A linear-time branching-time perspective on interface automata. (English) Zbl 1443.68099
Acta Inf. 57, No. 3-5, 513-550 (2020).

Summary: Over the past two decades, de Alfaro and Henzinger’s interface automata (IA) have become a popular formal framework for the component-based specification of concurrent systems. IA’s parallel composition assumes that a component may wait on inputs but never on outputs, implying that an output must be consumed immediately or a communication error occurs. By now, the literature contains a number of semantics for IA: linear-time semantics based on traces observing communication errors, quiescence and/or divergence, as well as branching-time semantics based on alternating simulation. This article surveys these semantics from Rob van Glabbeek’s linear-time branching-time perspective, which does not consider settings with communication errors. We shed light onto the subtleties implied by IA’s pruning of all behaviour that might lead a component to autonomously enter an error state, and investigate when exactly de Alfaro and Henzinger’s restriction of input-determinism is needed. In addition, we introduce several new semantics for IA, in particular the linear-time ready semantics and the branching-time ready simulation.

MSC:
68Q45 Formal languages and automata
68Q55 Semantics in the theory of computing
68Q85 Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.)

Software:
MIO Workbench

Full Text: DOI

References:


This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.