# Algorithmic Game Solving - from Theory towards Applications in Synthesis


Contact Information

Mr. Chih-Hong Cheng, M.Sc.

Boltzmann Str. 3 (Zi. 03.07.037)
Department of Informatics
Technische Universität München
Garching, D-85748 Germany

*Phone:* +49.89.289.18141
*Fax:* +49.89.289.18107
*E-mail:* chengch@in.tum.de
*WWW:* http://www6.in.tum.de/~chengch



Abstract

This talk summarizes our efforts bringing algorithmic game solving from theoretical results towards concrete applications. For this purpose, we first construct the tool GAVS+ which collects most types of games having practical interests with various winning conditions, starting with two-player turn-based games [ATVA'10], then expand its spectrum concerning concurrency, infinite state, distributivity, probability [TACAS'11], games of imperfect information and others [paper in preparation]. GAVS+ is the first library which summarizes existing efforts in such a comprehensive way. It targets to serve as an open platform for the research community in algorithmic game solving, and it is meant to work as a playground for researchers thinking of mapping interesting problems to game solving. With the library in hand, we start bridging games in theory and concrete applications; new algorithms are introduced when required.

We first investigate the problem concerning deadlock prevention in interaction systems composed by unmodifiable components; we use the BIP language developed at Verimag as an example. Tools based on verification (e.g., D-finder) are capable of deadlock finding while the burden of deadlock removal remains. By understanding the problem and combining concepts in games, we create a technique called *priority synthesis* [NFM'11], which results in a tool targeted for automatic supervisory control; it automatically adds static priorities over interactions to restrict the system behavior for safety. Second, we study how *behaviorial-level synthesis* can be connected to games. For this GAVS+ includes features such that the user can now process and synthesize planning (game) problems described in the established Strips/PDDL language by introducing a slight extension which allows to specify a second player [TACAS'11]. For the third application, we study how HW/SW level *fault-tolerant synthesis* can be combined with games [VMCAI'11, RTAS-WiP'10]. Under strict assumptions we are able to complete the whole flow, which is based on (a) timing-abstraction to convert systems to games, (b) game solving to select appropriate fault-tolerant mechanisms, and (c) constraint resolution to restore timing information on concrete hardware. This first result is implemented in a prototype software.

Joint work with (alphabetical order) S. Bensalem (Verimag), C. Buckl (fortiss GmbH), J. Esparza (TU Munich), B. Jobstmann (Verimag), A. Knoll (TU Munich), M. Luttenberger (TU Munich) and H. Ruess (fortiss GmbH).