Real Time Properties for Interrupt Timed Automata

B. Bérard[†] S. Haddad[‡] M. Sassolas[†]

[†]UPMC, LIP6/MoVe, CNRS UMR 7606, Paris, France
[‡]ENS de Cachan, LSV, CNRS UMR 8643, Cachan, France

MeFoSyLoMa
June 18, 2010

Outline

1. The context: timed and hybrid systems
2. The Interrupt Timed Automata Model
3. The model checking problem
4. Decidable fragments
5. Conclusion

Outline

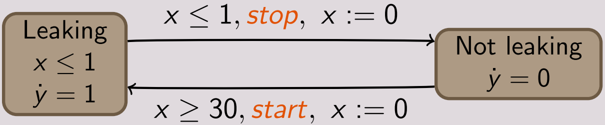
1. The context: timed and hybrid systems
2. The Interrupt Timed Automata Model
3. The model checking problem
4. Decidable fragments
5. Conclusion

Context

Modelling and verification of hybrid systems

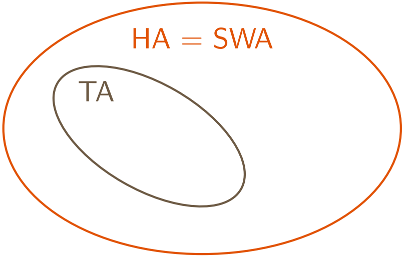
- Hybrid automaton = finite automaton + variables
  - Variables evolve in states and can be tested and updated on transitions.
  - Clocks are variables with slope 1 in all states
  - Stopwatches are variables with slope 0 or 1
- Timed automaton = finite automaton + clocks with guards $x + c \bowtie 0$ and resets $x := 0$

Example (The gas burner)



Previous results

Hybrid automata = Stopwatch automata [Cassez, Larsen 2000]



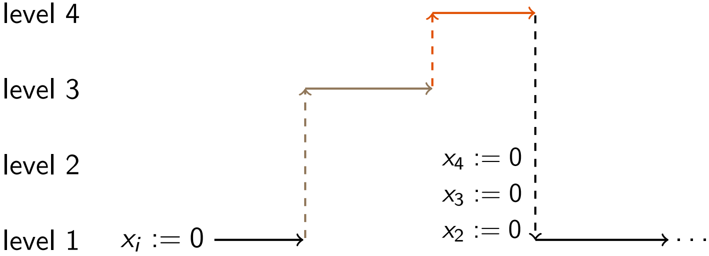
Timed automata

- The reachability problem is undecidable for a timed automaton with one stopwatch [Henzinger et al. 1998].
- Model checking timed automata with stopwatch observers is undecidable for WCTL (a weighted extension of CTL) [Bouyer et al. 2006].
- Reachability and model checking TCTL is decidable on TA [Alur, Dill 1990] [Alur, Courcoubetis, Dill 1993].

Motivations

- Theoretical
  - To express more than timed automata
  - To obtain decidability results
- Practical
  - In operating systems, tasks are scheduled according to their priority level.
  - A higher priority task can interrupt a lower priority task.
- An interrupt clock can be seen as a restricted type of stopwatch: only one evolves at a given time.

Clock interruptions

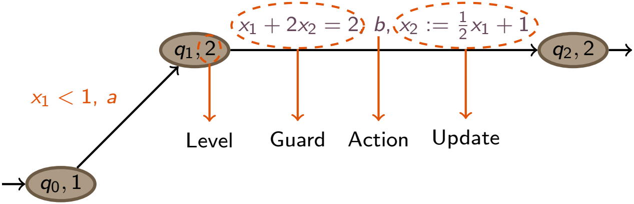


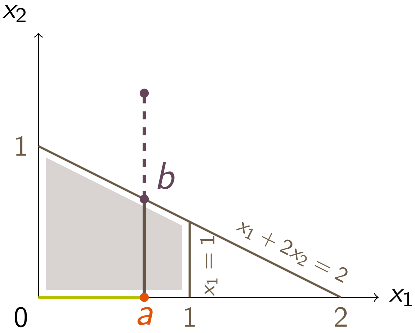
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{1.5} \begin{bmatrix} 1.5 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{2.1} \begin{bmatrix} 1.5 \\ 0 \\ 2,1 \\ 0 \end{bmatrix} \xrightarrow{1.7} \begin{bmatrix} 1.5 \\ 0 \\ 2.1 \\ 1.7 \end{bmatrix} \xrightarrow{2.2} \begin{bmatrix} 3.7 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Outline

1. The context: timed and hybrid systems
2. The Interrupt Timed Automata Model
3. The model checking problem
4. Decidable fragments
5. Conclusion

Interrupt Timed Automata

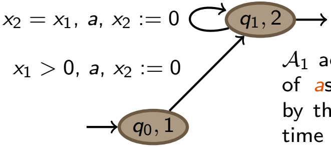




ITA and TA are incomparable

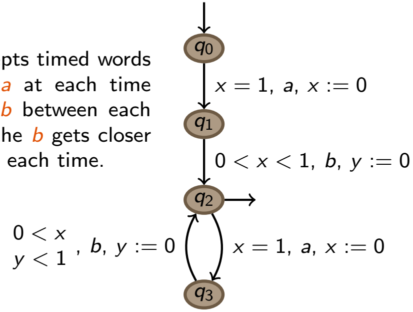
ITA $\mathcal{A}_1$ cannot be simulated by a TA

$(a, 0.7)\ (a, 1.4)\ (a, 2.1)$
$(a, 2.8)\ (a, 3.5)\ (a, 4.2)$



$\mathcal{A}_1$ accepts words made of as separated always by the same amount of time

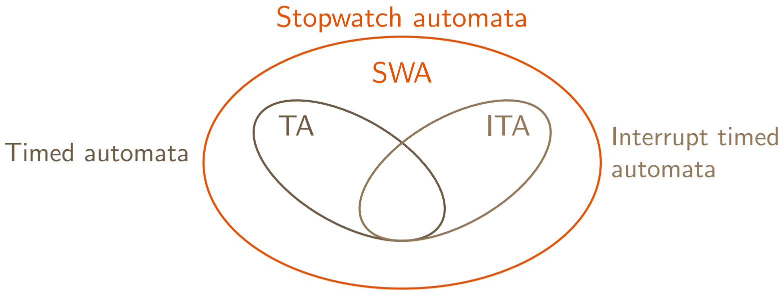
$\mathcal{A}_2$ accepts timed words with a a at each time unit, a b between each a, and the b gets closer to the a each time.



TA $\mathcal{A}_2$ cannot be simulated by an ITA

$(a, 1)\ (b, 1.87)$
$(a, 2)\ (b, 2.42)$
$(a, 3)\ (b, 3.37)$
$(a, 4)\ (b, 4.23)$

Expressiveness and decidability trade-off



Previous results

- *SWA: Reachability and model checking undecidable*
- *TA: Reachability and model checking decidable*
- *ITA: Reachability decidable*

What about model checking on ITA ?

Outline

1 The context: timed and hybrid systems

2 The Interrupt Timed Automata Model

3 The model checking problem

4 Decidable fragments

5 Conclusion

Timed CTL

- “No error in the first 50 time units”

  $$y.(\mathsf{A}\,\neg\,\mathrm{error}\,\mathsf{U}\,y > 50)$$

- “A normal state is reached when the clock of level 2 is greater than the one of level 1”

  $$\mathsf{E}\,\top\,\mathsf{U}\,\mathrm{normal} \wedge x_2 \geq x_1 \text{ or } \mathsf{EF}\,\mathrm{normal} \wedge x_2 \geq x_1$$

- “We never leave level 1 for more than 5 time units”

  $$\mathsf{AG}\,(\neg\ell_1 \Rightarrow z.(\mathsf{AF}\,\ell_1 \wedge z < 5))$$

- Timed CTL with explicit clocks:

  $$\psi ::= p \mid y + b \bowtie 0 \mid \sum_{i\in I} a_i \cdot x_i + b \bowtie 0 \mid y.\psi \mid$$

  $$\mathsf{A}\,\psi\,\mathsf{U}\,\psi \mid \mathsf{E}\,\psi\,\mathsf{U}\,\psi \mid \psi \wedge \psi \mid \neg\psi$$

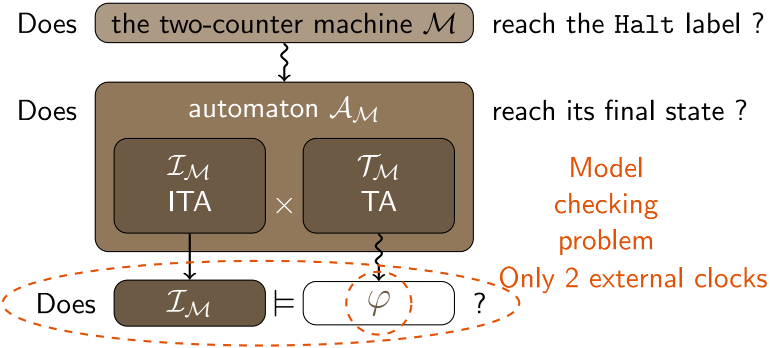
- Given a formula φ and an ITA $\mathcal{A}$, does $\mathcal{A} \models \varphi$?

Theorem

Model checking TCTL formula on ITA is undecidable.

Model checking TCTL on ITA is undecidable

- A two-counter machine: for e $\in$ {c, d}
  - “`e++ goto l`”,
  - “`if e > 0 then e-- goto l1 else goto l2`”,
  - “`Halt`”.
- The halting problem of a two-counter machine is undecidable



Does the two-counter machine $\mathcal{M}$ reach the `Halt` label ?

Does automaton $\mathcal{A}_{\mathcal{M}}$ reach its final state ?

$\mathcal{I}_{\mathcal{M}}$ ITA × $\mathcal{T}_{\mathcal{M}}$ TA

Model checking problem

Only 2 external clocks

Does $\mathcal{I}_{\mathcal{M}} \models \varphi$?

Outline

1. The context: timed and hybrid systems
2. The Interrupt Timed Automata Model
3. The model checking problem
4. Decidable fragments
5. Conclusion

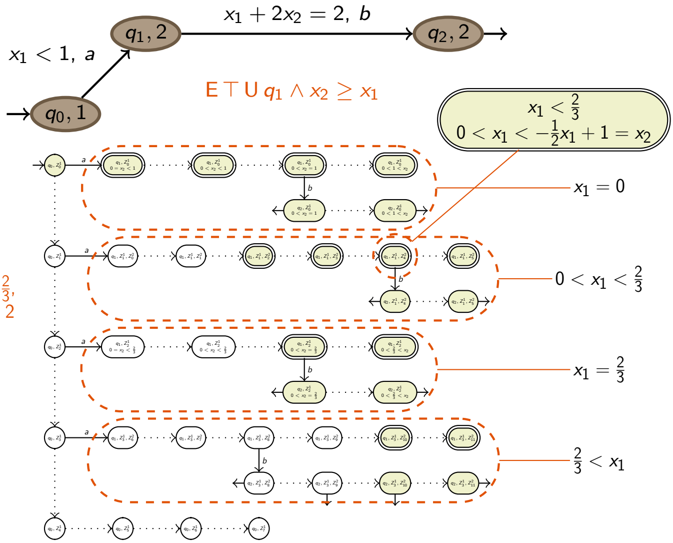
TCTL without external clocks

- Only $\sum_{i\in I} a_i \cdot x_i + b \bowtie 0$ comparisons.
- For example $\mathsf{E}\,\top\,\mathsf{U}\,\mathrm{normal} \wedge x_2 \geq x_1$
- The truth value of the comparison can be abstracted by *regions*.
- A classical CTL model checking algorithm can be applied.

Theorem

Model checking TCTL without external clocks on ITA can be done in 2-EXPSPACE and PSPACE when the number of clocks is fixed.

Example of model-checking procedure



when $x_1 = \frac{2}{3}$, $x_1 + 2x_2 = 2$ $\Updownarrow$ $x_1 = x_2$

A fragment of TCTL with only one external clock

- A particular case of TCTL with 1 external clock.
- Clock conditions can only restrict the *Until* operator with urgency ($y \leq b$ or $y < b$) or delay ($y \geq b$ or $y > b$).
- There can be no imbrication of *Until*s.
- For example $y.(\mathsf{A}\, \neg\, \mathrm{error}\, \mathsf{U}\, y > 50)$

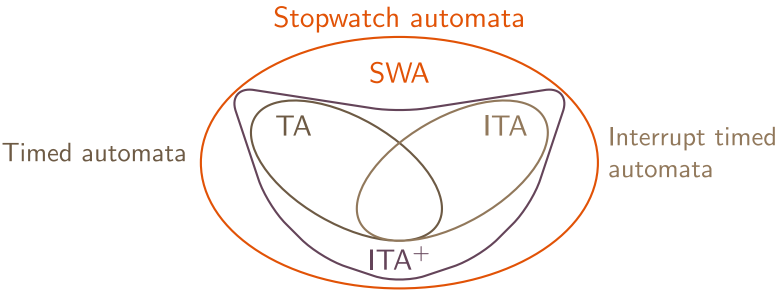
Theorem

Model checking this fragment of TCTL on ITA is decidable.

Outline

1. The context: timed and hybrid systems
2. The Interrupt Timed Automata Model
3. The model checking problem
4. Decidable fragments
5. Conclusion

Summary and future work



- ITA allow reasoning on systems with interruptions.
- Its expressive power is incomparable with the TA model.
- Unfortunately model checking of full TCTL is impossible.
- Nevertheless some interesting fragments are still decidable.

Thank you

Any questions ?