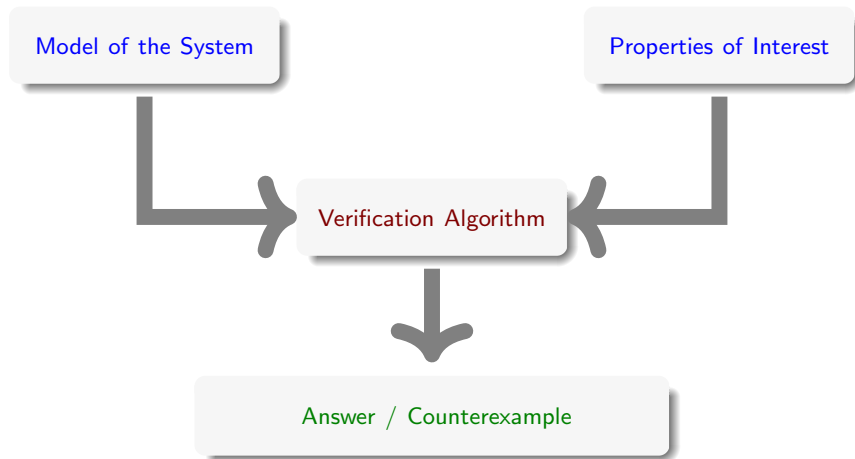


Formal verification of hybrid systems using ARIADNE

Luca Geretti and Tiziano Villa

October 28, 2019

The formal verification flow





Many real systems have a double nature. They:

- evolve in a **continuous** fashion
- are controlled by a **discrete** system



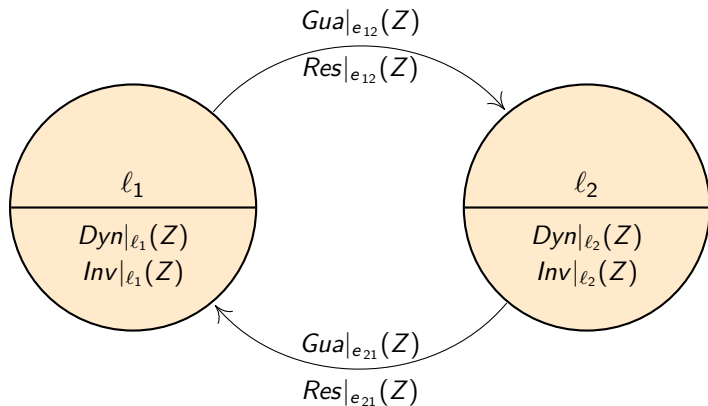
Such systems are called **hybrid systems** and may be modeled by **hybrid automata**

Hybrid automata

The intuition



A **hybrid automaton** H is a finite-state automaton with **continuous variables** Z



A **state** is a couple (ℓ, r) where r is a valuation for Z



- **dynamics** $Dyn|_{\ell}$: evolution of the variables in location ℓ
- **invariant** $Inv|_{\ell}$: conditions under which continuous evolution is allowed in location ℓ
- **guard** $Gua|_e$: conditions under which discrete evolution is allowed according to event e
- **reset** $Res|_e$: transformation of the continuous state after event e



To verify whether a dynamical system satisfies some properties, we describe its behaviour by computing the set of reached states (reachable set) Re .

- It allows full observation of system evolution (compared to abstraction methods).



To verify whether a dynamical system satisfies some properties, we describe its behaviour by computing the set of reached states (reachable set) Re .

- It allows full observation of system evolution (compared to abstraction methods).

Re is not computable in general, in particular for **nonlinear** systems.



To verify whether a dynamical system satisfies some properties, we describe its behaviour by computing the set of reached states (reachable set) Re .

- It allows full observation of system evolution (compared to abstraction methods).

Re is not computable in general, in particular for **nonlinear** systems.

Re can be approximated, but not in *both* an effective and efficient way.

- Some operations on accurate representations are still undecidable.
- Coarse approximations are problematic in terms of reliability of results.



Possible choices of approximating Re :

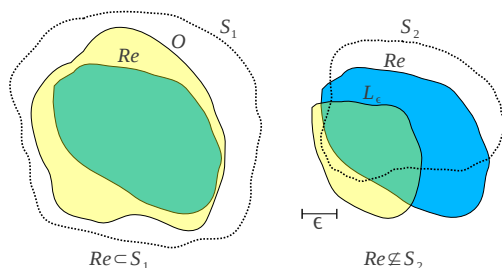
1. **Inner approximation** I : Re strictly contains I .
2. **Outer approximation** O : Re is strictly contained in O (an over-approximation of Re).
3. **ε -lower approximation** L_ε : every point of L_ε is at a distance less than ε from Re (an over-approximation of a subset of Re).



Possible choices of approximating Re :

1. **Inner approximation** I : Re strictly contains I .
2. **Outer approximation** O : Re is strictly contained in O (an over-approximation of Re).
3. **ε -lower approximation** L_ε : every point of L_ε is at a distance less than ε from Re (an over-approximation of a subset of Re).

- Inner approximation is not computable in general.
- Outer and ε -lower approximations can be used to verify/falsify properties.



- Re is the reachable set, which is unobservable.
- S_1, S_2 are sets satisfying given properties.
- O is the outer approximation of Re .
- L_ϵ is the ϵ -lower approximation of Re .



- Developed by a joint team including the University of Verona, the University of Maastricht, the University of Padova, and the University of Barry (Florida)
- Uses the formalism of **hybrid automata** to describe **nonlinear time-continuous** systems.
- Based on rigorous semantics paired with **interval arithmetics** to guarantee correctness of verification over **approximated sets**.
- Written as a C++ library, released as an open source distribution: <http://www.ariadne-cps.org>



Upper semantics

When numerical inaccuracies make the transition undecidable, all possible choices are taken. Computed sets do not need to include a point of Re .



Upper semantics

When numerical inaccuracies make the transition undecidable, all possible choices are taken. Computed sets do not need to include a point of Re .

Lower semantics

When numerical inaccuracies make the transition undecidable, evolution stops. Computed sets must include at least one point of Re .

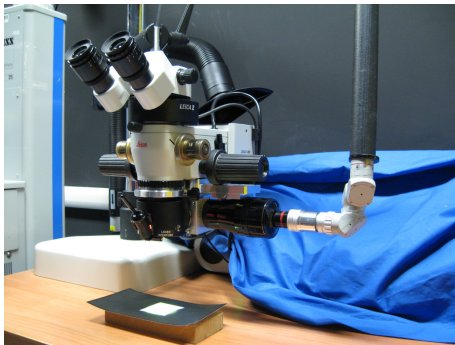


ARIADNE can compute the following approximations of the reachable set (available semantics under parentheses):

- An over-approximated subset, up to a given time t : for proving/disproving properties where a bound on the evolution time is identified [upper, lower];
- An outer approximation : for proving properties using infinite-time evolution [upper];
- For a given $\varepsilon > 0$, an ε -lower approximation : for disproving properties using infinite-time evolution [lower].



- Very strict safety requirements.
- Increasing reliance on assisted control for improved accuracy.
- Traditionally focused on control theory specifications, recently adopting formal verification approaches.





Combine hydraulic components to obtain a complex system.

- Focus on finite-time reachability

- Project instructions:

<http://www.ariadne-cps.org/files/acquamondo.pdf>