Introduction

The CIF language \[1\] is developed to establish the interoperability between a wide range of tools, needed for the development of complex embedded systems. The CIF language is extended with the concept of AND/OR superstates to form the Hierarchical CIF (HCIF) language. The added hierarchy allows for a hierarchical decomposition and stepwise refinement of models.

Hierarchy

In HCIF, hybrid automata are the basic constructs and hierarchy is established by means of AND/OR superstates. This means that a location of an automaton can contain another automaton (OR), or a parallel composition (AND).

A linearization procedure is defined to linearize HCIF models. This linearization procedure is implemented using QVTo. In this way, existing CIF tools such as the simulator, are available for HCIF models.

Process algebraic linearization

A linearization procedure is defined to eliminate hierarchy from HCIF models and rewrite them into a normal form. This normal form is a single location automaton with linear edges. The procedure is based on \[2\] and uses location pointers to combine all locations of an automaton into one. When eliminating hierarchy, first all substructures are linearized, after which they are merged with their superstate. This resulted flat automaton can again be rewritten into normal form.

Conclusions

- HCIF allows a hierarchical decomposition of models
- A procedure is defined to linearize HCIF models
- A tool chain for HCIF models is established

References
