Interval Temporal Logic, Satisfiability and Model Checking

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

MOVEP 2016
Genova (Italy), July 1st, 2016

Interval Temporal Logic, Satisfiability and Model Checking

Part II: model checking

- introduction to interval temporal logic model checking
- the general picture
- the case of the logic $A\overline{A}B\overline{BE}$
- what's next?

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

Model checking

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

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

Distinctive features of model checking:

- exaustive check of all the possible behaviours
- fully automatic process
- a counterexample is produced for a violated property

Point-based vs. interval-based model checking

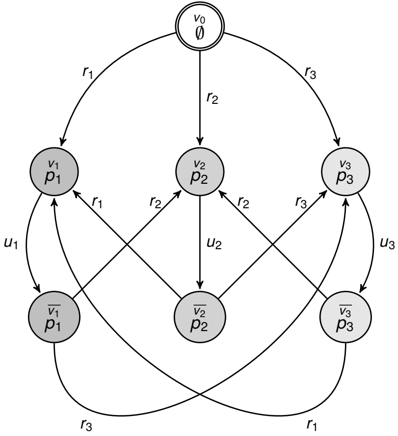
Model checking is usually point-based:

- 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 and CTL

Interval-based properties express conditions on computation stretches, e.g., accomplishments, actions with duration, and temporal aggregations

Little work has been done on interval temporal logic (ITL) model checking (Bozzelli, Lomuscio, Michaliszyn, Molinari, Montanari, Murano, Perelli, Peron, Sala)

Kripke structures



A finite Kripke structure $\mathcal{K}_{Sched}$

- HS 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

A short account of $\mathcal{K}_{Sched}$

$\mathcal{K}_{Sched}$ models the behaviour of a scheduler serving 3 processes which are continuously requesting the use of a common resource

Initial state: v_0 (no process is served in that state)

In v_i and $\overline{v}_i$ the i-th process is served (p_i holds in those states)

The scheduler cannot serve the same process twice in two successive rounds:

- process i is served in state v_i, then, after “some time”, a transition u_i from v_i to $\overline{v}_i$ is taken; subsequently, process i cannot be served again immediately, as v_i is not directly reachable from $\overline{v}_i$
- a transition r_j, with $j \neq i$, from $\overline{v}_i$ to v_j is then taken and process j is served

It can be easily generalised to an arbitrary number of processes

Some meaningful properties to be checked over $\mathcal{K}_{Sched}$

Validity of properties over all legal computation intervals can be forced by modality $[E]$ (they are suffixes of at least one initial trace)

Property 1: in any computation interval of length at least 4, at least 2 processes are witnessed (YES/no process can be executed twice in a row)

$$\mathcal{K}_{Sched} \models [E](\langle \mathrm{E} \rangle^3 \top \rightarrow (\chi(p_1, p_2) \vee \chi(p_1, p_3) \vee \chi(p_2, p_3))),$$

where $\chi(p, q) = \langle \mathrm{E} \rangle \langle \overline{\mathrm{A}} \rangle p \wedge \langle \mathrm{E} \rangle \langle \overline{\mathrm{A}} \rangle q$

Property 2: in any computation interval of length at least 11, process 3 is executed at least once (NO/the scheduler can postpone the execution of a process ad libitum)

$$\mathcal{K}_{Sched} \not\models [E](\langle \mathrm{E} \rangle^{10} \top \rightarrow \langle \mathrm{E} \rangle \langle \overline{\mathrm{A}} \rangle p_3)$$

Property 3: in any computation interval of length at least 6, all processes are witnessed (NO/the scheduler should be forced to execute them in a strictly periodic manner, which is not the case)

$$\mathcal{K}_{Sched} \not\models [E](\langle \mathrm{E} \rangle^5 \rightarrow (\langle \mathrm{E} \rangle \langle \overline{\mathrm{A}} \rangle p_1 \wedge \langle \mathrm{E} \rangle \langle \overline{\mathrm{A}} \rangle p_2 \wedge \langle \mathrm{E} \rangle \langle \overline{\mathrm{A}} \rangle p_3))$$

HS semantics and model checking

Truth of a formula ψ over a trace ρ of a Kripke structure $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ defined by induction on the complexity of ψ:

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

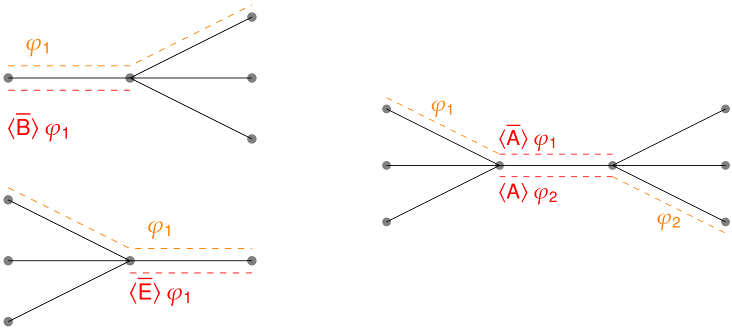
Model Checking

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

Possibly infinitely many traces!

Remark: HS state semantics

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



The key notion: BE_k-descriptor

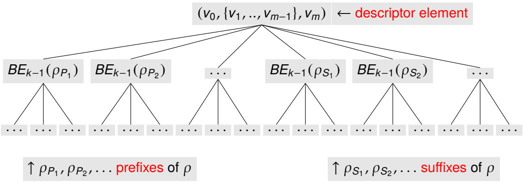
- The BE-nesting depth of an HS formula ψ ($\text{Nest}_{\text{BE}}(\psi)$) is the maximum degree of nesting of modalities B and E in ψ
- Two traces ρ and ρ' of a Kripke structure $\mathcal{K}$ are k-equivalent if and only if $\mathcal{K}, \rho \models \psi$ iff $\mathcal{K}, \rho' \models \psi$ for all HS-formula ψ with $\text{Nest}_{\text{BE}}(\psi) \leq k$

The key notion: BE_k-descriptor

- The BE-nesting depth of an HS formula ψ ($\text{Nest}_{\text{BE}}(\psi)$) is the maximum degree of nesting of modalities B and E in ψ
- Two traces ρ and ρ' of a Kripke structure $\mathcal{K}$ are k-equivalent if and only if $\mathcal{K}, \rho \models \psi$ iff $\mathcal{K}, \rho' \models \psi$ for all HS-formula ψ with $\text{Nest}_{\text{BE}}(\psi) \leq k$

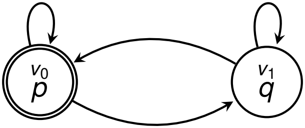
We provide a suitable tree representation for a trace, called a BE_k-descriptor

The BE_k-descriptor for a trace $\rho = v_0 v_1 .. v_{m-1}\ v_m$, denoted $BE_k(\rho)$, is defined as follows:

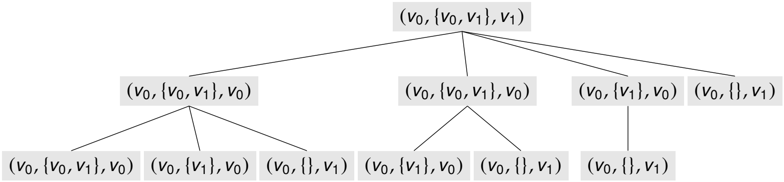


Remark: the descriptor has not sibling isomorphic subtrees

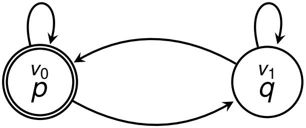
An example of a BE_2-descriptor



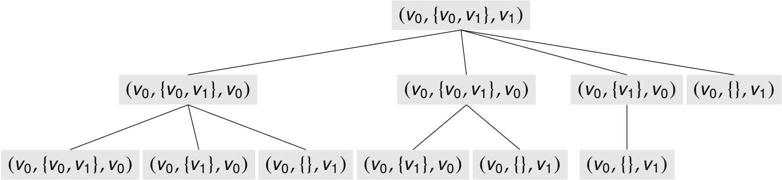
The BE_2-descriptor for the trace $\rho = v_0 v_1 v_0^4 v_1$ (for the sake of readability, only the subtrees for prefixes are displayed)



An example of a BE_2-descriptor



The BE_2-descriptor for the trace $\rho = v_0 v_1 v_0^4 v_1$ (for the sake of readability, only the subtrees for prefixes are displayed)



Remark: the subtree to the left is associated with both prefixes $v_0 v_1 v_0^3$ and $v_0 v_1 v_0^4$ (the descriptor has not sibling isomorphic subtrees)

Decidability of model checking for full HS

FACT 1: For any Kripke structure $\mathcal{K}$ and any BE-nesting depth $k \geq 0$, the number of different BE_k-descriptors is finite (and thus at least one descriptor has to be associated with infinitely many traces)

Decidability of model checking for full HS

FACT 1: For any Kripke structure $\mathcal{K}$ and any BE-nesting depth $k \geq 0$, the number of different BE_k-descriptors is finite (and thus at least one descriptor has to be associated with infinitely many traces)

FACT 2: Two traces ρ and ρ' of a Kripke structure $\mathcal{K}$ described by the same BE_k descriptor are k-equivalent

Decidability of model checking for full HS

FACT 1: For any Kripke structure $\mathcal{K}$ and any BE-nesting depth $k \geq 0$, the number of different BE_k-descriptors is finite (and thus at least one descriptor has to be associated with infinitely many traces)

FACT 2: Two traces ρ and ρ' of a Kripke structure $\mathcal{K}$ described by the same BE_k descriptor are k-equivalent

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 (to appear)

Decidability of model checking for full HS

FACT 1: For any Kripke structure $\mathcal{K}$ and any BE-nesting depth $k \geq 0$, the number of different BE_k-descriptors is finite (and thus at least one descriptor has to be associated with infinitely many traces)

FACT 2: Two traces ρ and ρ' of a Kripke structure $\mathcal{K}$ described by the same BE_k descriptor are k-equivalent

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 (to appear)

What about lower bounds?

The logic *BE*

Theorem

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

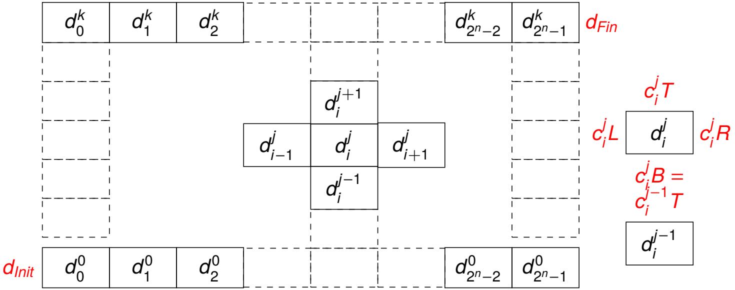
L. Bozzelli, A. Molinari, A. Montanari, A. Peron, and P. Sala, Interval Temporal Logic Model Checking: The Border Between Good and Bad HS Fragments, IJCAR 2016

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

- for an instance $\mathcal{I}$ of the problem, we build a Kripke structure $\mathcal{K}_{\mathcal{I}}$ and a formula $\varphi_{\mathcal{I}}$ in polynomial time
- there is an initial trace of $\mathcal{K}_{\mathcal{I}}$ satisfying $\varphi_{\mathcal{I}}$ iff there is a tiling of $\mathcal{I}$
- $\mathcal{K}_{\mathcal{I}} \models \neg\varphi_{\mathcal{I}}$ iff there exists no tiling of $\mathcal{I}$

BE hardness: encoding of the domino-tiling problem

Instance of the tiling problem: $(C, \Delta, n, d_{init}, d_{final})$, with C a finite set of colors and $\Delta \subseteq C \times C \times C \times C$ a set of tuples (c_B, c_L, c_T, c_R)

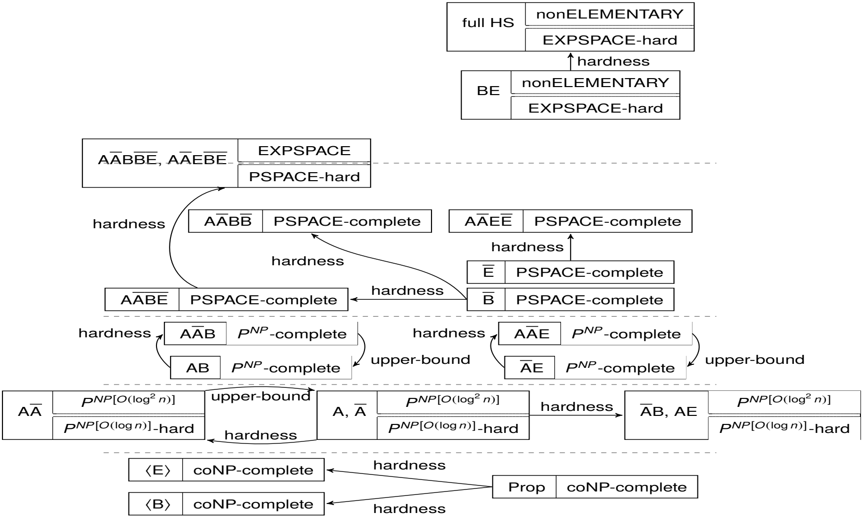


String (interval) encoding of the problem

row 0 — row 1

d^0_0	$0\cdots00$	d^0_1	$1\cdots00$	$\cdots$	$d^0_{2^n-1}$	$1\cdots11$	\$	d^1_0	$0\cdots00$	d^1_1	$1\cdots00$	$\cdots$	$d^1_{2^n-1}$	$1\cdots11$	\$
column 0		column 1			column 2^n-1			column 0		column 1			column 2^n-1		

The general picture



Three main gaps to fill

The picture shows that there three main gaps to fill:

- full HS and BE are in between nonELEMENTARY and EXPSPACE
- $A\overline{A}B\overline{BE}$, $A\overline{A}E\overline{BE}$, $AB\overline{BE}$, $AE\overline{BE}$, $\overline{A}B\overline{BE}$, and $\overline{A}E\overline{BE}$ are in between EXPSPACE and PSPACE
- A, $\overline{A}$, $A\overline{A}$, $\overline{A}B$, and AE are in between $P^{NP[O(\log^2 n)]}$ and $P^{NP[O(\log n)]}$

The logic $A\overline{A}B\overline{B}\overline{E}$

Let us consider the case of the logic $A\overline{A}B\overline{B}\overline{E}$, which is obtained from full HS ($A\overline{A}BE\overline{B}\overline{E}$) by removing modality $\langle E\rangle$

The logic $A\overline{A}B\overline{BE}$

Let us consider the case of the logic $A\overline{A}B\overline{BE}$, which is obtained from full HS ($A\overline{A}BE\overline{BE}$) by removing modality $\langle \mathrm{E} \rangle$

Some fundamental facts:

- we can restrict our attention on prefixes (B_k-descriptors suffice)

The logic $A\overline{A}B\overline{BE}$

Let us consider the case of the logic $A\overline{A}B\overline{BE}$, which is obtained from full HS ($A\overline{A}BE\overline{BE}$) by removing modality $\langle \mathrm{E} \rangle$

Some fundamental facts:

- we can restrict our attention on prefixes (B_k-descriptors suffice)
- the size of the tree representation of B_k-descriptors is larger than necessary (redundancy) and it prevents their efficient exploitation in model checking algorithms

The logic $A\overline{A}B\overline{BE}$

Let us consider the case of the logic $A\overline{A}B\overline{BE}$, which is obtained from full HS ($A\overline{A}BE\overline{BE}$) by removing modality $\langle \mathrm{E} \rangle$

Some fundamental facts:

- we can restrict our attention on prefixes (B_k-descriptors suffice)
- the size of the tree representation of B_k-descriptors is larger than necessary (redundancy) and it prevents their efficient exploitation in model checking algorithms
- a trace representative can be chosen to represent a (possibly infinite) set of traces with the same B_k-descriptor

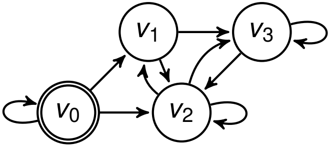
The logic $A\overline{A}B\overline{BE}$

Let us consider the case of the logic $A\overline{A}B\overline{BE}$, which is obtained from full HS ($A\overline{A}BE\overline{BE}$) by removing modality $\langle \mathrm{E} \rangle$

Some fundamental facts:

- we can restrict our attention on prefixes (B_k-descriptors suffice)
- the size of the tree representation of B_k-descriptors is larger than necessary (redundancy) and it prevents their efficient exploitation in model checking algorithms
- a trace representative can be chosen to represent a (possibly infinite) set of traces with the same B_k-descriptor
- a bound, which depends on both the number $|W|$ of states of the Kripke structure and the B-nesting depth k, can be given to the length of trace representatives

Traces and sequences of descriptor elements



Let us consider the trace
$\rho = v_0 v_0 v_0 v_1 v_2 v_1 v_2 v_3 v_3 v_2 v_2$

The descriptor element $DElem(\rho)$ for ρ:

$$(v_0, \{v_0, v_1, v_2, v_3\}, v_2)$$

The descriptor sequence ρ_{ds} for ρ (Δ_i stands for $\{v_0, \ldots, v_i\}$):

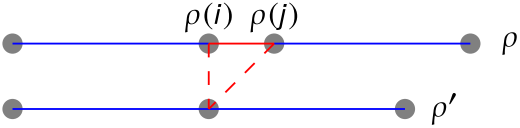
$$(v_0, \emptyset, v_0)(v_0, \Delta_0, v_0)(v_0, \Delta_0, v_1)(v_0, \Delta_1, v_2)(v_0, \Delta_2, v_1)(v_0, \Delta_2, v_2)$$
$$(v_0, \Delta_2, v_3)(v_0, \Delta_3, v_3)(v_0, \Delta_3, v_2)(v_0, \Delta_3, v_2)$$

The descriptor sequence is the sequence of the descriptor elements for ρ and for its prefixes in increasing order (from the one for the shortest prefix to the one for the whole trace)

A contraction method

- Repeated occurrences of the same descriptor element in a descriptor sequence ρ_{ds} represent prefixes of a trace which unwind a loop in a Krypke structure
- Two occurrences of the same descriptor element in a descriptor sequence ρ_{ds} are k-indistinguishable if the associated trace prefixes have the same B_k-descriptor
- If two repeated occurrences are k-indistiguinshable, we can contract the trace avoiding the second repetition

$$B_k(\rho(0,i)) = B_k(\rho(0,j))$$



$$B_k(\rho) = B_k(\rho')$$

Boundedness theorem

- given a trace ρ, we can repeatedly contract it until it has not occurrences of k-indistiguishable descriptor elements

Theorem (Boundedness theorem)

If ρ is a trace of a Kripke structure $\mathcal{K}$, with set of states W, and $k \geq 0$, then there exists a trace ρ', with the same B_k-descriptor as ρ, such that

$$|\rho'| \leq \tau(|W|, k) = min \left\{ \begin{array}{c} 1 + (1 + |W|)^{2k+4} + |W| \\ 1 + (k+3)^{|W|^2+1} + |W| \end{array} \right\}$$

- if $|\rho| > \tau(|W|, k)$, then ρ necessarily has some occurrences of k-indistiguishable descriptor elements
- termination criterion: when enumerating traces, it is enough to consider traces of length less then or equal to $\tau(|W|, k)$

The model checking algorithm

ModCheck$(\mathcal{K}, \psi)$

$k \leftarrow \text{Nest}_\text{B}(\psi)$
$u \leftarrow$ *New* $(\texttt{Unravel\_from}(\mathcal{K}, \texttt{init\_state}(\mathcal{K}), k, \text{FORWARD}))$
while $u.\texttt{hasMoreTraces}()$ **do**
  $\overline{\rho} \leftarrow u.\texttt{getNextTrace}()$
  if $\texttt{Check}(\mathcal{K}, k, \psi, \overline{\rho}) = 0$ **then return** 0: "$\mathcal{K}, \overline{\rho} \not\models \psi$"
return 1: "$\mathcal{K} \models \psi$" =0

EXPSPACE: $(|\psi| + 1) \cdot O(|W| + \text{Nest}_\text{B}(\psi)) \cdot \tau(|W|, \text{Nest}_\text{B}(\psi))$ bits

A. Molinari, A. Montanari, and A. Peron, A Model Checking Procedure for Interval Temporal Logics based on Track Representatives, CSL 2015

PSPACE-hardness of $A\overline{A}B\overline{B}\overline{E}$ model checking

PSPACE-hardness of the model checking problem for the fragment $\overline{B}$ (and thus for $A\overline{A}B\overline{B}\overline{E}$) can be proved by a reduction from the QBF problem

Theorem

The model checking problem for $\overline{B}$, and thus for $A\overline{A}B\overline{B}\overline{E}$, over finite Kripke structures is PSPACE-hard.

Remark: $A\overline{A}B\overline{B}\overline{E}$ model checking is in between PSPACE and EXPSPACE (remind: BE is EXPSPACE-hard)

A. Molinari, A. Montanari, A. Peron, and P. Sala, Model Checking Well-Behaved Fragments of HS: The (Almost) Final Picture, KR 2016

Current research agenda

- To complete the picture of interval temporal logic model checking under the homogeneity assumption (and the HS state semantics)

- To explore alternative HS semantics. In particular, the trace semantics, where the infinite paths (computations) of the Kripke structure are the main semantic entities, and the computation tree semantics, where future is branching, but past is linear (as well as finite and cumulative). Trace (resp., computation tree) semantics allows us to establish a bridge between HS (model checking) and LTL (resp., CTL) (model checking)

- To remove the homogeneity assumption

Epistemic HS (Lomuscio and Michaliszyn)

Distinctive feature of Epistemic HS (EHS for short): the labelling function is defined on the endpoints of the (finite) traces/intervals

Lomuscio and Michaliszyn proved that the local model checking problem (verification of a given specification against a single initial trace) for the fragment $EHS[BE]$ is PSPACE-complete

If epistemic modalities are removed, it is in PTIME (notice that modalities B and E allow one to access only sub-intervals of the given initial one, whose number is quadratic in the length of it)

A. Lomuscio and J. Michaliszyn, An Epistemic Halpern-Shoham Logic, IJCAI 2013

Epistemic HS (Lomuscio and Michaliszyn) - cont'd

Later on, they showed that the picture drastically changes with other fragments of HS that allow one to access infinitely many traces

They proved that the model checking problem for the HS fragment $A\overline{B}$, extended with epistemic modalities, is decidable, with a non-elementary upper bound

Notice that formulas of this logic can possibly refer to infinitely many (future) traces

A. Lomuscio and J. Michaliszyn, Decidability of model checking multi-agent systems against a class of EHS specifications, ECAI 2014

Epistemic HS (Lomuscio and Michaliszyn) - cont'd

In their most recent contribution, Lomuscio and Michaliszyn generalized the labeling function by allowing it to be given by any regular expression on the states of intervals

Such a generalization results in a considerable increase in the expressiveness of the specifications at no computational cost in terms of the corresponding model checking problem

A. Lomuscio and J. Michaliszyn, Model Checking Multi-Agent Systems against Epistemic HS Specifications with Regular Expressions, KR 2016

Mid- and long-term research agenda

- Systematic application of game-theoretic techniques in interval-based synthesis
- Quest for automaton-based techniques for interval temporal logic satisfiability and model checking
- Application of interval temporal logics to

  (i) system specification, verification, and synthesis

  (ii) planning and plan validation (to represent and to reason about actions/events with duration, accomplishments, and interval constraints)

  (iii) temporal databases (to deal with temporal aggregation) and workflow systems (to cope with additional temporal constraints)
- Application of interval temporal logic model checking to infinite state systems

People

- **Aceto, Luca; Ingólfsdóttir, Anna** — Reykjavik University, Iceland
- **Bozzelli, Laura** — Universidad Politécnica de Madrid, Spain
- **Bresolin, Davide** — University of Bologna, Italy
- **Conradie Willem** — University of Johannesburg, South Africa
- **Della Monica, Dario; Murano, Nello; Peron Adriano** — University of Napoli, Italy
- **Goranko, Valentin** — Stockholm University, Sweden
- **Hodkinson, Ian; Lomuscio, Alessio; Perelli Giuseppe** — Imperial College, UK
- **Kieroński, Emanuel; Marcinkowski, Jerzy; Michaliszyn, Jakub** — University of Wroclaw, Poland
- **Molinari Alberto; Montanari, Angelo; Vitacolonna Nicola** — University of Udine, Italy
- **Emilio Muñoz-Velasco** — University of Málaga, Spain
- **Pratt-Hartmann, Ian** — University of Manchester, UK
- **Puppis, Gabriele** — CNRS researcher at LaBRI, France
- **Sala, Pietro** — University of Verona, Italy
- **Sciavicco, Guido** — University of Ferrara, Italy
- and others (**Artale Alessandro, Durhan Salih, Kontchakov Roman , Ryzhikov Vladislav, Zakharyaschev Michael**, ...)