institute iMdea software

Online Simulation Reduction [1]

The Setting

Formal Verification: How do we know if some property holds on our system (programs, algorithms, etc...).

Which translates to verifying the validity of a formula on a model of our system (e.g. automata, transition systems...).

The Problem

State Explosion: The number of states in our model is often intractable.

The model can become

TOO BIG

to check formulas on it.

The Solution

Reduce our model so to get an equivalent (but smaller) one:

- by removing "useless" states
- by merging equivalent* states

* meaning they behave the same as long as the properties we are interested in are considered.

Reachability Analysis

1

Mark **blocks** of equivalent states which contain reachable nodes (from the initial ones).

! What if we have **infinitely** many reachable states?

OUR CONTRIBUTION

Interleaving reachability and behavioral refinement steps works better than computing one reduction after the other!

How to solve?
Overapproximate the set of reachable blocks.

4

Our algorithm addresses the problem for **simulation**, but:
Deciding reachability of a block in the simulation induced partition is **harder** than in the bisimulation one.

! And, said problem becomes generally undecidable.

Behavioral Equivalence

2

Starting from an initial relation, **refine** blocks of nodes, splitting states which are not equivalent.
Only sets currently marked as reachable are refined.

! What if the blocks of equivalent nodes are **infinitely** many?

3

[Lee, 1992] settled the problem when **bisimulation** is used as a behavioral equivalence notion.

What about **simulation**?

Pros of using Simulation

- Induces a **better reduction** (can lead to infinitely many less blocks)
- Precise enough to check many formulas (LTL...)

The Challenge:

- Simulation is harder to compute than bisimulation (refinements are computationally more expensive...)

Full Paper?

[1] *P. Ganty, N. Manini, F. Ranzato (2022). "Online Simulation Reduction". arXiv:2204.11804*