Model checking memoryful logics over one-counter automata

S. Demri\(^1\)  R. Lazić\(^3\)  A. Sangnier\(^1,2\)

\(^1\)LSV, ENS Cachan, CNRS, INRIA Saclay & \(^2\)EDF R&D

\(^3\)Department of Computer Science
University of Warwick, UK

Dagstuhl "Beyond the finite" – April 2008
Data Words/Trees

- Timed word [Alur & Dill, TCS 94]
  \[
  \begin{array}{cccccc}
  a & b & c & a & a & b \\
  0 & 0.3 & 1 & 2.3 & 3.5 & 3.51
  \end{array}
  \]

- Runs for infinite-state systems [Minsky, 67]
  \[
  \begin{array}{ccccccc}
  q_0 & q_2 & q_3 & q_2 & q_3 & q_2 \\
  0 & 0 & 1 & 2 & 3 & 4
  \end{array}
  \]

- Integer arrays [Habermehl & Iosif & Vojnar, FOSSACS 08]
  \[
  \]

- 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
  - Data automata
  - See the survey [Kaminski & Francez, TCS 94]
  - Bouyer & Petit & Thérien, IC 03
  - 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)

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

- Runs are of the form

  $\rho = q_0 = q_I \quad n_0 = 0 \quad q_1 \quad n_1 \quad q_2 \quad n_2 \quad \ldots$

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

Syntax

\[ \phi ::= q \mid \uparrow_r \mid \neg \phi \mid \phi \land \phi \mid \phi \cup \phi \mid X\phi \mid \downarrow_r \phi \]

- Models:

\[
\begin{array}{cccccc}
q_0 & q_2 & q_3 & q_2 & q_3 & q_2 \\
0 & 0 & 1 & 2 & 3 & 4
\end{array}
\]

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

Satisfaction relation

\[
\begin{align*}
\sigma, i \models_v q & \iff \sigma(i) \text{ has location } q \\
\sigma, i \models_v \uparrow_r & \iff \sigma(i) \text{ has counter value } v(r) \\
\sigma, i \models_v X\phi & \iff i + 1 < |\sigma| \text{ and } \sigma, i + 1 \models_v \phi \\
\sigma, i \models_v \downarrow_r \phi & \iff \sigma, i \models_{v[r \mapsto i]} \phi
\end{align*}
\]
Examples

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

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

$q_0 \ q_2 \ q_3 \ q_2 \ q_3 \ q_2 \ q_2 \ldots$

$q_0 \ q_2 \ q_3 \ q_2 \ q_3 \ q_2 \ q_2 \ldots$

• 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 $m$

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

$q \ q' \ q' \ q' \ q' \ q'' \ q'' \ldots$

$q \ q' \ q' \ q' \ q' \ q'' \ q'' \ldots$

50  60  1  50  60  4  5  \ldots
Model checking problems

- Finitary model-checking $\text{MC}(\text{LTL})^{<\omega}$:
  $\exists$ finite accepting run $\rho$ of $A$ such that $\rho, 0 \models \phi$?

- Infinitary model-checking $\text{MC}(\text{LTL})^{\omega}$:
  $\exists$ infinite accepting run $\rho$ of $A$ such that $\rho, 0 \models \phi$?

- Subproblems
  - $\text{MC}(\text{LTL})^\alpha_n$: restriction to $n$ registers.
  - Pure$\text{MC}(\text{LTL})^\alpha$: restriction without location.

- Deterministic 1CA:

```
q  OR  q  inc  q'  OR  q  ifzero  q''
```

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

- Formulae in $\text{FO}^\Sigma(\sim, <, +1)$:
  \[
  \phi ::= a(x) \mid x \sim y \mid x < y \mid x = y + 1 \mid \ldots \mid \exists x \phi
  \]
  \[(a \in \Sigma)\]

- Satisfaction relation:
  \[
  \sigma \models_v x \sim y \text{ def } \iff v(x), v(y) \text{ are defined and } v(x) \sim^\sigma v(y)
  \]

- 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])
  - FO3(\(\sim, <, +1\)) is undecidable.
  - FO2(\(\sim, <, +1\)) is decidable over finite/infinite data words.

- **LTL with registers** ([D. & Lazić, LICS 06])
  - LTL\(_{\downarrow}^1\) is undecidable over infinite data words.
  - LTL\(_{\downarrow}^1\) is decidable over finite data words.
  - LTL\(_{\downarrow}^2\) is undecidable over finite data words.
  - See preliminary undecidability results in
    [((D. & Lazić & Nowak; Lisitsa & Potapov), TIME 05)]
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 $q_i$
There exist constants $k_1, k_2, k_{inc}$ (polynomial in $|A|$) such that for $i \geq k_1$, $\langle q_{i+k_2}, n_{i+k_2} \rangle = \langle q_i, n_i + k_{inc} \rangle$. 

Prefix

Polynomial-size loop
Deciding when counter values are distinct
\((k_{inc} > 0)\)

- \(\exists\) constant / polynomial in \(|A|\) (easy to compute).

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

- \(P_\sim = \{(i, j) \in \{0, \ldots, k_1 + lk_2 - 1\}^2 : n_i = n_j\}\).

- \(n_i = n_j\) 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
    \(\langle k_1 + (i - k_1) \mod lk_2, k_1 + (j - k_1) \mod lk_2 \rangle \in P_\sim\)
  - (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

$$
(x < k_1 + lk_2 \land y < k_1 + lk_2 \land \bigvee_{\langle l, J \rangle \in P} l(x) \land J(y)) \lor \ldots \ldots
$$

+ 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

$$q_i \xrightarrow{\text{inc}} q_1 \xrightarrow{\text{inc}} q_2 \xrightarrow{\text{dec}} q_3 \xrightarrow{\text{dec}} q_4 \xrightarrow{\text{ifzero}} q_5$$

$$0 \rightarrow 1 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 0 \rightarrow 0 \rightarrow 0 \rightarrow 0 \rightarrow \ldots$$

becomes

$$q'_i \xrightarrow{\star} \langle q_i, \text{inc}, q_1 \rangle \xrightarrow{\star} \langle q_1, \text{inc}, q_2 \rangle \xrightarrow{\star} \langle q_2, \text{dec}, q_3 \rangle \xrightarrow{\star} \langle q_3, \text{dec}, q_4 \rangle$$

$$0 \rightarrow 3 \rightarrow 4 \rightarrow 4 \rightarrow 3 \rightarrow \ldots$$

- Reduction: $M$ reaches an halting state iff $A \models <\omega \phi$.

- The counter values in $A$ 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↓

Model-Checking nondeterministic one-counter automata
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

\[
\neg F(Inc_i \land \downarrow F(\uparrow \land B_{i,peak}^{down}) \land \neg \downarrow F(\uparrow \land Dec_i)) \land \\
\neg F(B_{i,peak}^{down} \land \downarrow F(\uparrow \land Dec_i))
\]

"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 $\Sigma_1^0$-complete.
  - $\text{MC}(\text{LTL})_1^{\omega}[X, F]$ and $\text{PureMC}(\text{LTL})_1^{\omega}$ are $\Sigma_1^1$-complete.

- The first-order side:
  - $\text{MC}(\text{FO2})^{<\omega}$ is $\Sigma_1^0$-complete.
  - $\text{MC}(\text{FO2})^{\omega}$ is $\Sigma_1^1$-complete.

(using proofs for $\text{LTL}_1^{\downarrow}$ and [D. & Lazić, LICS 06])
## What’s next?

<table>
<thead>
<tr>
<th></th>
<th>$\mathbb{PSPACE}$-c. 1DCA</th>
<th>$\Sigma^0_1$-c. 1NDCA</th>
<th>$\Sigma^1_1$-c. 1NDCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTL</td>
<td>MC(LTL)$^\omega$</td>
<td>MC(LTL)$^\omega$[X, F]</td>
<td>MC(LTL)$^\omega$[X, F]</td>
</tr>
<tr>
<td></td>
<td>MC(LTL)$^{&lt;\omega}$</td>
<td>PureMC(LTL)$^{&lt;\omega}$[X, F]</td>
<td>PureMC(LTL)$^{&lt;\omega}$[X, F]</td>
</tr>
<tr>
<td>FO</td>
<td>MC(FO)$^\omega$, MC(FO)$^{&lt;\omega}$</td>
<td>MC(FO2)$^{&lt;\omega}$[\sim, &lt;]</td>
<td>MC(FO2)$^{&lt;\omega}$[\sim, &lt;]</td>
</tr>
</tbody>
</table>
What's next?

<table>
<thead>
<tr>
<th></th>
<th>( \text{PSPACE-c.} )</th>
<th>( \Sigma^0_\text{-c.} )</th>
<th>( \Sigma^1_\text{-c.} )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1DCA</td>
<td>1NDCA</td>
<td>1NDCA</td>
</tr>
<tr>
<td>LTL</td>
<td>( \text{MC}(\text{LTL})^\omega )</td>
<td>( \text{MC}(\text{LTL})^\omega )</td>
<td>( \text{MC}(\text{LTL})^\omega )</td>
</tr>
<tr>
<td></td>
<td>( \text{MC}(\text{LTL})^{&lt;\omega} )</td>
<td>( \text{MC}(\text{LTL})^{&lt;\omega} )</td>
<td>( \text{MC}(\text{LTL})^{&lt;\omega} )</td>
</tr>
<tr>
<td>FO</td>
<td>( \text{MC}(\text{FO})^\omega, \text{MC}(\text{FO})^{&lt;\omega} )</td>
<td>( \text{MC}(\text{FO2})^{&lt;\omega} )</td>
<td>( \text{MC}(\text{FO2})^{&lt;\omega} )</td>
</tr>
</tbody>
</table>

- A selection of open problems:
  - Decidability status of \( \text{LTL}^\uparrow_1 [F] \).
    (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.).