

Model checking memoryful logics over one-counter automata

S. Demri¹ R. Lazić³ A. Sangnier^{1,2}

¹LSV, ENS Cachan, CNRS, INRIA Saclay & ²EDF R&D

³Department of Computer Science
University of Warwick, UK

Dagstuhl "Beyond the finite" – April 2008

Data Words/Trees

- Timed word [Alur & Dill, TCS 94]

<i>a</i>	<i>b</i>	<i>c</i>	<i>a</i>	<i>a</i>	<i>b</i>
0	0.3	1	2.3	3.5	3.51

- Runs for infinite-state systems [Minsky, 67]

q_0	q_2	q_3	q_2	q_3	q_2
0	0	1	2	3	4

- Integer arrays [Habermehl & Josif & Vojnar, FOSSACS 08]

$t[0]$ $t[1]$ $t[2]$ $t[3]$ $t[4]$ $t[5]$...

- Abstract data words [Bouyer & Petit & Thérien, IC 03]

- Data trees for XML documents

[Bojanczyk et al, PODS 06; Jurdzinski & Lazić, LICS 07]

Specifying classes of data words

- Register automata
 - Register automata [Kaminski & Francez, TCS 94]
 - Data automata [Bouyer & Petit & Thérien, IC 03]
 - See the survey [Segoufin, CSL 06]
- First-order languages [Bojańczyk et al., LICS 06]
- Temporal logics
 - Real-time logic TPTL [Alur & Henzinger, JACM 94]
 - LTL with freeze [D. & Lazić & Nowak, TIME 05]
- Many other formalisms
 - Rewriting systems with data [Bouajjani et al., FCT 07]
 - Hybrid logics [Schwentick & Weber, STACS 07]
 - ...

Motivations

- To analyze runs of operational models with focus on data values.
- Model-checking instead of satisfiability as done earlier.
- Our choices in this work:
 - Specification language: memoryful linear-time logic. (FO and LTL)
 - Operational models: one-counter automata (1CA)
 - Simple model but memoryful logics are expressive.
 - Numerous problems are decidable for 1CA.

One-counter automata (1CA)

- $\mathcal{A} = \langle Q, q_I, \delta, F \rangle$
 - Finite set of locations Q and initial location q_I ,
 - Set of accepting locations $F \subseteq Q$,
 - Transition relation $\delta \subseteq Q \times \{\text{inc}, \text{dec}, \text{ifzero}\} \times Q$.
- Runs are of the form

$$\rho = \begin{array}{ccccccc} q_0 = q_I & \rightarrow & q_1 & \rightarrow & q_2 & \rightarrow & \dots \\ n_0 = 0 & & n_1 & & n_2 & & \end{array}$$

- Accepting conditions:
 - Last location is accepting (finite runs).
 - Büchi acceptance condition (infinite runs).

LTL with registers

Syntax

$$\phi ::= q \mid \uparrow_r \mid \neg\phi \mid \phi \wedge \phi \mid \phi \cup \phi \mid \mathbb{X}\phi \mid \downarrow_r \phi$$

- **Models:**

q_0	q_2	q_3	q_2	q_3	q_2
0	0	1	2	3	4

- **Register valuation** v : (partial) map from registers to \mathbb{N} .

Satisfaction relation

$\sigma, i \models_v q$	$\stackrel{\text{def}}{\Leftrightarrow}$	$\sigma(i)$ has location q
$\sigma, i \models_v \uparrow_r$	$\stackrel{\text{def}}{\Leftrightarrow}$	$\sigma(i)$ has counter value $v(r)$
$\sigma, i \models_v \mathbb{X}\phi$	$\stackrel{\text{def}}{\Leftrightarrow}$	$i + 1 < \sigma $ and $\sigma, i + 1 \models_v \phi$
$\sigma, i \models_v \downarrow_r \phi$	$\stackrel{\text{def}}{\Leftrightarrow}$	$\sigma, i \models_{v[r \mapsto i]} \phi$

Examples

- There is a suffix such that all counter values are different

$$FG(\downarrow_1 \ XG\neg \uparrow_1)$$

$$\begin{array}{ccccccc} q_0 & q_2 & q_3 & q_2 & q_3 & q_2 & q_2 \dots \\ 0 & 0 & 1 & 2 & 3 & 4 & 5 \dots \end{array}$$

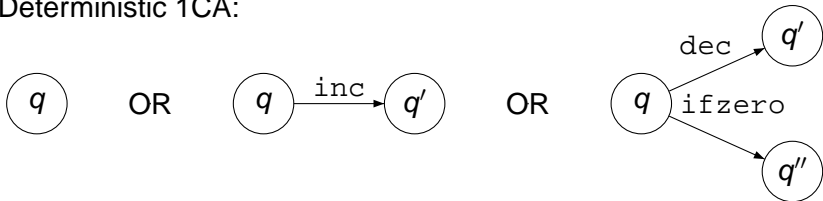
- Whenever location q is reached with current counter value n and next current counter value m , if there is a next occurrence of q , the two consecutive counter values are also n and m

$$G(q \Rightarrow \downarrow_1 \ X \downarrow_2 \ XG(q \Rightarrow \uparrow_1 \ \wedge X \uparrow_2))$$

$$\begin{array}{ccccccc} q & q' & q' & q & q' & q'' & q'' \dots \\ 50 & 60 & 1 & 50 & 60 & 4 & 5 \dots \end{array}$$

Model checking problems

- Finitary model-checking $\text{MC}(\text{LTL})^{<\omega}$:
 \exists finite accepting run ρ of \mathcal{A} such that $\rho, 0 \models \phi$?
- Infinitary model-checking $\text{MC}(\text{LTL})^\omega$:
 \exists infinite accepting run ρ of \mathcal{A} such that $\rho, 0 \models \phi$?
- Subproblems
 - $\text{MC}(\text{LTL})_n^\alpha$: restriction to n registers.
 - $\text{PureMC}(\text{LTL})^\alpha$: restriction without location.
- Deterministic 1CA:



FO over data words [Bojanczyk et al., LICS 06]

- Formulae in $\text{FO}^\Sigma(\sim, <, +1)$:

$$\phi ::= a(\mathbf{x}) \mid \mathbf{x} \sim \mathbf{y} \mid \mathbf{x} < \mathbf{y} \mid \mathbf{x} = \mathbf{y} + 1 \mid \dots \mid \exists \mathbf{x} \phi$$

$$(a \in \Sigma)$$

- Satisfaction relation:

$$\sigma \models_v \mathbf{x} \sim \mathbf{y} \stackrel{\text{def}}{\iff} v(\mathbf{x}), v(\mathbf{y}) \text{ are defined and } v(\mathbf{x}) \sim^\sigma v(\mathbf{y})$$

- Standard translation

$$\phi \text{ in } \text{LTL}^{\downarrow, \Sigma} \mapsto \phi' \text{ in } \text{FO}^\Sigma(\sim, <, +1)$$

Complexity of satisfiability problems

- FO over data words [Bojanczyk et al., LICS 06]
 - $\text{FO}_3(\sim, <, +1)$ is undecidable.
 - $\text{FO}_2(\sim, <, +1)$ is decidable over finite/infinite data words.
- LTL with registers [D. & Lazić, LICS 06]
 - $\text{LTL}_1^\downarrow[X, F]$ is undecidable over infinite data words.
 - LTL_1^\downarrow is decidable over finite data words.
(non-primitive recursive using [Schnoebelen, IPL 02])
 - LTL_2^\downarrow is undecidable over finite data words.
 - See preliminary undecidability results in
[(D. & Lazić & Nowak; Lisitsa & Potapov), TIME 05]

Purification

Purification lemma

There is a logspace reduction

- from $\text{MC}(\text{LTL})_n$ to $\text{PureMC}(\text{LTL})_{\max(n,1)}$
- from $\text{MC}(\text{FO})_n$ to $\text{PureMC}(\text{FO})_{n+2}$.

(determinism is preserved)

“Identify locations with patterns”

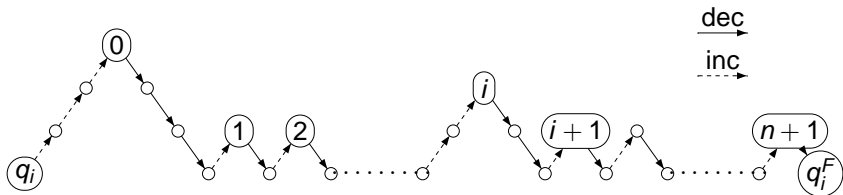
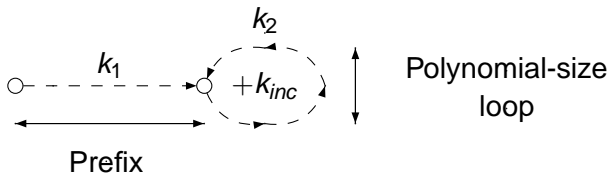


Figure: Pattern for q_i

Lasso runs for deterministic 1CA

- There exist constants k_1, k_2, k_{inc} (polynomial in $|\mathcal{A}|$) such that for $i \geq k_1$, $\langle q_{i+k_2}, n_{i+k_2} \rangle = \langle q_i, n_i + k_{inc} \rangle$.



Deciding when counter values are distinct

$$(k_{inc} > 0)$$

- \exists constant l polynomial in $|\mathcal{A}|$ (easy to compute).

“Passing l times in the loop guarantees that a value disappears.”

- $P_{\sim} = \{\langle i, j \rangle \in \{0, \dots, k_1 + lk_2 - 1\}^2 : n_i = n_j\}$.

- $n_i = n_j$ iff one of the conditions below holds true:

(I) $i, j < k_1 + lk_2$ and $\langle i, j \rangle \in P_{\sim}$,

(II) $i, j \geq k_1$, $|i - j| < lk_2$ and

$$\langle k_1 + (i - k_1) \bmod lk_2, k_1 + (j - k_1) \bmod lk_2 \rangle \in P_{\sim}$$

- (III) $i < k_1, j \geq k_1$ and $\langle i, j \rangle \in P_{\sim}$ (+ symmetrical case).

Encoding PureMC(FO)

- Reduction: $\mathcal{A} \models^\omega \phi$ iff $s \cdot t^\omega \models T(\phi)$.
- Model checking FO($<, +1$) over ultimately periodic words in PSPACE. [Markey & Schnoebelen, CONCUR 03]
- Ultimately periodic word
 - $s = \{0\} \cdot \{1\} \cdots \{k_1 - 1\}$.
 - $t = \{k_1\} \cdot \{k_1 + 1\} \cdots \{lk_2 - 1\}$.

- $x \sim y$ translated into

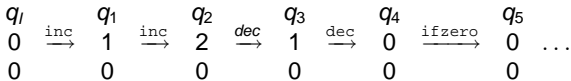
$$(x < k_1 + lk_2 \wedge y < k_1 + lk_2 \wedge \bigvee_{\langle I, J \rangle \in P_\sim} I(x) \wedge J(y)) \vee \dots$$

+ expression of previous cases (II) and (III).

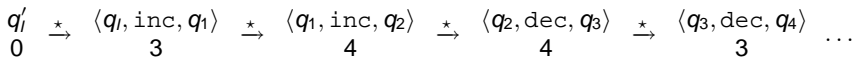
Complexity results for deterministic 1CA

- The first-order side:
 - MC(FO) is PSPACE-complete.
 - For every n , MC(FO)_n is in PTIME.
- The temporal side:
 - MC(LTL) is PSPACE-complete.
 - For every n , MC(LTL)_n is in PTIME.
- Lower bound from MC(LTL) and upper bounds from MC(FO) .

Standard principle to prove undecidability

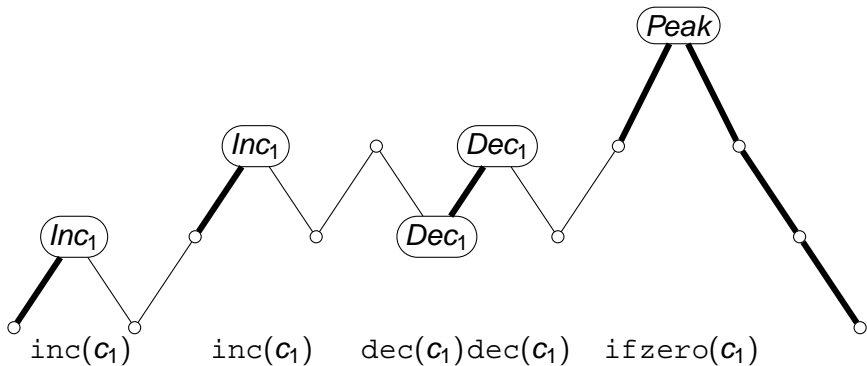


becomes

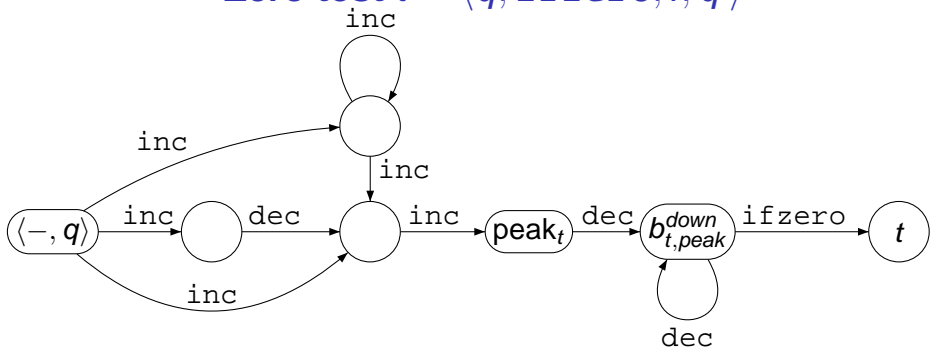


- Reduction: \mathcal{M} reaches an halting state iff $\mathcal{A} \models^{<\omega} \phi$.
- The counter values in \mathcal{A} are used as tags.
- Each counter value for decrementation corresponds to a counter value from a previous incrementation.
(similar principle in [David, MThesis 04])

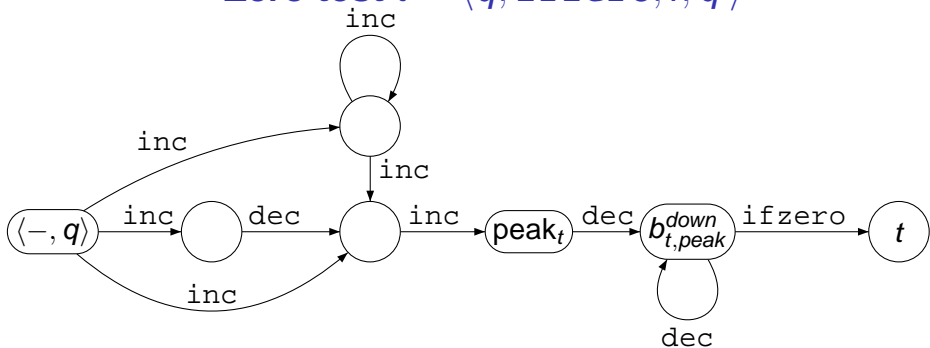
Encoding instructions in \mathcal{M} within \mathcal{A} and LTL \downarrow



Zero test $t = \langle q, \text{ifzero}, i, q' \rangle$



Zero test $t = \langle q, \text{ifzero}, i, q' \rangle$



incrementation and zero test imply decrementation

$$\overbrace{\neg F(Inc_i \wedge \downarrow F(\uparrow \wedge B_{i,peak}^{down})) \wedge \neg \downarrow F(\uparrow \wedge Dec_i))}^{\text{incrementation and zero test imply decrementation}} \wedge$$

$$\underbrace{\neg F(B_{i,peak}^{down} \wedge \downarrow F(\uparrow \wedge Dec_i))}_{\text{"no decrementation" after zero test}}$$

"no decrementation" after zero test

Undecidability results

- The temporal side:
 - $\text{MC}(\text{LTL})_1^{<\omega}[X, F]$ and $\text{PureMC}(\text{LTL})_1^{<\omega}$ are Σ_1^0 -complete.
 - $\text{MC}(\text{LTL})_1^\omega[X, F]$ and $\text{PureMC}(\text{LTL})_1^\omega$ are Σ_1^1 -complete.
 - The first-order side:
 - $\text{MC}(\text{FO2})^{<\omega}$ is Σ_1^0 -complete.
 - $\text{MC}(\text{FO2})^\omega$ is Σ_1^1 -complete.
- (using proofs for LTL_1^\downarrow and [D. & Lazić, LICS 06])

What's next?

	PSPACE-c. 1DCA	Σ_1^0 -c. 1NDCA	Σ_1^1 -c. 1NDCA
LTL	MC(LTL) $^\omega$ MC(LTL) $^{<\omega}$	MC(LTL) $_1^{<\omega}[X, F]$ PureMC(LTL) $_1^{<\omega}[X, F]$	MC(LTL) $_1^\omega[X, F]$ PureMC(LTL) $_1^\omega[X, F]$
FO	MC(FO) $^\omega$, MC(FO) $^{<\omega}$	MC(FO2) $^{<\omega}[\sim, <]$	MC(FO2) $^\omega[\sim, <]$

What's next?

	PSPACE-c. 1DCA	Σ_1^0 -c. 1NDCA	Σ_1^1 -c. 1NDCA
LTL	$MC(LTL)^\omega$ $MC(LTL)^{<\omega}$	$MC(LTL)_1^{<\omega}[X, F]$ PureMC(LTL) $_1^{<\omega}[X, F]$	$MC(LTL)_1^\omega[X, F]$ PureMC(LTL) $_1^\omega[X, F]$
FO	$MC(FO)^\omega, MC(FO)^{<\omega}$	$MC(FO2)^{<\omega}[\sim, <]$	$MC(FO2)^\omega[\sim, <]$

- A selection of open problems:
 - Decidability status of $LTL_1^{\downarrow}[F]$.
(for SAT and MC).
 - What about other syntactic fragments?
[Lazić, FSTTCS 06; D. & D'Souza & Gascon, LFCS'07]
 - Other classes of operational models.
(reversal-bounded counter machines, etc.).