A Coinductive Approach to Proving Reachability Properties in Logically Constrained Term Rewriting Systems

Ștefan Ciobâcă and Dorel Lucanu

Faculty of Computer Science
Alexandru Ioan Cuza University, Iași, Romania
stefan.ciobaca@info.uaic.ro

July 14th, 2018
IJCAR, FLOC

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS/CCCDI - UEFISCDI, project number PN-III-P2-2.1-BG-2016-0394, within PNCDI III.

Outline

What are LCTRSs

1. A way of combining term rewriting with SMT solving;
2. Introduced in a preliminary form by Kirchner et al. in 1990;
3. Rediscovered recently (e.g., by Kop et al. in FroCoS 2013).

Running example:

$$\begin{aligned}
&\mathit{init}(n) \twoheadrightarrow \mathit{loop}(n, 2) \ \texttt{if} \ \top, \\
&\mathit{loop}(i \times k, i) \twoheadrightarrow \mathit{comp} \ \texttt{if} \ k > 1, \\
&\mathit{loop}(n, i) \twoheadrightarrow \mathit{loop}(n, i + 1) \ \texttt{if} \ \neg(\exists k.k > 1 \wedge n = i \times k).
\end{aligned}$$

Features: free function symbols (e.g., *init*, *loop*, *comp*), interpreted function symbols (e.g., 2, $\times$, $+$), logical constraints (e.g., $k > 1$, $\neg(\exists k.k > 1 \wedge n = i \times k)$).

The Semantics of LCTRSs

We assume a *builtin model* M^{b} for a many-sorted *builtin signature* $\Sigma^{\mathsf{b}} = (S^{\mathsf{b}}, F^{\mathsf{b}})$.
Example: $S^{\mathsf{b}} = \{\mathit{Int}, \mathit{Bool}\}$, $F^{\mathsf{b}} = \{+, \times, \wedge, \vee, \ldots\}$.
Key feature: constraints over builtin model solvable by SMT.

We extend the builtin model M^{b} to M^{Σ}:

- $M^{\Sigma}_f = M^{\mathsf{b}}_f$ for each $f \in F^{\mathsf{b}}$;
- M^{Σ}_f is the term constructor $M^{\Sigma}_f(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$, for each $f \in F \setminus F^{\mathsf{b}}$.
- $M^{\Sigma}_s = T_{\left((F \setminus F^{\mathsf{b}}) \cup \bigcup_{u \in S^{\mathsf{b}}} F_{\varepsilon,u}\right),s}$, for each $s \in S \setminus S^{\mathsf{b}}$;

Semantics: A LCTRS $\mathcal{R}$ defines a sort-indexed transition system $(M^{\Sigma}, \rightsquigarrow_{\mathcal{R}})$.
For $\mathcal{R} \supseteq \mathit{loop}(n, i) \twoheadrightarrow \mathit{loop}(n, i+1)$ `if` $\neg(\exists k.k > 1 \wedge n = i \times k)$:

$$\mathit{loop}(10, 4) \rightsquigarrow_{\mathcal{R}} \mathit{loop}(10, 5)$$

Outline

Reachability Formulas for LCTRSs

Reachability formulas:

$$\underbrace{\langle \mathit{init}(n) \,|\, \exists u.1 < u < n \wedge n \bmod u = 0 \rangle}_{\textit{constrained term (lhs)}} \Rightarrow \quad \underbrace{\langle \mathit{comp} \,|\, \top \rangle}_{\textit{constrained term (rhs)}}$$

Intuitive meaning: any ground instance of the lhs such that the lh constraint is satisfied moves into a ground instance of the rhs such that the rh constraint is satisfied, along all terminating paths.

Coinductive Definition of Reachability (Preliminaries)

State predicate: $P \subseteq M$.
Reachability predicate: $P \Rightarrow Q$.

Derivative of a state predicate P:
$\partial(P) = \{\gamma' \mid \gamma \rightsquigarrow \gamma' \text{ for some } \gamma \in P\}$.

$(M, \rightsquigarrow) \models^{\forall} P \Rightarrow Q$ is coinductively defined by:

$[\![\text{Subsumption}]\!]\ \dfrac{}{P \Rightarrow Q}\ P \subseteq Q \qquad [\![\text{Step}]\!]\ \dfrac{\partial(P \setminus Q) \Rightarrow Q}{P \Rightarrow Q}\ P \setminus Q \text{ runnable.}$

Coinductive Definition of Reachability

Semantics: $\mathcal{R} \models^{\forall} \varphi \Rightarrow \varphi'$ if $(M^{\Sigma}, \rightsquigarrow_{\mathcal{R}}) \models^{\forall} [\![\sigma(\varphi)]\!] \Rightarrow [\![\sigma(\varphi')]\!]$ for each $\sigma : \mathit{var}(\varphi) \cap \mathit{var}(\varphi') \to M^{\Sigma}$.

Justification: we have $(M, \rightsquigarrow) \models^{\forall} P \Rightarrow Q$ iff any execution path τ *starting from* P ($\mathit{hd}(\tau) \in P$) satisfies $P \Rightarrow Q$, where $\tau \models^{\forall} P \Rightarrow Q$ is coinductively defined by:

$$\frac{}{\langle \tau, P \Rightarrow Q \rangle}\ \mathit{hd}(\tau) \in P \cap Q \qquad \frac{\langle \tau, \partial(P) \Rightarrow Q \rangle}{\langle \gamma_0 \circ \tau, P \Rightarrow Q \rangle}\ \gamma_0 \in P, \gamma_0 \rightsquigarrow \mathit{hd}(\tau).$$

Outline

Derivatives of Constrained Terms

Definition (Derivatives of Constrained Terms)

The *set of derivatives* of a constrained term $\varphi \triangleq \langle t \,|\, \phi \rangle$ w.r.t. a rule $l \twoheadrightarrow r$ `if` ϕ_{lr} is

$$\Delta_{l,r,\phi_{lr}}(\varphi) \triangleq \{\langle c[r] \,\big|\, \phi' \rangle \mid \phi' \triangleq \phi \wedge t = c[l] \wedge \phi_{lr}, \\ c[\cdot] \text{ an appropriate context} \\ \phi' \text{ is satisfiable}\}.$$

Example:

$$\Delta_{\mathcal{R}}(\langle init(n) \,|\, \exists u.1 < u < n \wedge n \bmod u = 0\rangle) = \\ \{\langle loop(n,2) \,|\, \exists u.1 < u < n \wedge n \bmod u = 0\rangle\}.$$

Proof System (DSTEP($\mathcal{R}$))

[axiom] $$\frac{}{\langle t_l \mid \bot \rangle \Rightarrow \langle t_r \mid \phi_r \rangle}$$

[subs] $$\frac{\langle t_l \mid \phi_l \wedge \neg(\exists \tilde{x}. t_l = t_r \wedge \phi_r) \rangle \Rightarrow \langle t_r \mid \phi_r \rangle}{\langle t_l \mid \phi_l \rangle \Rightarrow \langle t_r \mid \phi_r \rangle} \quad \begin{array}{l} \widetilde{x} \triangleq \mathit{var}(t_r, \phi_r) \setminus \mathit{var}(t_l, \phi_l) \\ \exists \tilde{x}. t_l = t_r \wedge \phi_r \text{ satisfiable} \end{array}$$

[der$^\forall$] $$\frac{\left\langle t^j \,\middle|\, \phi^j \right\rangle \Rightarrow \langle t_r \mid \phi_r \rangle, j \in \{1, \ldots, n\}}{\langle t_l \mid \phi_l \rangle \Rightarrow \langle t_r \mid \phi_r \rangle} \quad \begin{array}{l} \langle t_l \mid \phi_l \rangle \text{ is } \mathcal{R}-\text{derivable and} \\ \phi_l \rightarrow \bigvee_{j \in \{1,\ldots,n\}} \exists \widetilde{y}^j . \phi^j \text{ is valid} \end{array}$$

where $\Delta_{\mathcal{R}}(\langle t_l \mid \phi_l \rangle) = \{\left\langle t^1 \,\middle|\, \phi^1 \right\rangle, \ldots, \langle t^n \mid \phi^n \rangle\}$ and
$\widetilde{y}^j = \mathit{var}(t^j, \phi^j) \setminus \mathit{var}(t_l, \phi_l)$

Example

The proof tree for the reachability formula $\langle \mathit{init}(n) \,|\, \psi \rangle \Rightarrow \varphi_r$, where $\psi \triangleq \exists u.1 < u < n \wedge n \bmod u = 0$ and $\varphi_r \triangleq \langle \mathit{comp} \,|\, \top \rangle$:

$$\dfrac{\dfrac{\dfrac{}{\langle \mathit{comp} \,|\, \bot \rangle \Rightarrow \varphi_r}\,[\text{axiom}]}{\langle \mathit{comp} \,|\, \psi \wedge \phi_a \rangle \Rightarrow \varphi_r}\,[\text{subs}] \qquad \dfrac{\dfrac{\dfrac{}{\langle \mathit{comp} \,|\, \bot \rangle \Rightarrow \varphi_r}\,[\text{axiom}]}{\langle \mathit{comp} \,|\, \psi \wedge \phi_2 \wedge \phi_b \rangle \Rightarrow \varphi_r}\,[\text{subs}] \quad \vdots}{\langle \mathit{loop}(n,3) \,|\, \psi \wedge \phi_2 \rangle \Rightarrow \varphi_r}\,[\text{der}^{\forall}]}{\langle \mathit{loop}(n,2) \,|\, \psi \rangle \Rightarrow \varphi_r}\,[\text{der}^{\forall}]$$

$$\dfrac{\langle \mathit{loop}(n,2) \,|\, \psi \rangle \Rightarrow \varphi_r}{\langle \mathit{init}(n) \,|\, \psi \rangle \Rightarrow \varphi_r}\,[\text{der}^{\forall}]$$

Extending the Proof System to the Unbounded Case

Let G be a finite set reachability formulas (goals that we intend to prove).
Then the set of rules $\mathrm{DCC}(\mathcal{R}, G)$ consists of $\mathrm{DSTEP}(\mathcal{R})$, together with

$$[\text{circ}]\ \frac{\begin{array}{l}\langle t_r^c \mid \phi_I \wedge \phi \wedge \phi_r^c \rangle \Rightarrow \varphi_r, \\ \langle t_I \mid \phi_I \wedge \neg\phi \rangle \Rightarrow \varphi_r\end{array}}{\langle t_I \mid \phi_I \rangle \Rightarrow \varphi_r} \qquad \begin{array}{l}\phi \text{ is } \exists var(t_l^c, \phi_l^c).t_I = t_l^c \wedge \phi_l^c, \\ \langle t_l^c \mid \phi_l^c \rangle \Rightarrow \langle t_r^c \mid \phi_r^c \rangle \in G.\end{array}$$

Theorem (Circularity Principle)

Let $\mathcal{R}$ be a constrained rule system and G a set of goals. If $(\mathcal{R}, G) \vdash^{\forall} G$ then $\mathcal{R} \models^{\forall} G$.

Example

In order to prove $\langle init(n) \,|\, \psi \rangle \Rightarrow \langle comp \,|\, \top \rangle$, we choose the following set of circularities

$$G = \left\{ \begin{array}{l} \langle init(n) \,|\, \psi \rangle \Rightarrow \langle comp \,|\, \top \rangle, \\ \langle loop(n,i) \,|\, 2 \leq i \wedge \exists u.i \leq u < n \wedge n \bmod u = 0 \rangle \Rightarrow \langle comp \,|\, \top \rangle \end{array} \right\}.$$

The finite proof tree for the first circularity:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\langle comp \,|\, \bot \rangle \Rightarrow \varphi_r}\,[\text{axiom}]}{\langle comp \,|\, \psi \wedge \phi \wedge \top \rangle \Rightarrow \varphi_r}\,[\text{subs}] \qquad \dfrac{}{\langle loop(n,2) \,|\, \psi \wedge \neg\phi \rangle \Rightarrow \varphi_r}\,[\text{axiom}]}{\langle loop(n,2) \,|\, \psi \rangle \Rightarrow \varphi_r}\,[\text{circ}]}{\langle init(n) \,|\, \psi \rangle \Rightarrow \varphi_r}\,[\text{der}^{\forall}]$$

Example Proof Tree for the Second Circularity

$$\dfrac{\dfrac{}{\langle c \,|\, \bot \rangle \Rightarrow \varphi_r} \quad \dfrac{\dfrac{}{\langle c \,|\, \bot \rangle \Rightarrow \varphi_r}}{\langle c \,|\, \psi_i \wedge \psi_b \wedge \psi_c \rangle \Rightarrow \varphi_r} \quad \dfrac{}{\langle l(n, i+1) \,|\, \psi_i \wedge \psi_b \wedge \neg\psi_c \rangle \Rightarrow \varphi_r}}{\dfrac{\langle c \,|\, \psi_i \wedge \psi_a \rangle \Rightarrow \varphi_r \qquad \langle l(n, i+1) \,|\, \psi_i \wedge \psi_b \rangle \Rightarrow \varphi_r,}{\langle l(n, i) \,|\, \psi_i \rangle \Rightarrow \langle c \,|\, \top \rangle}\ [\mathrm{der}^{\forall}]}$$

Outline

Applications and Implementation (RMT tool)

`http://github.com/ciobaca/rmt`

LCTRS	Reachability Property
Computation of $1 + \ldots + n$	Result is $n * (n + 1)/2$
Comp. of $gcd(u, v)$ by *rptd. subtractions*	Result matches builtin *gcd* function
Comp. of $gcd(u, v)$ by *rptd. divisions*	Result matches builtin *gcd* function
Mult. of two naturals by *rptd. additions*	Result matches builtin $\times$ function
Comp. of $1^2 + \ldots + n^2$	Result is $n(n + 1)(2n + 1)/6$
Comp. of $1^2 + \ldots + n^2$ *w/out multiplications*	Result is $n(n + 1)(2n + 1)/6$
Semantics of an IMPerative language	Program computing $1 + \ldots + n$ is correct
Semantics of a FUNctional language	Program computing $1 + \ldots + n$ is correct
Semantics of a FUNctional language	Program computing $1^2 + \ldots + n^2$ is correct

Outline

Conclusion

1. LCTRSs are a rich model for describing computations, including programming languages.
2. Our approach shows that reachability in LCTRSs can be proved elegantly, using an SMT solver as an oracle (applications to program correctness).
3. Implementation: `http://github.com/ciobaca/rmt`.

4. Extended version: `https://arxiv.org/pdf/1804.08308.pdf`.
5. Future work: other applications, angelic reachability, strong reachability, other properties.