Symbolic Synthesis of Distributed Systems with Petri Games

Manuel Gieseking

SCARE
System Correctness under Adverse Conditions

Motivation

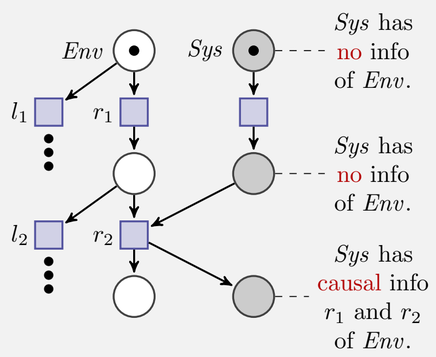
Distributed Synthesis

- Approach: Given a formal specification, automatically derive correct-by-construction implementation (if existent).
- Scenario: Distributed system with multiple concurrent processes acting independently.
- Goal: Derive local controller for each process, collectively fulfilling the global specification.

Petri games

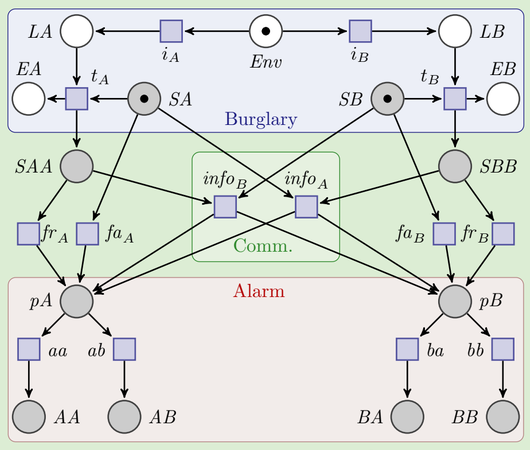
- Extension of Petri nets to multi-player games.
- Each token is a player.
- Players have causal memory (possibly obtained through synchronization).
- Solving takes only* single-exponential time.

* For n system players, 1 environment player, local safety objective.



B. Finkbeiner & E.-R. Olderog (GandALF 2014): Petri Games: Synthesis of Distributed Systems with Causal Memory.

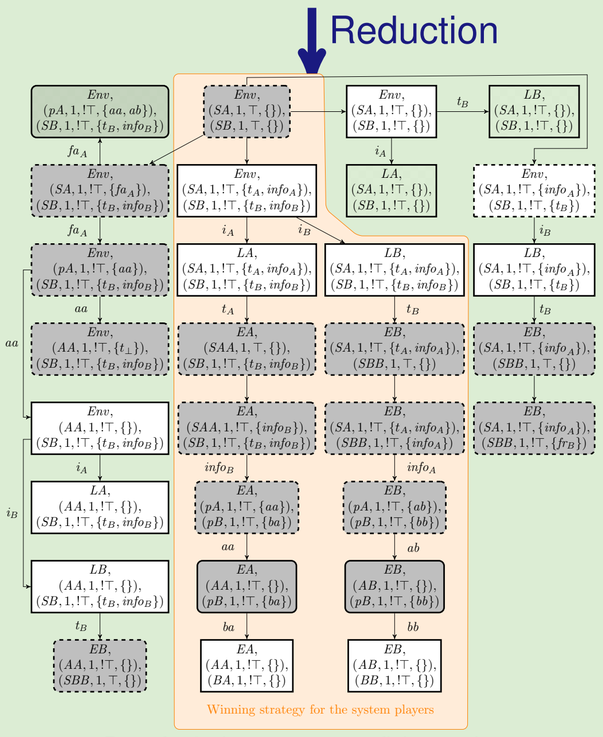
Deciding the Distributed Synthesis Problem – Symbolically Solving Petri Games



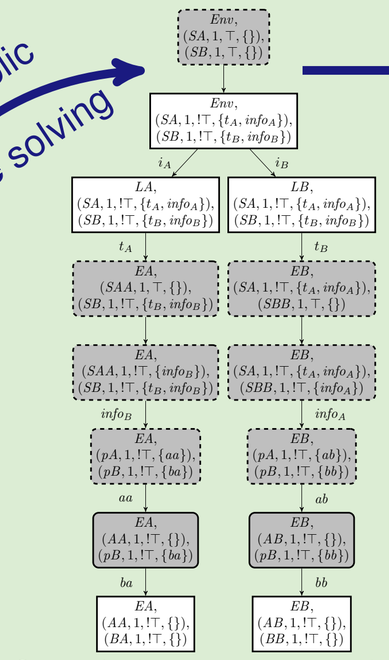
Petri game modeling AS.

Example:
Synthesis of local controllers for a distributed **alarm system (AS)**:

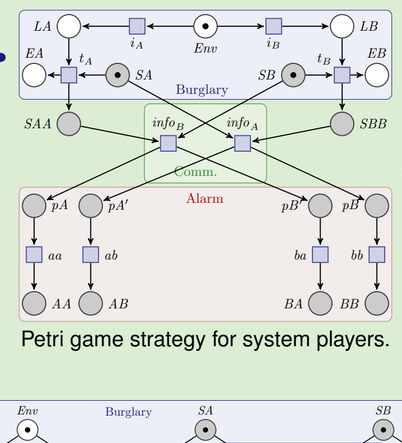
- A burglar (environment player) intrudes one of n secured locations (here $n = 2$).
- The corresponding local alarm system (system player) should detect the burglary and communicate the intruded location to the other alarm systems (system players).
- Goal: Each alarm system should correctly indicate the burglar's intrusion (no false alarm or false report).



Part of a 2-player game over finite graph.



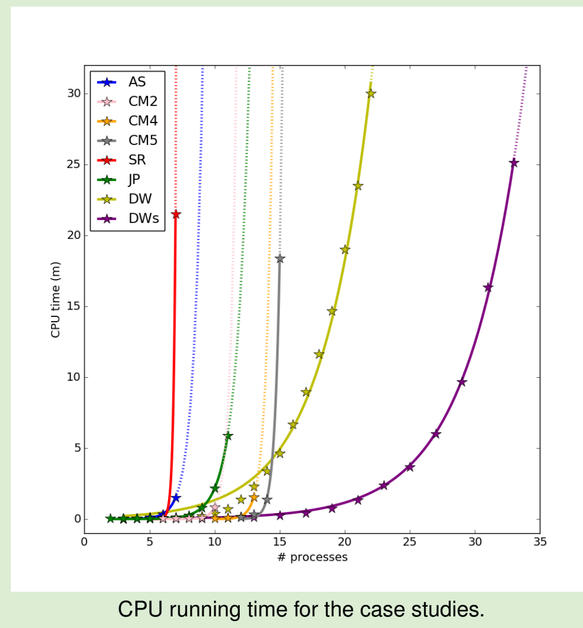
Strategy for the cooperative system players.



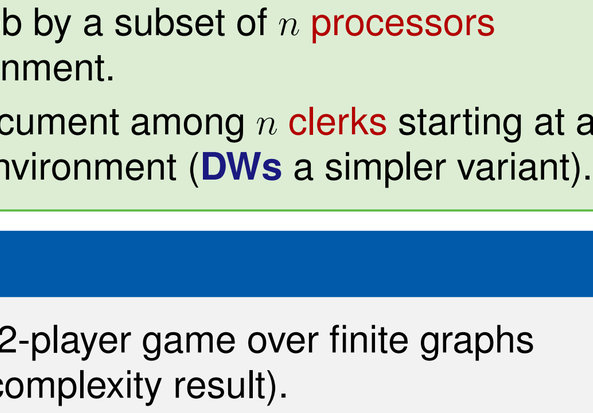
Petri game strategy for system players.

Distributed controllers.

Evaluation



CPU running time for the case studies.



Input size vs. strategy size (#transitions) for the case studies.

Tool Support: ADAM*

* ADAM is named in honor of Carl Adam Petri (1926–2010)

The tool ADAM (**A**nalyzer of **D**istributed **A**synchronous **M**odels)

- automatically synthesizes local controllers from a given Petri game,
- uses symbolic game solving algorithm with BDDs,
- handles case studies with up to 33 system processes (30 minutes time out),
- achieved artifact evaluation badge from CAV (Computer Aided Verification).

Available at `http://www.uni-oldenburg.de/csd/adam/`.

B. Finkbeiner, M. Gieseking, & E.-R. Olderog (CAV 2015): ADAM: Causality-Based Synthesis of Distributed Systems.
B. Finkbeiner, M. Gieseking, J. Hecking-Harbusch, & E.-R. Olderog (SYNT 2017): Symbolic vs. Bounded Synthesis for Petri Games

Case Studies

CM: n concurrent machines process k orders, each order by one machine. The environment decides which machines are functioning.

SR: Self-reconfiguration of n robots on which the environment destroys up to k tools.

JP: Processing of a job by a subset of n processors selected by the environment.

DW: Workflow of a document among n clerks starting at a clerk chosen by the environment (**DWs** a simpler variant).

Related Work: Other Frameworks for Distributed Synthesis

- **Pnueli-Rosner model:** synchronous concurrency with partial observation of shared variables. General: undecidable, pipelines and rings: nonelementary.
- **Zielonka automata**: asynchronous concurrency with shared actions and causal memory. General: decidability open, tree architectures: nonelementary.

A. Pnueli & R. Rosner (FOCS 1990): Distributed Reactive Systems are Hard to Synthesize.
B. Finkbeiner & S. Schewe (LICS 2005): Uniform Distributed Synthesis.
P. Gastin, B. Lerman, & M. Zeitoun (FFTTCS 2004): Distributed Games with Causal Memory are Decidable for Series-Parallel Systems.
B. Genest, H. Gimbert, A. Muscholl, & I. Walukiewicz (ICALP 2013): Asynchronous Games over Tree Architectures.

Future Work

- New reduction to a 2-player game over finite graphs (by preserving the complexity result).
- New winning conditions (reachability, Büchi, parity, global objectives).
- Extend level of informedness (add forgetful places, indistinguishable transitions (partial observation)).

Manuel Gieseking
gieseking@informatik.uni-oldenburg.de

Correct System Design
University of Oldenburg