Satisfiability Checking for Mission-Time LTL

Jianwen Li[1] Moshe Y. Vardi[2] Kristin Y. Rozier[1]

1. Iowa State University, Ames, IA, USA

2. Department of Computer Science, Rice University, USA

July 17, 2019

Mission-Time LTL (MLTL)

Application:

- NASA Robonat 2 system

- Runtime verification community interests - RV 2018 competition benchmarks

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

Mission-Time LTL (MLTL)

MLTL is designated for describing systems that focus on

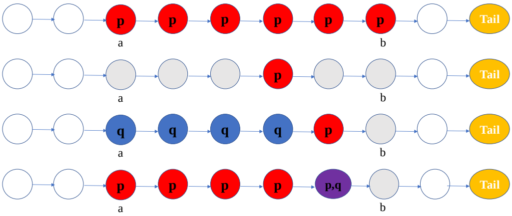
- finite behaviors with
- bounded and discrete time intervals.

Mission-Time LTL (MLTL)

MLTL formulas reason about linear timelines:

- finite set of atomic propositions {p q}
- Boolean connectives: $\neg$, $\wedge$, $\vee$, and $\rightarrow$
- temporal connectives:

$\Box_{[a,b]}p$	ALWAYS
$\Diamond_{[a,b]}p$	EVENTUALLY
$q\mathcal{U}_{[a,b]}p$	UNTIL
$q\mathcal{R}_{[a,b]}p$	RELEASE



IOWA STATE UNIVERSITY Laboratory for Temporal Logic

Mission-Time LTL (MLTL)

	MLTL	MTL	LTL	LTL_f
Model Length	finite	infinite	infinite	finite
Interval Domain	integer	real	-	-
Interval Range	bounded	unbounded	-	-

- MTL: Metric Temporal Logic
- LTL: Linear Temporal Logic
- LTL_f: LTL over finite traces

MLTL Satisfiability Checking (MLTLSAT)

Given an MLTL formula φ, is there a finite trace that is a model of φ?

- $\Diamond_{[0,3]} p \wedge \Box_{[0,2]} \neg p$ is satisfiable;
- $\Diamond_{[0,3]} p \wedge \Box_{[0,4]} \neg p$ is unsatisfiable;

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

MLTL Satisfiability Checking (MLTLSAT)

Given an MLTL formula φ, is there a finite trace that is a model of φ?

- $\Diamond_{[0,3]} p \wedge \Box_{[0,2]} \neg p$ is satisfiable;
- $\Diamond_{[0,3]} p \wedge \Box_{[0,4]} \neg p$ is unsatisfiable;

- MLTLSAT is a fundamental problem of MLTL reasoning;
- MLTLSAT is helpful for writing consistent MLTL specifications.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

Contributions

- Prove MLTLSAT is NEXPTIME-complete;
- Reduce MLTLSAT to LTL_fSAT, LTLSAT and LTL model checking;
- Present a practical SMT-based checking algorithm for MLTLSAT;
- Show the SMT-based approach has the most scalable performance.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

MLTLSAT Complexity

Theorem 1

The complexity of MLTL *satisfiability checking is NEXPTIME-complete.*

Upper: For an MLTL formula φ, there is an LTL$_f$ formula ψ s.t.

- φ and ψ are equi-satisfiable;
- $|\psi| = K \times |\varphi|$ (K is the maximal decimal integer in the intervals of φ).
- ψ contains only $\mathcal{X}/\mathcal{N}$;
- A model of length $O(|\psi|)$ exists iff ψ is satisfiable.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

MLTLSAT Complexity

Theorem 1

The complexity of MLTL *satisfiability checking is NEXPTIME-complete.*

Lower: Given a nondeterministic Turing machine M and an integer k,

- Construct the MLTL formula φ_M with length of $O(k)$;
- φ_M is satisfiable iff M accepts the empty tape in 2^k steps;
- MLTL intervals are written in decimal, so we can ensure $|\varphi_M|$ is in $O(k)$.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

MLTLSAT via Reductions

- MLTLSAT via LTL_fSAT (Theorem 1)
- MLTLSAT via LTLSAT (LTL_fSAT is reducible to LTLSAT)
- MLTLSAT via LTL model checking (LTLSAT is reducible to LTL model checking)

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

SMT-based MLTLSAT

Given an MLTL formula φ,

1. $f_p : Int \rightarrow Bool$, a monadic predicate representing $p \in \Sigma_\varphi$;
2. $\mathrm{fol}(\varphi, k, len)$ for φ ($k, len \in N$):
   - $\mathrm{fol}(p, k, len) = (len > k) \wedge f_p(k)$ for $p \in \Sigma$;
   - $\mathrm{fol}(\neg\xi, k, len) = (len > k) \wedge \neg\mathrm{fol}(\xi, k, len)$;
   - $\mathrm{fol}(\xi \wedge \psi, k, len) = (len > k) \wedge \mathrm{fol}(\xi, k, len) \wedge \mathrm{fol}(\psi, k, len)$;
   - $\mathrm{fol}(\xi\ \mathcal{U}_{[a,b]}\ \psi, k, len) = (len > a + k) \wedge \exists i.(\ (a + k \leq i \leq b + k) \wedge$ $\mathrm{fol}(\psi, i, len - i) \wedge\ \forall j.((a + k \leq j < i) \rightarrow \mathrm{fol}(\xi, j, len - j)))$.

k: Index where the formula is evaluated;

len: Model length.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

SMT-based MLTLSAT

$S(\mathrm{fol}(\varphi, k, n))$: SMT-LIB v2 encoding.

- $S(\mathrm{fol}(p, k, len)) \longrightarrow (\text{and } (> len\ k)\ (f_p\ k))$
- $S(\neg\mathrm{fol}(\varphi, k, len)) \longrightarrow (\text{and } (> len\ k)\ (\text{not } S(\mathrm{fol}(\varphi, k))))$
- $S(\mathrm{fol}(\varphi_1 \wedge \psi, k, len) \longrightarrow (\text{and } (>\ len\ k)\ (\text{and } S(\mathrm{fol}(\varphi_1, k, len))\ S(\mathrm{fol}(\psi, k, len))))$
- $S(\mathrm{fol}(\varphi_1\ \mathcal{U}_{[a,b]}\ \psi, k, len)) \longrightarrow (\text{and } (>\ len\ a+k)\ (\text{exists } (i\ Int)\ (\text{and } (\leq\ (+\ a\ k)\ i)\ (\geq\ i\ (+\ b\ k))\ S(\mathrm{fol}(\psi, i, len - i))\ (\text{forall } (j\ Int)\ (\Rightarrow (\text{and } (\leq\ (+\ a\ k)\ j)\ (<\ j\ i))\ S(\mathrm{fol}(\varphi_1, j, len - j))))))))$

Theories used: Uninterpreted functions and quantifiers

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

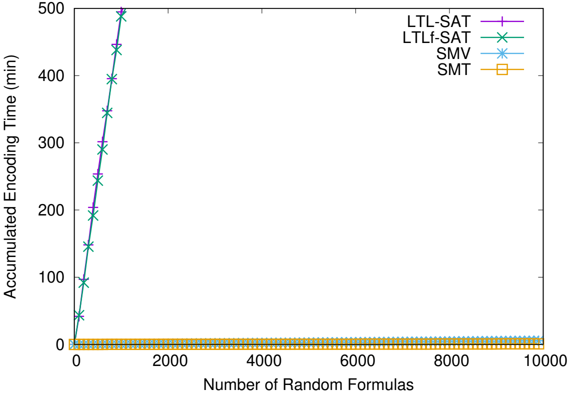
Experimental Results

- Benchmarks:
  - 10,000 Random MLTL formulas: interval ranges in [0,100] (R);
  - 3 group of 63 NASA-Boeing MLTL formulas: interval ranges in [0, 1000], [0,10000] and [0, 100000] respectively (NB);
- Testing tools
  - Aalta-finite: LTL_f satisfiability checker;
  - Aalta-infinite: LTL satisfiability checker;
  - nuXmv (BMC and KLIVE): LTL Model Checker for the model-checking approach;
  - Z3: SMT solver for the SMT-based approach.
- Platform: NOTS cluster of Rice University;
- Time limit: 1 hour for each instance

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

Experimental Results

- Evaluating encoding (R benchmarks)

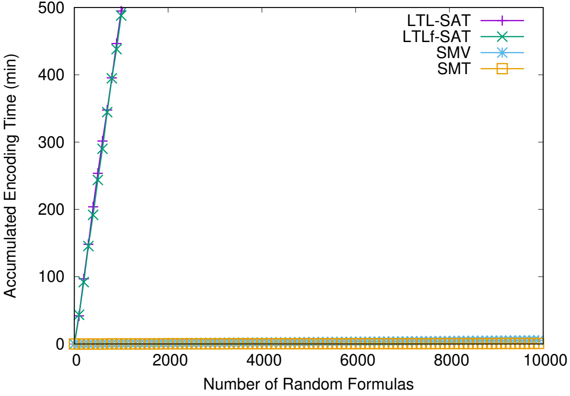


LTLSAT and LTL$_f$SAT lines overlap;
SMV and SMT lines overlap.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

Experimental Results

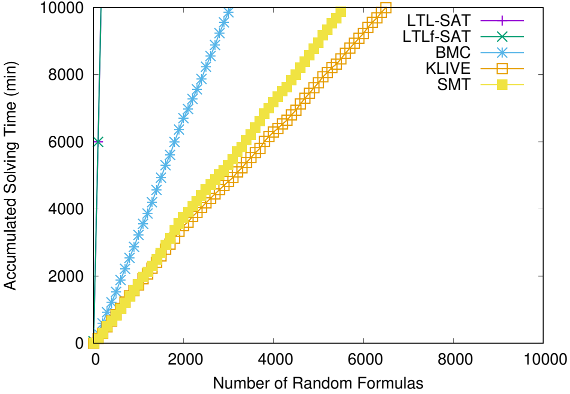
- Evaluating encoding (R benchmarks)



SMV encoding is more compact than LTL/LTL$_f$ encoding.

Experimental Results

- Evaluating solving (R benchmarks)

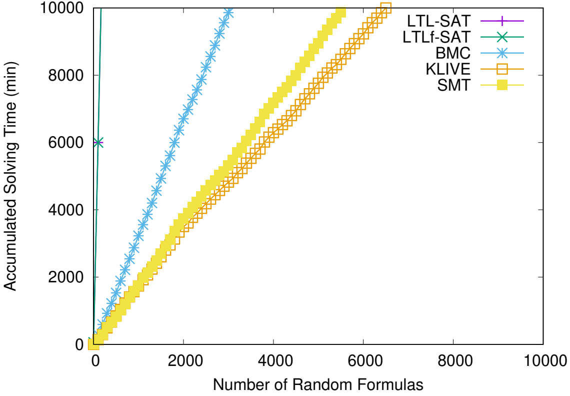


LTLSAT and LTL$_f$SAT lines overlap.

IOWA STATE UNIVERSITY Laboratory for Temporal Logic

Experimental Results

- Evaluating solving (R benchmarks)

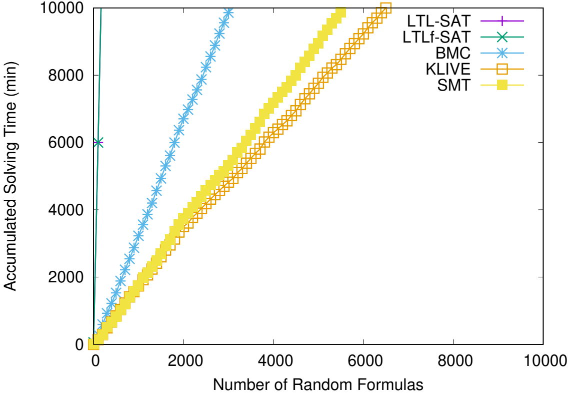


1. Reduction to LTLSAT/LTL$_f$SAT is not practical.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

Experimental Results

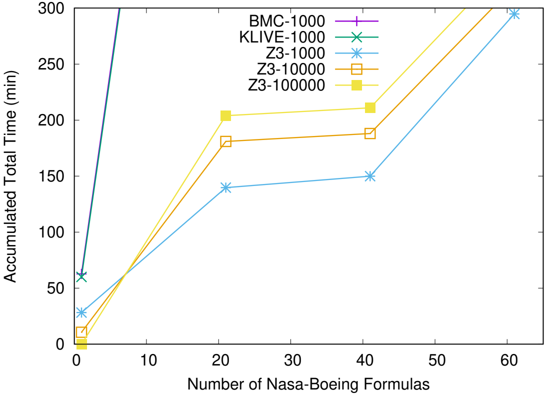
- Evaluating solving (R benchmarks)



2. KLIVE model checking performs best.

Experimental Results

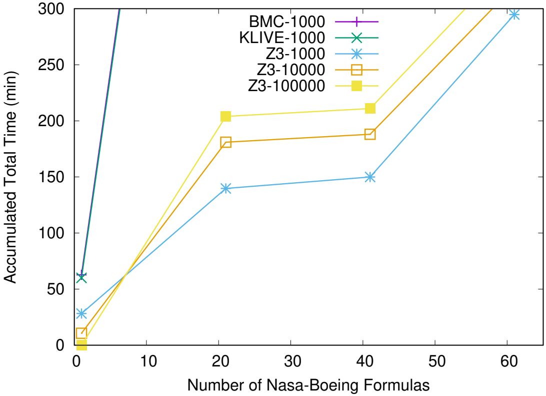
- Evaluating scalability (NB benchmarks)



BMC and KLIVE overlap.

Experimental Results

- Evaluating scalability (NB benchmarks)



3. The SMT approach is the most scalable.

IOWA STATE UNIVERSITY | Laboratory for Temporal Logic

Summary

- We prove MLTLSAT is NEXPTIME-complete;
- MLTLSAT via LTL_fSAT/LTLSAT is not practical at all;
- MLTLSAT via LTL model checking performs best when interval ranges are small;
- MLTLSAT via SMT has the most scalable performance;