technische universiteit eindhoven

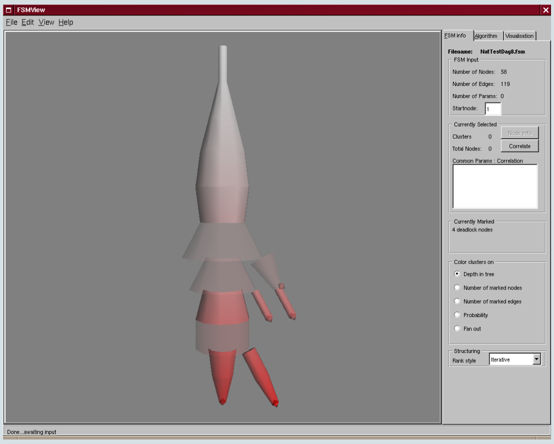
Process algebraic system verification

Muck van Weerdenburg

M.J.van.Weerdenburg@tue.nl

Verification of systems with process algebra is a formal and effective method. Toolsets, like mCRL2, have proven to be extremely effective at showing whether protocols and distributed systems behave as desired. On a regular basis design flaws in real life systems are exposed.

Given a specification of the system under study, we transform it to an intermediate format, a *linear process equation* (or *LPE*), which allows for effective analysis. Typically, one then generates the state space of the system, after which formulas, describing the desired properties of the system, can be verified.



Filename: NatTestDag8.fsm

FSM Input

Number of Nodes: 58

Number of Edges: 119

Number of Params: 0

Startnode: 1

Currently Selected

Clusters 0

Total Nodes: 0

Common Params : Correlation

Currently Marked

4 deadlock nodes

Color clusters on

● Depth in tree

○ Number of marked nodes

○ Number of marked edges

○ Probability

○ Fan out

Structuring

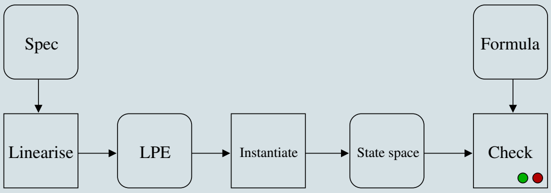
Rank style Iterative

Done...awaiting input

Visualised state space of a Petri net

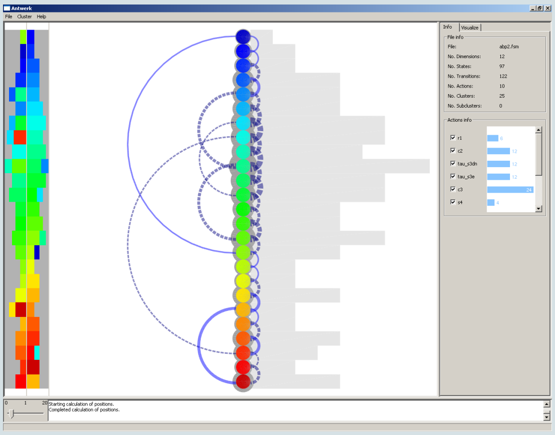
But, with state space visualisation we can also easily detect irregular behaviour. For example, in the visualised system above there are three branches which seem to indicate anomalous behaviour.

Special tools can manipulate LPEs such that, for example, unreachable states are removed beforehand and performance of state space generation increases significantly. State spaces of over 10^9 states can be verified by using distributed tools. Larger systems can be handled by using *confluence* or *abstraction*.


Spec
Formula
Linearise
LPE
Instantiate
State space
Check

Verification process

We are continuously working on new and better techniques to be able to handle even larger systems. One aim is to avoid generating state spaces altogether and use *symbolic reasoning* on the higher level behaviour descriptions (especially LPEs). For example, we can combine an LPE and a requirement formula into a *parameterised boolean equation system* (*PBES*) and try to solve it.



File info

File: abp2.fsm

No. Dimensions: 12

No. States: 97

No. Transitions: 122

No. Actions: 10

No. Clusters: 25

No. Subclusters: 0

Actions info

☑ r1	6
☑ c2	12
☑ tau_s3dn	12
☑ tau_s3e	12
☑ c3	24
☑ s4	4

Starting calculation of positions.
Completed calculation of positions.

Clustering of state space of a simple protocol

Tools shown are made by Frank van Ham (left) and Hannes Pretorius (right).

/ department of mathematics and computer science