarrow \Sigma \times \mathrm{STypes}(\beta, \mathfrak{d}_1, \mathfrak{d}_\alpha)$. $\mathrm{STypes}(\beta, \mathfrak{d}_1, \mathfrak{d}_\alpha)$ = set of satisfiable types.

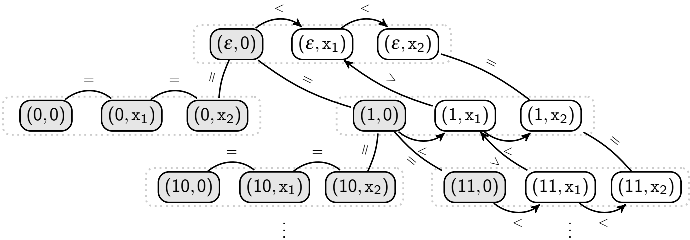
Satisfiable Symbolic Trees

- A symbolic tree t is satisfiable if there is a tree $\mathbf{t} : [0, D-1]^* \to \mathbb{Z}^\beta$ satisfying the constraints from t.
- A symbolic tree t is $\mathbb{A}$-consistent if it can be accepted symbolically by $\mathbb{A}$ and it is locally consistent.
- $\mathbb{A}$-consistency is a regular property captured by a Büchi tree automaton $A_{\mathrm{cons}(\mathbb{A})}$.
- $\mathrm{L}(\mathbb{A}) \neq \emptyset$ iff there is a satisfiable (and $\mathbb{A}$-consistent) symbolic tree.
- ... but the set of satisfiable symbolic trees is not regular.

Symbolic Graph G_t^c



$$\Theta \quad : \quad 0 = \mathtt{x}_1^{-1} = \mathtt{x}_2^{-1} < \mathtt{x}_1 < \mathtt{x}_2$$
$$\Theta_0 \quad : \quad 0 = \mathtt{x}_1 = \mathtt{x}_2 < \mathtt{x}_1^{-1} < \mathtt{x}_2^{-1}$$
$$\Theta_1 \quad : \quad 0 < \mathtt{x}_1 < \mathtt{x}_1^{-1} < \mathtt{x}_2^{-1} = \mathtt{x}_2$$



- Constant nodes $(\mathbf{n}, \mathfrak{d}_1)$ and $(\mathbf{n}, \mathfrak{d}_\alpha)$ and other nodes $(\mathbf{n}, \mathtt{x}_i)$.

(above $0 = \mathfrak{d}_1 = \mathfrak{d}_\alpha$)

Characterisation of Satisfiability

- Strict length $\mathrm{slen}(\pi)$ of

$$\pi = (\mathbf{n}_0, \mathtt{xd}_0) \xrightarrow{\sim_1} (\mathbf{n}_1, \mathtt{xd}_1) \cdots \xrightarrow{\sim_n} (\mathbf{n}_n, \mathtt{xd}_n)$$

  in G_t^{C} is the number of edges labelled by '$<$' in π.

- $\mathrm{slen}((\mathbf{n}, \mathtt{xd}), (\mathbf{n}', \mathtt{xd}'))$, is the supremum of all the strict lengths of paths from $(\mathbf{n}, \mathtt{xd})$ to $(\mathbf{n}', \mathtt{xd}')$.

- t is satisfiable iff for all $(\mathbf{n}, \mathtt{xd}), (\mathbf{n}', \mathtt{xd}')$,

$$\mathrm{slen}((\mathbf{n}, \mathtt{xd}), (\mathbf{n}', \mathtt{xd}')) < \omega.$$

(can be simplified a bit)

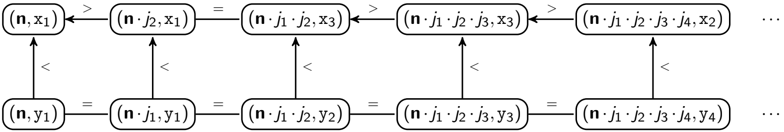
- This type of characterisation (re)proved many times in the literature. See e.g. [Čerāns, ICALP'94].

Condition ($\bigstar^{\mathsf{c}}$)

- ($\bigstar^{\mathsf{c}}$): variant of ($\bigstar$) from [Labai & Ortiz & Šimkus, KR'20].

- For every regular locally consistent symbolic tree t, G_t^{c} satisfies ($\bigstar^{\mathsf{c}}$) iff t is satisfiable.

- Condition ($\bigstar^{\mathsf{c}}$): for no branch $\mathscr{B} = \mathbf{n} \cdot j_1 \cdot j_2 \cdot j_3 \cdots$, a subgraph similar to the one below can be found:



- Condition ($\bigstar^{\mathsf{c}}$) is a regular property.

Reduction from TCA to Rabin Tree Automata

- $A_{\mathrm{cons}(\mathbb{A})}$: accepts all the $\mathbb{A}$-consistent symbolic trees.

- Construction of the Rabin tree automata $A_{\bigstar C}$ in several steps:

  - the number of Rabin pairs is bounded by $8(\beta+2)^2+3$,
  - the number of locations is exponential in β,
  - the transition relation can be decided in polynomial-time in

$$\max(\lceil log(|\mathfrak{d}_1|)\rceil, \lceil log(|\mathfrak{d}_\alpha|)\rceil) + \beta + \mathrm{card}(\Sigma) + D.$$

- $\mathrm{L}(\mathbb{A}) \neq \emptyset$ iff $\mathrm{L}(A_{\mathrm{cons}(\mathbb{A})}) \cap \mathrm{L}(A_{\bigstar C}) \neq \emptyset$.

- Nonemptiness for TCA can be solved in time
  - exponential in D and β for Büchi TCA,
  - exponential in D, β, and the number of Rabin acceptance pairs for Rabin TCA,

  hence in EXPTIME.

Main Steps to Decide $\mathrm{SAT}(\mathbf{CTL}^*(\mathbb{Z}))$ in 2ExpTime

- $\mathsf{CTL}^*(\mathbb{Z})$ formulae in special form

  $$\mathsf{E}\ (\mathtt{x}_1 = 0)\ \wedge\ \Big(\bigwedge_{i\in[1,D-1]} \mathsf{AGE}\ \Phi_i\Big)\ \wedge\ \Big(\bigwedge_{j\in[1,D']} \mathsf{A}\ \Phi'_j\Big)$$

  with $\mathsf{LTL}(\mathbb{Z})$ formulae Φ_i's and Φ'_j's

- Given $\mathsf{LTL}(\mathbb{Z})$ formula Φ in simple form, there is an equivalent deterministic Rabin word constraint automaton $\mathbb{A}_\Phi$ with
  - number of locations double-exponential in $\mathtt{size}(\Phi)$,
  - number of Rabin pairs exponential in $\mathtt{size}(\Phi)$,
  - $+$ quantitative properties useful for final complexity analysis.

  (extension of Safra's construction)

- Rabin TCA $\mathbb{A}$ such that

  $$\mathrm{L}(\mathbb{A}) = \mathrm{L}(\mathbb{A}_0) \bigcap_{i\in[1,D-1]} \mathrm{L}(\mathbb{A}_i) \bigcap_{j\in[1,D']} \mathrm{L}(\mathbb{A}'_j),$$

- $+$ final complexity analysis to get 2ExpTime.

Low-Hanging Fruits

- $\mathrm{SAT}(\mathsf{CTL}^*(\mathbb{N}))$ in $2\textsc{ExpTime}$.
- $\mathrm{SAT}(\mathsf{CTL}^*(\Sigma^*, <_{\mathsf{pre}}, =, =_{\mathfrak{w}})_{\mathfrak{w} \in \Sigma^*}))$ in $2\textsc{ExpTime}$.
- Description logics with concrete domains $\mathbb{N}$ or $\mathbb{Z}$.

Open Problem Related to Gap-Order Constraints

- Gap-order constraint "$\mathtt{x} - \mathtt{y} \geq k$".
- How to add gap-order constraints to CTL($\mathbb{Z}$) and CTL*($\mathbb{Z}$) without jeopardizing decidability/complexity?
- Tree constraints automata with gap-order constraints in a restricted way?

Concluding Remarks

- Use of tree constraint automata accepting infinite data trees to solve $\mathrm{SAT}(\mathsf{CTL}(\mathbb{Z}))$ and $\mathrm{SAT}(\mathsf{CTL}^*(\mathbb{Z}))$ (with constraints).
- Sharp complexity upper bounds for $\mathsf{CTL}(\mathbb{Z})$ and $\mathsf{CTL}^*(\mathbb{Z})$.
- Similar results for $(\Sigma^*, <_{\mathsf{pre}}, =, (=_{\mathfrak{w}})_{\mathfrak{w} \in \Sigma^*})$ and for description logics (see previous GT FM & AI).
- Open problems: other logics, other relational structures ...