

On Distributed Verification and Verified Distribution

Simona Orzan

Verification is ...

proving correctness by precise formal means,
or detecting errors in the design of

- embedded systems
- communication protocols, complex algorithms
- software architectures
- security protocols (e-commerce, e-voting, bank transactions)
- safety-critical systems (nuclear power plants, air traffic control, medical equipments)

Distribution is ...



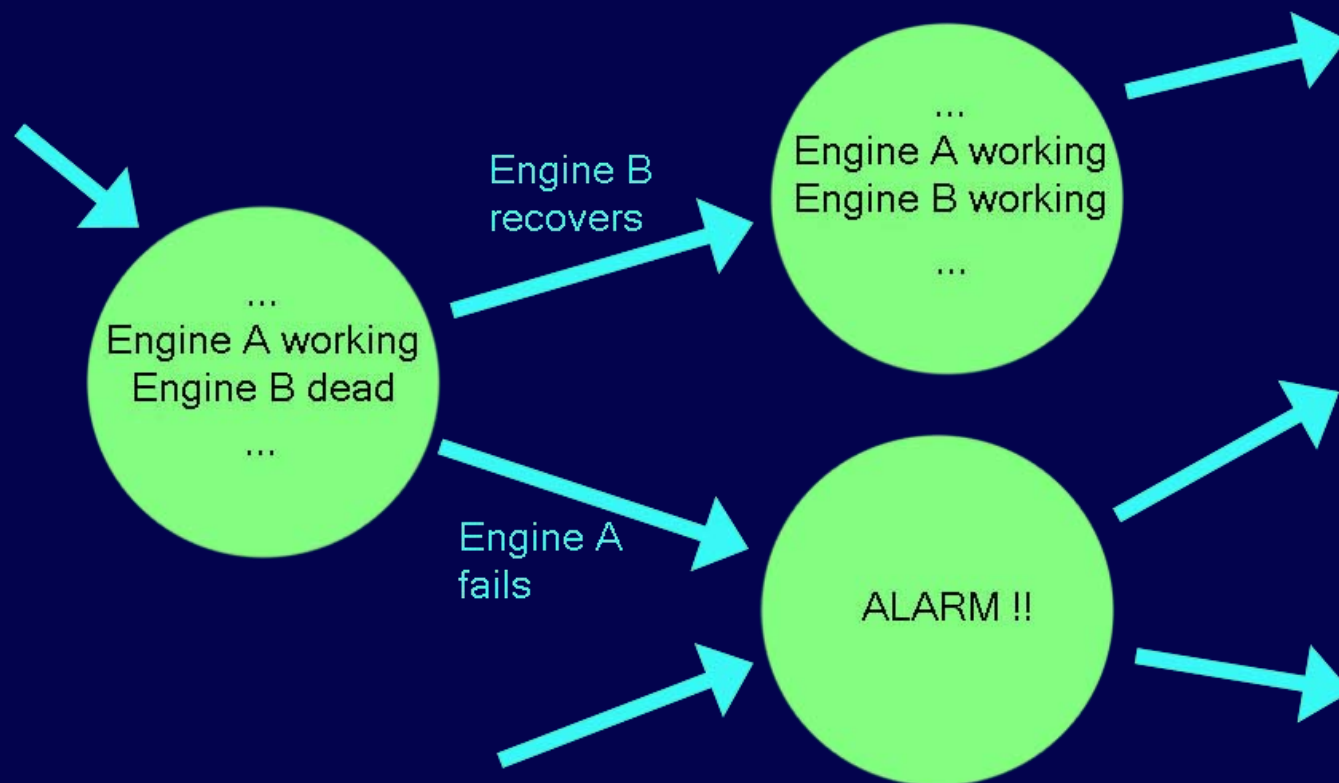
Distributed Algorithms and Tools for Verification

Verified Design of Data Distribution Architectures

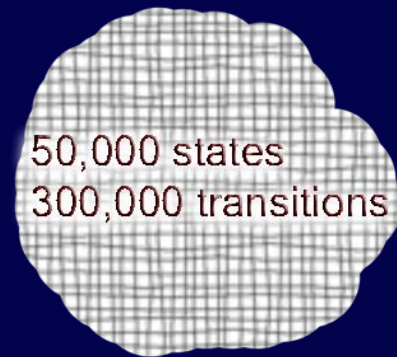
Enumerative model checking is...

generating (enumerating) all possible behaviors and look for bad or for desired ones.

states and **transitions** of an Automatic Pilot



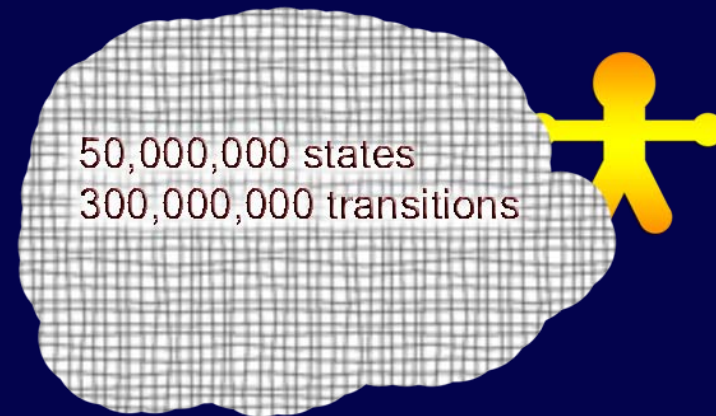
Big state spaces



computer program

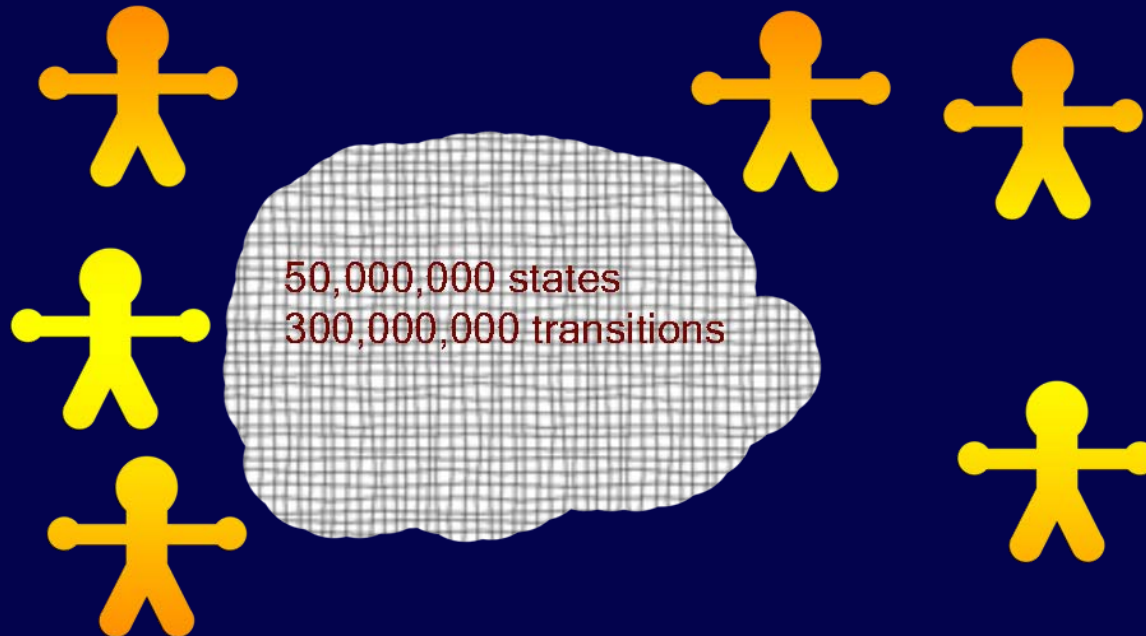
Computer support is needed!

Huge state spaces



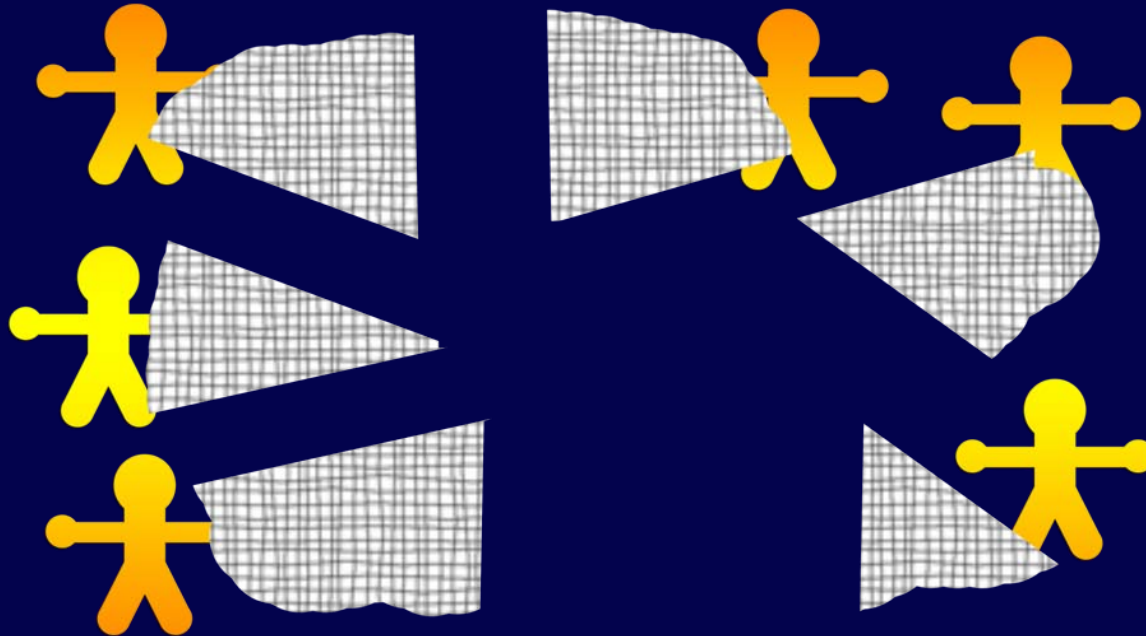
Too big for one computer!

Distributed verification



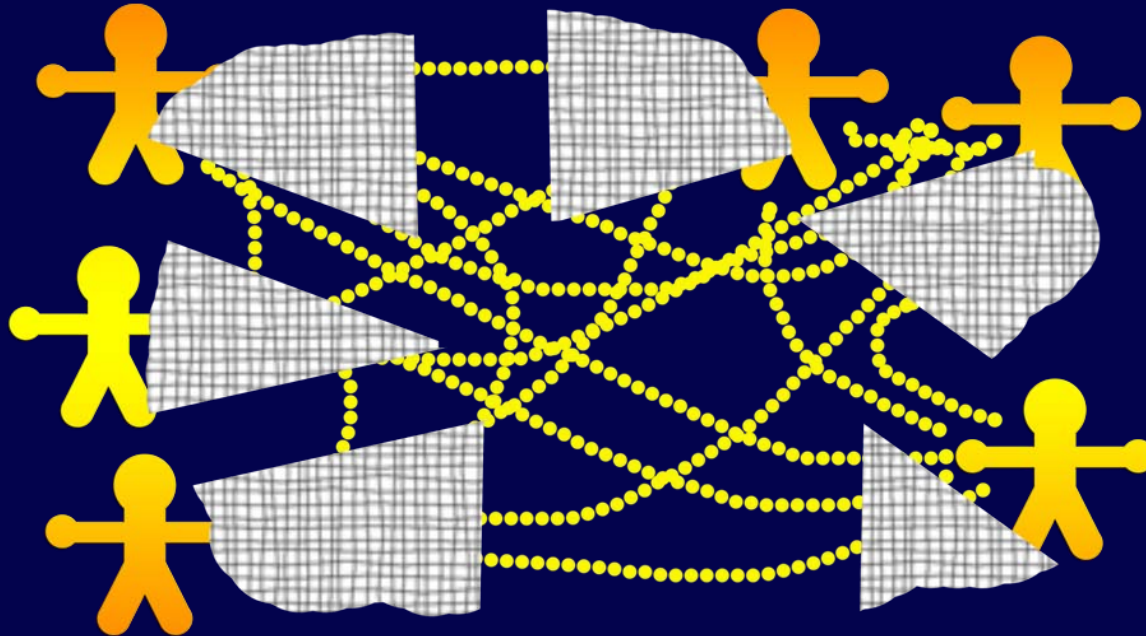
Use more computers!

Distributed verification



Let each have a piece.

Distributed verification



Now and then they need to synchronize.

This thesis

distributed algorithms and tools for

- reduction of large state spaces modulo strong bisimulation equivalence
- reduction of large state spaces modulo branching bisimulation equivalence
- detection of strongly connected components

Prototype implementations show that the algorithms scale well in both time and memory (more important!).

Distributed systems based on shared dataspace



Characteristics: time decoupling, anonymity of components.

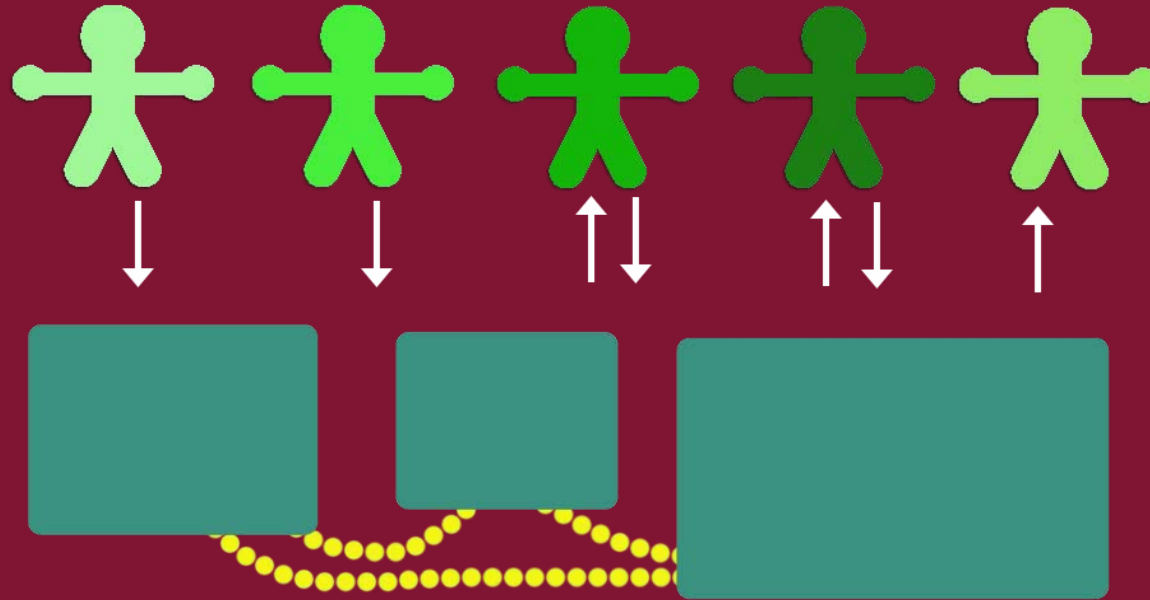
Design = from requirements to a set of applications and a global dataspace implementation.

Design decisions



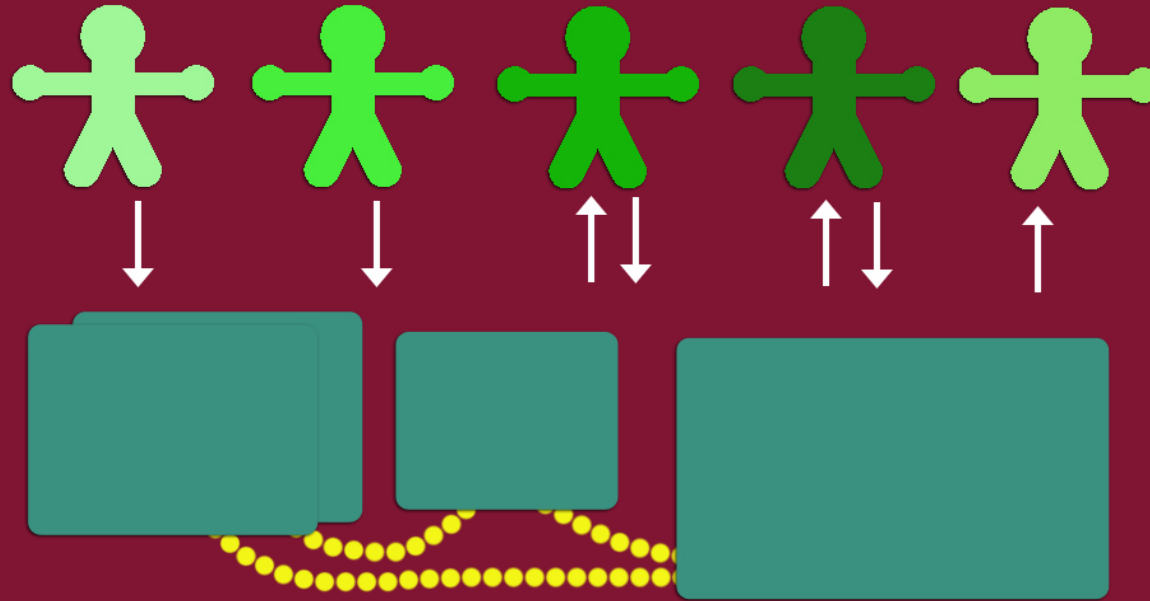
Transparent replication?

Design decisions



distributed implementation of the global dataspace

Design decisions



Transparent distribution?

This thesis

verification techniques applied to the design of distributed dataspace systems

- expressiveness study of a simple shared dataspace model, using process algebra techniques
- modeling and analysis framework for distributed systems based on shared dataspace

This thesis

distributed algorithms and tools for

- reduction of large state spaces modulo strong bisimulation equivalence
- reduction of large state spaces modulo branching bisimulation equivalence
- detection of strongly connected components

verification techniques applied to the design of distributed dataspace systems

- expressiveness study of a simple shared dataspace model, using process algebra techniques
- modeling and analysis framework for distributed systems based on shared dataspace