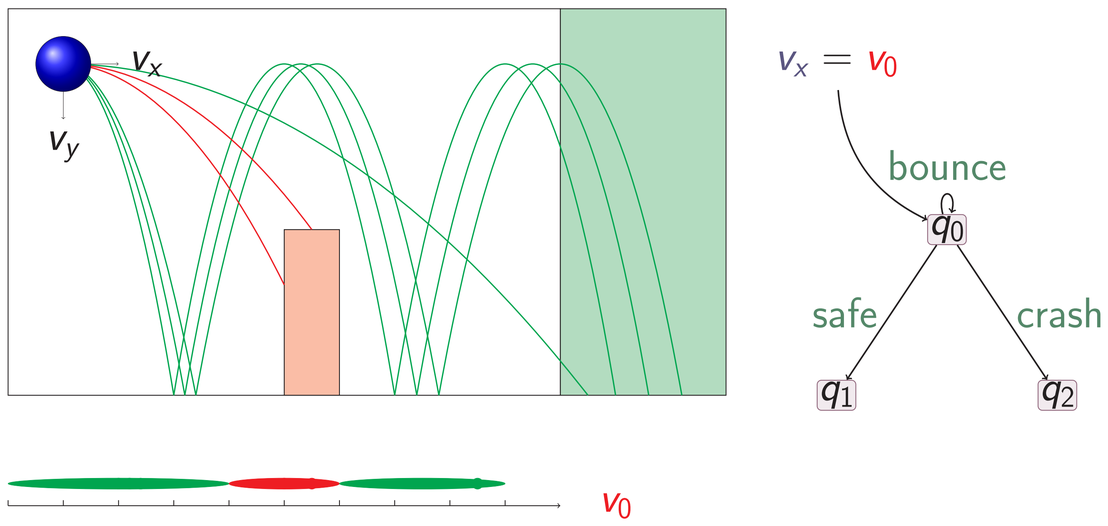
HyMITATOR
Parametric Analysis of Hybrid Systems

Étienne André[1] and Ulrich Kühne[2]

[1]LIPN, CNRS UMR 7030, Université Paris 13, France
[2]Group for Computer Architecture, University of Bremen, Germany

An Example of Hybrid System: The Bouncing Ball

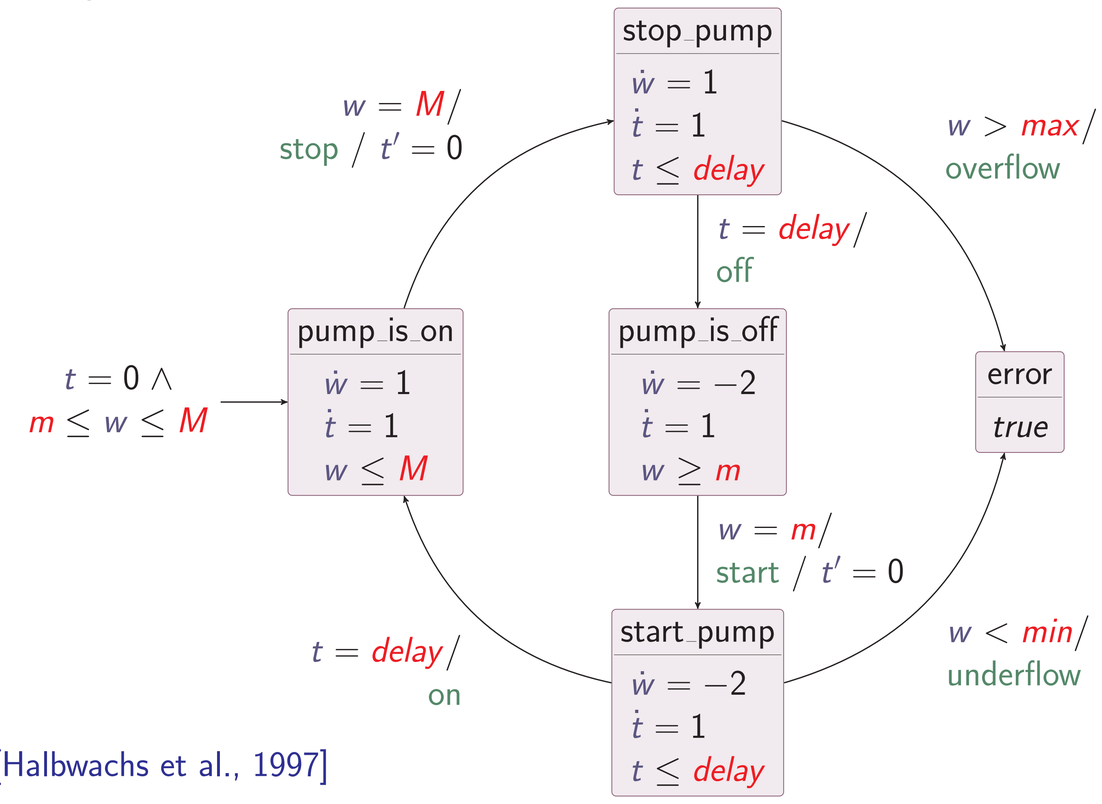
Hybrid systems combine
- Discrete behavior
- Continuous behavior



Parameterized Hybrid Automata

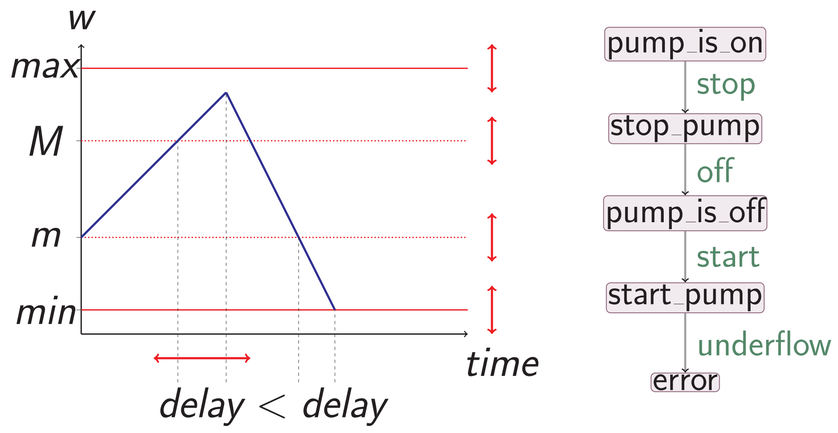
- Hybrid Automata (HA): Sets of variables, actions, locations, and discrete transitions
- Parameterized Hybrid Automata: HA augmented with a set of timing parameters (unknown constants)

- Example: Water Tank



[Halbwachs et al., 1997]

The Parameter Synthesis Problem



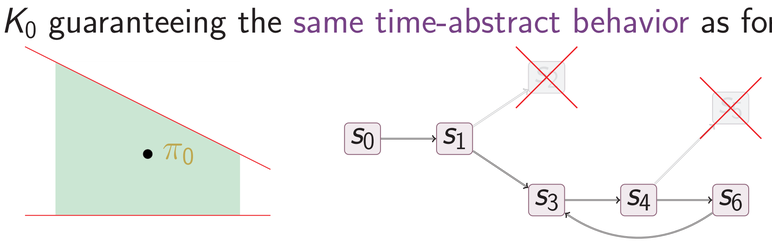
- How to choose min, max, m, M and delay, such that always $min < w < max$?

Synthesis problem: "find values for the timing parameters such that the system behaves well".

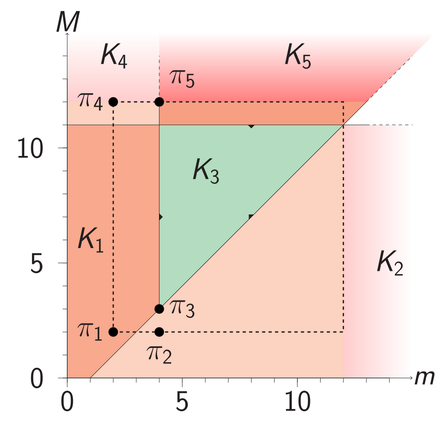
We will synthesize here a constraint, viz., a convex and dense set of values.

Parameter Synthesis for Hybrid Automata

- Inverse Method [Fribourg and Kühne, 2011]
  - Given an HA and a reference valuation π_0 for the parameters, synthesize a constraint K_0 guaranteeing the same time-abstract behavior as for π_0



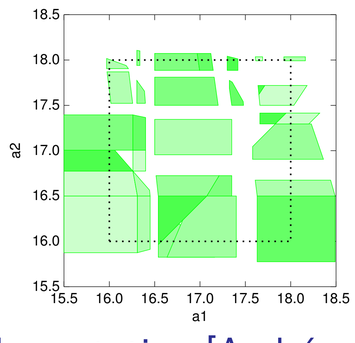
  - K_0 obtained by iterative removal of states incompatible with π_0
- Behavioral Cartography [André and Fribourg, 2010]
  - Performs a tiling of the parametric space, and partition it between good and bad tiles w.r.t. a given property

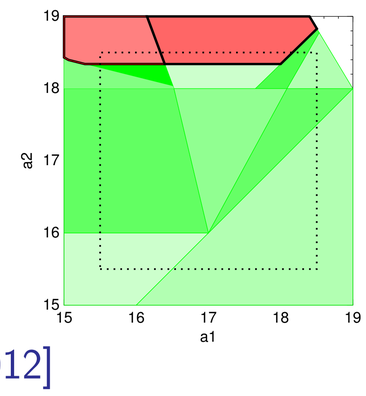


Example of "good" constraint for the water tank:
$M + delay \geq m \ \wedge \ m \geq min + 2 \cdot delay \ \wedge \ max \geq M + delay$

Features of HyMITATOR

- Algorithms of Parameter Synthesis for Hybrid Systems
  - Implements the inverse method and the behavioral cartography
  - Includes local partitioning with linear over-approximations
  - Makes use of predicate abstraction techniques
  - Features an efficient merging technique [André et al., 2012]
- User-friendly Features
  - Numerous options for analysis
  - Graphical output





- Implementation [André and Kühne, 2012]
  - Implemented in OCaml, using the Parma Polyhedra Library

Try it!

- Distributed under the GNU General Public License
- `www.lsv.ens-cachan.fr/Software/hymitator/`

References

André, É. and Fribourg, L. (2010).
Behavioral cartography of timed automata.
In *RP'10*, volume 6227 of *LNCS*, pages 76–90. Springer.

André, É., Fribourg, L., and Soulat, R. (2012).
Enhancing the inverse method with state merging.
In *NFM'12*, volume 7226 of *LNCS*, pages 100–105. Springer.

André, É. and Kühne, U. (2012).
Parametric analysis of hybrid systems using HyMITATOR.
In *Posters & Tool demos Session of iFM'12*, pages 16–19. CNR and ISTI.

Fribourg, L. and Kühne, U. (2011).
Parametric verification and test coverage for hybrid automata using the inverse method.
In *RP'11*, volume 6945 of *LNCS*, pages 191–204. Springer.

Halbwachs, N., Proy, Y.-E., and Roumanoff, P. (1997).
Verification of real-time systems using linear relation analysis.
Formal Methods In System Design, 11(2):157–185.