Model Checking for Reaction and Multi-Agent Systems

PhD dissertation

Artur Męski

supervisor:

prof. dr hab. inż. Wojciech Penczek

IPI PAN, 2020-09-17

Reaction systems:

Reaction systems, Fundamenta Informaticae 75(1-4), 2007
A. Ehrenfeucht and G. Rozenberg

- Formal model of computation
- Based on the mechanism of facilitation and inhibition
- Inspired by the biochemistry of living cells

Goal: verification methods for reaction systems

Complexity of model checking for reaction systems, TCS 623, 2016
S. Azimi, C. Gratie, S. Ivanov, L. Manzoni, I. Petre, A. E. Porreca

Reaction system models for the heat shock response,
Fundamenta Informaticae 11(3), 2014
S. Azimi, B. Iancu, I. Petre

Reaction systems

Main results of the PhD thesis

1. Model checking for:
    a. rsCTL
    b. rsCTLK
    c. rsLTL
2. Parametric model checking for rsLTL
3. Toolkit and experiments

$\mathcal{R} = (S, A)$ – reaction system (RS):

- S – finite background set *entities (e.g., molecules)*
- A – set of reactions over S

Reaction: $b = (R, I, P)$ such that R, I, P are nonempty subsets of S with $R \cap I = \varnothing$

- R – reactants
- I – inhibitors
- P – products

Example

$$(S, A) = (\{1, 2, 3, 4\}, \{a, b, c, d\})$$

$$\begin{aligned} a &= (\{1, 4\}, \{2\}, \{1, 2\}) \\ b &= (\{2\}, \{4\}, \{1, 3, 4\}) \\ c &= (\{1, 3\}, \{2\}, \{1, 2\}) \\ d &= (\{3\}, \{2\}, \{1\}) \end{aligned}$$

In state $\{1, 3, 4\}$:

- a, c, d – enabled reactions

Individual results for the reactions:

$$\begin{array}{ll} a \longrightarrow \{1, 2\} & b \longrightarrow \varnothing \\ c \longrightarrow \{1, 2\} & d \longrightarrow \{1\} \end{array}$$

Result state: $res_A(\{1, 3, 4\}) = \{1, 2\}$

- Execution of reaction systems depends on their environment (context)
- Context: sequence of sets of entities, supplied at each step of execution
- Affects reaction enablement: states are extended with context

Interactive process ($n \geqslant 1$ steps): $\pi = (\gamma, \delta)$

- context sequence: $\gamma = (C_0, C_1, \cdots, C_n)$, where $C_0, C_1, \cdots, C_n \subseteq S$
- result sequence: $\delta = (D_0, D_1, \cdots, D_n)$, where $D_0 = \varnothing$ and $D_1, \cdots, D_n \subseteq S$
- $D_i = res_A(D_{i-1} \cup C_{i-1})$ for all $i \in \{1, \cdots, n\}$

The state sequence of π is the sequence $\tau = (W_0, W_1, \cdots, W_n)$ such that:

$$W_i = C_i \cup D_i \text{ for all } i \in \{0, \ldots, n\}.$$

Automatic verification method for reaction systems

Input:

- F – extended reaction system: MARS, RSC, PRS
- $L \in \{\mathrm{rsCTL}, \mathrm{rsCTLK}, \mathrm{rsLTL}\}$ – language of properties

$\mathcal{M}_L^F$ – model for RS extension F and logic L

ϕ_L – property expressed in L

Model checking – decision problem:

$$\mathcal{M}_L^F \models \phi_L \ ?$$

ICR-$\mathcal{R} = (\mathcal{R}, \mathcal{E}, S_0)$ – initialised context-restricted RS (ICRRS):

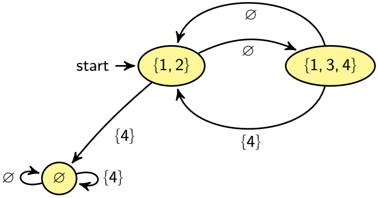
- $\mathcal{R} = (S, A)$ – reaction system
- $\mathcal{E} \subseteq S$ – context entities
- $S_0 \subseteq 2^S$ – initial states, $S_0 \neq \emptyset$

$(S, A) = (\{1, 2, 3, 4\}, \{a, b, c, d\})$

$a = (\{1, 4\}, \{2\}, \{1, 2\})$ $b = (\{2\}, \{4\}, \{1, 3, 4\})$
$c = (\{1, 3\}, \{2\}, \{1, 2\})$ $d = (\{3\}, \{2\}, \{1\})$

assumed **start state**: $\{1, 2\}$

environment (context): $2^{\{4\}} = \{\varnothing, \{4\}\}$

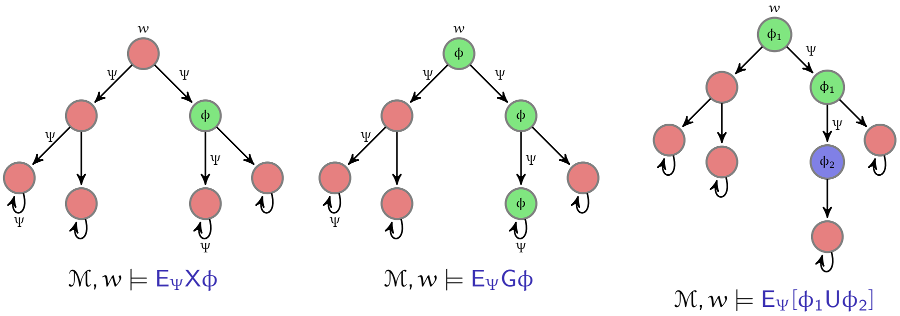


rsCTL – temporal logic based on CTL

Syntax of rsCTL

$$\phi := ent \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \mathsf{E}_\Psi\mathsf{X}\phi \mid \mathsf{E}_\Psi\mathsf{G}\phi \mid \mathsf{E}_\Psi[\phi\mathsf{U}\phi]$$

- $ent \in S$
- $\Psi \subseteq 2^{\mathcal{E}}$ – restriction on the set of considered paths ($\Psi \neq \emptyset$)



Model checking problem

Input: ICRRS, rsCTL formula

Decision problem:

$$\mathcal{M}^{\text{ICRRS}}_{\text{rsCTL}} \models \phi_{\text{rsCTL}} \quad \text{iff} \quad \phi_{\text{rsCTL}} \text{ holds in all the initial states}$$

Theorem 1

Model checking for rsCTL is PSPACE-complete

- PSPACE: construction of nondeterministic model checking algorithm using polynomial space
- PSPACE-hard: reduction of TQBF (QSAT) to rsCTL MC

$\mathcal{A} = \{1, \ldots, m\}$ – *agents* $(\mathcal{A} \neq \varnothing)$

$\mathcal{D} = (S, \{A_i\}_{i \in \mathcal{A}})$ – *multi-agent reaction system* (MARS):

- S – is a finite nonempty set
- $A_i \subseteq rac(S)$ for each $i \in \mathcal{A}$

Each agent maintains its own local state

- $St_{\mathcal{D}} = \underbrace{2^S \times \cdots \times 2^S}_{m \textit{ times}}$ – states of $\mathcal{D}$
- $Ct_{\mathcal{D}} = St_{\mathcal{D}} \times 2^{\mathcal{A}}$ – contexts

$\mathbf{C} \in Ct_{\mathcal{D}}$, $\mathbf{C} = (\mathbf{C}^c, \mathbf{C}^a)$

- $\mathbf{C}^c \in St_{\mathcal{D}}$ – tuple of sets of context entities
- $\mathbf{C}^a \subseteq \mathcal{A}$ – *activated* agents

Semantics – intuition

- if $i \in \mathcal{A}$ is activated, we take its context set and the union of all the local states of the activated agents
- if $i \in \mathcal{A}$ is not activated, then the local state remains unchanged

Here we replace $\mathcal{E} \subseteq S$ (and $S_0 \subseteq 2^S$) from ICRRS with context automata:

Context automaton (CA) over Σ: $\mathfrak{A} = (\mathcal{Q}, \mathfrak{q}^{init}, R)$

- Σ – finite set of labels
- $\mathcal{Q}$ – finite set of *locations*
- $\mathfrak{q}^{init} \in \mathcal{Q}$ – *initial location*
- $R \subseteq \mathcal{Q} \times \Sigma \times \mathcal{Q}$ – *transition relation* (serial)

Context-restricted reaction system (CRRS): $\text{CR-}\mathcal{R} = (\mathcal{R}, \mathfrak{A})$

- $\mathcal{R} = (S, A)$ – reaction system
- $\mathfrak{A}$ – *context automaton* over 2^S

- Extended CA (ECA): $\mathfrak{E} = (\mathcal{Q}, \mathfrak{q}^{init}, R)$
  extended with transition guards (conditions on states of MARS)
- Context-restricted MARS (CRMARS): $\text{CR-}\mathcal{D} = (\mathcal{D}, \mathfrak{E})$

Syntax of rsCTLK:

$$\phi ::= \mathtt{i}.ent \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \mathsf{E}_{\mathfrak{sc}}\mathsf{X}\phi \mid \mathsf{E}_{\mathfrak{sc}}\mathsf{G}\phi \mid \mathsf{E}_{\mathfrak{sc}}[\phi\mathsf{U}\phi] \mid \overline{\mathsf{K}}_{\mathtt{i}}\phi \mid \overline{\mathsf{C}}_{\Gamma}\phi$$

where $\mathtt{i} \in \mathcal{A}$, $\Gamma \subseteq \mathcal{A}$, $ent \in \mathcal{S}$, $\mathfrak{sc} \in \mathcal{SC}_{\mathcal{D}}$

$\mathfrak{sc} \in \mathcal{SC}_{\mathcal{D}}$ – Boolean formulae used to restrict contexts in path selection

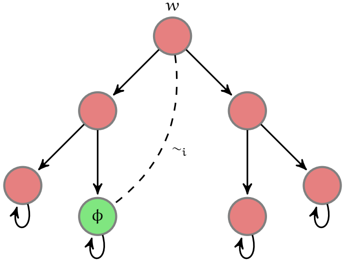
Epistemic indistinguishability relation

For each $\mathtt{i} \in \mathcal{A}$ we define $\sim_{\mathtt{i}} \subseteq St_{\text{CR-}\mathcal{D}} \times St_{\text{CR-}\mathcal{D}}$:

$\mathcal{S} \sim_{\mathtt{i}} \mathcal{S}'$ iff:

- $\mathcal{D}(\mathcal{S})[\mathtt{i}] = \mathcal{D}(\mathcal{S}')[\mathtt{i}]$
- $\mathcal{S}, \mathcal{S}' \in Reach(\text{CR-}\mathcal{D})$

$\sim^{\mathsf{C}}_{\Gamma}$ – transitive closure of $\bigcup_{\mathtt{i}\in\Gamma} \sim_{\mathtt{i}}$



$\mathcal{M}, w \models \overline{\mathsf{K}}_{\mathtt{i}}\phi$

$\mathsf{K}_{\mathtt{i}}\phi \stackrel{def}{=} \neg\overline{\mathsf{K}}_{\mathtt{i}}\neg\phi$

Syntax of rsCTLK:

$$\phi := \mathtt{i}.ent \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \mathsf{E}_{\mathfrak{sc}}\mathsf{X}\phi \mid \mathsf{E}_{\mathfrak{sc}}\mathsf{G}\phi \mid \mathsf{E}_{\mathfrak{sc}}[\phi \mathsf{U} \phi] \mid \overline{\mathsf{K}}_{\mathtt{i}}\phi \mid \overline{\mathsf{C}}_{\Gamma}\phi$$

where $\mathtt{i} \in \mathcal{A}$, $\Gamma \subseteq \mathcal{A}$, $ent \in \mathcal{S}$, $\mathfrak{sc} \in \mathcal{SC}_{\mathcal{D}}$

$\mathfrak{sc} \in \mathcal{SC}_{\mathcal{D}}$ – Boolean formulae used to restrict contexts in path selection

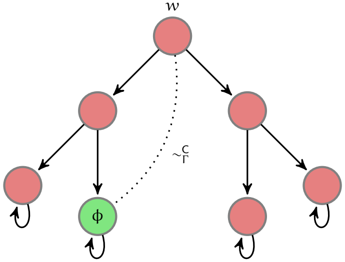
Epistemic indistinguishability relation

For each $\mathtt{i} \in \mathcal{A}$ we define $\sim_{\mathtt{i}} \subseteq St_{\text{CR-}\mathcal{D}} \times St_{\text{CR-}\mathcal{D}}$:

$\mathcal{S} \sim_{\mathtt{i}} \mathcal{S}'$ iff:

- $\mathcal{D}(\mathcal{S})[\mathtt{i}] = \mathcal{D}(\mathcal{S}')[\mathtt{i}]$
- $\mathcal{S}, \mathcal{S}' \in Reach(\text{CR-}\mathcal{D})$

$\sim^{\mathsf{C}}_{\Gamma}$ – transitive closure of $\bigcup_{\mathtt{i}\in\Gamma} \sim_{\mathtt{i}}$



$$\mathcal{M}, w \models \overline{\mathsf{C}}_{\Gamma}\phi$$

$$\mathsf{C}_{\Gamma}\phi \overset{def}{=} \neg\overline{\mathsf{C}}_{\Gamma}\neg\phi$$

Problem definition

Input: CRMARS, rsCTLK formula

Decision problem:

$$\mathcal{M}^{\mathrm{CRMARS}}_{\mathrm{rsCTLK}} \models \phi_{\mathrm{rsCTLK}} \quad \text{iff} \quad \phi_{\mathrm{rsCTLK}} \text{ holds in the initial state}$$

Theorem 2

Model checking for rsCTLK is PSPACE-complete

- PSPACE: similar to rsCTL; construction of a nondeterministic model checking algorithm using polynomial space
- PSPACE-hard: rsCTL model checking (PSPACE-complete) can be translated to rsCTLK model checking

Concentrations of entities:

$s \mapsto \mathtt{i}$ – multiplicity of s $\quad$ $\mathtt{i} \in \mathbb{N}$ $\quad$ e.g. $\{s \mapsto 2, x \mapsto 3, y\}$

Reaction system with (discrete) concentrations (RSC): $\mathcal{C} = (S, A)$

- S – finite **background set**
- A – nonempty finite set of **c-reactions** over S

$\mathcal{B}(S)$ – set of all multisets over S

$a = (\mathbf{r}, \mathbf{i}, \mathbf{p}) \in A$ – c-reaction

- $\mathbf{r}$, $\mathbf{i}$, $\mathbf{p}$ – reactant, inhibitor, and product concentration levels
- denoted: $\mathbf{r}_a$, $\mathbf{i}_a$, and $\mathbf{p}_a$
- $\mathbf{r}, \mathbf{i}, \mathbf{p} \in \mathcal{B}(S)$ with $\mathbf{r}(e) < \mathbf{i}(e)$, for every $e \in \mathit{carr}(\mathbf{i})$

$(\mathit{carr}(\mathbf{b}) = \{s \in S \mid \mathbf{b}(s) > 0\})$

$\mathbf{t} \in \mathcal{B}(S)$ – state (with concentrations)

$\mathfrak{a} \in A$ is enabled by $\mathbf{t}$: (denoted $en_{\mathfrak{a}}(\mathbf{t})$)

if $\mathbf{r}_{\mathfrak{a}} \leqslant \mathbf{t}$ and $\mathbf{t}(e) < \mathbf{i}_{\mathfrak{a}}(e)$, for every $e \in carr(\mathbf{i}_{\mathfrak{a}})$

Result of $\mathfrak{a}$ on $\mathbf{t}$: $res_{\mathfrak{a}}(\mathbf{t})$

- $res_{\mathfrak{a}}(\mathbf{t}) = \mathbf{p}_{\mathfrak{a}}$ if $en_{\mathfrak{a}}(\mathbf{t})$
- $res_{\mathfrak{a}}(\mathbf{t}) = \varnothing_S$ otherwise

Result of applying multiple reactions

Union of multisets representing individual results of c-reactions:

$$res_A(\mathbf{t}) = \bigcup\!\!\!\!\bigcup\{res_{\mathfrak{a}}(\mathbf{t}) \mid \mathfrak{a} \in A\}$$

Multiset expressions: $\mathfrak{a} \in BE(S)$

$$\mathfrak{a} ::= \mathit{true} \mid e \sim c \mid e \sim e \mid \neg\mathfrak{a} \mid \mathfrak{a} \vee \mathfrak{a}$$

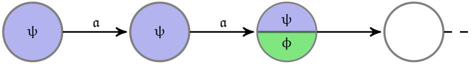
where: $\sim \in \{<, \leqslant, =, \geqslant, >\}$, $e \in S$, $c \in \mathbb{N}$

Syntax of rsLTL

$$\phi ::= \mathfrak{a} \mid \phi \wedge \phi \mid \phi \vee \phi \mid \mathsf{X}_{\mathfrak{a}}\phi \mid \phi \mathsf{U}_{\mathfrak{a}}\phi \mid \phi \mathsf{R}_{\mathfrak{a}}\phi$$

where $\mathfrak{a} \in BE(S)$

$$\sigma \models \phi \mathsf{R}_{\mathfrak{a}} \psi$$



Input: CRRSC, rsLTL formula

Decision problem:

$\mathcal{M}^{CRRSC}_{rsLTL} \models \phi_{rsLTL}$ iff ϕ_{rsLTL} holds for all the paths (starting in the initial state)

Existential model checking for rsLTL

$\mathcal{M}^{CRRSC}_{rsLTL} \models_{\exists} \phi_{rsLTL}$ iff ϕ_{rsLTL} holds for a path (starting in the initial state)

Theorem 3

Model checking for rsLTL is PSPACE-complete

- PSPACE: by reduction to LTL model checking
- PSPACE-hardness:
  - reachability for CRRSC is PSPACE-hard
  - existential rsLTL model checking (MC) is PSPACE-hard
  - coPSPACE=PSPACE: rsLTL MC is PSPACE-hard

- Reactions defined partially: missing reactants or inhibitors

Parametric reaction system (PRS): $\mathcal{P} = (S, P, A)$

- S – *background* set
- P – set of *parameters*
- A – set of *parametric reactions*, $A \neq \emptyset$

S, P, A are finite

Let $a = (\mathfrak{r}, \mathfrak{i}, \mathfrak{p}) \in A$: $\mathfrak{r}, \mathfrak{i}, \mathfrak{p} \in \mathcal{B}(S) \cup P$

- $\mathfrak{r}$, $\mathfrak{i}$, $\mathfrak{p}$ — denoted by $\mathfrak{r}_a$, $\mathfrak{i}_a$, and $\mathfrak{p}_a$
- reactants, inhibitors, and products of parametric reaction a

Example: let $\lambda \in P$

Parametric reaction: $(\{x, y\}, \lambda, \{z\})$

Parameter valuation of $\mathcal{P}$: $v : P \cup \mathcal{B}(S) \to \mathcal{B}(S)$

- $\mathcal{P}^{\leftarrow v}$ – $\mathcal{P}$ with concretised parameters in reactions (parameters become multisets)
- $\mathrm{PV}_{\mathcal{P}}$ – all the parameter valuations for $\mathcal{P}$
- $v \in \mathrm{PV}_{\mathcal{P}}$ is a valid parameter valuation if $\mathcal{P}^{\leftarrow v}$ yields a RSC

Parameter constraints: $\mathfrak{c} \in PC(\mathcal{P})$

$$\mathfrak{c} ::= true \mid \lambda[e] \sim c \mid \lambda[e] \sim \lambda[e] \mid \neg\mathfrak{c} \mid \mathfrak{c} \vee \mathfrak{c},$$

$$\lambda \in P, \quad e \in S, \quad c \in \mathbb{N}, \quad \sim \in \{<, \leqslant, =, \geqslant, >\}$$

Constrained parametric reaction system (CPRS): $\mathcal{CP} = (S, P, A, \mathfrak{c})$

- $\mathcal{P} = (S, P, A)$
- $\mathfrak{c} \in PC(\mathcal{P})$

- for $v \in \mathtt{PV}_{\mathcal{P}}$ we define $\mathcal{CP}^{\leftarrow v} = \mathcal{P}^{\leftarrow v}$
- $v \in \mathtt{PV}_{\mathcal{P}}$ is *valid* in $\mathcal{CP}$ if it is valid in $\mathcal{P}$ and it satisfies $\mathfrak{c}$

Context-restricted cprs (CR-CPRS): $\textsc{cr}\text{-}\mathcal{CP} = (\mathcal{CP}, \mathfrak{A})$

- $\mathcal{CP} = (S, P, A, \mathfrak{c})$
- $\mathfrak{A} = (Q, q_0, R)$ – context automaton over $\mathcal{B}(S)$

- $\textsc{cr}\text{-}\mathcal{CP}^{\leftarrow v} = (\mathcal{CP}^{\leftarrow v}, \mathfrak{A})$ – parameter substitution for CR-CPRS

Input:

- CR-$\mathcal{CP}$ – CR-CPRS
- $F = \{\phi_1, \ldots, \phi_n\}$ – rsLTL formulae (observations)
- $\mathfrak{c}$ – parameter constraint

Calculate a valid (under $\mathfrak{c}$) parameter valuation v of CR-$\mathcal{CP}$ such that:

$$(\mathcal{M}(\text{CR-}\mathcal{CP}^{\leftarrow \mathrm{v}}) \models_{\exists} \phi_1) \wedge \cdots \wedge (\mathcal{M}(\text{CR-}\mathcal{CP}^{\leftarrow \mathrm{v}}) \models_{\exists} \phi_n)$$

Theorem 4

The problem whether there is a valid parameter valuation is PSPACE*-complete.*

- PSPACE: nondeterministic algorithm guessing a valid valuation
- PSPACE-hardness: follows from PSPACE-hardness of rsLTL MC

RS verification toolkit

ReactICS – toolkit that implements verification methods presented in the thesis

- RSSL – Reaction Systems Specification Language
- ReactICS-BDD – MC and BMC methods based on BDDs (CUDD library)
- ReactICS-SMT – BMC and synthesis methods based on SMT (Z3 SMT-solver)
- Open Source, MIT licence: `https://arturmeski.github.io/reactics/`

The methods presented in the thesis were evaluated on different benchmarks:

rsCTL

- Heat shock response
- Binary counter
- Mutual exclusion protocol
- Abstract pipeline system

rsCTLK

- Train-gate-controller
- Distributed abstract pipeline

rsLTL

- Heat shock response (modified)
- Scalable chain

Synthesis with rsLTL

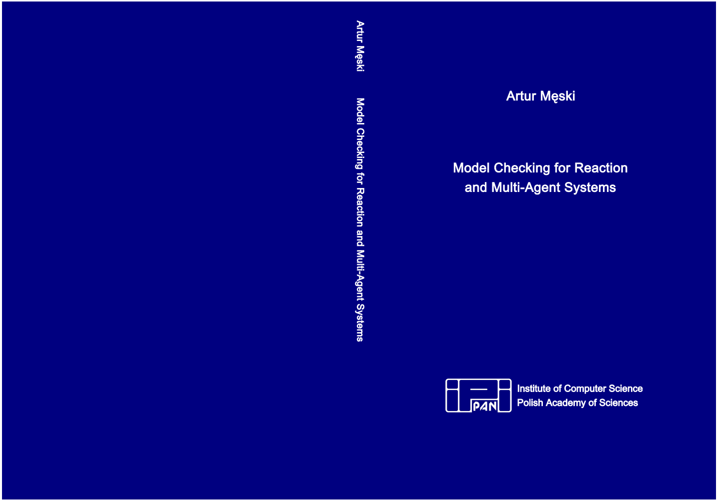
- (Parametric) mutual exclusion protocol

Contributions

- Model checking methods for reaction systems
  — The first (based on logic) model checking method for reaction systems
- Extensions of reaction systems
- Logics for expressing properties of reaction systems
- Parametric model checking approach for reaction synthesis
- Reaction systems verification toolkit

Publications:

1. *Model checking temporal properties of reaction systems*
   Information Sciences 313, 2015; A. Męski, W. Penczek, G. Rozenberg
2. *Towards Quantitative Verification of Reaction Systems*
   Unconventional Computation and Natural Computation, 2016; A. Męski, M. Koutny, W. Penczek
3. *Verification of Linear-Time Temporal Properties for Reaction Systems with Discrete Concentrations*
   Fundamenta Informaticae, 2017; A. Męski, M. Koutny, W. Penczek
4. *Reaction Mining for Reaction Systems*
   Unconventional Computation and Natural Computation, 2018; A. Męski, M. Koutny, W. Penczek


Artur Męski
Model Checking for Reaction and Multi-Agent Systems
Artur Męski
Model Checking for Reaction
and Multi-Agent Systems
PAN
Institute of Computer Science
Polish Academy of Sciences

Reviews

Remark 1:

- The results are valuable as they demonstrate the behaviour of the proposed algorithms
- Comparisons with other results/tools are missing
- Reviewer's suggestions:
    1. select problems for which there exist experimental results and which are using competing formalisms;
    2. formalise these problems using extensions of reaction systems;
    3. compare the results.

Remark 1:

- The results are valuable as they demonstrate the behaviour of the proposed algorithms
- Comparisons with other results/tools are missing
- Reviewer's suggestions:
  1. select problems for which there exist experimental results and which are using competing formalisms;
  2. formalise these problems using extensions of reaction systems;
  3. compare the results.

BDD- versus SAT-based bounded model checking for the existential fragment of linear temporal logic with knowledge: algorithms and their performance.
Auton. Agents Multi Agent Syst. 28(4), 2014
A. Męski, W. Penczek, M. Szreter, B. Woźna-Szcześniak, A. Zbrzezny

---

Remark 2:

1. The usage of the term *Multi-agent Systems* (MAS), especially in Chapter 4, is debatable.
2. These are in fact classical distributed systems with shared memory, which should exclude the usage of the term MAS.
3. Aspects of communication in MAS (ACL, KQML), cooperation, models such as BDI, and many others were omitted.
4. Reaction systems are low-level when compared with modern MAS (their formal specification) for which a high-level specification and modelling is expected.
5. The investigated epistemic properties seem natural. Are there any other properties typical for MAS, but not occurring in classical distributed systems can be expressed and investigated for the proposed formalisms?

Remark 3:

1. The approach to parametric reaction systems is interesting but its experimental evaluation is not satisfying.
    - How adequate are the results in the case of biochemical reactions that are the inspiration for reaction systems?
    - To what extent is it possible to extract reactions from (typically very large) data sets obtained from real reactions?

- Original contributions:
  - Lists one journal article, one conference paper, and two technical reports
  - Missing:
    1. Journal article:
       Model checking temporal properties of reaction systems
       Information Sciences 313, 2015; A. Męski, W. Penczek, G. Rozenberg
    2. Conference paper:
       Towards Quantitative Verification of Reaction Systems
       UCNC, 2016; A. Męski, M. Koutny, W. Penczek

S. Azimi, C. Gratie, S. Ivanov, L. Manzoni, I. Petre, A. E. Porreca
Complexity of model checking for reaction systems, TCS 623:103–113, 2016

- Different approach: does not specify model/logic
- Problems of complexity ranging from P to PSPACE-complete:
  - Different methods for lower complexity problems
- Some of their properties can be specified in rsCTL
- The paper refers to the model from:

  S. Azimi, B. Iancu, I. Petre
  Reaction system models for the heat shock response, Fundam. Informat. 11(3), 2014

  Using rsCTL we specify the properties listed in this paper:

  - mass-conservation of hse (heat shock element)
  - a single form of hse
  - mass-conversation of proteins
  - *misfolded proteins must be addressed*
  - a single form of hsf (heat shock factors)
  - stability of hsp (heat shock proteins)

Artur Męski

Model Checking for Reaction and Multi-Agent Systems

Artur Męski

Model Checking for Reaction and Multi-Agent Systems

Institute of Computer Science
Polish Academy of Sciences

Experimental evaluation

- Two implementations: for crrs and for crrsc
- crrs obtained from $\Theta(crrsc)$
- Incremental approach: unrolling of interactive processes

- Z3 SMT-solver, Python
- crrs – boolean variables
- crrsc – integers

M_e – maximal allowed value of e

- **Incrementation** of $e \in S$ when $g \in S$ is present:

$$\uparrow_e^g \stackrel{\mathrm{def}}{=} \{(\{e \mapsto j, g \mapsto 1\}, \emptyset_S, \{e \mapsto j+1\}) \mid 1 \leqslant j < M_e\}.$$

- **Decrementation** of $e \in S$ when $g \in S$ is present:

$$\downarrow_e^g \stackrel{\mathrm{def}}{=} \{(\{e \mapsto j, g \mapsto 1\}, \emptyset_S, \{e \mapsto j-1\}) \mid 2 < j \leqslant M_e\}.$$

- **Permanency** of $e \in S$, inhibitable by $\mathbf{i} \in \mathcal{B}(S)$:

$$\Diamond_e^{\mathbf{i}} \stackrel{\mathrm{def}}{=} \{(\{e \mapsto j\}, \mathbf{i}, \{e \mapsto j\}) \mid 1 \leqslant j \leqslant M_e\}.$$

- Encoded as simple/single operations on integer variables
- Allowed only when no ordinary reaction is enabled

Heat shock response

- Eukaryotic heat shock response
- Internal repair mechanism triggered when a cell is subjected to an environmental stressor

Properties:

- It is possible to enter the state where HSR may become stable:
  $\rho_1 = (\mathbf{x}_1, \mathbf{y}_1)$ where:
  - $\mathbf{x}_1 = \{hsp : hsf \mapsto 1, hse \mapsto 1, prot \mapsto 1\}$
  - $\mathbf{y}_1 = \{temp \mapsto 42\}$
- It is possible for the proteins to eventually misfold:
  $\rho_2 = (\mathbf{x}_2, \mathbf{y}_2)$ where:
  - $\mathbf{x}_2 = \{mfp \mapsto 1\}$
  - $\mathbf{y}_2 = \emptyset_S$

Heat shock response: results

	ρ_1		ρ_2	
	time [s]	memory [MB]	time [s]	memory [MB]
crrs	17.32	25.08	38.78	28.38
crrsc	**0.35**	**24.87**	**0.93**	**24.99**
improvement	49.48×	1.01×	41.69×	1.13×

n-reachability: ρ_1 is proved for $n = 4$, ρ_2 for $n = 9$

Scalable chain

- Abstract scalable system — $\text{CR-}\mathcal{C}_{\text{SC}} = (\mathcal{C}_{\text{SC}}, \mathfrak{A}_{\text{SC}})$
- Reactions incrementing concentration levels of m molecules up to a maximal concentration level k

$\mathcal{C}_{\text{SC}} = (S, \mathcal{P} \cup \mathcal{O} \cup \mathcal{F})$, where: $S = \{e_1, e_2, \ldots, e_m, inc, dec\}$,

- $\mathcal{P} = \{(\{e_i \mapsto k\}, \emptyset_S, \{e_{i+1} \mapsto 1)\}) \mid 1 \leqslant i < m\}$,
- $\mathcal{O} = \{ \uparrow^{inc}_{e_i}, \downarrow^{dec}_{e_i} \mid 1 \leqslant i \leqslant m\}$,
- $\mathcal{F} = \{(\{e_m \mapsto k\}, \{dec \mapsto 1\}, \{e_m \mapsto k\})\}$.

$\mathfrak{A}_{\text{SC}} = (Q, q_0, R)$, where: $Q = \{0, 1\}$, $q_0 = 0$,

- $(0, \{e_1 \mapsto 1, inc \mapsto 1\}, 1)$,
- $(1, \{inc \mapsto 1\}, 1)$,
- $(1, \{dec \mapsto 1\}, 1)$.

Scalable chain

Property: $\rho = (\{e_m \mapsto \mathrm{k}\}, \emptyset)$

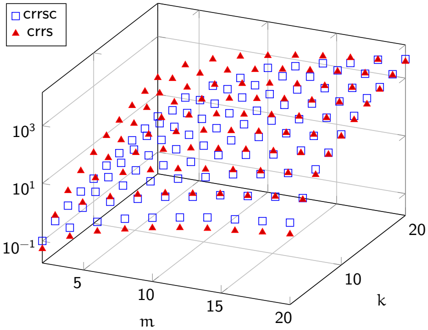


Figure: Time (in seconds)

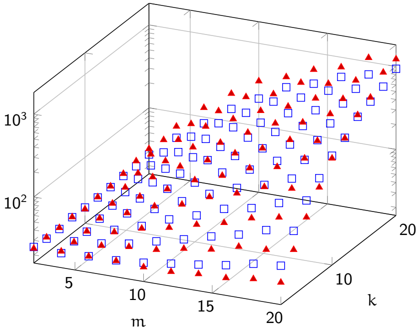


Figure: Memory (in MB)

Scalable chain

Property: $\rho = (\{e_m \mapsto k\}, \emptyset)$

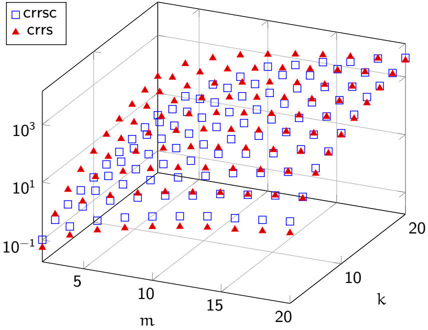


Figure: Time (in seconds)

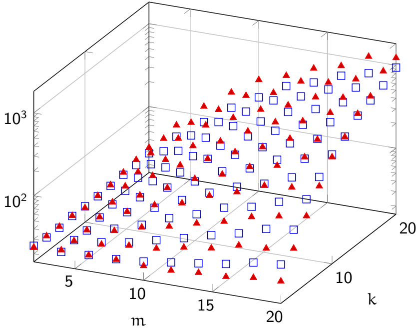


Figure: Memory (in MB)

$m = 8$, $k = 20$ – crrsc 5.6x faster

Scalable chain

Property: $\rho = (\{e_m \mapsto k\}, \emptyset)$

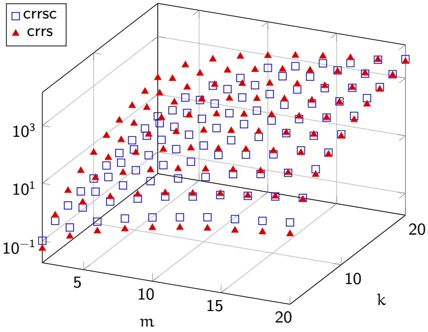


Figure: Time (in seconds)

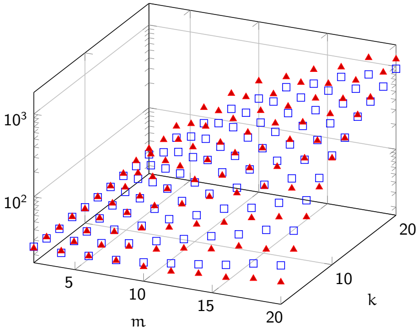


Figure: Memory (in MB)

$m = 10$, $k = 14$ – crrs 1.6x faster (1334 vs 2155 sec)

Conclusions

- Introduced reaction systems with discrete concentrations which support direct quantitative modelling
- Translation of reachability into SMT. Verification method
- One-to-one correspondence of crrs and crrsc processes
- Explicit concentration levels:
  - easier modelling
  - more efficient verification
  - possibilities for optimisations, e.g., macro-reactions

- Assumption: open system
- n-th process: additional (malicious) reaction with parameters:

$$P = \{\lambda_r, \lambda_i, \lambda_p\}$$

$$\textsc{cr}\text{-}\mathcal{CP}_M = ((S, P, A \cup \{(\lambda_r, \lambda_i, \lambda_p)\}, \mathfrak{c}), \mathfrak{A})$$

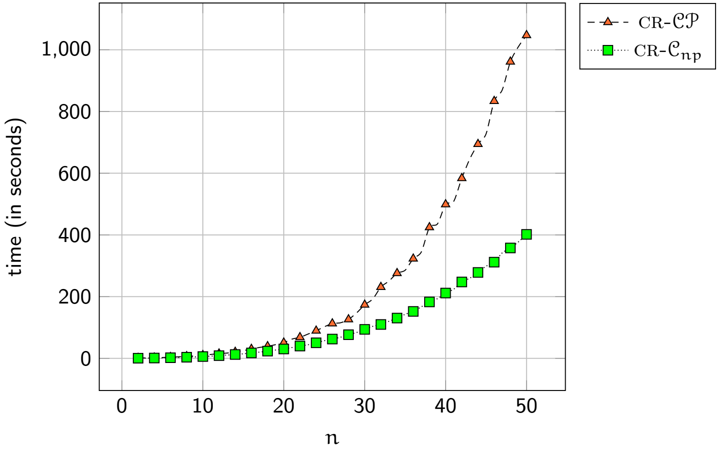
- $\mathfrak{c} = (\lambda_p[in_n] = 0) \wedge \bigwedge_{\lambda \in P, e \in S \setminus S_n} (\lambda[e] = 0)$ – additional reaction:
  - produces only entities related to the n-th process
  - cannot produce in_n (to avoid trivial solutions)

Synthesis: parameter valuation v of $\textsc{cr}\text{-}\mathcal{CP}_M$:

- $\phi = \mathsf{F}(in_1 \wedge in_n)$ – violation of mutual exclusion

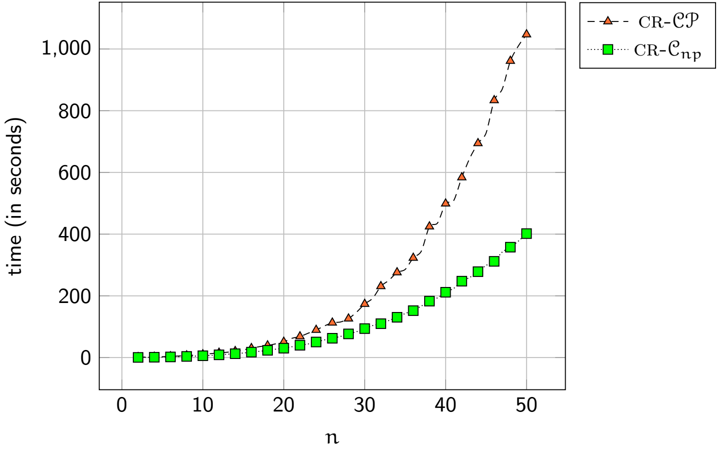
$$\mathcal{M}(\textsc{cr}\text{-}\mathcal{CP}_M^{\leftarrow v}) \models_\exists \phi$$

Results: time



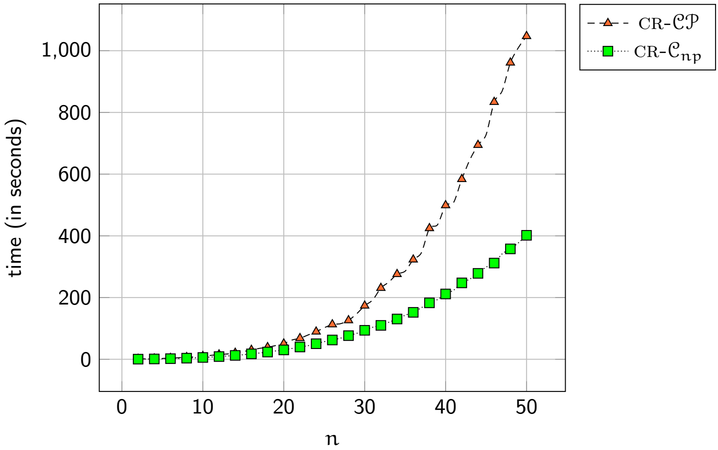
$\lambda_r^{\leftarrow v} = \{out_n\}$, $\lambda_i^{\leftarrow v} = \{s\}$, and $\lambda_p^{\leftarrow v} = \{req_n, done\}$

Results: time



CR-$\mathcal{CP}$ – parametric implementation

Results: time



CR-$\mathcal{C}_{np}$ – non-parametric (hard-coded valuation)

Results: memory