A Semantic Approach to the Analysis of Rewriting-Based Systems

Salvador Lucas

DSIC, Universitat Politècnica de València, Spain

27th International Symposium on Logic-Based Program Synthesis and Transformation, LOPSTR 2017
Is the following \textit{true}?

\[(\forall x) \quad x + 0 \geq x \] (1)

Yes!... provided that the \textit{standard} (arithmetic) interpretation $\mathcal{A}$ is assumed for all symbols: $\mathcal{A} \models (1)$. 
Is the following true?

\[(\forall x) \quad x + 0 \geq x \quad (1)\]

Yes!... provided that the standard (arithmetic) interpretation \(\mathcal{A}\) is assumed for all symbols: \(\mathcal{A} \models (1)\).

What about this?

\[(\forall x_1) \quad A_1^2(f_1^2(x_1, a_1), x_1) \quad (2)\]

(1) and (2) are ‘syntactically equivalent’ under renaming of symbols.

Viewed as first-order logic (FOL) formulas, non-logic symbols occurring in (1) (e.g., ‘0’, ‘+’, and ‘\(\geq\)’) have no special meaning!

Many interpretations of \(a_1\), \(f_1^2\) and \(A_1^2\) in (2) do not satisfy (2), i.e.,

\[\not\models (2) \quad \text{and even} \quad \not\models (1)!\]
How to use FOL in the analysis of computational properties of rewriting-based systems?

For instance, *confluence* can be expressed as follows:

\[
(\forall x, y, z) \ (x \rightarrow^* y \land x \rightarrow^* z \Rightarrow (\exists u) (y \rightarrow^* u \land z \rightarrow^* u))
\]  
(3)
Given a Term Rewriting System $\mathcal{R}$, how do we say “$\mathcal{R}$ is confluent” using FOL?

1. $\overline{\mathcal{R}} \vdash (3)$, i.e., (3) can be proved from some theory $\overline{\mathcal{R}}$ associated to $\mathcal{R}$?
2. $\overline{\mathcal{R}} \models (3)$, i.e., every model of $\overline{\mathcal{R}}$ satisfies (3)?
3. $\mathcal{A}_\mathcal{R} \models (3)$, i.e., (3) is satisfied by some special interpretation $\mathcal{A}_\mathcal{R}$ associated to $\mathcal{R}$?

Dauchet and Tison’s first-order theory of rewriting uses with the standard interpretation $\mathcal{H}_\mathcal{R}$ where predicate symbols $\rightarrow$ and $\rightarrow^*$ are interpreted as the one-step and many-step rewrite relations on ground terms $\rightarrow_\mathcal{R}$ and $\rightarrow^*_\mathcal{R}$, respectively.

Problems

- In general, $\mathcal{H}_\mathcal{R}$ is not computable, and $\mathcal{H}_\mathcal{R} \models (3)$ is undecidable!
- Can we use other (computable!) interpretations? How?
Summary

1. Preservation of first-order formulas
2. Application to Horn theories
3. Rewriting-based systems as Horn theories
4. Examples of use
5. Related work
6. Conclusions and future work
Our approach is based on two well-known facts:

[Hodges97, Theorem 1.5.2]

Every set $S$ of ground atoms has an initial (Herbrand) model $\mathcal{I}_S$, i.e.,

- $\mathcal{I}_S \models S$ and
- for all models $\mathcal{A}$ of $S$, there is a homomorphism $h : \mathcal{I}_S \to \mathcal{A}$.

A positive boolean combination of atoms is a formula

$$
\bigvee_{i=1}^{m} \bigwedge_{j=1}^{n_i} A_{ij}
$$

(4)

where the $A_{ij}$ are atoms. Satisfiability of the existential closure of (4) is preserved under homomorphism

[Hodges97, Theorem 2.4.3(a)]

Given interpretations $\mathcal{A}$ and $\mathcal{A}'$ with an homomorphism $h : \mathcal{A} \to \mathcal{A}'$,

$$
\mathcal{A} \models (\exists x_1) \cdots (\exists x_k) \bigvee_{i=1}^{m} \bigwedge_{j=1}^{n_i} A_{ij} \implies \mathcal{A}' \models (\exists x_1) \cdots (\exists x_k) \bigvee_{i=1}^{m} \bigwedge_{j=1}^{n_i} A_{ij}
$$

(5)
According to these results, we have the following:

**Corollary**

Let $S$ be a set of ground atoms, and $A_{ij}$ be atoms with variables $x_1, \ldots, x_k$. Then,

$$
\mathcal{I}_S \models (\exists x_1) \cdots (\exists x_k) \bigvee_{i=1}^{m} \bigwedge_{j=1}^{n_i} A_{ij} \implies S \models (\exists x_1) \cdots (\exists x_k) \bigvee_{i=1}^{m} \bigwedge_{j=1}^{n_i} A_{ij} \quad (6)
$$
If the set of atoms $S$ is generated by a set $S_0$ of Horn sentences, then the interpretation of each predicate symbol $P$ by $I$ consists of the set of ground atoms $P(t_1, \ldots, t_n)$ such that $S_0 \vdash P(t_1, \ldots, t_n)$.

**Corollary (Semantic criterion)**

Let $S$ be a Horn theory, $\varphi$ be the existential closure of a positive boolean combination of atoms, and $A$ be a model of $S$. If $A \models \neg \varphi$, then $I_S \models \neg \varphi$.

**Many-sorted theories**

The previous corollaries easily generalize to many-sorted signatures: as usual, we just treat sorted variables $x_i : s_i$ by using atoms $S_i(x_i)$ which are added as a new conjunction $\bigwedge_{i=1}^{k} S_i(x_i)$ to the matrix formula (4).
In the following, we focus on oriented CTRSs $\mathcal{R}$, with rules

$$\ell \rightarrow r \iff s_1 \rightarrow t_1, \ldots, s_n \rightarrow t_n$$

whose operational semantics is given by the following inference system:

- **(Rf)**: $x \rightarrow^* x$
- **(C)**: $x_i \rightarrow y_i$ for all $f \in \mathcal{F}$ and $1 \leq i \leq k = \text{arity}(f)$
- **(T)**: $x \rightarrow z \quad z \rightarrow^* y \quad x \rightarrow^* y$
- **(Rp)**: $s_1 \rightarrow^* t_1 \quad \ldots \quad s_n \rightarrow^* t_n$
The Horn theory $\overline{R}$ for a CTRS $R$ is obtained by specializing $(C)$ and $(Rp)$. Inference rules $\frac{B_1 \cdots B_n}{A}$ become universally quantified implications $\forall \vec{x} \; B_1 \land \cdots \land B_n \Rightarrow A$.

Example

For the CTRS $R$ (from [Giesl & Arts, AAECC’01])

$$
\begin{align*}
    a & \rightarrow b \\
    f(a) & \rightarrow b \\
    g(x) & \rightarrow g(a) \iff f(x) \rightarrow x
\end{align*}
$$

its associated theory $\overline{R}$ is

$$
\begin{align*}
    (\forall x) \: x & \rightarrow^* x \\
    (\forall x, y, z) \: x & \rightarrow y \land y \rightarrow^* z \Rightarrow x \rightarrow^* z \\
    (\forall x, y) \: x & \rightarrow y \Rightarrow f(x) \rightarrow f(y) \\
    (\forall x, y) \: x & \rightarrow y \Rightarrow g(x) \rightarrow g(y) \\
    a & \rightarrow b \\
    f(a) & \rightarrow b \\
    (\forall x) \: f(x) & \rightarrow^* x \Rightarrow g(x) \rightarrow g(a)
\end{align*}
$$
Infeasibility of conditional rules

For infeasibility of $\ell \rightarrow r \iff s_1 \rightarrow t_1, \ldots, s_n \rightarrow t_n$ we use $\varphi_{Feas}$ given by:

$$(\exists \bar{x})s_1 \rightarrow^* t_1 \land \cdots \land s_n \rightarrow^* t_n$$

The following structure $\mathcal{A}$ over $\mathbb{N} - \{0\}$:

- $a^A = 1$
- $b^A = 2$
- $f^A(x) = x + 1$
- $g^A(x) = 1$
- $x \rightarrow^A y \iff y \geq x$
- $x (\rightarrow^*)^A y \iff y \geq x$

is a model of $\overline{\mathcal{R}} \cup \{\neg (\exists x) f(x) \rightarrow^* x\}$ for our running CTRS $\mathcal{R}$.

Automation

This model has been automatically generated by using the tool AGES: [http://zenon.dsic.upv.es/ages/](http://zenon.dsic.upv.es/ages/)

Thus, rule

$$g(x) \rightarrow g(a) \iff f(x) \rightarrow x$$

is proved $\mathcal{R}$-infeasible.
The following CTRS $\mathcal{R}$ (Example 23 in [Sternagel & Sternagel, FSCD’16])

\begin{align*}
g(x) & \to f(x, x) \quad (7) \\
g(x) & \to g(x) \leftrightarrow g(x) \to f(a, b) \quad (8)
\end{align*}

has a conditional critical pair $f(x, x) \downarrow g(x) \leftrightarrow g(x) \to f(a, b)$. The following structure $\mathcal{A}$ over the finite domain $\{0, 1\}$:

\begin{align*}
a^\mathcal{A} & = 1 \\
b^\mathcal{A} & = 0 \\
f^\mathcal{A}(x, y) & = \begin{cases} 
  x - y + 1 & \text{if } x \geq y \\
  y - x + 1 & \text{otherwise}
\end{cases} \\
g^\mathcal{A}(x) & = 1 \\
x \to^\mathcal{A} y & \equiv x = y \\
x (\to^* \mathcal{A}) y & \equiv x \geq y
\end{align*}

is a model $\overline{\mathcal{R}} \cup \{\neg (\exists x) \; g(x) \to^* f(a, b)\}$. The critical pair is infeasible.

In the FSCD’16 paper, this is proved by using unification tests together with a transformation. It is discussed that the alternative tree automata techniques investigated in the paper do not work for this example.
A term \( t \) loops if there is a rewrite sequence \( t = t_1 \rightarrow_R \cdots \rightarrow_R t_n \) for some \( n > 1 \) such that \( t \) is a (non-necessarily strict) subterm of \( t_n \), written \( t_n \triangleright t \). A CTRS is non-looping if no term loops.

We can check (non)loopingness of terms \( t \) or CTRSs \( R \) by using

\[
\varphi_{\text{Loop}} \iff (\exists x, y) \ t \rightarrow x \land x \rightarrow^* y \land y \triangleright t
\]

\[
\varphi_{\text{Loopt}} \iff (\exists x, y, z) \ x \rightarrow y \land y \rightarrow^* z \land z \triangleright x
\]

for \( \overline{R} \cup H_{\triangleright} \) where \( H_{\triangleright} \) describe the subterm relation \( \triangleright \):

\[
(\forall x) \ x \triangleright x \tag{9}
\]

\[
(\forall x, y, z) \ x \triangleright y \land y \triangleright z \Rightarrow x \triangleright z \tag{10}
\]

\[
(\forall x_1, \ldots, x_k) \ f(x_1, \ldots, x_k) \triangleright x_i \tag{11}
\]

for each \( k \)-ary function symbol \( f \in \mathcal{F} \) and argument \( i, 1 \leq i \leq k \).
Example (A non-looping term)

For $\mathcal{R} = \{a \rightarrow c(b), b \rightarrow c(b)\}$, $\overline{\mathcal{R}} \cup H_{\succ}$ is:

$$(\forall x)\ x \rightarrow^* x$$ (12)

$$(\forall x)\ x \succ x$$ (17)

$$(\forall x, y, z)\ (x \rightarrow y \land y \rightarrow^* z \Rightarrow x \rightarrow^* z)$$ (13)

$$(\forall x, y, z)\ x \succ y \land y \succ z \Rightarrow x \succ z$$ (18)

$$(\forall x, y)\ (x \rightarrow y \Rightarrow c(x) \rightarrow c(y))$$ (14)

$$(\forall x)\ c(x) \succ x$$ (19)

$a \rightarrow c(b)$ (15)

$b \rightarrow c(b)$ (16)

The following structure over $\mathbb{N} \cup \{-1\}$:

$$a^A = -1$$

$$x \rightarrow^A y \iff x \leq 1 \land y \geq 1$$

$$b^A = 1$$

$$x (\rightarrow^*)^A y \iff x \leq y$$

$$c^A(x) = x$$

$$x \succ^A y \iff x \leq y$$

satisfies $\overline{\mathcal{R}} \cup H_{\succ} \cup \{\neg \varphi_{Loopt}\}$ where

$$\varphi_{Loopt} \iff (\exists x, y)\ a \rightarrow x \land x \rightarrow^* y \land y \succ a.$$ 

Therefore, $a$ is non-looping.
Example (A non-cycling TRS)

Although $b$ is a looping term (for $\mathcal{R} = \{a \rightarrow c(b), b \rightarrow c(b)\}$), we can prove it non-cycling (i.e., it does not rewrite into itself in at least one step).

Actually, we can prove $\mathcal{R}$ non-cycling (i.e., no term rewrites into itself in at least one step) with the following structure over $\mathbb{N} \cup \{-1\}$

$$
\begin{align*}
 a^A &= -1 \\
 b^A &= -1 \\
 c^A(x) &= 2x + 2 \\
 x \rightarrow^A y &\iff x < y \\
 x (\rightarrow^*)^A y &\iff x \leq y
\end{align*}
$$

which is a model of $\overline{\mathcal{R}} \cup \{\neg \varphi_{Cycl}\}$ where

$$
\varphi_{Cycl} \iff (\exists x, y) \ x \rightarrow y \land y \rightarrow^{*} x.
$$
We have presented a semantic approach to disprove properties of Horn theories which can be expressed as the satisfiability of the existential closure of a positive boolean combination of atoms.

We can apply this approach to rewriting-based systems with

- many-sorted signatures,
- alternative satisfiability notions for the conditions (e.g., joinability), or
- more general components there (e.g., memberships).

We could handle many examples coming from papers developing different specific techniques to deal with these problems.
We have presented a semantic approach to **disprove** properties of Horn theories which can be expressed as the satisfability of the existential closure of a positive boolean combination of atoms.

We can apply this approach to rewriting-based systems with

- many-sorted signatures,
- alternative **satisfiability notions** for the conditions (e.g., joinability), or
- more general **components there** (e.g., memberships).

We could handle many examples coming from papers developing different specific techniques to deal with these problems.

**Future work**

- Use other **preservation** results for FOL.
- Use these techniques in **tools** for proving computational properties of rewriting-based systems (e.g., confluence, termination, etc.)
Thanks!