A Linear-Time Nominal μ-Calculus with Name Allocation

Daniel Hausmann, Stefan Milius and Lutz Schröder

Gothenburg University, Sweden and University Erlangen-Nürnberg, Germany

FM Seminar, Gothenburg

17 August 2021

Model Checking for Data Languages

- Linear-time (e.g. LTL) vs. branching-time (CTL, μ-calculus)

Basic linear-time model checking principle:

Transform φ to automaton $A(\varphi)$, check inclusion of model in $A(\varphi)$

Inclusion checking for “data automata” (infinite alphabet $\rightsquigarrow$ data):

- Register Automata (RA) [Kaminski et al. 1994] undecidable
- Nondeterministic Orbit-finite Automata (NOFA) [Neven et al. 2004, Boyańczyk et al. 2014] undecidable
- Variable Automata [Grumberg et al. 2010] undecidable

Logics with Freeze Quantification

Freeze LTL [Demri, Lazić, 2007]:

- paths: data words $(P_1, d_1), (P_2, d_2), \dots$
- operators $\downarrow_r \varphi$: "$r \leftarrow d_i; \varphi$", $\uparrow_r$: "$d_i = r$?"

Flat Freeze LTL [Bollig et al. 2019]:

- for all subformulae $\phi_1 \, \mathsf{U} \, \phi_2$, no freeze operator in ϕ_1

Model Checking for Freeze LTL:

- Freeze LTL over RA [Demri, Lazić, 2007] undecidable
- Flat Freeze LTL over OCA [Bollig et al. 2019] $\textsc{NExpTime}$

One-Counter Automata

Model Checking for Bar Strings

[Schröder et al. 2017]: Regular bar expressions and Regular Nondeterministic Nominal Automata (RNNA), using nominal sets

- RNNA inclusion checking is in ExpSpace

Our progress here: A linear-time fixpoint logic for RNNA

- Define Logic
- Introduce extended RNNA (ERNNA)
- Translate formulae to equivalent ERNNA
- Generalize RNNA inclusion checking to ERNNA and obtain 2ExpTime model checking and validity checking

Nominal Sets

Fix countable set $\mathbb{A}$ of names, G: group of fin. permutations on $\mathbb{A}$

Nominal sets

Action of G on set X:
$\cdot : G \times X \to X$ s.t. for all $x \in X$, $\pi, \pi' \in G$

$$\mathsf{id} \cdot x = x \qquad\qquad \pi \cdot (\pi' \cdot x) = (\pi\pi') \cdot x$$

Given set X, set $S \subseteq \mathbb{A}$ is a support of $x \in X$ if $\pi(x) = x$ for all $\pi \in G$ such that $\pi(a) = a$ for all $a \in S$

Nominal set: $(X, \cdot)$ s.t. all $x \in X$ have min. finite support $\mathsf{supp}(x)$

Orbit of $x \in X$: $\{\pi \cdot x \mid \pi \in G\}$

Bar Strings

Abstraction set: $[\mathbb{A}]X = (\mathbb{A} \times X)/\sim$ where

$$(a, x) \sim (b, y) \text{ if and only if } (ac) \cdot x = (bc) \cdot y \text{ for any fresh } c$$

$\langle a \rangle x$: $\sim$-equivalence class of (a, x)

Bar strings

Set of finite bar strings: $\overline{\mathbb{A}}^*$ where $\overline{\mathbb{A}} = \mathbb{A} \cup \{|a \mid a \in \mathbb{A}\}$

$\equiv_\alpha$ on bar strings: equivalence generated by

$$w|av \equiv_\alpha w|bu \text{ iff } \langle a \rangle v = \langle b \rangle u \text{ in } [\mathbb{A}]\overline{\mathbb{A}}^*$$

E.g. $|a|bb \equiv_\alpha |a|aa \not\equiv_\alpha |a|ba$

Bar languages: subsets of $\overline{\mathbb{A}}^*/\equiv_\alpha$

Data languages from bar languages

Put $\mathsf{ub}(a) = \mathsf{ub}(|a) = a$, extend ub to bar strings

Given bar language L, put

no name bound twice

$N(L) = \{\mathsf{ub}(w) \mid [w]_\alpha \in L, w \text{ clean}\}$ global freshness semantics
$D(L) = \{\mathsf{ub}(w) \mid [w]_\alpha \in L\}$ local freshness semantics

E.g.

$$D(|a|ba) = N(|a|ba) = \{aba \mid a, b \in \mathbb{A}, a \neq b\},$$

while

$$N(|a|b) = \{ab \mid a, b \in \mathbb{A}, a \neq b\} \quad \text{but}$$
$$D(|a|b) = \{ab \mid a, b \in \mathbb{A}\}$$

A Linear-time Logic for Bar Strings

Syntax of Bar-μTL

$$\varphi, \psi ::= \epsilon \mid \neg\varphi \mid \varphi \wedge \psi \mid \Diamond_a \varphi \mid \Diamond_{|a} \varphi \mid X \mid \mu X.\varphi \quad (a \in \mathbb{A}, X \in \mathsf{V})$$

requiring positivity and guardedness of fixpoint variables

Put $\Box_\sigma \psi := \neg \Diamond_\sigma \neg \psi$ for $\sigma \in \overline{\mathbb{A}}$

Define $\equiv_\alpha$ on formulae, e.g. $\Diamond_{|a}(\Diamond_a \epsilon \vee \Box_b \neg\epsilon) \equiv_\alpha \Diamond_{|c}(\Diamond_c \epsilon \vee \Box_b \neg\epsilon)$

Semantics of Bar-μTL

Interpret in context $S \subseteq \mathbb{A}$ over bar strings w s.t. $\mathsf{FN}(w) \subseteq S$:

$$
\begin{aligned}
S, w \models \epsilon \quad &\Leftrightarrow \quad w = \epsilon \\
S, w \models \mu X.\, \varphi \quad &\Leftrightarrow \quad S, w \models \varphi[X/\mu X.\, \varphi] \\
S, w \models \Diamond_a \varphi \quad &\Leftrightarrow \quad w = av \text{ and } S, v \models \varphi \\
S, w \models \Diamond_{|a} \varphi \quad &\Leftrightarrow \quad \exists b \in \mathbb{A}, v \in \overline{\mathbb{A}}^*, \psi.\, w \equiv_\alpha |bv, \\
&\qquad \langle a \rangle \varphi = \langle b \rangle \psi \text{ and } S \cup \{b\}, v \models \psi
\end{aligned}
$$

Put $[\![\varphi]\!] = \{w \in \overline{\mathbb{A}}^* \mid \emptyset, w \models \varphi\}/\equiv_\alpha$

Extended Regular Nondeterministic Nominal Automata

Set $S \subseteq X$ is equivariant if $\pi \cdot x \in S$ for all $\pi \in G$, $x \in S$

Extended Regular Nondeterministic Nominal Automata

Set $S \subseteq X$ is equivariant if $\pi \cdot x \in S$ for all $\pi \in G$, $x \in S$

Extended Regular Nondeterministic Nominal Automata (ERNNA)

$A = (Q, \to, s, f)$ with

- orbit-finite nominal set Q of states, initial state $s \in Q$
- equivariant transition relation $\to \subseteq Q \times (\overline{\mathbb{A}} \cup \epsilon) \times Q$
- equivariant acceptance function $f : Q \to \{0, 1, \top\}$

such that $q \overset{|a}{\to} q'$ and $\langle a \rangle q' = \langle b \rangle q''$ imply $q \overset{|b}{\to} q''$ (α-invariance) and such that $\{(a, q') \mid q \overset{a}{\to} q'\}$ and $\{\langle a \rangle q' \mid q \overset{|a}{\to} q'\}$ are finite.

Degree of A: maximal size of support of some state $q \in Q$

ERNNAs, acceptance

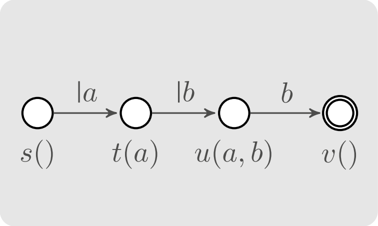
Definition (ERNNA acceptance)

Bar string $w \in \overline{\mathbb{A}}^*$ is accepted by $A = (Q, \to, s, f)$ if

- $\exists q \in Q.\, s \xrightarrow{w} q$ and $f(q) = 1$, or
- $\exists q \in Q$, prefix u of $w.\, s \xrightarrow{u} q$ and $f(q) = \top$.

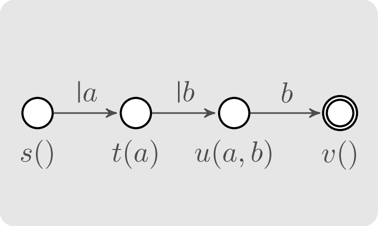
Literal acceptance: $L_0(A) = \{w \in \overline{\mathbb{A}}^* \mid A \text{ accepts } w\}$
Accepted bar language: $L_\alpha(A) = L_0/{\equiv_\alpha}$

(Name dropping) ERNNA, Example

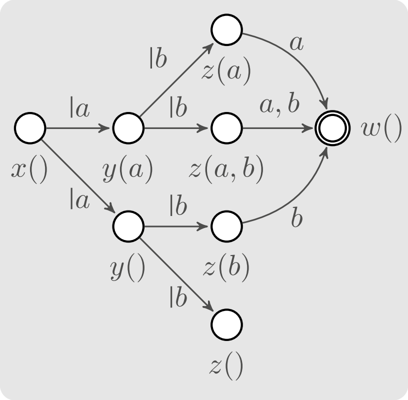


$s()$ accepts $|a|bb$ but not $|a|aa$

(Name dropping) ERNNA, Example



$\rightsquigarrow$



$s()$ accepts $|a|bb$ but not $|a|aa$

$x()$ accepts both $|a|bb$ and $|a|aa$

Name Dropping ERNNA, Result

Lemma [following Schröder et al. 2017]

For all ERNNAs A of degree k and with n orbits, there is ERNNA nd(A) of degree $k+1$ and with $n \cdot 2^{k+1}$ orbits, s.t.

1. $L_\alpha(A) = L_\alpha(\mathsf{nd}(A))$ and
2. $L_0(\mathsf{nd}(A))$ is closed under α-equivalence of bar strings.

Corollary [following Schröder et al. 2017]

Inclusion checking for ERNNAs is in $\textsc{ExpSpace}$ / para-$\textsc{PSpace}$.

Translating Formulae to ERNNA

Problem

Let $\varphi(b) = \mu Y. (\Box_b \bot \wedge \Box_{|c} Y)$ and $\psi = \mu X. (\Box_{|a} X \wedge \Box_{|b} \varphi(b))$

To check $|a_1|a_2 \ldots |a_n a_i v \models \psi$, have to check $a_i v \models \varphi(a_n)$ for all n

Solution: use nondeterminism to guess relevant letter a_i, keep just one copy $\varphi(a_i)$ of $\varphi(\_)$.

Further complication: Elimination of $\Box$-formulae.

Given φ of size n and degree k, define formula automaton $A(\varphi)$

Theorem

We have $L_\alpha(A(\varphi)) = [\![\varphi]\!]$ and $A(\varphi)$ has $2^{\mathcal{O}(n^2 \cdot 2^k)}$ orbits.

Model Checking

Input: RNNA M, formula φ of size n and degree k

- Model checking: check whether

$$L_\alpha(M) \subseteq [\![\varphi]\!] = L_\alpha(A(\varphi)) = L_\alpha(\mathsf{nd}(A(\varphi)))$$

formulae are ERNNA   name dropping construction

- $A(\varphi)$ has at most $2^{\mathcal{O}(n^2 \cdot 2^k)}$ orbits,
  $\mathsf{nd}(A(\varphi))$ has at most $2^{k+1} \cdot 2^{\mathcal{O}(n^2 \cdot 2^k)}$ orbits

Theorem

Model checking and validity checking for BarμTL are in 2ExpSpace (para-ExpSpace with parameter k). Satisfiability checking is in ExpSpace (para-PSpace with parameter k).

Conclusion

Results

- Linear-time logic for finite bar strings
- Extended regular nominal automata (ERNNA)
  - inclusion checking for ERNNA in $\textsc{ExpSpace}$
- Non-trivial translation of formulae into ERNNA, removing universal branching by nondeterminism
- Model / validity checking over RNNA is elementary!
  (in $2\textsc{ExpSpace}$ and para-$\textsc{ExpSpace}$)

Future work:

- Extend this to infinite bar strings (nominal Büchi automata, see [Urbat, H, Milius, Schröder, CONCUR 2021])
- Is translation from formulae to RNNA a nondeterminisation procedure for alternating nominal automata?