Verification of Embedded Software: From Mars to Actions

Charles Pecheur

Catholic University of Louvain, Belgium
charles.pecheur@uclouvain.be
http://www.info.ucl.ac.be/~pecheur

Embedded controllers are more and more pervasive and feature more and more advanced capabilities. For space applications in particular, the development of autonomous controllers is seen as a critical technology to enable new mission objectives and scale down operating costs. On the flip side, the validation of intelligent control software poses a huge challenge, both due to the increased complexity of the system itself and the broad spectrum of normal and abnormal conditions in which it has to be able to operate. This talk will follow our journey in applying some modern, analytical verification technologies and tools to the validation of autonomy software, in the context of space applications, at NASA Ames Research Center in California. This route will take us from the concrete, practical dependability requirements for space-bound software that motivated the work down to the deeper, broader issues in formal methods that emerged as part of that work.

Our work has focused on analysing model-based approaches to fault diagnosis, as exemplified by NASA Ames' Livingstone system. We developed two lines of verification tools for model-based diagnosis applications, and experimented with those tools on real-size problems taken from NASA applications. Under different angles, both approaches stem from model checking principles. Verifying diagnosis systems and models has led to considering the issue of diagnosability: given a partially observable dynamic system, and a diagnosis system observing its evolution over time, how to verify (at design time) whether the system provides sufficient observations to determine and track (at run-time) its internal state with sufficient accuracy. This kind of question can be reduced to a state reachability problem, which can be solved using (symbolic) model checking. In turn, diagnosability can be phrased as a particular temporal and epistemic property ("the diagnoser always knows the faults"), and we have carried experiments in applying a generic temporal-epistemic model checker to our diagnosis applications. Finally, epistemic models and logics themselves can, under some assumptions, be reduced to labelled transition systems and action-based temporal logics. We implemented this reduction and added support for this logics in NuSMV, thereby leveraging NuSMV's language, features and ecosystem to the analysis of epistemic (and, more broadly, action-based) properties.

S. Leue and P. Merino (Eds.): FMICS 2007, LNCS 4916, p. 1, 2008.