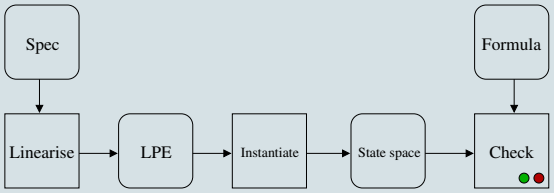
System verification with μCRL

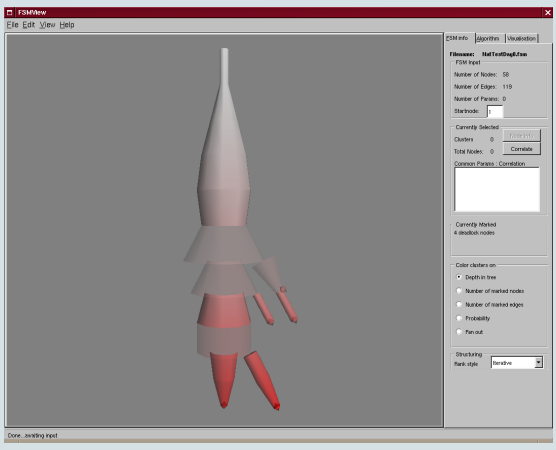
Muck van Weerdenburg
M.J.van.Weerdenburg@tue.nl

Verification of systems with process algebra is a formal and effective method, which has proven to be useful in checking real life cases, such as protocols and distributed systems.

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.

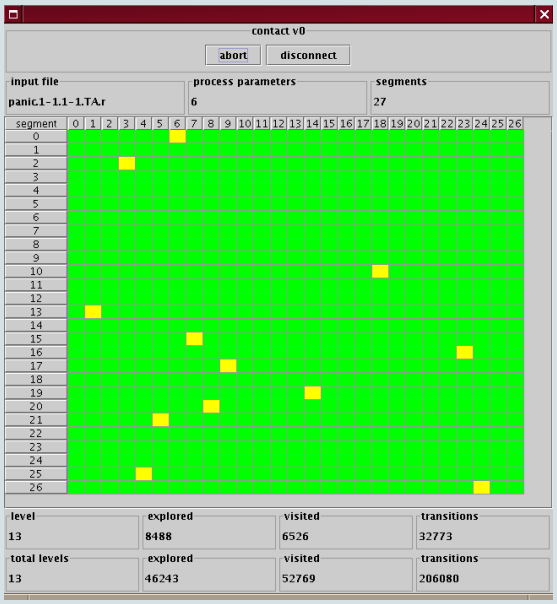


But, with state space visualisation we can also easily detect irregular behaviour. For example, in the visualised system below it is clear that there are three branches which seem to indicate anomalous behaviour.



Visualised state space of a Petri net

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 up to 10^9 can be generated, for instance by using distributed generation tools. But also larger systems can be handled by using *confluence* or *abstraction*.



Status display of distributed state space generation

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 more high level behaviour descriptions (especially LPEs). For example, we can combine a LPE and a requirement formula into a *parameterised boolean equation system* (or *PBES*) and try to solve it. Also mCRL2, follow up of μCRL and part of the GenSpect project, could help to tackle more complicated systems.