Linear-time (e.g. LTL) vs. branching-time (CTL, $\mu$-calculus)

A basic linear-time model checking principle:
Transform $\varphi$ to automaton $A(\varphi)$, check inclusion of model in $A(\varphi)$

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

- nondeterministic Register Automata (RA)  
  [Kaminski et al. 1994]  undecidable
- deterministic / unambiguous RA  
  [Mottet, Quaas 2019, Colcombet 2015]  decidable
- 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 \((P_1, d_1), (P_2, d_2), \ldots\)
- operators \(\downarrow_r \varphi: "r \leftarrow d_i; \varphi"\), \(\uparrow_r: "d_i = r?"\)

Flat Freeze LTL [Bollig et al. 2019]:
- for all subformulae \(\phi_1 U \phi_2\), no freeze operator in \(\phi_1\)

Model Checking for Freeze LTL:
- Freeze LTL over RA [Demri, Lazić, 2007] undecidable
- Flat Freeze LTL over OCA [Bollig et al. 2019] NExpTime
Contributions

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

- RNNA inclusion checking is in $\text{ExpSpace}$

**Bar-$\mu$TL**: a linear-time fixpoint logic for RNNA

- safety and liveness (via fixpoints), full nondeterminism
- closure under complement
- no restriction on number of registers
- expresses e.g. “some letter occurs twice” (unlike deterministic or unambiguous RA)

**Results**

The main reasoning problems of Bar-$\mu$TL are decidable.
Fix countable set $\mathbb{A}$ of names, $G$: group of fin. permutations on $\mathbb{A}$

**Nominal sets**

- **Action** $\cdot : G \times X \rightarrow X$ of $G$ on $X$

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

- **Nominal set**: $(X, \cdot)$ s.t. all $x \in X$ have finite support

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

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

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

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

Bar strings / languages

- Set of finite bar strings: \( \overline{A}^* \) where \( \overline{A} = A \cup \{ |a | \ a \in A \} \)

- Standard \( \alpha \)-equivalence on \( \overline{A}^* \), e.g. \( |a|bb \equiv_\alpha |a|aa \not\equiv_\alpha |a|ba \)

- Bar languages: subsets of \( \overline{A}^*/\equiv_\alpha \)

Put \( ub(a) = ub(|a) = a \), extend \( ub \) to bar strings

Data languages from bar language \( L \)

\[
D(L) = \{ ub(w) | [w]_\alpha \in L \} \quad \text{local freshness semantics}
\]

\[
N(L) = \{ ub(w) | [w]_\alpha \in L, w \text{ clean} \} \quad \text{global freshness semantics}
\]

E.g. \( D(|a|b) = \{ ab \mid a, b \in A \} \), \( N(|a|b) = \{ ab \mid a, b \in A, a \neq b \} \),
A Linear-time Logic for Bar Strings

| Syntax of Bar-$\mu$TL |
|-----------------------|
| $\varphi, \psi ::= \epsilon \mid \neg \varphi \mid \varphi \land \psi \mid \Diamond_a \varphi \mid \Diamond_{1a} \varphi \mid X \mid \mu X. \varphi \ (a \in A, X \in V)$ |

requiring positivity and guardedness of fixpoint variables

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

Define $\equiv_\alpha$ on formulae, e.g. $\Diamond_{1a}(\Diamond_a \epsilon \lor \Box_b \neg \epsilon) \equiv_\alpha \Diamond_{1c}(\Diamond_c \epsilon \lor \Box_b \neg \epsilon)$
Semantics of Bar-\(\mu\)TL

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

\[
S, w \models \diamond_a \varphi \iff w = av \text{ and } S, v \models \varphi
\]

\[
S, w \models \diamond_{|a}\varphi \iff \exists b \in A, v \in \overline{A}^*, \psi. \ w \equiv_a \vdash bv,
\]

\[
\diamond_{|a}\varphi \equiv_a \diamond_{|b}\psi \text{ and } S \cup \{b\}, v \models \psi
\]

\[
S, w \models \mu X. \varphi \iff S, w \models \varphi[X/\mu X. \varphi]
\]

Put \([\varphi] = \{w \in \overline{A}^* \mid \emptyset, w \models \varphi\}/\equiv_a\)

E.g. \(\{b\}, \{ccb\} \models \diamond_{|b}\diamond_{|b}\neg\epsilon\) since \(\{ccb\} \equiv_a \vdash \{d\}, \{db\} \models \diamond_a \neg\epsilon\)
Set $S \subseteq X$ is equivariant if $\pi \cdot x \in S$ for all $\pi \in G$, $x \in S$. 
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)**

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

- orbit-finite nominal set $Q$ of states, initial state $s \in Q$
- equivariant transition relation $\rightarrow \subseteq Q \times \bar{\Delta} \times Q$
- equivariant acceptance function $f : Q \rightarrow \{0, 1, \top\}$

s.t. $q \xrightarrow{a} q'$ and $\langle a \rangle q' = \langle b \rangle q''$ imply $q \xrightarrow{b} q''$ ($\alpha$-invariance) and s.t. \{(a, q') | q \xrightarrow{a} q'\} and \{\langle a \rangle q' | q \xrightarrow{la} q'\} are finite

Degree of $A$: maximal size of support of some state $q \in Q$
Definition (ERNNA acceptance)

Bar string $w \in \overline{A}^*$ is accepted by $A = (Q, \rightarrow, s, f)$ 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: $L_0(A) = \{ w \in \overline{A}^* | A$ accepts $w \}$

Accepted bar language: $L_\alpha(A) = L_0/\equiv_\alpha$
(Name dropping) ERNNA, Example

\[ s() \text{ accepts } |a|bb \text{ but not } |a|aa \]
$s()$ accepts $|a|bb$ but not $|a|aa$

$x()$ accepts both $|a|bb$ and $|a|aa$
### Lemma [following Schröder et al. 2017]

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

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

### Corollary [following Schröder et al. 2017]

Inclusion checking for ERNNAs is in $\text{ExpSpace} / \text{para-PSpace}$.
Translating Formulae to ERNNA

Problem

Let \( \varphi(b) = \mu Y. (\Box b \perp \wedge \Box c Y) \) and \( \psi = \mu X. (\Box a X \wedge \Box b \varphi(b)) \)

To check \( |a_1|a_2 \ldots |a_n a_{i}v | = \psi \), have to check \( a_{i}v | = \varphi(a_n) \) 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 \( \varphi \) of size \( n \) and degree \( k \), define formula automaton \( A(\varphi) \)

Theorem

We have \( L_{\alpha}(A(\varphi)) = [\varphi] \) and \( A(\varphi) \) has \( 2^{O(n^2 \cdot 2^{k})} \) orbits.
Input: RNNA $M$, formula $\varphi$ of size $n$ and degree $k$

- Model checking: check whether
\[
L_\alpha(M) \subseteq \mathbb{[}\varphi\mathbb{]} = L_\alpha(A(\varphi)) = L_\alpha(\text{nd}(A(\varphi)))
\]

formulæ are ERNNA  

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

Main results:

|                | global freshness | local freshness |
|----------------|------------------|-----------------|
| validity checking | ExpSpace          | 2ExpSpace       |
| satisfiability checking | ExpSpace          | 2ExpSpace       |
| model checking       | ExpSpace          | 2ExpSpace       |
Conclusion

Results

- Linear-time logic for finite bar strings
- Extended regular nominal automata (ERNNA)
  - inclusion checking for ERNNA in \( \text{ExpSpace} \)
- Non-trivial translation of formulae into ERNNA, removing universal branching by nondeterminism
- Model / validity / sat. checking over RNNA decidable!

Future work:

- Extend logic to infinite bar strings (nominal Büchi automata, see [Urbat, H, Milius, Schröder, CONCUR 2021])