GCR Examples from Transformation of Cyclic Proofs into RI Proofs

Shujun Zhang
joint work with Naoki Nishida
Nagoya University
February 22, 2024, ARI meeting, Kira Yosida

Contents

Background: Comparison of RI with cyclic proof systems

Research Purpose

Comparison of RI with cyclic proof systems

- Both RI with cyclic proof systems have very similar inference rules (e.g. case analysis, rule application)

Approach to Comparison of RI with Cyclic Proof Systems

1. Transform an IDS Φ into an equivalent TRS $\mathcal{R}$ such that [ZN, 21 & 22]
   - $\mathcal{R}$ is GSC-terminating and confluent
2. Transform a cyclic proof $\mathcal{P}$ into an RI proof [ZN, 22]
   - Add rewrite rules of sequent calculus into TRS $\mathcal{R}$
   - The result TRS is also confluent
3. Transform an RI proof into a cyclic proof [ZN, 22]

- We would like to use GCR tools on Cocoweb to prove confluence of transformed TRS

Contents

Term Rewrite Systems and Inductive Theorems

- Term rewrite system (TRS) is a set $\mathcal{R}$ of rewrite rules $\ell \to r$
- Equation $s \approx t$ is an inductive theorem of $\mathcal{R}$ if

$$\forall \theta : \text{ground substitution. } s\theta \leftrightarrow^*_{\mathcal{R}} t\theta$$

that is, $s \approx t$ is valid w.r.t. $\to_{\mathcal{R}}$

Example

$$\mathcal{R}_1 = \left\{ \begin{array}{c} 0 + y \to y \\ \mathsf{s}(x) + y \to \mathsf{s}(x + y) \end{array} \right\}$$

- $x + 0 \approx x$ is not a theorem of $\mathcal{R}_1$ because $x + 0 \not\leftrightarrow^*_{\mathcal{R}_1} x$
- $x + 0 \approx x$ is an inductive theorem of $\mathcal{R}_1$ because

$$\begin{array}{c} 0 + 0 \leftrightarrow^*_{\mathcal{R}_1} 0 \\ \mathsf{s}(0) + 0 \leftrightarrow^*_{\mathcal{R}_1} \mathsf{s}(0) \\ \vdots \end{array}$$

Rewriting Induction (RI) [Reddy, 90]

- A method of proving equations to be inductive theorems of $\mathcal{R}$
- Consist of inference rules for $(\mathcal{E}, \mathcal{H})$, e.g.,
  - **Expand** for case analysis with setting IH to $\mathcal{H}$
  - **Simplify** by rule application with $\mathcal{R}$ and $\mathcal{H}$
  - **Delete** to remove trivial inductive theorems

Example

$$\mathcal{R}_1 = \left\{ \begin{array}{c} 0 + y \to y \\ \mathsf{s}(x) + y \to \mathsf{s}(x + y) \end{array} \right\}$$

- RI proves $x + 0 \approx x$ to be an inductive theorem of $\mathcal{R}_1$:

$$\begin{array}{rl} (\{((\varepsilon)\ x + 0 \approx x\}, \emptyset) & \Rrightarrow_{\mathsf{E}} (\{\ (1)\ 0 \approx 0, (2)\ \mathsf{s}(x_1 + 0) \approx \mathsf{s}(x_1)\ \},\ \{((\varepsilon)\ x + 0 \to x\}) \\ & \Rrightarrow_{\mathsf{D}} (\{\ (2)\ \mathsf{s}(x_1 + 0) \approx \mathsf{s}(x_1)\ \},\ \{((\varepsilon)\ x + 0 \to x\}) \\ & \Rrightarrow_{\mathsf{S}} (\{\ (2.1)\ \mathsf{s}(x_1) \approx \mathsf{s}(x_1)\ \},\ \{((\varepsilon)\ x + 0 \to x\}) \\ & \Rrightarrow_{\mathsf{D}} (\{\ \},\ \{((\varepsilon)\ x + 0 \to x\}) \end{array}$$

Inductive Definition Sets (IDS)

- IDS Φ is a finite set of productions

$$\frac{P_1(\vec{t_1}) \quad \dots \quad P_n(\vec{t_n})}{P(\vec{t})}$$

where $P, P_1, \dots, P_n$ are predicate symbols

- We do not deal with ordinary predicates
- $P(\vec{t})\theta$ is true iff $P_1(\vec{t_1})\theta, \ \dots, \ P_n(\vec{t_n})\theta$ are true

Example ([Brotherston, 05])

$$\Phi_1 = \left\{ \frac{}{\mathsf{N}(0)} \quad \frac{\mathsf{N}(x)}{\mathsf{N}(\mathsf{s}(x))} \quad \frac{}{\mathsf{E}(0)} \quad \frac{\mathsf{O}(x)}{\mathsf{E}(\mathsf{s}(x))} \quad \frac{\mathsf{E}(x)}{\mathsf{O}(\mathsf{s}(x))} \right\}$$

- $\mathsf{E}(\mathsf{s}(\mathsf{s}(0)))$ is true
- $\mathsf{O}(0)$ is false
- $\mathsf{E}(x) \vee \mathsf{O}(x)$ is valid w.r.t. Φ_1

Sequents and Cyclic Proofs

- $\Gamma \vdash \Delta$ is a sequent, where Γ, Δ are multi-sets of formulas
- $\Gamma \vdash \Delta$ is valid w.r.t. Φ if $(\bigwedge_{F \in \Gamma} F) \Rightarrow (\bigvee_{F' \in \Delta} F')$ is valid w.r.t. Φ
- Cyclic proof is a finite proof tree consiting of
  - (P_i-App), (Case P), (Subst), and rules for sequent calculus
  - pairs of bud nodes and companions

Example (Cyclic Proof)

$$\dfrac{\dfrac{\dfrac{}{(1.1)\ \vdash \mathsf{N}(0)}(\mathsf{N}_1\text{-App}) \quad \dfrac{\dfrac{\dfrac{(1.2.1.1)\ \mathsf{O}(x) \vdash \mathsf{N}(x)\ \dagger}{(1.2.1)\ \mathsf{O}(y) \vdash \mathsf{N}(y)}(\text{Subst})}{(1.2)\ \mathsf{O}(y) \vdash \mathsf{N}(\mathsf{s}(y))}(\mathsf{N}_2\text{-App})}{(1)\ \mathsf{E}(x) \vdash \mathsf{N}(x)\ \ddagger}(\text{Case E}) \quad \dfrac{\dfrac{\dfrac{(2.1.1.1)\ \mathsf{E}(x) \vdash \mathsf{N}(x)\ \ddagger}{(2.1.1)\ \mathsf{E}(y) \vdash \mathsf{N}(y)}(\text{Subst})}{(2.1)\ \mathsf{E}(y) \vdash \mathsf{N}(\mathsf{s}(y))}(\mathsf{N}_2\text{-App})}{(2)\ \mathsf{O}(x) \vdash \mathsf{N}(x)\ \dagger}(\text{Case O})}{(\varepsilon)\ \mathsf{E}(x) \vee \mathsf{O}(x) \vdash \mathsf{N}(x)}(\vee\text{L})$$

- $(1.2.1.1)\ \mathsf{O}(x) \vdash \mathsf{N}(x)\ \dagger$ and $(2.1.1.1)\ \mathsf{E}(x) \vdash \mathsf{N}(x)\ \ddagger$ are bud nodes
- $(2)\ \mathsf{O}(x) \vdash \mathsf{N}(x)\ \dagger$ and $(1)\ \mathsf{E}(x) \vdash \mathsf{N}(x)\ \ddagger$ are their companions, resp.

Theorem ([Brotherston, 05])

$\Gamma \vdash \Delta$ is valid w.r.t. Φ if there is a cyclic proof for $\Gamma \vdash \Delta$

Contents

From IDS to TRS [ZN, 22]

Transform IDS Φ into TRS

$$\mathcal{R} \quad := \quad \mathcal{R}_\Phi \cup \mathcal{R}_\Phi^{co} \cup \mathcal{R}_{pl} \cup \mathcal{R}_{seq}$$

where

- $\mathcal{R}_\Phi = \{\ A \to \mathsf{and}(A_1, \mathsf{and}(\dots, \mathsf{and}(A_{n-1}, A_n)\dots)) \mid \frac{A_1 \ \cdots \ A_n}{A} \in \Phi\ \}$
  - If $n = 1$, then $A \to A_1$ is generated
  - If $n = 0$, then $A \to \top$ is generated
- $\mathcal{R}_\Phi^{co} = \{\ t \to \mathsf{false} \mid t \text{ is a co-pattern of } \mathcal{R}_\Phi\ \}$
- $\mathcal{R}_{pl} = \left\{ \begin{array}{lll} \mathsf{and}(\mathsf{false},\mathsf{false}) \to \mathsf{false}, & \mathsf{or}(\mathsf{false},\mathsf{false}) \to \mathsf{false}, & \mathsf{not}(\mathsf{false}) \to \mathsf{true}, \\ \mathsf{and}(\mathsf{false},\mathsf{true}) \to \mathsf{false}, & \mathsf{or}(\mathsf{false},\mathsf{true}) \to \mathsf{true}, & \mathsf{not}(\mathsf{true}) \to \mathsf{false}, \\ \mathsf{and}(\mathsf{true},\mathsf{false}) \to \mathsf{false}, & \mathsf{or}(\mathsf{true},\mathsf{false}) \to \mathsf{true}, & \\ \mathsf{and}(\mathsf{true},\mathsf{true}) \to \mathsf{true}, & \mathsf{or}(\mathsf{true},\mathsf{true}) \to \mathsf{true} & \end{array} \right\}$
- $\mathcal{R}_{seq} = \{\, \mathsf{seq}(\mathsf{false},\mathsf{false}) \to \top \ \mathsf{seq}(\mathsf{false},\mathsf{true}) \to \top \ \mathsf{seq}(\mathsf{true},\mathsf{false}) \to \bot \ \mathsf{seq}(\mathsf{true},\mathsf{true}) \to \top \,\}$
  - $\top, \bot$ mean validity and invalidity of sequents, respectively.

Example: Transformation of IDS into TRS

Example (cont'd)

$$\Phi_1 = \left\{ \quad \frac{}{\mathsf{N}(0)} \qquad \frac{\mathsf{N}(x)}{\mathsf{N}(\mathsf{s}(x))} \qquad \frac{}{\mathsf{E}(0)} \qquad \frac{\mathsf{O}(x)}{\mathsf{E}(\mathsf{s}(x))} \qquad \frac{\mathsf{E}(x)}{\mathsf{O}(\mathsf{s}(x))} \quad \right\}$$

is transformed into

$$\mathcal{R} = \left\{ \begin{array}{lll} \mathsf{N}(0) \to \mathsf{true}, & \mathsf{E}(0) \to \mathsf{true}, & \mathsf{O}(0) \to \mathsf{false}, \\ \mathsf{N}(\mathsf{s}(x)) \to \mathsf{N}(x), & \mathsf{E}(\mathsf{s}(x)) \to \mathsf{O}(x), & \mathsf{O}(\mathsf{s}(x)) \to \mathsf{E}(x) \end{array} \right\} \cup \mathcal{R}_{pl} \cup \mathcal{R}_{seq}$$

Properties of the Resulting TRS $\mathcal{R}$

Suppose

- $\mathcal{R}$ is confluent
  - Orthogonality of Φ is assumed to imply confluence of $\mathcal{R}$
- $\mathcal{R}$ is terminating

Lemma ([ZN, 21])

For any ground formula F,

- $\Phi \models F$ *iff* $F \to^*_{\mathcal{R}} \mathsf{true}$
- $\Phi \not\models F$ *iff* $F \to^*_{\mathcal{R}} \mathsf{false}$

Theorem ([ZN, 21])

$\Gamma \vdash \Delta$ *is valid w.r.t.* Φ *iff* $\mathsf{seq}(\Gamma, \Delta) \approx \mathsf{true}$ *is an inductive theorem of* $\mathcal{R}$

- Ground confluence and ground termination of $\mathcal{R}$ are enough for the above claims
- Use GCR tools to prove ground confluence of $\mathcal{R}$

Example: Confluence and Ground Confluence of $\mathcal{R}$

Example

Define the type of $\mathcal{R}$ as following:

```
(SIG
   (E      Nat -> Bool)
   (O      Nat -> Bool)
   (N      Nat -> Bool)
   (s      Nat -> Nat)
   (0      -> Nat)
   (true   -> Bool)
   (false  -> Bool)
   (and    Bool Bool -> Bool)
   (or     Bool Bool -> Bool)
   (not    Bool -> Bool)
   (seq    Bool Bool -> Prop)
   (top    -> Prop)
   (bot    -> Prop)
   (AND    Porp Prop -> Prop)
)
```

```
(VAR
   t : Nat
   x : Bool
   y : Bool
   z : Bool
)
```


Example: Confluence and Ground Confluencr of $\mathcal{R}$

Example

$$\mathcal{R} = \left\{ \begin{array}{ccc} \mathsf{N}(0) \to \mathsf{true}, & \mathsf{E}(0) \to \mathsf{true}, & \mathsf{O}(0) \to \mathsf{false}, \\ \mathsf{N}(\mathsf{s}(x)) \to \mathsf{N}(x), & \mathsf{E}(\mathsf{s}(x)) \to \mathsf{O}(x), & \mathsf{O}(\mathsf{s}(x)) \to \mathsf{E}(x) \end{array} \right\} \cup \mathcal{R}_{pl} \cup \mathcal{R}_{seq}$$

```
(RULES
   N(0) -> true
   E(0) -> true
   O(0) -> false
   N(s(t)) -> N(t)
   E(s(t)) -> O(t)
   O(s(t)) -> E(t)
   and(false,false) -> false
   and(false,true) -> false
   and(true,false) -> false
   and(true,true) -> true
   or(false,false) -> false
   or(false,true) -> true
   or(true,false) -> true
   or(true,true) -> true
   not(true) -> false
   not(false) -> true
   seq(false,false) -> top
   seq(false,true) -> top
   seq(true,false) -> bot
   seq(true,true) -> top
)
```


Example: Confluence of $\mathcal{R}$

Example

	CR of $\mathcal{R}$ (TRS)	GCR of $\mathcal{R}$ (MSTRS)
ACP	YES	/
AGCP	/	YES
CONFident	YES	/
CSI	YES	/
FORT-h	MAYBE	MAYBE
Hakusan	YES	/

- FORT-h only support for LV-TRSs

Contents

Term Coding of Sequents [ZN, 21]

- Transform

$$
\begin{array}{rcc}
\Gamma \vdash \Delta & \text{into} & \mathsf{seq}(\Gamma, \Delta) \\
\Gamma = \{F_1, \ldots, F_n\} & \text{into} & \mathsf{and}(F_1, \mathsf{and}(\ldots, \mathsf{and}(F_{n-1}, F_n)\ldots)) \\
\Delta = \{F_1, \ldots, F_n\} & \text{into} & \mathsf{or}(F_1, \mathsf{or}(\ldots, \mathsf{or}(F_{n-1}, F_n)\ldots)) \\
\neg F & \text{into} & \mathsf{not}(F) \\
F_1 \wedge F_2 & \text{into} & \mathsf{and}(F_1, F_2) \\
F_1 \vee F_2 & \text{into} & \mathsf{or}(F_1, F_2)
\end{array}
$$

Example

- $\mathsf{E}(x) \vee \mathsf{O}(x) \vdash \mathsf{N}(x)$ is transformed into $\mathsf{seq}(\mathsf{or}(\mathsf{E}(x), \mathsf{O}(x)), \mathsf{N}(x))$

Rewrite Rules of Sequent-Calculus Rules

Sequent Calculus Rules

$$
\frac{}{F \vdash F}\,(\text{Axiom}) \qquad \frac{F_1 \vdash \Delta \quad F_2 \vdash \Delta}{F_1 \vee F_2 \vdash \Delta}\,(\vee\text{L}) \qquad \frac{\Gamma \vdash F_1 \quad \Gamma \vdash F_2}{\Gamma \vdash F_1 \wedge F_2}\,(\wedge\text{R})
$$

$$
\frac{F_1 \vdash \Delta}{F_1 \wedge F_2 \vdash \Delta}\,(\wedge\text{L}_1) \qquad \frac{F_2 \vdash \Delta}{F_1 \wedge F_2 \vdash \Delta}\,(\wedge\text{L}_2) \qquad \frac{\Gamma \vdash F_1}{\Gamma \vdash F_1 \vee F_2}\,(\vee\text{R}_1) \qquad \frac{\Gamma \vdash F_2}{\Gamma \vdash F_1 \vee F_2}\,(\vee\text{R}_2) \quad \ldots
$$

Problem

Above sequent-calculus rules have not been transformed into rewrite rules

- Transform these sequent-calculus rules into rewrite rules

Transforming Sequent-Calculus Rules into Rewrite Rules

Rewrite Rules for Sequent Calculus Rules

$$
\frac{}{F \vdash F}\,(\text{Axiom}) \qquad \frac{F_1 \vdash F \quad F_2 \vdash F}{F_1 \vee F_2 \vdash F}\,(\vee\text{L}) \qquad \frac{F \vdash F_1 \quad F \vdash F_2}{F \vdash F_1 \wedge F_2}\,(\wedge\text{R})
$$

$$
\frac{F_1 \vdash \Delta}{F_1 \wedge F_2 \vdash \Delta}\,(\wedge\text{L}_1) \qquad \frac{F_2 \vdash \Delta}{F_1 \wedge F_2 \vdash \Delta}\,(\wedge\text{L}_2) \qquad \frac{\Gamma \vdash F_1}{\Gamma \vdash F_1 \vee F_2}\,(\vee\text{R}_1) \qquad \frac{\Gamma \vdash F_2}{\Gamma \vdash F_1 \vee F_2}
$$

are transformed into:

$$
\mathcal{R}_{scr} = \left\{
\begin{array}{lr}
(\text{Axiom}) & \mathsf{seq}(x, x) \to \top \\
(\vee\text{L}) & \mathsf{seq}(\mathsf{or}(x, y), z) \to \mathsf{seq}(x, z) \mathbin{\&} \mathsf{seq}(y, z) \\
(\wedge\text{R}) & \mathsf{seq}(x, \mathsf{and}(y, z)) \to \mathsf{seq}(x, y) \mathbin{\&} \mathsf{seq}(x, z) \\
(\wedge\text{L}_1) & \mathsf{seq}(\mathsf{and}(x, y), z) \to \mathsf{seq}(x, z) \\
(\wedge\text{L}_2) & \mathsf{seq}(\mathsf{and}(x, y), z) \to \mathsf{seq}(y, z) \\
(\vee\text{R}_1) & \mathsf{seq}(x, \mathsf{or}(y, z)) \to \mathsf{seq}(x, y) \\
(\vee\text{R}_2) & \mathsf{seq}(x, \mathsf{or}(y, z)) \to \mathsf{seq}(x, z)
\end{array}
\right\}
$$

- For RI proofs, $\mathcal{R} \cup \mathcal{R}_{scr}$ may be used instead of $\mathcal{R}$
- Use GCR tools to prove ground confluence of $\mathcal{R} \cup \mathcal{R}_{scr}$

Confluence and Ground Confluence of $\mathcal{R} \cup \mathcal{R}_{scr}$

Example

We add rules of AND and $\mathcal{R}_{scr}$ into TRS

```
(RULES
  N(0) -> true
  E(0) -> true
  O(0) -> false
  N(s(t)) -> N(t)
  E(s(t)) -> O(t)
  O(s(t)) -> E(t)
  and(false,false) -> false
  and(false,true) -> false
  and(true,false) -> false
  and(true,true) -> true
  or(false,false) -> false
  or(false,true) -> true
  or(true,false) -> true
  or(true,true) -> true
  not(true) -> false
  not(false) -> true
  seq(false,false) -> top
  seq(false,true) -> top
  seq(true,false) -> bot
  seq(true,true) -> top
  AND(bot,bot) -> bot
  AND(bot,top) -> bot
  AND(top,bot) -> bot
  AND(top,top) -> top
  seq(x,x) -> top
  seq(or(x,y),z) -> AND(seq(x,z),seq(y,z))
  seq(x,and(y,z)) -> AND(seq(x,y),seq(x,z))
  seq(and(x, y), z) -> seq(x,z)
  seq(and(x, y), z) -> seq(y,z)
  seq(x, or(y, z)) -> seq(x,y)
  seq(x, or(y, z)) -> seq(x,z)
)
```

Confluence and Ground Confluence of $\mathcal{R} \cup \mathcal{R}_{scr}$

Example

- ACP and SCI prove that $\mathcal{R} \cup \mathcal{R}_{scr}$ is not confluent
- FORT-h only support for LV-TRSs
- CONFident seems TIMEOUT

	CR of $\mathcal{R} \cup \mathcal{R}_{scr1} \cup \mathcal{R}_{scr2}$ (TRS)	GCR of $\mathcal{R} \cup \mathcal{R}_{scr1} \cup \mathcal{R}_{scr2}$ (MSTRS)
ACP	NO	/
AGCP	/	NO
CONFident	MAYBE	/
CSI	NO	/
FORT-h	MAYBE	MAYBE
Hakusan	MAYBE	/

Ground Confluence of $\mathcal{R} \cup \mathcal{R}_{scr}$

Example

- From critical pairs we know $\mathcal{R} \cup \mathcal{R}_{scr}$ is not ground confluent (following is one of the counterexample)

$$\mathsf{top} = \mathsf{seq}(?x, \mathsf{and}(?x, ?y))$$

Use GCR tools to prove CR and GCR of $\mathcal{R} \cup \mathcal{R}_{scr1}$

Example

Let $\mathcal{R}_{scr1}$ includes (Axiom), ($\vee$L) and ($\wedge$R)

$$\mathcal{R}_{scr1} = \left\{ \begin{array}{cc} \text{(Axiom)} & \mathsf{seq}(x,x) \to \top \\ (\vee\text{L}) & \mathsf{seq}(\mathsf{or}(x,y),z) \to \mathsf{seq}(x,z)\ \&\ \mathsf{seq}(y,z) \\ (\wedge\text{R}) & \mathsf{seq}(x,\mathsf{and}(y,z)) \to \mathsf{seq}(x,y)\ \&\ \mathsf{seq}(x,z) \end{array} \right\}$$

We add AND rules, (Axiom), ($\vee$L) and ($\wedge$R) into TRS

```
(RULES
   N(0) -> true
   E(0) -> true
   O(0) -> false
   N(s(t)) -> N(t)
   E(s(t)) -> O(t)
   O(s(t)) -> E(t)
   and(false,false) -> false
   and(false,true) -> false
   and(true,false) -> false
   and(true,true) -> true
   or(false,false) -> false
   or(false,true) -> true
   or(true,false) -> true
   or(true,true) -> true
   not(true) -> false
   not(false) -> true
   seq(false,false) -> top
   seq(false,true) -> top
   seq(true,false) -> bot
   seq(true,true) -> top
   AND(bot,bot) -> bot
   AND(bot,top) -> bot
   AND(top,bot) -> bot
   AND(top,top) -> top
   seq(x,x) -> top
   seq(or(x,y),z) -> AND(seq(x,z),seq(y,z))
   seq(x,and(y,z)) -> AND(seq(x,y),seq(x,z))
)
```


Confluence and Ground Confluence of $\mathcal{R} \cup \mathcal{R}_{scr1}$

Example

- ACP and SCI prove that $\mathcal{R} \cup \mathcal{R}_{scr1}$ is not confluent
- FORT-h only support for LV-TRSs
- CONFident seems TIMEOUT
- AGCP prove that $\mathcal{R} \cup \mathcal{R}_{scr1}$ is ground confluent

	CR of $\mathcal{R} \cup \mathcal{R}_{scr1}$ (TRS)	GCR of $\mathcal{R} \cup \mathcal{R}_{scr1}$ (MSTRS)
ACP	NO	/
AGCP	/	YES
CONFident	MAYBE	/
CSI	NO	/
FORT-h	MAYBE	MAYBE
Hakusan	MAYBE	/