A new approach for showing termination of parameterized transition systems

Alpine Verification Meeting '24

Roland Herrmann
University of Regensburg - Theoretical Computer Science

Joint work:

Philipp Rümmer

Universität Regensburg

What I will talk about?

A transition system is broadly speaking just a set of **states** and **transitions** between those states.

(Informal) Example: Token Passing

Consider n people standing in a row. Each of them can either hold a token (T) or not (0). They can hand their token to their respective right neighbour, receive a token from their left neighbour or stay without a token. For $n = 4$ an illustration of the transitions can look as follows:

$$T000 \to 0T00 \to 00T0 \to 000T$$

This will always stop when the token arrives at the rightmost position.

Question: How can we verify, whether a transition system terminates on any input?
Approach: Regular model checking techniques: Consider transition systems, which are described by an *automaton* and exploit the *automatic framework* to construct an *automaton*, that searches for a proof for termination *automatically*.

Regular Transition Systems

Definition

Definition: (Regular Transition System)

A regular transition system (RTS) is a pair $(\Sigma, \mathcal{T})$, which consists of a finite set Σ and a (length preserving) $\Sigma - \Sigma$-transducer, that is a deterministic finite automaton with $\Sigma \times \Sigma$ as its alphabet. We call

- $x \in \Sigma$ a state
- $w \in \Sigma^*$ a configuration
- $u_1 \dots u_n \otimes v_1 \dots v_n := (u_1, v_1) \dots (u_n, v_n) = \binom{u_1}{v_1} \cdots \binom{u_n}{v_n} \in \mathcal{L}(\mathcal{T})$ a transition from u to v
- the $i-th$ position in a configuration the $i-th$ agent

$(\Sigma, \mathcal{T})$ describes infinitely many transition systems, with the length $n \in \mathbb{N}$ as a parameter, hence RTS are a subset of **parameterized transition systems**.
Since $\mathcal{T}$ is length preserving and Σ finite, the set of reachable configurations is finite $\Rightarrow (\Sigma, \mathcal{T})$ is **weakly-finite**.

Regular Transition Systems

Example

Example: Token Passing

- $\Sigma = \{T, 0\}$, $T =$ "token" , $0 =$ "no token"
- $\mathcal{T} = \binom{0}{0}^* \binom{T}{0} \binom{0}{T} \binom{0}{0}^*$

For example

$$0T00$$

is a configuration for $n = 4$, the first, third and fourth agent are in state "no token" , whereas the second agent is in state "token". The word $\binom{0}{0}\binom{T}{0}\binom{0}{T}\binom{0}{0}$ describes the only possible transition $0T00 \to 00T0$

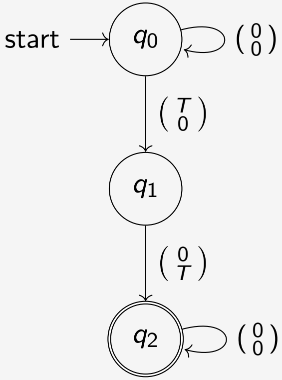


Figure: $\mathcal{T}$

Inductive Invariants

How can we find invariants in RTS?

Usually: We want to construct an automaton $\mathcal{I}$ that describes an invariant ($\mathcal{L}(\mathcal{I})$) as a regular language. An invariant should satisfy

$$\mathcal{I}(x) \wedge \mathcal{T}(x, y) \longrightarrow \mathcal{I}(y)$$

Problem: The construction of $\mathcal{I}$ is hard. The question if there exists an automaton $\mathcal{I}$ is undecideable.
Solution: Describe invariants through words (instead of a whole automaton). Then checking if a word describes an invariant reduces to whether the word is accepted by a fixed automaton, that accepts words that describe invariants. This approach is due to "Regular Model Checking Upside-Down: An Invariant-Based Approach", Javier Esparza, Mikhail Raskin, Christoph Welzel-Mohr.

Inductive Invariants

Example

Statements on $(\Sigma, \mathcal{T})$ are words in another alphabet Γ. A fixed $\Sigma - \Gamma$ transducer $\mathcal{V}$ is called an interpretation. A statement $I \in \Gamma^*$ holds for a configuration $w \in \Sigma^*$ if $|w| = |I|$ and $w \otimes I \in \mathcal{L}(\mathcal{V})$.

Example: Token Passing

- $\Gamma = 2^{\Sigma}$
- $\mathcal{V} = \mathcal{V}_{Trap}$

Then $00T0$ satisfies $\{T\}\{T\}\{T\}\{T\}$ in the interpretation $\mathcal{V}_{Trap}$. In everyday language $\{T\}^n$ can be read as "there is at least one token".

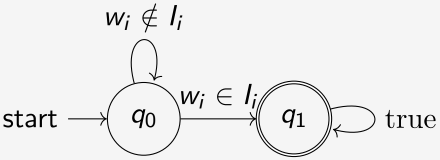


Figure: $\mathcal{V}_{Trap}$

Inductive Invariants

Construction of Automaton

Roland Herrmann

Write $I(x)$ if $x \otimes I \in \mathcal{V}$. The property of an invariant is then

$$\forall_{x,y \in \Sigma^*} I(x) \wedge \mathcal{T}(x,y) \longrightarrow I(y)$$

Construct an automaton that accepts exactly those statements that satisfy this formula.
$\rightarrow$ get rid of universal quantifier by constructing the complement $\mathcal{A}_{ind}^{C}$ first.
$\rightarrow$ $\mathcal{A}_{ind}^{C}$ accepts exactly those statements that are **not** an invariant, i.e.

$$\exists_{x,y \in \Sigma^*} I(x) \wedge \mathcal{T}(x,y) \wedge \neg I(y)$$

Inductive Invariants

Construction of Automaton

The construction of $\mathcal{A}_{ind}^{C}$ reflects the formula for not being an inductive invariant.

$$Q_{\mathcal{A}_{ind}^{C}} = \underbrace{Q_{\mathcal{T}}}_{\mathcal{T}(x,y)} \times \underbrace{Q_{\mathcal{V}}}_{I(x)} \times \underbrace{Q_{\mathcal{V}}}_{I(y)}$$

We have $((p_{\mathcal{T}}, p_1, p_2), l_i, (q_{\mathcal{T}}, q_1, q_2)) \in \delta_{\mathcal{A}_{ind}^{C}}$ iff there exist x_i, y_i such that

$$\begin{aligned} (p_{\mathcal{T}}, (x_i, y_i), q_{\mathcal{T}}) &\in \delta_{\mathcal{T}} && (\mathcal{T}(x,y)) \\ (p_1, (x_i, l_i), q_1) &\in \delta_{\mathcal{V}} && (I(x)) \\ (p_2, (y_i, l_i), q_2) &\in \delta_{\mathcal{V}} && (I(y)) \end{aligned}$$

$$F_{\mathcal{A}_{ind}^{C}} = \underbrace{F_{\mathcal{T}}}_{\mathcal{T}(x,y)} \times \underbrace{F_{\mathcal{V}}}_{I(x)} \times \underbrace{Q_{\mathcal{V}} \setminus F_{\mathcal{V}}}_{\neg I(y)}$$

All words accepted by $\mathcal{A}_{ind}$ correspond to inductive invariants, e.g. $\{T\}^n$ for Token passing and $\mathcal{V}_{Trap}$ is an inductive invariant.

Statements

Towards termination

Can we adjust this setting $(\Gamma, \mathcal{V})$ to prove termination?

- $\Gamma = 2^{\Sigma \times \Sigma}$
- $\mathcal{V} = ?$ ($\mathcal{V}_{Trap}$ is possible)
- Consider the induced relation $R_I = \{(u, v) \in \Sigma^* \times \Sigma^* \mid (u \otimes v) \otimes I \in \mathcal{L}(\mathcal{V})\}$
- $\mathcal{T} \subseteq R_I$
- "R_I is a proof for termination"

Example: Token passing

- $\Gamma = \mathcal{P}(\{\binom{0}{0}, \binom{T}{T}, \binom{T}{0}, \binom{0}{T}\})$
- $\mathcal{V} = \mathcal{V}_{Trap}$
- $I = \{\binom{T}{0}\}^4$
- $R_I = \bigcup_{n=0}^{3} \binom{\Sigma}{\Sigma}^n \binom{T}{0} \binom{\Sigma}{\Sigma}^{3-n} \supseteq (\mathcal{T} \cap \Sigma^4 \times \Sigma^4)$

⚠ R_I is not a proof for termination here

Termination

Proof Setup

Roland Herrmann

In order to show that a RTS terminates, it suffices to find a **well-founded** relation on the set of configurations that **overapproximates the transition relation**.
Weakly-finiteness gives the following result:

Lemma

$R \subseteq S \times S$ an irreflexive, transitive relation a finite set S (e.g. Σ^n), then R is well-founded.

Proof conditions

Write $R(x, y) \equiv \mathit{true} :\Leftrightarrow (x, y) \in R$. If $I \in (2^{\Sigma \times \Sigma})^*$, R_I has to be

1. irreflexive: $R_I(x, y) \longrightarrow x \neq y$
2. transitive: $R_I(x, y) \wedge R_I(y, z) \longrightarrow R_I(x, z)$
3. containing the transition relation $\mathcal{T}(x, y) \longrightarrow R_I(x, y)$.

We still lack an interpretation $\mathcal{V}$!

Lexicographically ordered systems

Definition

Definition: Lexicographic order

Let $>_\Sigma$ be a strict order relation on Σ. Then the following induced strict order relation is called lexicographic order relation

$$u_1 \dots u_n >_{lex} v_1 \dots v_m :\Leftrightarrow \exists i \in \{1, \dots, n\}. \ (u_i >_\Sigma v_i \wedge \forall j < i. \ u_j = v_j) \\ \vee (n > m \wedge u_1 \dots u_m = v_1 \dots v_m).$$

Lexicographic orders are the way words are arranged in a dictionary. ($T > 0$ for Token passing)

Example: Counting down in binary

If we consider $1 > 0$, then counting down in binary is lexicographically ordered.

$$\underline{1}000 >_{lex} 011\underline{1} >_{lex} 01\underline{1}0 >_{lex} 010\underline{1} >_{lex} 01\underline{0}0 >_{lex} 001\underline{1} >_{lex} 00\underline{1}0 >_{lex} 000\underline{1} >_{lex} 0000$$

Lexicographically ordered systems

Proof setup

$\mathcal{V}$ should interpret statements $I \in \Gamma^*$ as lexicographic orders!

- $\Delta = \{\binom{x}{x} \mid x \in \Sigma\}$
- $\mathcal{V}_{lex}$ accepts words $w \otimes I$, where w models a transition with respect to the lexicographic order given by I
- Analogously to the inductive invariants case, one can construct an automaton $\mathcal{A}_{lex}$ which accepts those statements which satisfy the proof conditions (irreflexive, transitive, contain $\mathcal{T}$) with respect to $\mathcal{V}_{lex}$

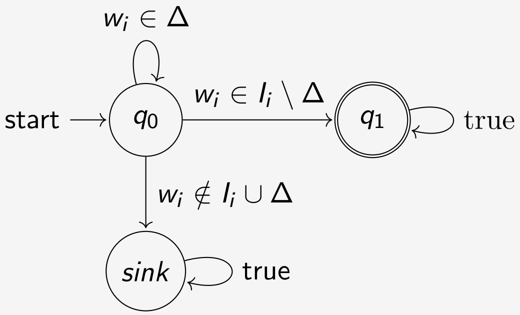


Figure: $\mathcal{V}_{lex}$

Lexicographically ordered systems

Construction of search automaton

Roland Herrmann

- $I \in \mathcal{L}(\mathcal{A}_{lex}) \Leftrightarrow I$ satisfies the proof conditions
- $\Gamma = 2^{\Sigma \times \Sigma}$ is the alphabet of $\mathcal{A}_{lex}$
- proof conditions are (implicitly) universally quantified $\rightarrow$ construct complement $\mathcal{A}_{lex}^C$ to eliminate the existential quantifier and take complement again
- $\mathcal{A}_{lex}^C$ accepts a statement I if there **exists** configurations $x, y, z \in \Sigma^*$, such that R_I fails to satisfy at least one of the proof conditions

$$Q_{\mathcal{A}_{lex}^C} = \underbrace{Q_{\mathcal{T}}}_{\mathcal{T}(x,y)} \times \underbrace{Q_{=}}_{x=y} \times \underbrace{Q_{\mathcal{V}_{lex}}}_{R_I(x,y)} \times \underbrace{Q_{\mathcal{V}_{lex}}}_{R_I(y,z)} \times \underbrace{Q_{\mathcal{V}_{lex}}}_{R_I(x,z)}$$

Each factor in $Q_{\mathcal{A}_{lex}^C}$ models one of the predicates occurring in the proof conditions.

Lexicographically ordered systems

Construction of search automaton

The transition relation $\delta_{\mathcal{A}^C_{lex}}$ is given by

$$((p_{\mathcal{T}}, p_{=}, p_1, p_2, p_3), l_i, (q_{\mathcal{T}}, q_{=}, q_1, q_2, q_3)) \in \delta_{\mathcal{A}^C_{lex}}$$

if and only if there exist $x_i, y_i, z_i \in \Sigma$, such that

$$(p_{\mathcal{T}}, (x_i, y_i, l_i), q_{\mathcal{T}}) \in \delta_{\mathcal{T}} \qquad (\mathcal{T}(x, y))$$

$$(p_{=}, (x_i, y_i), q_{=}) \in \delta_{=} \qquad (x = y)$$

$$(p_1, (x_i, y_i, l_i), q_1) \in \delta_{\mathcal{V}_{lex}} \qquad (R_l(x, y))$$

$$(p_2, (y_i, z_i, l_i), q_2) \in \delta_{\mathcal{V}_{lex}} \qquad (R_l(y, z))$$

$$(p_3, (x_i, z_i, l_i), q_3) \in \delta_{\mathcal{V}_{lex}} \qquad (R_l(x, z))$$

Lexicographically ordered systems

Construction of search automaton

The accepting states correspond $1:1$ to the negated proof conditions

$$\underbrace{(R_l(x,y) \wedge x = y)}_{\neg 1.} \vee \underbrace{(R_l(x,y) \wedge R_l(y,z) \wedge \neg R_l(x,z))}_{\neg 2.} \vee \underbrace{(\mathcal{T}(x,y) \wedge \neg R_l(x,y))}_{\neg 3.}.$$

$$\begin{aligned} F_{\mathcal{A}^C_{lex}} =& Q_{\mathcal{T}} \times F_{=} \times F_{\mathcal{V}_{lex}} \times Q_{\mathcal{V}_{lex}} \times Q_{\mathcal{V}_{lex}} && \neg 1. \\ & \cup Q_{\mathcal{T}} \times Q_{=} \times F_{\mathcal{V}_{lex}} \times F_{\mathcal{V}_{lex}} \times (Q_{\mathcal{V}_{lex}} \setminus F_{\mathcal{V}_{lex}}) && \neg 2. \\ & \cup F_{\mathcal{T}} \times Q_{=} \times (Q_{\mathcal{V}_{lex}} \setminus F_{\mathcal{V}_{lex}}) \times Q_{\mathcal{V}_{lex}} \times Q_{\mathcal{V}_{lex}} && \neg 3. \end{aligned}$$

We summarize: $\mathcal{A}^C_{lex} = (Q_{\mathcal{A}^C_{lex}}, \Gamma, (s_{\mathcal{T}}, s_{=}, s_{\mathcal{V}_{lex}}, s_{\mathcal{V}_{lex}}, s_{\mathcal{V}_{lex}}), \delta_{\mathcal{A}^C_{lex}}, F_{\mathcal{A}^C_{lex}})$.

Theorem:

Let $(\Sigma, \mathcal{T})$ be a RTS, $\mathcal{A}_{lex}^{C}$ the corresponding automaton according to our construction above. If ${\mathcal{A}_{lex}^{C}}^{C} = \mathcal{A}_{lex}$ has a word of every length, i.e. $\mathcal{L}(\mathcal{A}_{lex}) \cap \Sigma^n \neq \emptyset$ for all $n \in \mathbb{N}$, then $(\Sigma, \mathcal{T})$ terminates.

For the Token passing example, we obtain $\{\binom{T}{0}\}^n \in \mathcal{A}_{lex}$ for all $n \in \mathbb{N}$.

Corollary:

Let $(\Sigma, \mathcal{T})$ be a lexicographically ordered RTS. Then $\mathcal{A}_{lex} \cap \Sigma^n \neq \emptyset$ for all $n \in \mathbb{N}$.

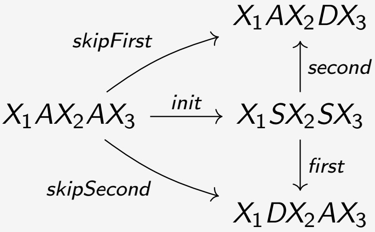
Note that $\mathcal{A}_{lex}$ does also prove RTS terminating, which allow different letterwise orders at different positions, e.g. even agents counting down and odd agents counting up.

Letterwise ordered systems

Motivating Example

Example: Polite Mexican Standoff

n armed agents are alive (state A) and each of them wants to kill the others to be the only one left alive. In order to bring the whole thing to a neat and tidy end, they randomly pick two of them to transition into a shooting state (S). After that one of them transitions into a dead state (D) and the other goes back to his alive state. The transitive closure of the transition relation $\mathcal{T}$ is described on the right, where $X_i \in (A + D)^*$ for $i \in \{1, \ldots, 3\}$



Letterwise ordered systems

Adjustments

Problem: The polite mexican standoff is not lexicographically ordered.
Solution: Define $\mathcal{V}_{All}$ to detect changes at all positions.

- Construct again a order relation on Σ^n
- An outer relation (induced by) $A >_1 D$ can model the coarse structure.
- Loops $A \to S \to A$ are possible at some positions, the outer relation does not need to distinguish them $A =_1 S$
- All transitions are covered except *init*. In the case $x =_1 y$ another relation should cover *init* by $A >_2 S$

For a relation $R \subseteq S \times S$ we write

- $x \geq_R y :\Leftrightarrow (x, y) \in R$
- $x >_R y :\Leftrightarrow (x, y) \in R \wedge (y, x) \notin R$
- $x =_R y :\Leftrightarrow (x, y), (y, x) \in R$

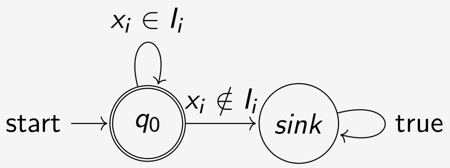


Figure: $\mathcal{V}_{All}$

Letterwise ordered systems

Proof Conditions

Synthesize an irreflexive, transitive relation from two preorders (reflexive and transitive), the "outer relation" R_1 and the "inner relation" R_2

$$x >_3 y :\Leftrightarrow x >_1 y \vee (x =_1 y \wedge x >_2 y)$$

Lemma:

Let R_1, R_2 be two preorders, R_3 as above, then R_3 is irreflexive and transitive.

Proof Conditions

If $I_1, I_2 \in 2^{\Sigma \times \Sigma}$, R_1, R_2 has to be

1. reflexive: $x = y \longrightarrow R_i(x, y)$
2. transitive: $R_i(x, y) \wedge R_i(y, z) \longrightarrow R_i(x, z)$
3. containing the transition relation:
   $\mathcal{T}(x, y) \rightarrow R_1(x, y) \wedge \neg R_1(y, x) \vee (R_1(x, y) \wedge R_1(y, x) \wedge R_2(x, y) \wedge \neg R_2(y, x))$

A corresponding automaton $\mathcal{A}_{All}$, that accepts exactly those pairs (l_1, l_2) that satisfy the proof conditions can be constructed analogously to $\mathcal{A}_{lex}$. Note that now we have $2^{\Sigma\times\Sigma} \times 2^{\Sigma\times\Sigma}$ as alphabet and eight copies of $\mathcal{V}_{All}$ for the predicates.

Theorem:

Let $(\Sigma, \mathcal{T})$ be a RTS, $\mathcal{A}_{All}$ the corresponding automaton according to our construction. If $\mathcal{A}_{All}$ has a word of every length, i.e., $\mathcal{L}(\mathcal{A}_{All}) \cap \Sigma^n \neq \emptyset$ for all $n \in \mathbb{N}$, then $(\Sigma, \mathcal{T})$ terminates.

One can iterate this process with $n \in \mathbb{N}$ nested relations

$$
\begin{aligned}
x >_{n+1} y :\Leftrightarrow\ & x >_1 y \\
& \vee (x =_1 y \wedge x >_2 y) \\
& \vee (x =_1 y \wedge x =_2 y \wedge x >_3 y) \vee \dots \\
& \vee \Big(\bigwedge_{i=1}^{n-1} x =_i y \wedge x >_n y\Big)
\end{aligned}
$$

Example: Polite Mexican Standoff

For the polite mexican standoff, the following words are accepted by $\mathcal{A}_{All}$ for all $n \in \mathbb{N}$

$$\left\{\binom{A}{A},\binom{S}{S},\binom{D}{D},\binom{A}{S},\binom{S}{A},\binom{A}{D},\binom{S}{D}\right\}^n \otimes \left\{\binom{A}{A},\binom{S}{S},\binom{D}{D},\binom{A}{S},\binom{S}{D},\binom{A}{D},\binom{D}{S}\right\}^n$$

Conclusion

- The construction of the desired automaton follows a general pattern (once reasonable $\mathcal{V}$ and proof conditions are found)
- ⚠ Many choices of $\mathcal{V}$ and Γ result in empty or universal automata.
- ⚠ We need to complement our automaton at some point to get rid of the universal quantifier of the proof conditions, hence it is feasible to use parametric automata to handle infinite alphabets Σ.
- The whole setup with Γ and $\mathcal{V}$ can possibly be used to tackle other verification problems due to its flexibility.

A new approach for showing termination of parameterized transition systems

Thank you for listening !
Any Questions ?

Universität Regensburg