Disproving Confluence of Term Rewriting Systems by Interpretation and Ordering

FroCoS 2013

Takahito Aoto (Tohoku University)

Outline

1. Backgrounds: TRS and Confluence
2. Backgrounds: Proving (Non)-Confluence
3. Proving Non-Joinability by Interpretation
4. Proving Non-Joinability by Ordering
5. Implementation and Experiments

Term Rewriting Systems (TRSs)

Example: TRS modelling addition of natural numbers

$$\mathcal{R} = \left\{ \begin{array}{lcl} +(0, y) & \to & y \\ +(\mathsf{s}(x), y) & \to & \mathsf{s}(+(x, y)) \end{array} \right\}$$

Natural numbers $0, 1, 2, \ldots$ are represented by $0, \mathsf{s}(0), \mathsf{s}(\mathsf{s}(0)), \ldots$.

- **Computational model:**
  Equational logic + Functional programs
- **Automated theorem proving: KB-completion, etc.**
- **Automated verification: Termination, Confluence, etc.**

$$\mathcal{R} = \left\{ \begin{array}{lcl} +(0, y) & \to & y \\ +(\mathsf{s}(x), y) & \to & \mathsf{s}(+(x, y)) \end{array} \right\}$$

Computation by reduction ("2 + 2 = 4")

$$\begin{array}{lll} \underline{+(\mathsf{s}(\mathsf{s}(0)), \mathsf{s}(\mathsf{s}(0)))} & \to_{\mathcal{R}} & \mathsf{s}(\underline{+(\mathsf{s}(0), \mathsf{s}(\mathsf{s}(0)))}) \\ & \to_{\mathcal{R}} & \mathsf{s}(\mathsf{s}(\underline{+(0, \mathsf{s}(\mathsf{s}(0)))})) \\ & \to_{\mathcal{R}} & \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))) \quad \textbf{normal form} \end{array}$$

LHS patterns of rewrite rules are replaced by the corresponding RHS patterns.

$$\mathcal{R} = \left\{ \begin{array}{lcl} +(0, y) & \to & y \\ +(\mathsf{s}(x), y) & \to & \mathsf{s}(+(x, y)) \end{array} \right\}$$

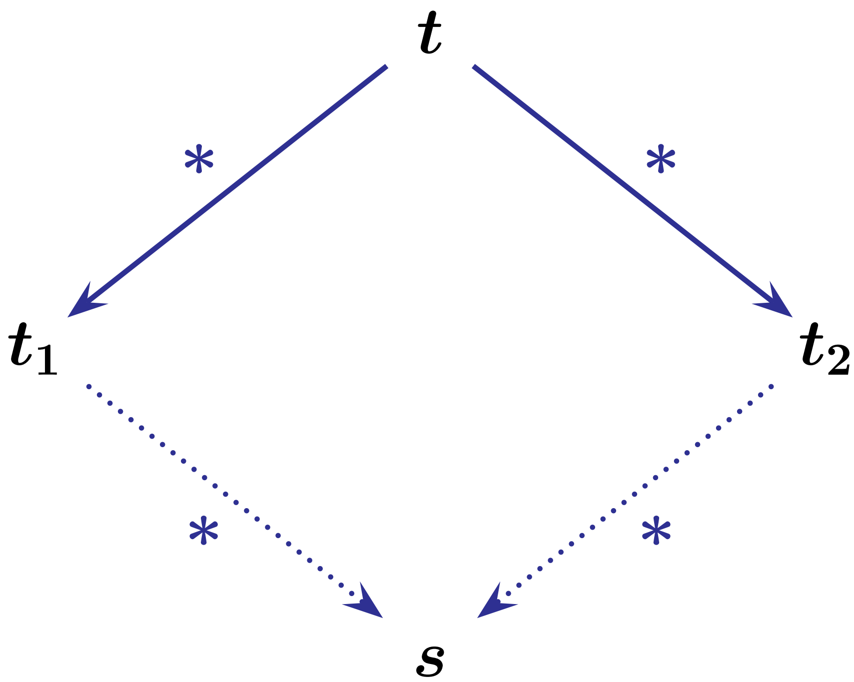
Computation by reduction ("2 + 2 = 4")

$$\begin{array}{rll} \underline{+(\mathsf{s}(\mathsf{s}(0)), \mathsf{s}(\mathsf{s}(0)))} & \to_{\mathcal{R}} & \mathsf{s}(\underline{+(\mathsf{s}(0), \mathsf{s}(\mathsf{s}(0)))}) \\ & \to_{\mathcal{R}} & \mathsf{s}(\mathsf{s}(\underline{+(0, \mathsf{s}(\mathsf{s}(0)))})) \\ & \to_{\mathcal{R}} & \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))) \quad \text{normal form} \end{array}$$

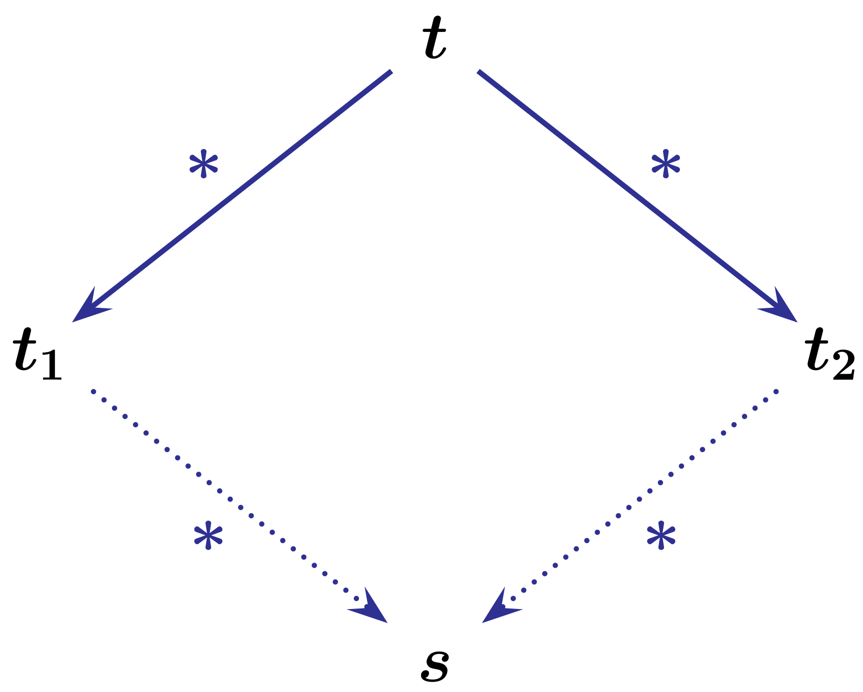
LHS patterns of rewrite rules are replaced by the corresponding RHS patterns.

$$t_0 \xrightarrow{*}_{\mathcal{R}} t_n \overset{\text{def}}{\Leftrightarrow} t_0 \to_{\mathcal{R}} t_1 \to_{\mathcal{R}} \cdots \to_{\mathcal{R}} t_n$$

Confluence (Church-Rosser)

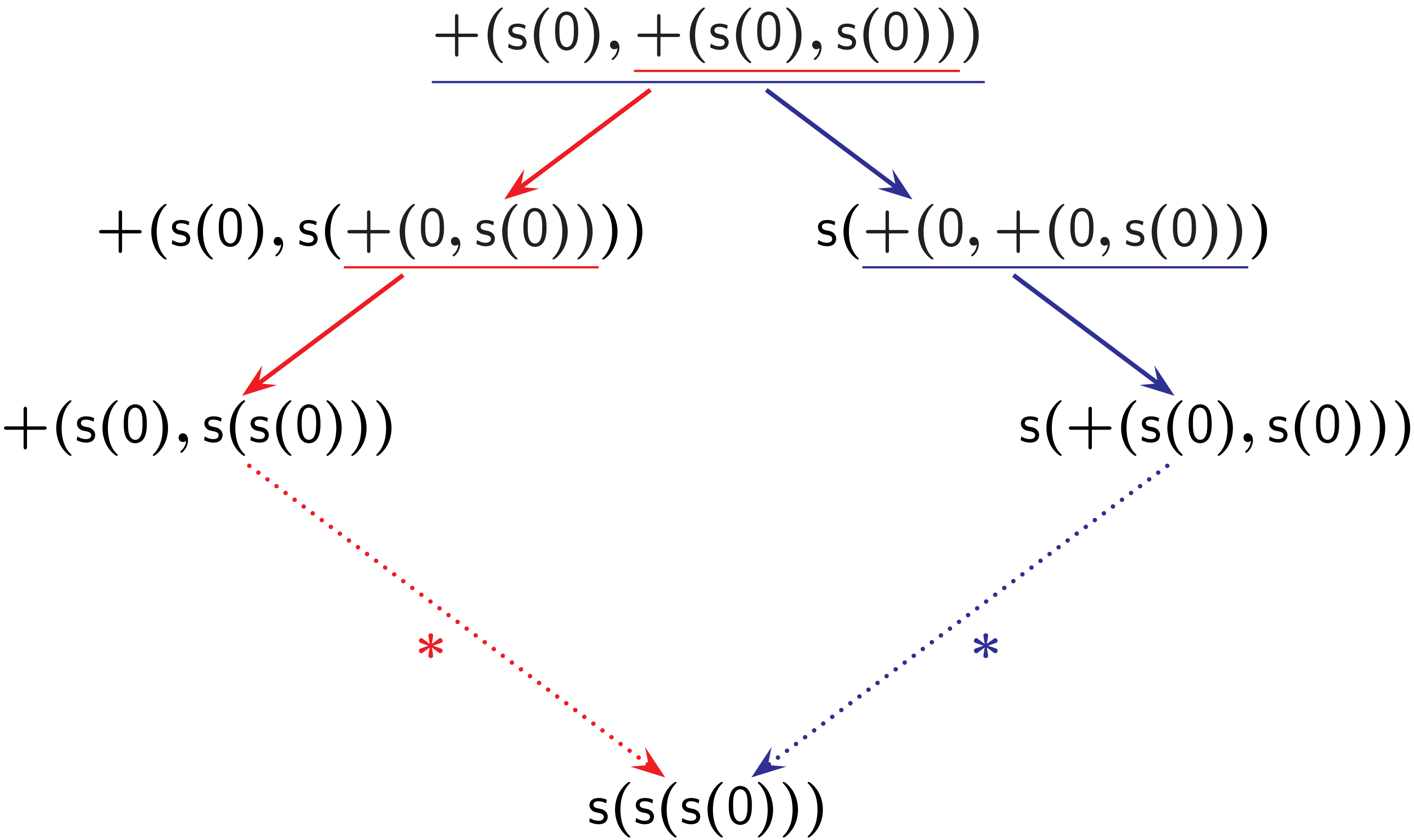


Confluence (Church-Rosser)



A TRS $\mathcal{R}$ is confluent if $\overset{*}{\leftarrow}_{\mathcal{R}} \circ \overset{*}{\rightarrow}_{\mathcal{R}} \subseteq \overset{*}{\rightarrow}_{\mathcal{R}} \circ \overset{*}{\leftarrow}_{\mathcal{R}}$,
i.e. any two terms obtained from one term by reduction are joinable by reduction.

Confluence (Church-Rosser)



$$\mathcal{R} = \left\{ \begin{array}{lcl} +(0, y) & \to & y \\ +(\mathsf{s}(x), y) & \to & \mathsf{s}(+(x, y)) \end{array} \right\}$$

Confluent

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{f}(x) & \to & \mathsf{g}(x) \\ \mathsf{g}(x) & \to & \mathsf{f}(x) \\ \mathsf{f}(x) & \to & \mathsf{a} \\ \mathsf{g}(x) & \to & \mathsf{b} \end{array} \right\}$$

Not Confluent

(Non-)Confluence Criteria

Long history of development...

Decidable classes: Terminating [Knuth&Bendix, 1970], Ground [Oyamaguchi, 1987; Dauchet et al., 1990], Right-ground [Kaiser, 2005; Tiwari et al., 2005], Right-linear shallow [Tiwari, 2002; Godoy et al, 2003; Godoy&Tiwari, 2005]. **Critical Pair Conditions for Left-linear TRSs**: Orthogonal [Rosen, 1973], Left-linear development closed [Huet, 1980; Toyama, 1988; van Oostrom, 1997], Linear strongly closed [Huet, 1980], Parallel critical pairs [Toyama, 1981], Simultaneous critical pairs [Okui, 1998], Upside-parallel-closed or Outside-closed [Oyamaguchi&Ohta, 2004]. **Modularity**: Persistency [Toyama,1987; Aoto&Toyama,1997], Commutativity [Toyama,1988], Layer-preservation [Ohlebusch,1994]

Conditions for Non-E-Overlapping TRSs: Simple-right-linear [Ohta&Oyamaguchi&Toyama, 1995], Strongly depth-preserving [Gomi&Oyamaguchi&Ohta, 1996], Strongly weight-preserving/depth-preserving root-E-closed [Gomi&Oyamaguchi&Ohta, 1998]. **Decreasing Diagram [van Oostrom, 1997] Approach**: rule-labelling [van Oostrom, 1997; Aoto, 2010; Hirokawa&Middeldorp; Zankle&Middeldorp, 2011]. **Others**: Weakly-non-overlapping non-collapsing shallow [Sakai&Ogawa, 2010], Reduction-preserving completion [Aoto&Toyama, 2012], Condition for relatively terminating TRSs [Klein&Hirokawa, 2012], Quasi-left-linear and parallel-closed [Suzuki&Aoto&Toyama, 2013].

Tools for proving/disproving confluence of TRSs:
ACP, CSI, Saigawa, . . .

Outline

1. **Backgrounds: TRS and Confluence**
2. **Backgrounds: Disproving Confluence**
3. **Proving Non-Joinability by Interpretation**
4. **Proving Non-Joinability by Ordering**
5. **Implementation and Experiments**

Disproving Confluence of TRSs (1)

- terminating TRSs (Knuth&Bendix, 1970)

A terminating TRS is confluent iff all critical pairs are joinable.

- relatively terminating TRSs (Klein&Hirokawa, 2012)

Suppose $\mathcal{S}$ is confluent, $\mathcal{R}$ is terminating relative to $\mathcal{S}$, and $\mathcal{R}$ and $\mathcal{S}$ are strongly non-overlapping. Then $\mathcal{R} \cup \mathcal{S}$ is confluent iff all $\mathcal{S}$-critical pairs of $\mathcal{R}$ are $(\mathcal{R} \cup \mathcal{S})$-joinable.

$\mathcal{S}$-critical pairs include non-minimal instances. In general, $\mathcal{S}$-critical pairs are not effectively computed.

Disproving Confluence of TRSs (2)

Decidable Classes

- ground TRSs: polynomial [Comon et al., 2001] [Tiwari, 2002]; cubic [Falgenhauer, 2012]

  . . . .

- linear shallow TRSs: polynomial [Godoy et al., 2003]
- right-ground TRSs: exponential [Tiwari et al., 2005]
- right-linear shallow TRSs: [Godoy&Tiwari, 2005]

These decidable classes are rather restrictive. Except for some basic classes, decision procedures are very complex. So far, only implemented procedure seems to be the one for ground TRSs. Needs more investigation for using in confluence tools.

Disproving Confluence of TRSs (3)

Find terms t_1, t_2 such that

(1) $s \xrightarrow{*} t_1$ and $s \xrightarrow{*} t_2$ for some s, and
(finding 'candidates' for non-confluence witness)

(2) $t_1 \xrightarrow{*} u$ and $t_2 \xrightarrow{*} u$ for no u,

i.e. $\{u \mid t_1 \xrightarrow{*} u\} \cap \{v \mid t_2 \xrightarrow{*} v\} = \emptyset$.
(proving non-joinability of 'candidates')

Disproving Confluence of TRSs (3)

Find terms t_1, t_2 such that

(1) $s \xrightarrow{*} t_1$ and $s \xrightarrow{*} t_2$ for some s, and
(finding 'candidates' for non-confluence witness)

(2) $t_1 \xrightarrow{*} u$ and $t_2 \xrightarrow{*} u$ for no u,

i.e. $\{u \mid t_1 \xrightarrow{*} u\} \cap \{v \mid t_2 \xrightarrow{*} v\} = \emptyset$.
(proving non-joinability of 'candidates')

We let the problem (1) untouched, and consider the problem (2).

Disproving Confluence of TRSs (3)

Find terms t_1, t_2 such that

(1) $s \xrightarrow{*} t_1$ and $s \xrightarrow{*} t_2$ for some s, and
(finding 'candidates' for non-confluence witness)

(2) $t_1 \xrightarrow{*} u$ and $t_2 \xrightarrow{*} u$ for no u,

i.e. $\{u \mid t_1 \xrightarrow{*} u\} \cap \{v \mid t_2 \xrightarrow{*} v\} = \emptyset$.
(proving non-joinability of 'candidates')

We let the problem (1) untouched, and consider the problem (2).

We abbreviate non-joinability of terms t_1 and t_2 (i.e. $\{u \mid t_1 \xrightarrow{*} u\} \cap \{v \mid t_2 \xrightarrow{*} v\} = \emptyset$) as $\mathrm{NJ}(t_1, t_2)$.

Proving Non-Joinability by Tree Automata

So far, the only serious approach for proving non-joinability is using tree automata approximation [Durand&Middeldorp, 1997] [Genet, 1998].

(1) Construct tree automata $\mathcal{A}_1, \mathcal{A}_2$ such that $\{u \mid t_i \xrightarrow{*} u\} \subseteq \mathcal{L}(\mathcal{A}_i)$ $(i = 1, 2)$ by tree automata approximation.

(2) Check $\mathcal{L}(\mathcal{A}_1) \cap \mathcal{L}(\mathcal{A}_2) = \emptyset$.

Proving Non-Joinability by Tree Automata

So far, the only serious approach for proving non-joinability is using tree automata approximation [Durand&Middeldorp, 1997] [Genet, 1998].

(1) Construct tree automata $\mathcal{A}_1, \mathcal{A}_2$ such that $\{u \mid t_i \xrightarrow{*} u\} \subseteq \mathcal{L}(\mathcal{A}_i)$ $(i = 1, 2)$ by tree automata approximation.

(2) Check $\mathcal{L}(\mathcal{A}_1) \cap \mathcal{L}(\mathcal{A}_2) = \emptyset$.

Sometimes it is difficult to construct a well-approximated tree automaton.

This work: another approach for proving non-joinability.

Outline

1. Backgrounds: TRS and Confluence
2. Backgrounds: Disproving Confluence
3. Proving Non-Joinability by Interpretation
4. Proving Non-Joinability by Ordering
5. Implementation and Experiments

Interpretation

We first recall some standard definitions.

An $\mathcal{F}$-algebra $\mathcal{A} = \langle A, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$ is a set A equipped with functions $f^{\mathcal{A}} : A^n \to A$ for each n-ary function symbol $f \in \mathcal{F}$.

A valuation σ on a $\mathcal{F}$-algebra $\mathcal{A}$ is a mapping $\sigma : \mathcal{V} \to A$.

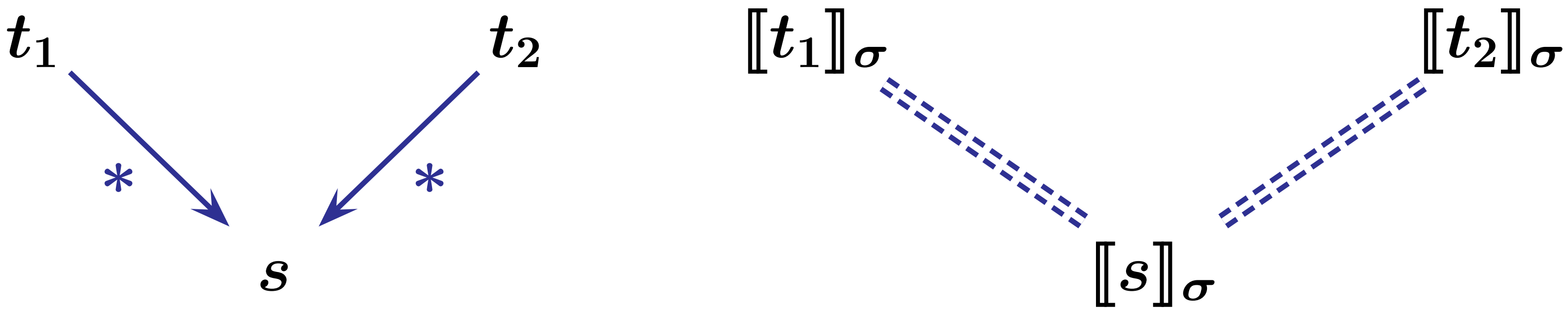
The interpretation $[\![t]\!]_\sigma \in A$ of a term $t \in \mathrm{T}(\mathcal{F}, \mathcal{V})$ is given by

$$[\![x]\!]_\sigma = \sigma(x)$$

$$[\![f(t_1, \ldots, t_n)]\!]_\sigma = f^{\mathcal{A}}([\![t_1]\!]_\sigma, \ldots, [\![t_n]\!]_\sigma)$$

Interpretation for Non-Joinability

If there exist an $\mathcal{F}$-algebra and a valuation σ such that
(i) $u \to_{\mathcal{R}} v$ implies $[\![u]\!]_\sigma = [\![v]\!]_\sigma$ and (ii) $[\![t_1]\!]_\sigma \neq [\![t_2]\!]_\sigma$,
then $\mathrm{NJ}(t_1, t_2)$.



Interpretation for Non-Joinability

If there exist an $\mathcal{F}$-algebra and a valuation σ such that

(i) $u \to_{\mathcal{R}} v$ implies $[\![u]\!]_\sigma = [\![v]\!]_\sigma$ and (ii) $[\![t_1]\!]_\sigma \neq [\![t_2]\!]_\sigma$,

then $\mathrm{NJ}(t_1, t_2)$.

But, since $t_0 \xrightarrow{*} t_1$ and $t_0 \xrightarrow{*} t_2$ for some t_0, there is no such an $\mathcal{F}$-algebra for our candidates t_1, t_2.

Interpretation for Non-Joinability

If there exist an $\mathcal{F}$-algebra and a valuation σ such that

(i) $u \to_{\mathcal{R}} v$ implies $[\![u]\!]_\sigma = [\![v]\!]_\sigma$ and (ii) $[\![t_1]\!]_\sigma \neq [\![t_2]\!]_\sigma$,

then $\mathrm{NJ}(t_1, t_2)$.

But, since $t_0 \xrightarrow{*} t_1$ and $t_0 \xrightarrow{*} t_2$ for some t_0, there is no such an $\mathcal{F}$-algebra for our candidates t_1, t_2.

Idea: replace (i) by the following (i′)

(i′) $u \to_{\{l \to r\}} v$ implies $[\![u]\!]_\sigma = [\![v]\!]_\sigma$ for any usable rule $l \to r \in \mathcal{R}$.

Here, *usable* means it can happen $t_1 \xrightarrow{*}_{\mathcal{R}} \circ \to_{\{l \to r\}} u$ or $t_2 \xrightarrow{*}_{\mathcal{R}} \circ \to_{\{l \to r\}} u$ for some u (given in the next slide).

Usable Rules for Reachability

Definition. The set of **usable rules** $\mathcal{U}(s) \subseteq \mathcal{R}$ is the smallest set satisfying:
(i) for any non-variable subterm $f(u_1, \ldots, u_n)$ of s and $l \to r \in \mathcal{R}$, if $f(\mathrm{TCAP}(u_1), \ldots, \mathrm{TCAP}(u_n))$ and l are unifiable then $l \to r \in \mathcal{U}(s)$; and
(ii) if $l' \to r' \in \mathcal{U}(s)$ and $l \to r \in \mathcal{U}(r')$, then $l \to r \in \mathcal{U}(s)$.

Lemma. If $s \xrightarrow{*}_{\mathcal{R}} \circ \to_{\{l \to r\}} t$ then $l \to r \in \mathcal{U}(s)$.

Here, we assume variable conditions of rewrite rules. It is straightforward to generalize usable rules to the case variable conditions do not hold.

Non-Joinability by Interpretation

Theorem 1. Let $\mathcal{A} = \langle A, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$ be an $\mathcal{F}$-algebra with $A = \biguplus_{i \in I} A_i$ (i.e. disjoint union of A_i's), and s, t terms. Suppose
(i) $[\![l]\!]_\sigma \in A_i$ implies $[\![r]\!]_\sigma \in A_i$ for any $l \to r \in \mathcal{U}(s) \cup \mathcal{U}(t)$,
(ii) if $a \in A_i$ implies $f^{\mathcal{A}}(\ldots, a, \ldots) \in A_j$, then for any $b \in A_i$, $f^{\mathcal{A}}(\ldots, b, \ldots) \in A_j$, and
(iii) $[\![s]\!]_\rho \in A_i$ and $[\![t]\!]_\rho \in A_j$ with $i \neq j$ for some ρ.
Then $\mathrm{NJ}(s, t)$.

(Proof Sketch) (i),(ii) imply that for any $s \xrightarrow{*}_{\mathcal{R}} u \to_{\mathcal{R}} v$, $[\![u]\!]_\rho \in A_i$ implies $[\![v]\!]_\rho \in A_i$. □

Non-Joinability by Interpretation

Theorem 1. Let $\mathcal{A} = \langle A, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$ be an $\mathcal{F}$-algebra with $A = \biguplus_{i \in I} A_i$ (i.e. disjoint union of A_i's), and s, t terms. Suppose
(i) $[\![l]\!]_\sigma \in A_i$ implies $[\![r]\!]_\sigma \in A_i$ for any $l \to r \in \mathcal{U}(s) \cup \mathcal{U}(t)$,
(ii) if $a \in A_i$ implies $f^{\mathcal{A}}(\ldots, a, \ldots) \in A_j$, then for any $b \in A_i$, $f^{\mathcal{A}}(\ldots, b, \ldots) \in A_j$, and
(iii) $[\![s]\!]_\rho \in A_i$ and $[\![t]\!]_\rho \in A_j$ with $i \neq j$ for some ρ.
Then $\mathrm{NJ}(s, t)$.

(Proof Sketch) (i),(ii) imply that for any $s \xrightarrow{*}_{\mathcal{R}} u \to_{\mathcal{R}} v$, $[\![u]\!]_\rho \in A_i$ implies $[\![v]\!]_\rho \in A_i$. $\square$

Example 1.

$$\mathcal{R} = \left\{ \begin{array}{llll} (1) & \mathsf{a} \to \mathsf{h}(\mathsf{c}) & (3) & \mathsf{h}(x) \to \mathsf{h}(\mathsf{h}(x)) \\ (2) & \mathsf{a} \to \mathsf{h}(\mathsf{f}(\mathsf{c})) & (4) & \mathsf{f}(x) \to \mathsf{f}(\mathsf{g}(x)) \end{array} \right\}.$$

Take candidates $\mathsf{h}(\mathsf{c})$, $\mathsf{h}(\mathsf{f}(\mathsf{c}))$. **Usable rules are** $\{(3), (4)\}$.

Take an $\mathcal{F}$**-algebra** $\mathcal{A} = \langle \{0, 1\}, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$ **as**

$\mathsf{a}^{\mathcal{A}} = \mathsf{c}^{\mathcal{A}} = 0$,

$\mathsf{f}^{\mathcal{A}}(n) = 1 - n$,

$\mathsf{h}^{\mathcal{A}}(n) = \mathsf{g}^{\mathcal{A}}(n) = n$.

Then $[\![\mathsf{h}(x)]\!]_\sigma = [\![\mathsf{h}(\mathsf{h}(x))]\!]_\sigma$, $[\![\mathsf{f}(x)]\!]_\sigma = [\![\mathsf{f}(\mathsf{g}(x))]\!]_\sigma$ **and** $[\![\mathsf{h}(\mathsf{c})]\!] \neq [\![\mathsf{h}(\mathsf{f}(\mathsf{c}))]\!]$. **Hence,** $\mathrm{NJ}(\mathsf{h}(\mathsf{c}), \mathsf{h}(\mathsf{f}(\mathsf{c})))$.

Example 2.

$$\mathcal{R} = \left\{ \begin{array}{llll} (\mathbf{1}) & \mathsf{a} \to \mathsf{f}(\mathsf{c}) & (\mathbf{3}) & \mathsf{f}(x) \to \mathsf{h}(\mathsf{g}(x)) \\ (\mathbf{2}) & \mathsf{a} \to \mathsf{h}(\mathsf{c}) & (\mathbf{4}) & \mathsf{h}(x) \to \mathsf{f}(\mathsf{g}(x)) \end{array} \right\}.$$

Take candidates $\mathsf{f}(\mathsf{c})$ and $\mathsf{h}(\mathsf{c})$. Usable rules are $\{(\mathbf{3}), (\mathbf{4})\}$.

Take an $\mathcal{F}$-algebra $\mathcal{A} = \langle \mathbb{N}, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$ as

$\mathsf{a}^{\mathcal{A}} = \mathsf{c}^{\mathcal{A}} = 0$

$\mathsf{g}^{\mathcal{A}}(n) = n + 1$

$\mathsf{f}^{\mathcal{A}}(n) = n$

$\mathsf{h}^{\mathcal{A}}(n) = n + 1$

Then $[\![\mathsf{f}(x)]\!]_\sigma \equiv [\![\mathsf{h}(\mathsf{g}(x))]\!]_\sigma \pmod{2}$, $[\![\mathsf{h}(x)]\!]_\sigma \equiv [\![\mathsf{f}(\mathsf{g}(x))]\!]_\sigma \pmod{2}$ **and** $[\![\mathsf{f}(\mathsf{c})]\!] \not\equiv [\![\mathsf{h}(\mathsf{c})]\!] \pmod{2}$. **Hence** $\mathrm{NJ}(\mathsf{f}(\mathsf{c}), \mathsf{h}(\mathsf{c}))$.

Outline

1. Backgrounds: TRS and Confluence
2. Backgrounds: Disproving Confluence
3. Proving Non-Joinability by Interpretation
4. **Proving Non-Joinability by Ordering**
5. Implementation and Experiments

Non-Joinability by Ordered $\mathcal{F}$-algebras

For a set of integers, an obvious choice of partition is $A = \{n \in A \mid n < k\} \uplus \{n \in A \mid k \leq n\}$ for some fixed k. More generally, one can use ordered $\mathcal{F}$-algebras $\mathcal{A} = \langle A, \leq, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$, where $\leq$ is a partial order on A.

Non-Joinability by Ordered $\mathcal{F}$-algebras

For a set of integers, an obvious choice of partition is $A = \{n \in A \mid n < k\} \uplus \{n \in A \mid k \leq n\}$ for some fixed k. More generally, one can use ordered $\mathcal{F}$-algebras $\mathcal{A} = \langle A, \leq, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$, where $\leq$ is a partial order on A.

Theorem 2. Let $\mathcal{A}$ be a weakly monotone ordered $\mathcal{F}$-algebra and s, t be terms. Suppose
(i) $[\![l]\!]_\sigma \leq [\![r]\!]_\sigma$ for any valuation σ and any $l \to r \in \mathcal{U}(s)$,
(ii) $[\![l]\!]_\sigma \geq [\![r]\!]_\sigma$ for any valuation σ and any $l \to r \in \mathcal{U}(t)$,
(iii) $[\![s]\!]_\rho > [\![t]\!]_\rho$ for some valuation ρ.
Then $\mathrm{NJ}(s, t)$.

Non-Joinability by Ordered $\mathcal{F}$-algebras

For a set of integers, an obvious choice of partition is $A = \{n \in A \mid n < k\} \uplus \{n \in A \mid k \leq n\}$ for some fixed k. More generally, one can use ordered $\mathcal{F}$-algebras $\mathcal{A} = \langle A, \leq, \langle f^{\mathcal{A}} \rangle_{f \in \mathcal{F}} \rangle$, where $\leq$ is a partial order on A.

Theorem 2. Let $\mathcal{A}$ be a weakly monotone ordered $\mathcal{F}$-algebra and s, t be terms. Suppose
(i) $[\![l]\!]_\sigma \leq [\![r]\!]_\sigma$ for any valuation σ and any $l \to r \in \mathcal{U}(s)$,
(ii) $[\![l]\!]_\sigma \geq [\![r]\!]_\sigma$ for any valuation σ and any $l \to r \in \mathcal{U}(t)$,
(iii) $[\![s]\!]_\rho > [\![t]\!]_\rho$ for some valuation ρ.
Then $\mathrm{NJ}(s, t)$.

Discrimination Pair

We now take term algebras for $\mathcal{F}$-algebras, and ordering on terms.

Definition. A pair $\langle \gtrsim, \succ \rangle$ of two relations $\gtrsim$ and $\succ$ is said to be a discrimination pair if (i) $\gtrsim$ is a rewrite relation, (ii) $\succ$ is an irreflexive relation and (iii) $\gtrsim \circ \succ \subseteq \succ$ and $\succ \circ \gtrsim \subseteq \succ$.

Theorem 3. Let $\mathcal{R}$ be a TRS and s, t terms. Suppose there exists a discrimination pair $\langle \gtrsim, \succ \rangle$ such that $\mathcal{U}(s) \subseteq \lesssim$, $\mathcal{U}(t) \subseteq \gtrsim$ and $s \succ t$. Then $\mathrm{NJ}(s, t)$.

In particular, various path orderings (developed for termination proving) can be used for discrimination pair.

Argument Filtering for Non-Joinability

One can incorporates the same notion of argument filtering in a termination proving technique. Argument filtering cuts off subterms in a consistent way.

An argument filtering is a mapping such that $\pi(f) \in \{[i_1, \ldots, i_k] \mid 1 \leq i_1 < \cdots < i_k \leq \mathrm{arity}(f)\} \cup \{i \mid 1 \leq i \leq \mathrm{arity}(f)\}$ for each $f \in \mathcal{F}$. We define $f(t_1, \ldots, t_n)^\pi = f(t_{i_1}^\pi, \ldots, t_{i_k}^\pi)$ if $\pi(f) = [i_1, \ldots, i_k]$, $f(t_1, \ldots, t_n)^\pi = t_i^\pi$ if $\pi(f) = i$. For TRS $\mathcal{R}$, we put $\mathcal{R}^\pi = \{l^\pi \to r^\pi \mid l \to r \in \mathcal{R}\}$.

Theorem 4. Let $\mathcal{R}$ be a TRS and s, t terms. Suppose there exists a discrimination pair $\langle \gtrsim, \succ \rangle$ and argument filtering π such that $\mathcal{U}_{\mathcal{R}^\pi}(s^\pi) \subseteq \lesssim$, $\mathcal{U}_{\mathcal{R}^\pi}(t^\pi) \subseteq \gtrsim$ and $s^\pi \succ t^\pi$. Then $\mathrm{NJ}(s, t)$.

Example 3.

$$\mathcal{R} = \left\{ \begin{array}{llll} (1) & \mathsf{c} \to \mathsf{f}(\mathsf{c}, \mathsf{d}), & (3) & \mathsf{f}(x, y) \to \mathsf{h}(\mathsf{g}(y), x), \\ (2) & \mathsf{c} \to \mathsf{h}(\mathsf{c}, \mathsf{d}) & (4) & \mathsf{h}(x, y) \to \mathsf{f}(\mathsf{g}(y), x) \end{array} \right\}.$$

Take candidates $\mathsf{h}(\mathsf{f}(\mathsf{c}, \mathsf{d}), \mathsf{d})$ **and** $\mathsf{f}(\mathsf{c}, \mathsf{d})$**.**

Take $\pi(\mathsf{g}) = 1$, $\pi(\mathsf{f}) = [2]$ **and** $\pi(\mathsf{h}) = [1]$**. Then** $\mathcal{U}(s^{\pi}) = \{(3)^{\pi}, (4)^{\pi}\}$ **and** $\mathcal{U}(t^{\pi}) = \{(3)^{\pi}, (4)^{\pi}\}$**.**

Then we obtain the constraint

$$\mathsf{h}(\mathsf{f}(\mathsf{d})) \succ \mathsf{f}(\mathsf{d}), \quad \mathsf{f}(y) \simeq \mathsf{h}(y), \quad \mathsf{h}(x) \simeq \mathsf{f}(x)$$

which is satisfied by a discrimination pair $\langle \gtrsim_{rpo}, \gtrsim_{rpo} \cap \lesssim_{rpo} \rangle$ **with precedence** $\mathsf{f} \simeq \mathsf{h}$**. Thus** $\mathrm{NJ}(s, t)$**.**

Outline

1. Backgrounds: TRS and Confluence
2. Backgrounds: Disproving Confluence
3. Proving Non-Joinability by Interpretation
4. Proving Non-Joinability by Ordering
5. Implementation and Experiments

Implementation

We implemented our techniques on the confluence prover ACP.

- **Interpretation by $\mathcal{F}$-algebras (Theorem 1) using the polynomial interpretation with linear polynomials and partition $\mathbb{N} = \biguplus_{0 \leq i < k} \{n \mid n \bmod k = i\}$ $(k = 2, 3)$.**

- **Interpretation by ordered $\mathcal{F}$-algebras (Theorem 2) with polynomial interpretation via linear polynomials.**

- **Discrimination pair (Theorem 4) using recursive path order with argument filtering.**

Criteria are encoded as a constraint and an external SMT-solver is called to check whether it has a solution.

Experiments

	Th.1 ($k=2$)	Th.1 ($k=3$)	Th.2 (poly)	Th.4 (rpo)	all
Example 1	✓	✓	✓	✓	✓
Example 2	✓	✓	×	×	✓
Example 3	×	×	×	✓	✓
23 ex. (success/t.o.)	16/0	16/3	14/0	19/0	21/1
23 ex. (time)	25	293	206	26	84
35 ex. (success/t.o.)	17/5	16/8	17/3	17/1	16/9
35 ex. (time)	318	562	446	106	761

	ACP	CSI	Saigawa
Example 1	×	×	×
Example 2	×	×	×
Example 3	×	×	×
23 ex. (success/t.o.)	9/0	12/–	3/1
23 ex. (time)	2	2107	228
35 ex. (success/t.o.)	18/1	21/–	17/6
35 ex. (time)	71	485	482

23 new examples
35 examples from Cops
ACP v.0.31
CSI v.0.2
Saigawa v.1.4

Conclusion

Disproving confluence by showing non-joinability of candidates.

- Proving non-joinability by interpretation
  $\mathcal{F}$-algebra, usable rules
- Proving non-joinability by ordering
  ordered $\mathcal{F}$-algebra
  discrimination pairs, argument filtering
- Implementation and experiments

Future Works

- More effective interpretation and ordering