Towards a Regular Theory of Parameterized Concurrent Systems

Benedikt Bollig

Laboratoire Spécification et Vérification
ENS Cachan & CNRS, France

Reports on joint works with Paul Gastin, Akshay Kumar, and Jana Schubert.

Highlights 2014
When is an automata model «regular» or «robust»?
When is an automata model «regular» or «robust»?

- Decidable emptiness problem.
When is an automata model «regular» or «robust»?

- Decidable emptiness problem.
- Closed under union, intersection, and complementation.
When is an automata model «regular» or «robust»?

- Decidable emptiness problem.
- Closed under union, intersection, and complementation.
- Equivalent characterizations (MSO logic, ...).
When is an automata model «regular» or «robust»?

- Decidable emptiness problem.
- Closed under union, intersection, and complementation.
- Equivalent characterizations (MSO logic, …).

Here: A robust automata model for *parameterized concurrent systems*. 
When is an automata model «regular» or «robust»?

- Decidable emptiness problem.
- Closed under union, intersection, and complementation.
- Equivalent characterizations (MSO logic, ...).

Here: A robust automata model for parameterized concurrent systems.

In particular: Büchi-Elgot-Trakhtenbrot Theorems
When is an automata model «regular» or «robust»?

- Decidable emptiness problem.
- Closed under union, intersection, and complementation.
- Equivalent characterizations (MSO logic, ...).

Here: A robust automata model for parameterized concurrent systems.

In particular: Büchi-Elgot-Trakhtenbrot Theorems

Focus of previous work has been on verification:
- ...
Theorem [Büchi-Elgot-Trakhtenbrot 1960s]:
Finite Automata = MSO

word

Finite Automata
Theorem [Büchi-Elgot-Trakhtenbrot 1960s]:
Finite Automata = MSO

MSO over finite words:
- $a(x)$: position $x$ carries letter $a$
- $\text{succ}(x,y)$: $x$ and $y$ are successive word positions
Theorem [Büchi-Elgot-Trakhtenbrot 1960s]:
Finite Automata = MSO

**MSO over finite words:**
- $a(x)$: position $x$ carries letter $a$
- $\text{succ}(x, y)$: $x$ and $y$ are *successive* word positions

$$\forall x(a(x) \rightarrow \exists y(\text{succ}^*(x, y) \land b(y)))$$
Theorem [Büchi-Elgot-Trakhtenbrot 1960s]:
Finite Automata = MSO

MSO over finite words:
- $a(x)$: position $x$ carries letter $a$
- $\text{succ}(x,y)$: $x$ and $y$ are successive word positions

\[
\forall x (a(x) \rightarrow \exists y (\text{succ}^*(x,y) \land b(y)))
\]
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO

assume rendez-vous communication
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO

MSO:
- \text{succ}(x,y) \quad x \text{ and } y \text{ are successors on a process}
- c_1(y,z) \quad y \text{ and } z \text{ form a message exchange through channel } c_1

assume rendez-vous communication
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO

MSO:
succ(x,y)  x and y are successors on a process
c1(y,z)  y and z form a message exchange through channel c1

Communicating Automata

\( S_1 \xrightarrow{!m,c_3} S_2 \)
assume rendez-vous communication
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]: Communicating Automata = MSO

MSO:
- \( \text{succ}(x, y) \) x and y are successors on a process
- \( c_1(y, z) \) y and z form a message exchange through channel \( c_1 \)

Communicating Automata

assume rendez-vous communication
Theorem [Mukund et al. 2005; Genest-Kuske-Muscholl 2006]:
Communicating Automata = MSO

MSO:
- \(\text{succ}(x,y)\) \(x\) and \(y\) are successors on a process
- \(c_1(y,z)\) \(y\) and \(z\) form a message exchange through channel \(c_1\)

fixed topology
assume rendez-vous communication
Parameterized Communicating Automata (PCAs)

parameterized topology: fixed finite set of channel names $c_1, c_2$ structures of bounded degree
Parameterized Communicating Automata (PCAs)

Parameterized topology: fixed finite set of channel names → structures of bounded degree

Acceptance condition: (restricted) MSO formula over «topology + final states»
Parameterized Communicating Automata (PCAs)

Theorem [B, Gastin, Kumar 2014]:
Over pipelines, PCAs are **not complementable.**
Parameterized Communicating Automata (PCAs)

Theorem [B, Gastin, Kumar 2014]:
Over pipelines, PCAs are not complementable.

Theorem [B 2014]:
PCAs are expressively equivalent to $\text{EMSO}(msg_c, \text{succ}^*)$ on «unambiguous» topology classes
(such as pipelines, trees, rings, and grids).

B: Logic for Communicating Automata with Parameterized Topology. CSL-LICS 2014.
Theorem [B, Gastin, Kumar 2014]:
Over pipelines, PCAs are **not complementable**.

Theorem [B 2014]:
PCAs are expressively equivalent to $\text{EMSO}(\text{msg}_c, \text{succ}^*)$ on «unambiguous» topology classes (such as pipelines, trees, rings, and grids).

Theorem [B, Gastin, Kumar 2014]:
PCAs are expressively equivalent to $\text{MSO}(\text{msg}_c, \text{succ})$ on the class of all topologies when processes are **context-bounded** (e.g., bounded number of channel switches).

Parameterized Communicating Automata (PCAs)

Theorem [B, Gastin, Kumar 2014]:
Over pipelines, PCAs are **not complementable**.

Theorem [B 2014]:
PCAs are expressively equivalent to \( \text{EMSO}(msg_c, \text{succ}^*) \) on «unambiguous» topology classes (such as pipelines, trees, rings, and grids).

Theorem [B, Gastin, Kumar 2014]:
PCAs are expressively equivalent to \( \text{MSO}(msg_c, \text{succ}) \) on the class of **all topologies** when processes are **context-bounded** (e.g., bounded number of channel switches).

Theorem [B, Gastin, Schubert 2014]:
**Emptiness** of context-bounded PCAs is decidable on the classes of pipelines, trees, rings.

Parameterized Communicating Automata (PCAs)

Theorem [B 2014]:
PCAs are expressively equivalent to $\text{EMSO}(\text{msg}_c, \text{succ}^*)$ on «unambiguous» topology classes (such as pipelines, trees, rings, and grids).

Theorem [B, Gastin, Kumar 2014]:
PCAs are expressively equivalent to $\text{MSO}(\text{msg}_c, \text{succ})$ on the class of all topologies when processes are context-bounded (e.g., bounded number of channel switches).

Proof: By induction, using complementability (via determinization).

Parameterized Communicating Automata (PCAs)

Theorem [B 2014]:
PCAs are expressively equivalent to $\text{EMSO}(\text{msg}_c, \text{succ}^*)$ on «unambiguous» topology classes (such as pipelines, trees, rings, and grids).

**Proof:** Normal form due to [Schwentick-Barthelmann 1999].

Theorem [B, Gastin, Kumar 2014]:
PCAs are expressively equivalent to $\text{MSO}(\text{msg}_c, \text{succ})$ on the class of all topologies when processes are context-bounded (e.g., bounded number of channel switches).

**Proof:** By induction, using complementability (via determinization).

Every EMSO($msg_\mathfrak{c}$, $succ^*$) formula is equivalent to

«... s.t. all bounded portions satisfy $\phi \in FO(msg_\mathfrak{c}, succ^*)».
Parameterized Communicating Automata (PCAs)

Every EMSO(msg, succ*) formula is equivalent to
«... s.t. all bounded portions satisfy \( \phi \in \text{FO}(\text{msg}, \text{succ}^*) \)». 
Every EMSO(msg_c, succ*) formula is equivalent to
«... s.t. all bounded portions satisfy φ ∈ FO(msg_c, succ*)».
Every EMSO(msg_c, succ*) formula is equivalent to
«… s.t. all bounded portions satisfy \( \phi \in FO(msg_c, succ^*) \)>>.
Every process traverses a bounded number of «zones».

Parameterized Communicating Automata (PCAs)

Every EMSO(msg_c, succ*) formula is equivalent to
«… s.t. all bounded portions satisfy \( \phi \in FO(msg_c, succ^*) \)». 
Future Work

- Topologies of unbounded degree (unranked trees, stars, ...)

![Diagram of unbounded degree topologies]
Future Work

- Topologies of unbounded degree (unranked trees, stars, …)
- Include data in messages (e.g., pids)
Future Work

- Topologies of unbounded degree (unranked trees, stars, ...)

- Include data in messages (e.g., pids)

Thank You!