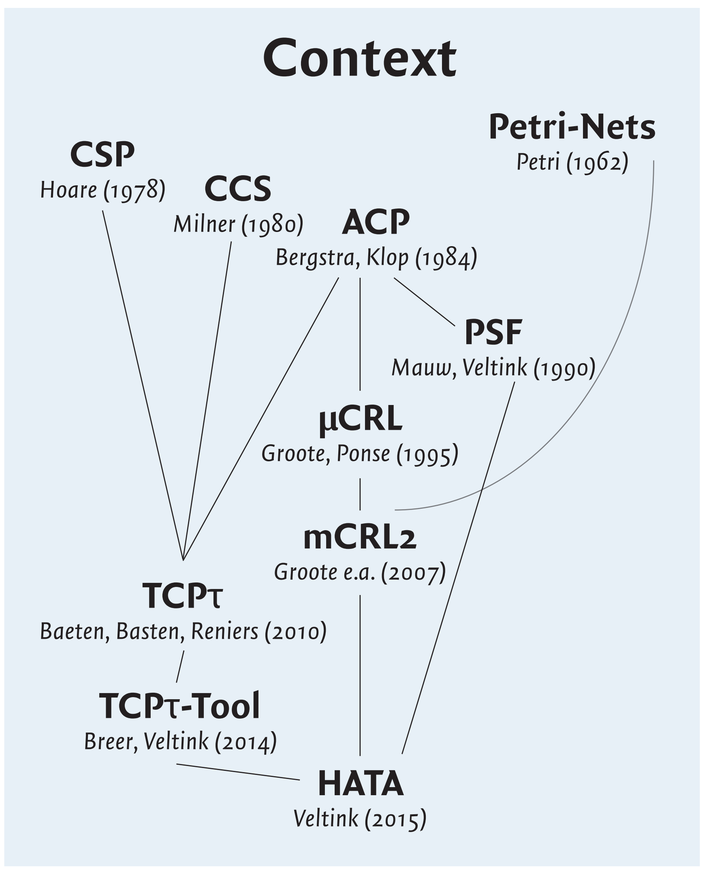
University of Applied Sciences
HOCHSCHULE EMDEN·LEER

Hierarchical Algebraic Transaction Architecture

Towards executable formal specifications - Closing the gap with HATA

Goals

The HATA Project at the University of Applied Sciences Emden/Leer aims at providing a framework for the systematical integration of formal methods into the specification and verification of automation projects and cyber-physical systems in the context of „smart factories“.



As a result of the „Industry 4.0“ high-tech initiative of the German Government the interoperability of existing systems, especially regarding the compositionality of existing sub-systems, has come into the focus of research, which is also indicated by the rising use of the two keywords *orchestration* and *choreography* in current automation projects.

The initial target of the HATA project is to develop a programming language that uses aspects from process algebras (most notably ACP) and to use this language for the modular specification and verification of implementations to be installed at the „digital factory“ of the I²AR-Institute.

Results

The HATA Framework provides tools that support the specification, verification and execution of formal specifications.
At current a Java Library has been developed, that embodies the core semantics of the process algebra ACP and allows for adding a description of concurrent behaviour on top of a normal Java program. The library is able to analyze the concurrent behaviour of a system by generating a so-called labelled transition system (LTS) from the algebraic specification. The LTS can be exported into the Aldebaran ‚.aut‘-Format, which in turn can be read by the mCRL2-Tools in order to visualize the LTS or to analyze it further using Hennessy-Milner Logic. Finally, the library maps the concurrent parts of a specification to separate Java-Threads, such that it can be executed in parallel.

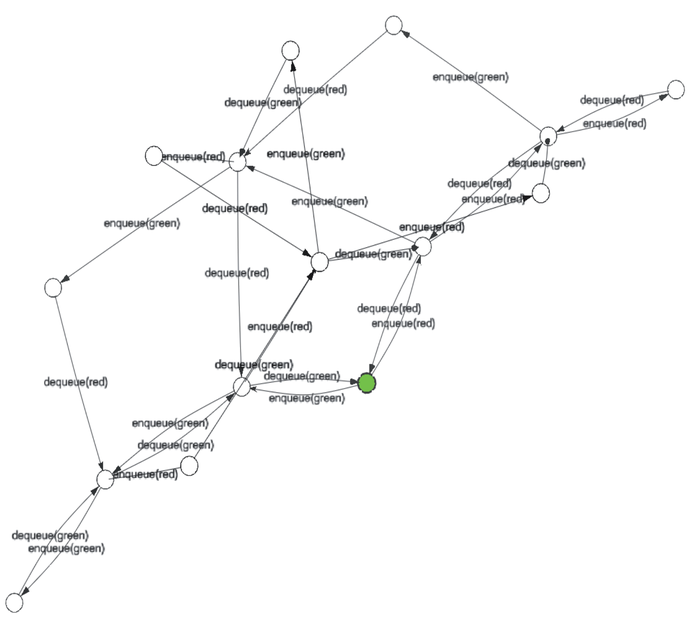
Specification

The specification of a system is done in standard Java through a DSL (Domain Specific Language) that somewhat resembles ACP. The image below shows an excerpt of such a specification.

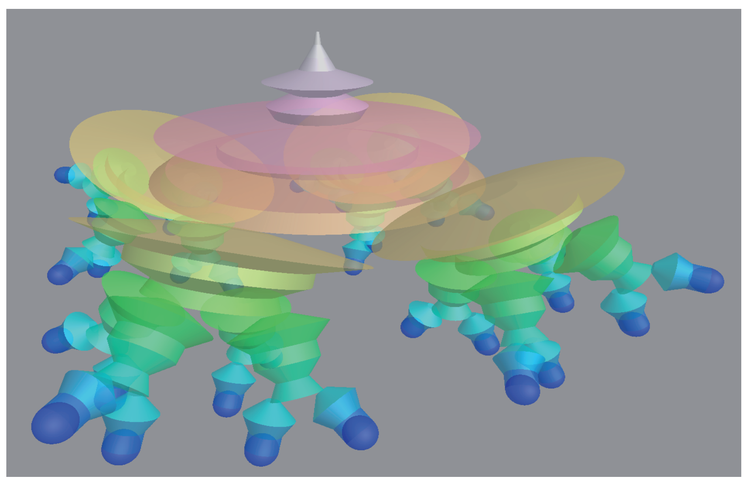
```
128    /**
129     * The sub-process that waits for the acknowledgement
130     * and decides how to continue execution,
131     * based on the value of the acknowledgement.
132     *
133     * @param b  the expected acknowledgement
134     * @param d  the data item being sent
135     *
136     * @return   the process term representing the sub-process
137     */
138    public PExp _WaitForAck(Boolean b, Integer d) {
139    return
140    _Alt(                    // either ...
141        readAck(b),          // ... receive the correct ack
142        _Seq(                // or ...
143            readAck(!b),     // ... receive the incorrect ack
144            Sender(b, d)     // and resend
145        ),
146        _Seq(                // or ...
147            readErr(),       // ... receive an error
148            Sender(b, d)     // and resend
149        )
150    );
151    }
```

Analysis and Verification

The generated LTS can be visualized using the mCRL2-Toolkit. If the number of states is not too large, the entire LTS can be shown.

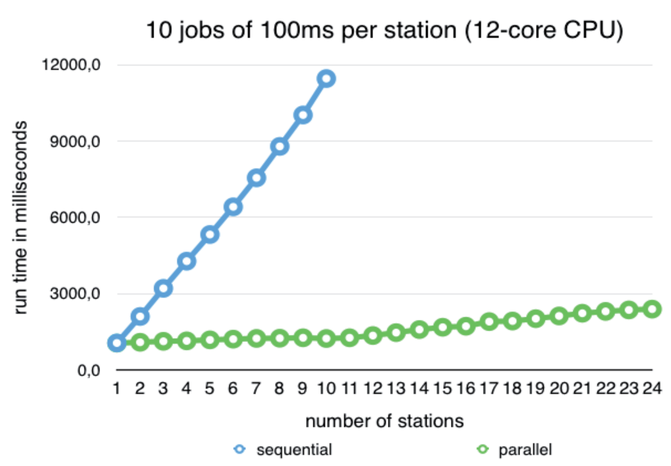


However, if the LTS get larger a different visualization, based on clustering, becomes necessary. A clustering visualisation is shown below.



Execution

In the final step the specification can be executed, possibly using threads to implement concurrency. The performance gain that can be achieved through parallel instead of sequential execution is shown in the diagram below.



Conclusion

The HATA Library sucessfully integrates concepts from the theoretical field of process algebras into an existing all-purpose programming language (Java). To enable visualisation, an interface to the mCRL2-Tools is provided. This integration allows for a seamless transition between specification, analysis/verification and execution of formal specifications of cyber-physical systems.

Future Developments

There are three future research topics that can be identified.
The DSL used in the current version can be improved and possible implemented as a separate dedicated language using *Xtext* and *Xtend*.
The second topic is to create a logical extension of the multi-threaded execution, namely the distributed execution over a network.
Finally, we intend to carry out case studies using HATA, targeting the systems in the „digital factory“ of the I²AR-Institute.

Prof. Dr. Gerrit Jan Veltink
gert.veltink@hs-emden-leer.de

Hochschule Emden/Leer
Constantiaplatz 4
26723 Emden