On Finite Domains in First-Order Linear Temporal Logic

Denis Kuperberg [2] **Julien Brunel** [1] David Chemouil [1]

[1]ONERA, Université Fédérale de Toulouse

[2]TU Munich

LOGICAL BACKGROUND OF ELECTRUM: FO-LTL

The logic FO-LTL

$\varphi ::= (x_1 = x_2) \mid P_i(x_1, \ldots, x_n) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x.\varphi \mid \mathsf{X}\varphi \mid \varphi \mathsf{U} \varphi.$
We also define $\mathsf{F}\varphi = \mathit{true}\mathsf{U}\varphi$ and $\mathsf{G}\varphi = \neg\mathsf{F}(\neg\varphi)$.

We use FO-LTL as underlying logic of the language Electrum.

Finite domain semantics

- First-Order variables x_i: finite domain
- Implicit time: infinite domain $\mathbb{N}$

LTL: Good properties of expressiveness and complexity, widely used in verification.

What is the theoretical cost of adding LTL to Alloy's logic ?

FO-LTL ON FINITE FO DOMAINS

1. Complexity of “bounded SAT” (*i.e.* given a bound on the FO domain)

2. Finite model property of FO-LTL

   Considering finite FO domain can be enough in some fragments.

COMPLEXITY

Definition (BSAT Problem)

Given φ and N, is there a model for φ, for which the size of the first-order domain is at most N ?

Parameters

- Logic: FO versus FO-LTL
- Encoding of N: unary versus binary
- Rank of formulas (nested quantifiers): bounded ($\perp$) versus unbounded ($\top$).

COMPLEXITY

Definition (BSAT Problem)

Given φ and N, is there a model for φ, for which the size of the first-order domain is at most N ?

Theorem

	N unary	*N binary*
FO $\bot$	*NP-complete*	*NEXPTIME-complete*
FO $\top$	*NEXPTIME-complete*	*NEXPTIME-complete*
FO-LTL $\bot$	*PSPACE-complete*	*EXPSPACE-complete*
FO-LTL $\top$	*EXPSPACE-complete*	*EXPSPACE-complete*

IDEAS OF THE PROOFS

Membership:

- Guess a structure and verify it,
- Unfold the formula according to the elements of this structure,
- Use PSPACE LTL Satisfiability.

Hardness

- Reduce from Turing machines or SAT for NP-hardness,
- Encode states and alphabet in the signature,
- Structure encodes space/time for FO and space for FO-LTL,
- Formula in the studied fragment encode run of the machine.

Finite Model Theory

Definition (Finite Model Property (FMP))

If there is a model for φ, then there is a finite one.

Some First-Order Fragments with FMP

- $[\exists^*\forall^*, \mathit{all}]_=$ (Ramsey 1930)
- $[\exists^*\forall\exists^*, \mathit{all}]_=$ (Ackermann 1928)
- $[\exists^*, \mathit{all}, \mathit{all}]_=$ (Gurevich 1976)
- FO_2 (Mortimer 1975) : 2 variables.
- $[\exists^*\forall, \mathit{all}, (1)]_=$ (Grädel 1996)
- $[\mathit{all}, (\omega), (\omega)]$ (Gurevich 1969, Löb 1967)

LIFTING FMP TO FO-LTL: A GENERAL RESULT

Definition (FMP for FO-LTL)

If there is a model for φ, then there is a model with finite FO-domain.

Theorem

Adding X, F *to FO preserves FMP if the fragment imposes no constraint on the number and arity of predicates/functions.*

Applies to the above-mentioned fragments except:

- $[\exists^*\forall, \textit{all}, (\textit{1})]_=$ only one function of arity one.
- $[\textit{all}, (\omega), (\omega)]$ only predicates and functions of arity one.

Ideas of the proof

- Consider an FO fragment *Frag* that has the FMP
- Suppose that $\varphi \in \textit{Frag} + \{\mathsf{X}, \mathsf{F}\}$ has a model.
- We translate φ into a pure FO (in *Frag*) formula ψ (also satisfiable)
  Example: $\mathsf{X}p \wedge \mathsf{XX}p \rightsquigarrow p_1 \wedge p_2$
- Since $\psi \in \textit{Frag}$, ψ has a finite model M
- We build a finite model of φ from M

LIFTING FMP TO FO-LTL: AD-HOC RESULTS

Theorem (Extension of the Gurevich fragment)

$[\textit{all}, (\omega), (\omega)] + \{\mathsf{X}, \mathsf{F}\}$ *has the FMP.*

Theorem (Extension of the Ramsey fragment)

The FO-LTL fragment of formulas of the form $\exists x_1 \ldots \exists x_n.\psi$, where ψ is a FO-LTL formula without any $\exists$ quantifiers, has the FMP.

Axioms of infinity

In general, adding LTL allows to write axioms of infinity:

Wrong extension of the Ramsey fragment

$$\mathsf{G}(\exists x.P(x) \wedge \mathsf{X}(\mathsf{G}\neg P(x)))).$$

(only one existential variable!)

Axioms of infinity

In general, adding LTL allows to write axioms of infinity:

Wrong extension of the Ramsey fragment

$$\mathrm{G}(\exists x. P(x) \wedge \mathrm{X}(\mathrm{G}\neg P(x)))).$$

(only one existential variable!)

Without nesting quantifiers in temporal operators

$$\forall x \exists y. P(c) \wedge \mathrm{G}(P(x) \Rightarrow \mathrm{X}(P(y) \wedge \mathrm{G}\neg P(x))).$$

Axioms of infinity

In general, adding LTL allows to write axioms of infinity:

Wrong extension of the Ramsey fragment

$$G(\exists x. P(x) \wedge X(G \neg P(x)))).$$

(only one existential variable!)

Without nesting quantifiers in temporal operators

$$\forall x \exists y. P(c) \wedge G(P(x) \Rightarrow X(P(y) \wedge G \neg P(x))).$$

Without G

$$\forall x \exists y. P(c) \wedge ((P(x) \wedge P(y)) U (\neg P(x) \wedge P(y))).$$

CONCLUSION

Theoretical study of FO-LTL on finite domain

- Complexity
- Finite model property

Open questions:

- Complexity of BSAT for FO-LTL[1] with n in binary
- Can we drop (or weaken) the condition for adding X and F to a fragment that has the FMP?
- Can we find a reasonable condition to extend the FO fragments that have the FMP with G and/or U?
- Decidability of FO-LTL fragments

Backup slides

Proof scheme for hardness

Idea : encode runs of Turing Machines via formulas.

For FO, unbounded rank, binary encoding :

Reduction :

- Start from non-deterministic M running in time 2^n on inputs of size n. States Q and alphabet A.
- Consider the first-order structure $\{1, \ldots, 2^n\}$ with predicate successor, representing both time and space of the machine.
- Predicate $a(x, t)$ with $a \in A$: the cell x is labeled a at time t
- Predicate $q(x, t)$: M is in state q in position x at time t

For any word u of size n, we can now write a formula φ_u of size polynomial in n, stating that:

- The initial configuration of the tape is u:
  $a_1(1,1) \wedge a_2(2,1) \wedge \cdots \wedge a_n(n,1)$
- For all time t, the tape is updated from t to $t+1$ according to the transition table of M
- there is a time t_f where M is in its accepting state.

Correctness: φ_u has a model of size $2^n \iff u$ is accepted by M

Size 2^n is given in binary $\rightarrow$ polynomial reduction.

For any word u of size n, we can now write a formula φ_u of size polynomial in n, stating that:

- The initial configuration of the tape is u:
  $a_1(1,1) \wedge a_2(2,1) \wedge \cdots \wedge a_n(n,1)$
- For all time t, the tape is updated from t to $t+1$ according to the transition table of M
- there is a time t_f where M is in its accepting state.

Correctness: φ_u has a model of size $2^n \iff u$ is accepted by M

Size 2^n is given in binary $\rightarrow$ polynomial reduction.

Extension to FO-LTL: LTL uses implicit time $\rightarrow$ we can start from an EXPSPACE machine.
Constraint on transitions is now of the form
$\mathrm{G}(\forall x, q(x) \implies \mathrm{X}\varphi_q(x))$

Tricky case: unbounded rank but unary N.
$\rightarrow$ We can no longer use the domain as a model for the tape.

Tricky case: unbounded rank but unary N.
$\rightarrow$ We can no longer use the domain as a model for the tape.

Solution: Use a structure of size 2, and binary encoding to point to a cell or time instant : $a(\vec{x}, \vec{t})$ for FO and $a(\vec{x})$ for FO-LTL.

Example: For size 8, $a(0, 1, 1, 1, 0, 1)$ means that the 3^{th} cell is labeled by a at instant 5.