Interval Temporal Logic

Angelo Montanari
Dept. of Mathematics, Computer Science, and Physics
University of Udine, Italy

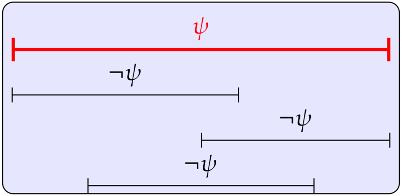
Dagstuhl Seminar 20071
Schloss Dagstuhl (Germany)
February 11, 2020

Outline of the talk

- A 2-page introduction to interval temporal logic (ITLs)
- The satisfiability checking problem for ITLs
- The model checking problem for ITLs
- Expressiveness: interval vs. point temporal logic model checking
- ITL model checking with regular expressions
- Recent and future developments
- An open question

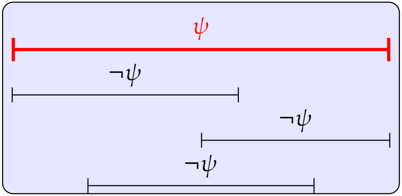
The distinctive features of interval temporal logics

Truth of formulae is defined over intervals (not points).



The distinctive features of interval temporal logics

Truth of formulae is defined over intervals (not points).

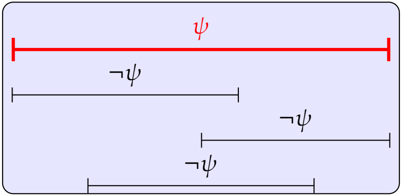


Interval temporal logics are very expressive (compared to point-based temporal logics). They allow one express actions/events with duration, accomplishments, temporal aggregations

In particular, formulas of interval logics express properties of pairs of time points rather than of single time points, and are evaluated as sets of such pairs, i.e., as binary relations

The distinctive features of interval temporal logics

Truth of formulae is defined over intervals (not points).



Interval temporal logics are very expressive (compared to point-based temporal logics). They allow one express actions/events with duration, accomplishments, temporal aggregations

In particular, formulas of interval logics express properties of pairs of time points rather than of single time points, and are evaluated as sets of such pairs, i.e., as binary relations

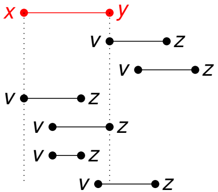
In general, there is no reduction of the satisfiability/validity in interval logics to monadic second-order logic, and therefore Rabin's theorem is not applicable here

HS: the modal logic of Allen's interval relations

Allen's interval relations: the 13 binary ordering relations between 2 intervals on a linear order. They give rise to corresponding unary modalities over frames where intervals are primitive entities:

- HS features a modality for any Allen ordering relation between pairs of intervals (except for equality)

Allen rel.	HS	Definition	Example
meets	$\langle A \rangle$	$[x,y]\mathcal{R}_A[v,z] \iff y = v$	
before	$\langle L \rangle$	$[x,y]\mathcal{R}_L[v,z] \iff y < v$	
started-by	$\langle B \rangle$	$[x,y]\mathcal{R}_B[v,z] \iff x = v \wedge z < y$	
finished-by	$\langle E \rangle$	$[x,y]\mathcal{R}_E[v,z] \iff y = z \wedge x < v$	
contains	$\langle D \rangle$	$[x,y]\mathcal{R}_D[v,z] \iff x < v \wedge z < y$	
overlaps	$\langle O \rangle$	$[x,y]\mathcal{R}_O[v,z] \iff x < v < y < z$	



All modalities can be expressed by means of $\langle A \rangle$, $\langle B \rangle$, $\langle E \rangle$, and their transposed modalities only (if point intervals are admitted, $\langle B \rangle$, $\langle E \rangle$, and their transposed modalities suffice)

(Un)decidability of HS fragments: main parameters

The satisfiability/validity problem for HS is highly undecidable over all standard classes of linear orders. What about its fragments?

(Un)decidability of HS fragments: main parameters

The satisfiability/validity problem for HS is highly undecidable over all standard classes of linear orders. What about its fragments?

Research agenda:

- search for maximal decidable HS fragments
- search for minimal undecidable HS fragments

(Un)decidability of HS fragments: main parameters

The satisfiability/validity problem for HS is highly undecidable over all standard classes of linear orders. What about its fragments?

Research agenda:

- search for maximal decidable HS fragments
- search for minimal undecidable HS fragments

The large majority of HS fragments turns out be undecidable, but some meaningful exceptions exist

(Un)decidability of HS fragments: main parameters

The satisfiability/validity problem for HS is highly undecidable over all standard classes of linear orders. What about its fragments?

Research agenda:

- search for maximal decidable HS fragments
- search for minimal undecidable HS fragments

The large majority of HS fragments turns out be undecidable, but some meaningful exceptions exist

(Un)decidability of HS fragments depends on two factors:

- the set of interval modalities
- the class of interval structures (linear orders) over which the logic is interpreted

A real character: the logic D

The logic D of the subinterval relation (Allen's relation *during*) is quite interesting from the point of view of (un)decidability

The satisfiability problem for D, interpreted over the class of dense linear orders, is PSPACE-complete

I. Shapirovsky, On PSPACE-decidability in Transitive Modal Logic, Advances in Modal Logic 2005

It is undecidable, when D is interpreted over the classes of finite and discrete linear orders

J. Marcinkowski and J. Michaliszyn, *The Ultimate Undecidability Result for the Halpern-Shoham Logic*, LICS 2011

It is unknown, when D is interpreted over the class of all linear orders

A real character: the logic D

The logic D of the subinterval relation (Allen's relation *during*) is quite interesting from the point of view of (un)decidability

The satisfiability problem for D, interpreted over the class of dense linear orders, is PSPACE-complete

I. Shapirovsky, On PSPACE-decidability in Transitive Modal Logic, Advances in Modal Logic 2005

It is undecidable, when D is interpreted over the classes of finite and discrete linear orders

J. Marcinkowski and J. Michaliszyn, *The Ultimate Undecidability Result for the Halpern-Shoham Logic*, LICS 2011

It is unknown, when D is interpreted over the class of all linear orders

The complete picture can be found at:
https://users.dimi.uniud.it/ angelo.montanari/Movep2016-partI.pdf

The interval way to model checking

Model checking: the desired properties of a system are checked against a model of it

- the model is usually a (finite) state-transition system
- system properties are specified by a temporal logic (LTL, CTL, CTL* and the like)

Distinctive features of model checking: (i) exaustive (check of all the possible behaviours), (ii) fully automatic process, amd (iii) a counterexample is produced for a violated property.

The interval way to model checking

Model checking: the desired properties of a system are checked against a model of it

- the model is usually a (finite) state-transition system
- system properties are specified by a temporal logic (LTL, CTL, CTL* and the like)

Distinctive features of model checking: (i) exaustive (check of all the possible behaviours), (ii) fully automatic process, amd (iii) a counterexample is produced for a violated property.

Model checking is usually point-based:

- properties express requirements over points (snapshots) of a computation (states of the state-transition system)
- they are specified by means of point-based temporal logics such as LTL, CTL, and CTL*

The interval way to model checking

Model checking: the desired properties of a system are checked against a model of it

- the model is usually a (finite) state-transition system
- system properties are specified by a temporal logic (LTL, CTL, CTL* and the like)

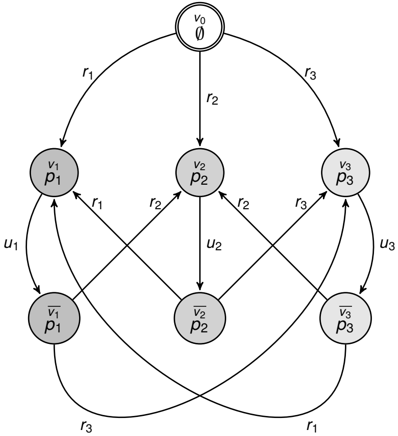
Distinctive features of model checking: (i) exaustive (check of all the possible behaviours), (ii) fully automatic process, amd (iii) a counterexample is produced for a violated property.

Model checking is usually point-based:

- properties express requirements over points (snapshots) of a computation (states of the state-transition system)
- they are specified by means of point-based temporal logics such as LTL, CTL, and CTL*

Interval properties express conditions on computation stretches instead of on computation states

The modeling of the system: Kripke structures



An example of Kripke structure

- ITL formulas are interpreted over (finite) state-transition systems, whose states are labeled with sets of proposition letters (Kripke structures)
- An interval is a trace (finite path) in a Kripke structure

HS semantics and model checking

Truth of a formula ψ over a trace ρ of a Kripke structure $\mathcal{K}$ defined by induction on the complexity of ψ:

- $\mathcal{K}, \rho \models p$ iff $p \in \bigcap_{w \in \text{states}(\rho)} \mu(w)$, for any letter $p \in \mathcal{AP}$, where $\mu(w)$ is the set of proposition letters true at w (homogeneity assumption);
- clauses for negation, disjunction, and conjunction are standard;
- $\mathcal{K}, \rho \models \langle A \rangle\, \psi$ iff there is a trace ρ' s.t. $\text{lst}(\rho) = \text{fst}(\rho')$ and $\mathcal{K}, \rho' \models \psi$;
- $\mathcal{K}, \rho \models \langle B \rangle\, \psi$ iff there is a proper prefix ρ' of ρ s.t. $\mathcal{K}, \rho' \models \psi$;
- $\mathcal{K}, \rho \models \langle E \rangle\, \psi$ iff there is a proper suffix ρ' of ρ s.t. $\mathcal{K}, \rho' \models \psi$;
- the semantic clauses for $\langle \overline{A} \rangle$, $\langle \overline{B} \rangle$, and $\langle \overline{E} \rangle$ are similar

Model Checking

$\mathcal{K} \models \psi \iff$ for all initial traces ρ of $\mathcal{K}$, it holds that $\mathcal{K}, \rho \models \psi$

Possibly infinitely many traces!

Decidability of model checking for HS

Theorem. The model checking problem for full HS on finite Kripke structures is decidable (with a *non-elementary* algorithm)

A. Molinari, A. Montanari, A. Murano, G. Perelli, and A. Peron, Checking Interval Properties of Computations, Acta Informatica, Special Issue: Temporal Representation and Reasoning (TIME'14), 56(6-8), 2016

Decidability of model checking for HS

Theorem. The model checking problem for full HS on finite Kripke structures is decidable (with a *non-elementary* algorithm)

A. Molinari, A. Montanari, A. Murano, G. Perelli, and A. Peron, Checking Interval Properties of Computations, Acta Informatica, Special Issue: Temporal Representation and Reasoning (TIME'14), 56(6-8), 2016

What about lower bounds?

Decidability of model checking for HS

Theorem. The model checking problem for full HS on finite Kripke structures is decidable (with a *non-elementary* algorithm)

A. Molinari, A. Montanari, A. Murano, G. Perelli, and A. Peron, Checking Interval Properties of Computations, Acta Informatica, Special Issue: Temporal Representation and Reasoning (TIME'14), 56(6-8), 2016

What about lower bounds?

Theorem. The model checking problem for BE, over finite Kripke structures, is EXPSPACE-hard

Bozzelli L., Molinari A., Montanari A., Peron A., and Sala P., "Which Fragments of the Interval Temporal Logic HS are Tractable in Model Checking?", Theoretical Computer Science, 764, 2019

Proof: a polynomial-time reduction from a domino-tiling problem for grids with rows of single exponential length

Decidability of model checking for HS

Theorem. The model checking problem for full HS on finite Kripke structures is decidable (with a *non-elementary* algorithm)

A. Molinari, A. Montanari, A. Murano, G. Perelli, and A. Peron, Checking Interval Properties of Computations, Acta Informatica, Special Issue: Temporal Representation and Reasoning (TIME'14), 56(6-8), 2016

What about lower bounds?

Theorem. The model checking problem for BE, over finite Kripke structures, is EXPSPACE-hard

Bozzelli L., Molinari A., Montanari A., Peron A., and Sala P., "Which Fragments of the Interval Temporal Logic HS are Tractable in Model Checking?", Theoretical Computer Science, 764, 2019

Proof: a polynomial-time reduction from a domino-tiling problem for grids with rows of single exponential length

The complete picture can be found at:
https://users.dimi.uniud.it/ angelo.montanari/TheIntervalWayTrieste2019.pdf

Point vs. interval temporal logic model checking

Question: is there any advantage in replacing points by intervals as the primary temporal entities, or is it just a matter of taste?

In order to compare the expressiveness of HS in model checking with those of LTL, CTL, and CTL*, we consider three semantic variants of HS:

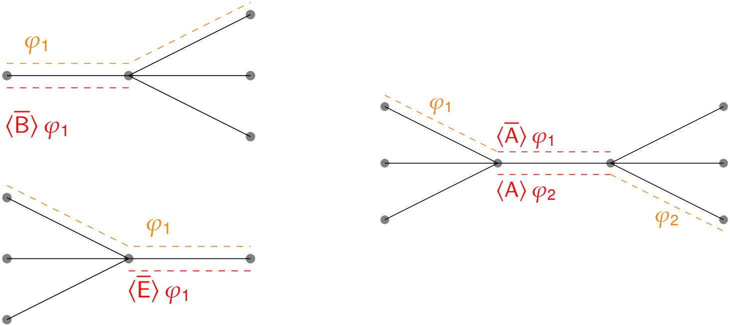
- HS with state-based semantics (the original one)
- HS with computation-tree-based semantics
- HS with trace-based semantics

These variants are compared with the above-mentioned standard temporal logics and among themselves

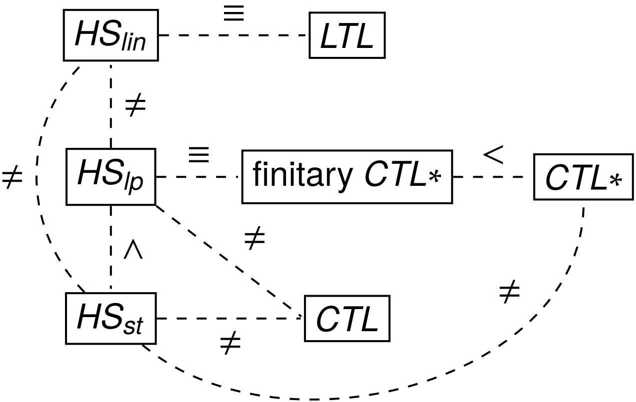
L. Bozzelli, A. Molinari, A. Montanari, A. Peron, and P. Sala, Interval vs. Point Temporal Logic Model Checking: an Expressiveness Comparison. ACM Transactions on Computational Logic, Volume 20(1), Article No. 4, 2019.

The state semantics of HS (HS_{st})

- According to the given semantics, HS modalities allow one to branch both in the past and in the future



The expressiveness picture



ITL model checking with regular expressions

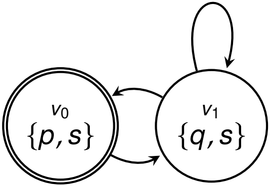
Can we relaxe the homogeneity assumption? The addition of regular expressions:

$$r ::= \varepsilon \mid \phi \mid r \cup r \mid r \cdot r \mid r^*$$

where ϕ is a Boolean (propositional) formula over $\mathcal{AP}$

Examples:

- $r_1 = (\mathbf{p} \wedge \mathbf{s}) \cdot \mathbf{s}^* \cdot (\mathbf{p} \wedge \mathbf{s})$
- $r_2 = (\neg \mathbf{p})^*$



- $\rho = v_0 v_1 v_0 v_1 v_1$
- $\mu(\rho) = \{p, s\}\{q, s\}\{p, s\}\{q, s\}\{q, s\}$
- $\rho' = v_0 v_1 v_1 v_1 v_0$
- $\mu(\rho') = \{p, s\}\{q, s\}\{q, s\}\{q, s\}\{p, s\}$
  - $\mu(\rho) \notin \mathcal{L}(r_1)$, but $\mu(\rho') \in \mathcal{L}(r_1)$
  - $\mu(\rho) \notin \mathcal{L}(r_2)$ and $\mu(\rho') \notin \mathcal{L}(r_2)$

ITL model checking with regular expressions

In the definition of the truth of a formula ψ over a trace ρ of a Kripke structure $\mathcal{K}$, we replace the clause for proposition letters by a clause for regular expressions:

- $\mathcal{K}, \rho \models r$ iff $\mu(\rho) \in \mathcal{L}(r)$

Homogeneity can be recovered as a special case. To force it, all regular expressions in the formula must be of the form:

$$p \cdot (p)^*$$

Solution: given $\mathcal{K}$ and an HS formula φ over $\mathcal{AP}$, we build a nondeterministic finite state automaton over $\mathcal{K}$ accepting the set of traces ρ such that $\mathcal{K}, \rho \models \varphi$

Bozzelli L., Molinari A., Montanari A., Peron A., "Model Checking Interval Temporal Logics with Regular Expressions", Information and Computation, 2020 (to appear).

Temporal logics of prefixes, suffixes, and infixes

The satisfiability/model checking problems for D (infixes) over finite linear orders, under the homogeneity assumption, are PSPACE-complete

L. Bozzelli, A. Molinari, A. Montanari, A. Peron, and P. Sala, Satisfiability and Model Checking for the Logic of Sub-Intervals under the Homogeneity Assumption, Proc. of the 44th International Colloquium on Automata, Languages, and Programming (ICALP), LIPIcs 80, 2017

The same problems for BD (prefixes and infixes) belong to EXPSPACE. The interplay of modalities B and D make the proofs harder (in preparation)

There is no a natural way to generalize the above proofs to BE (prefixes and suffixes)

Going beyond finite Kripke structures

We are looking for possible replacements of Kripke structures by more expressive system models

- inherently interval-based models, that allows one to describe systems on the basis of their interval behavior/properties, such as, e.g., those involving accomplishments, actions with duration, or temporal aggregations (no restriction on the evaluation of proposition letters)
  - timeline-based (planning) systems: a set of timelines (transition functions) plus a set of synchronization rules
- visibly pushdown systems, that can encode recursive programs and infinite state systems

L. Bozzelli, A. Montanari, and A. Peron, Interval Temporal Logic for Visibly Pushdown Systems, Proc. of the 39th Annual Conference on Foundations of Software Technology and Theoretical Computer Science, (FSTTCS), LIPIcs 150, 2019

Model checking a single interval model

A different direction: model checking a single interval model (for temporal dataset evaluation)

Complexity turns out to be much better: the problem of checking an HS formula against a single interval model (a finite history) can be solved by a deterministic algorithm that runs in polynomial time in the size of the input

D. Della Monica, D. de Frutos-Escrig, A. Montanari, A. Murano, and G. Sciavicco, Evaluation of Temporal Datasets via Interval Temporal Logic Model Checking, Proc. of the 24th International Symposium on Temporal Representation and Reasoning (TIME), LIPIcs 90, 2017

An open question

Is not interval temporal logic the right logic for composite event recognition?

Truth of formulae is defined over intervals (not points).