New ideas on transformations of CTRSs

Karl Gmeiner

Vienna University of Technology, Austria

July 4, 2012

Outline

Transformations of CTRSs

Conditional term rewriting

- Rules bound to conditions: $l \to r \Leftarrow s_1 = t_1, \ldots, s_n = t_n$
- = here interpreted as reductability $\to^*$ (joinability $\downarrow$ can be easily simulated).

Transformations

- (1) Unravelings $\mathbb{U}$ defined by [Marchiori'96]
- (2) Transformations stemming from [Viry'99]
- Properties:
  Completeness ($s \to_{\mathcal{R}}^* t \Rightarrow s \to_{\mathbb{U}(\mathcal{R})}^* t$),
  Soundness ($s \to_{\mathcal{R}}^* t \Leftarrow s \to_{\mathbb{U}(\mathcal{R})}^* t$)

Ultra-properties (wrt. a transformation)

- $\mathcal{R}$ has ultra-property $\mathbb{U} - \mathcal{P}$, if $\mathbb{U}(\mathcal{R})$ has property $\mathcal{P}$.

Unravelings

Definition of unravelings

- Rule $\alpha : l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n$

Unravelings

Definition of unravelings

- Rule $\alpha : l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n$
- $\mathbb{U}_{seq}(\alpha)$: All variables are stored along with conditions:

$$
\begin{array}{rcl|l}
l & \to & U_1^\alpha(s_1, \vec{X}_1) & X_1 = \mathcal{V}ars(l) \\
U_1^\alpha(t_1, \vec{X}_1) & \to & U_2^\alpha(s_2, \vec{X}_2) & X_2 = \mathcal{V}ars(l, t_1) \ldots \\
\vdots & & \vdots & X_n = \mathcal{V}ars(l, t_1, \ldots, t_{n-1}) \\
U_n^\alpha(t_n, \vec{X}_n) & \to & r &
\end{array}
$$

Unravelings

Definition of unravelings

- Rule $\alpha : l \to r \Leftarrow s_1 \to^* t_1, \ldots, s_n \to^* t_n$
- $\mathbb{U}_{seq}(\alpha)$: All variables are stored along with conditions:

$$\begin{array}{rcl|l} l & \to & U_1^\alpha(s_1, \vec{X}_1) & X_1 = \mathcal{V}ars(l) \\ U_1^\alpha(t_1, \vec{X}_1) & \to & U_2^\alpha(s_2, \vec{X}_2) & X_2 = \mathcal{V}ars(l, t_1) \ldots \\ \vdots & & \vdots & X_n = \mathcal{V}ars(l, t_1, \ldots, t_{n-1}) \\ U_n^\alpha(t_n, \vec{X}_n) & \to & r & \end{array}$$

- $\mathbb{U}_{opt}(\alpha)$: Variables are only encoded if they are needed:

$$\begin{array}{rcl|l} l & \to & U_1^\alpha(s_1, \vec{X}_1) & \\ U_1^\alpha(t_1, \vec{X}_1) & \to & U_2^\alpha(s_2, \vec{X}_2) & X_1 = \mathcal{V}ars(l) \cap \mathcal{V}ars(t_1, s_2, t_2, \ldots, s_n, t_n, r) \\ \vdots & & \vdots & X_2 = \mathcal{V}ars(l, t_1) \cap \mathcal{V}ars(t_2, s_3, t_3, \ldots, s_n, t_n, r) \ldots \\ & & & X_n = \mathcal{V}ars(l, t_1, \ldots, t_{n-1}) \cap \mathcal{V}ars(t_n, r) \\ U_n^\alpha(t_n, \vec{X}_n) & \to & r & \end{array}$$

Example

Example

$$\mathcal{R}_{or} = \left\{ \alpha : \quad or(x, y) \to true \Leftarrow x \to^* true \right\}$$

Example

Example

$$\mathcal{R}_{or} = \left\{ \alpha : \quad or(x,y) \to true \Leftarrow x \to^* true \right\}$$

$$\mathbb{U}_{seq}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x,y) \to U_1^\alpha(x,x,y) \\ U_1^\alpha(true,x,y) \to true \end{array} \right\}$$

Example

Example

$$\mathcal{R}_{or} = \left\{ \alpha : \quad or(x, y) \to true \Leftarrow x \to^{*} true \right\}$$

$$\mathbb{U}_{seq}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x, y) \to U_1^{\alpha}(x, x, y) \\ U_1^{\alpha}(true, x, y) \to true \end{array} \right\}$$

$$\mathbb{U}_{opt}(\mathcal{R}_{or}) = \left\{ \begin{array}{l} or(x, y) \to U_1^{\alpha}(x) \\ U_1^{\alpha}(true) \to true \end{array} \right\}$$

Example

Example

$$\mathcal{R}_{or} = \left\{ \alpha : \quad or(x,y) \to true \Leftarrow x \to^* true \right\}$$

$$\mathbb{U}_{seq}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x,y) \to U_1^{\alpha}(x,x,y) \\ U_1^{\alpha}(true,x,y) \to true \end{array} \right\}$$

$$\mathbb{U}_{opt}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x,y) \to U_1^{\alpha}(x) \\ U_1^{\alpha}(true) \to true \end{array} \right\}$$

Properties of example

- $\mathbb{U}_{opt}$ is unsound for joinability:
  $or(false, false) \not\downarrow_{\mathcal{R}} or(false, true)$, but
  $or(false, false) \downarrow_{\mathbb{U}_{opt}(\mathcal{R})} or(false, true)$
- Causes unsoundness for WLL CTRSs.

When are we allowed to drop variables?

- $\mathbb{U}_{opt}$ does not preserve of soundness for joinability: $s \downarrow_{\mathcal{R}} \Leftarrow s \downarrow_{\mathbb{U}(\mathcal{R})} t$ (also for left-linear systems).
- We only remove variables if soundness of joinability is not affected
- If a variable x occurs on the lhs s_i of a condition, and if the condition is injective wrt. the variable ($s_i\sigma \rightarrow^* u \leftarrow^* s_i\sigma'$) and it is not used anymore, we do not encode it:

$$
\begin{aligned}
l &\rightarrow U_1^{\alpha}(s_1, \vec{X}_1) \\
U_1^{\alpha}(t_1, \vec{X}_1) &\rightarrow U_2^{\alpha}(s_2, \vec{X}_2) \\
&\vdots \\
U_n^{\alpha}(t_n, \vec{X}_n) &\rightarrow r
\end{aligned}
$$

When are we allowed to drop variables?

- $\mathbb{U}_{opt}$ does not preserve of soundness for joinability: $s \downarrow_{\mathcal{R}} \Leftarrow s \downarrow_{\mathbb{U}(\mathcal{R})} t$ (also for left-linear systems).
- We only remove variables if soundness of joinability is not affected
- If a variable x occurs on the lhs s_i of a condition, and if the condition is injective wrt. the variable ($s_i\sigma \to^* u \leftarrow^* s_i\sigma'$) and it is not used anymore, we do not encode it:

$$\begin{aligned} l &\to U_1^\alpha(s_1, \vec{X}_1) \\ U_1^\alpha(t_1, \vec{X}_1) &\to U_2^\alpha(s_2, \vec{X}_2) \\ \vdots \quad & \quad \vdots \\ U_n^\alpha(t_n, \vec{X}_n) &\to r \end{aligned}$$

$X_1 = \{x \in \mathcal{V}ars(l) \mid x \in \mathcal{V}ars(t_1, s_2, t_2, \ldots, s_n, t_n, r) \vee s_1\{x \mapsto u\}\sigma \downarrow s_1\{x \mapsto v\}\sigma\}$
$X_2 = \{x \in X_1 \cup \mathcal{V}ars(t_1) \mid x \in \mathcal{V}ars(t_2, \ldots, s_n, t_n, r) \vee s_2\{x \mapsto u\}\sigma \downarrow s_2\{x \mapsto v\}\sigma\} \ldots$
$X_n = \{x \in X_{n-1} \cup \mathcal{V}ars(t_{n-1}) \mid x \in \mathcal{V}ars(t_n, r) \vee s_n\{x \mapsto u\}\sigma \downarrow s_n\{x \mapsto v\}\sigma\} \ldots$

for some σ and $u \not\downarrow v$

Example

Example

$$\mathcal{R}_{or} = \left\{ \alpha : \quad or(x, y) \to true \Leftarrow x \to^* true \right\}$$

$$\mathbb{U}_{seq}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x, y) \to U_1^\alpha(x, x, y) \\ U_1^\alpha(true, x, y) \to true \end{array} \right\}$$

$$\mathbb{U}_{opt}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x, y) \to U_1^\alpha(x) \\ U_1^\alpha(true) \to true \end{array} \right\}$$

$$\mathbb{U}_{osafe}(\mathcal{R}_{or}) = \left\{ \begin{array}{r} or(x, y) \to U_1^\alpha(x, y) \\ U_1^\alpha(true, y) \to true \end{array} \right\}$$

Transformation of [Viry99][AntoyBrasselHanus03] - Motivation

Example

Goal: Find a transformation $\mathbb{T}$ that is sound for preserving normal forms [GmeinerGramlich08]/computationally equivalent ([SerbanutaRosu06]).

$$\mathcal{R}_{not} = \left\{ \begin{array}{l} not(x) \to true \Leftarrow x \to^* false \\ not(x) \to false \Leftarrow x \to^* true \end{array} \right\}$$

$$\mathbb{U}_{osafe}(\mathcal{R}_{not}) = \left\{ \begin{array}{l} not(x) \to U_1^{\alpha}(x) \\ U_1^{\alpha}(false) \to true \\ not(x) \to U_1^{\beta}(x) \\ U_1^{\beta}(true) \to false \end{array} \right\}$$

$$not(true) \to U_1^{\alpha}(true) \not\to$$

Transformation of [Viry99][AntoyBrasselHanus03] - Motivation

Example

Goal: Find a transformation $\mathbb{T}$ that is sound for preserving normal forms [GmeinerGramlich08]/computationally equivalent ([SerbanutaRosu06]).

$$\mathcal{R}_{not} = \left\{ \begin{aligned} not(x) &\to true \Leftarrow x \to^* false \\ not(x) &\to false \Leftarrow x \to^* true \end{aligned} \right\}$$

$$\mathbb{U}_{osafe}(\mathcal{R}_{not}) = \left\{ \begin{aligned} not(x) &\to U_1^\alpha(x) \\ U_1^\alpha(false) &\to true \\ not(x) &\to U_1^\beta(x) \\ U_1^\beta(true) &\to false \end{aligned} \right\}$$

$$\begin{array}{l} not(true) \to U_1^\alpha(true) \not\to \\ \qquad\qquad\quad \searrow \\ \qquad\qquad\qquad U_1^\beta(true) \to false \end{array}$$

Appending the conditional argument to *not*-terms

Approach: Extend arity of defined symbols.

Example

$$\mathcal{R}_{not} = \left\{ \begin{array}{l} not(x) \to true \Leftarrow x \to^* false \\ not(x) \to false \Leftarrow x \to^* true \end{array} \right\}$$

$$\mathbb{T}_{abh}(\mathcal{R}_{not}) = \left\{ \begin{array}{rl} not'(x, \bot, z) & \to not'(x, \langle x \rangle, z) \\ not'(x, \langle false \rangle, z) & \to true \\ not'(x, z, \bot) & \to not'(x, z, \langle x \rangle) \\ not'(x, z, \langle true \rangle) & \to false \end{array} \right\}$$

$not'(true, \bot, \bot) \to not'(true, \langle true \rangle, \bot)$

Appending the conditional argument to *not*-terms

Approach: Extend arity of defined symbols.

Example

$$\mathcal{R}_{not} = \left\{ \begin{array}{l} not(x) \to true \Leftarrow x \to^* false \\ not(x) \to false \Leftarrow x \to^* true \end{array} \right\}$$

$$\mathbb{T}_{abh}(\mathcal{R}_{not}) = \left\{ \begin{array}{rl} not'(x, \bot, z) \to & not'(x, \langle x \rangle, z) \\ not'(x, \langle false \rangle, z) \to & true \\ not'(x, z, \bot) \to & not'(x, z, \langle x \rangle) \\ not'(x, z, \langle true \rangle) \to & false \end{array} \right\}$$

$$\begin{array}{l} not'(true, \bot, \bot) \to not'(true, \langle true \rangle, \bot) \to not'(true, \langle true \rangle, \langle true \rangle) \to false \\ \quad\searrow \qquad\qquad\qquad\qquad\qquad\qquad \nearrow \\ \qquad not'(true, \bot, \langle true \rangle) \to false \end{array}$$

Appending the conditional argument to *not*-terms

Approach: Extend arity of defined symbols.

Example

$$\mathcal{R}_{not} = \left\{ \begin{array}{l} not(x) \to true \Leftarrow x \to^* false \\ not(x) \to false \Leftarrow x \to^* true \end{array} \right\}$$

$$\mathbb{T}_{abh}(\mathcal{R}_{not}) = \left\{ \begin{array}{rl} not'(x, \bot, z) \to & not'(x, \langle x \rangle, z) \\ not'(x, \langle false \rangle, z) \to & true \\ not'(x, z, \bot) \to & not'(x, z, \langle x \rangle) \\ not'(x, z, \langle true \rangle) \to & false \end{array} \right\}$$

$$\begin{array}{ccccccc} not'(true, \bot, \bot) & \to & not'(true, \langle true \rangle, \bot) & \to & not'(true, \langle true \rangle, \langle true \rangle) & \to & false \\ & \searrow & & \nearrow & & & \\ & & not'(true, \bot, \langle true \rangle) & \to & false & & \end{array}$$

Observe, that $\mathbb{T}_{abh}(\mathcal{R}_{not})$ is not confluent: $not'(x, \langle false \rangle, \langle true \rangle)$.

The problem of overlapping CTRSs

Example

$$\mathcal{R} = \left\{ \begin{array}{l} f(g(x)) \to x \Leftarrow x \to^* 0 \\ g(s(0)) \to g(0) \end{array} \right\}$$

$$\mathbb{T}_{abh}(\mathcal{R}) = \left\{ \begin{array}{r} f'(g(x), \bot) \to f'(g(x), \langle x \rangle) \\ f'(g(x), \langle 0 \rangle) \to x \\ g(s(0)) \to g(0) \end{array} \right\}$$

$$\begin{array}{l} f'(g(s(0)), \bot) \to f'(g(s(0)), \langle s(0) \rangle) \to f'(g(0), \langle s(0) \rangle) \not\to \\ \qquad\qquad\qquad \searrow \\ \qquad\qquad\qquad f'(g(0), \bot) \to f'(g(0), \langle 0 \rangle) \to 0 \end{array}$$

- Similar example leads to unsoundness.

Preserve "computational equivalence": [SR06]

Extension of [SR06]

- Reset conditional argument by propagating rewrite steps to outer term positions.
- Rewrite steps have non-local impact

Example

$$\mathcal{R} = \left\{ \begin{array}{l} f(g(x)) \to x \Leftarrow x \to 0 \\ g(s(0)) \to g(0) \end{array} \right\}$$

$$\mathbb{T}_{sr}(\mathcal{R}) = \left\{ \begin{array}{rr} f'(g(x),\bot) \to f'(g(x),\langle\{x\}\rangle) & f'(\{x\},z) \to \{f'(x,\bot)\} \\ f'(g(x),\langle\{0\}\rangle) \to \{x\} & g(\{x\}) \to \{g(x)\} \\ g(s(0)) \to \{g(0)\} & s(\{x\}) \to \{s(x)\} \\ & \{\{x\}\} \to \{x\} \end{array} \right\}$$

$$\begin{aligned} f'(g(s(0)),\bot) &\to f'(g(s(0)),\langle\{s(0)\}\rangle) \to f'(\{g(0)\},\langle\{s(0)\}\rangle) \\ &\to \{f'(g(0),\bot)\} \to \{f'(g(0),\langle\{0\}\rangle)\} \to \{\{0\}\} \to \{0\} \end{aligned}$$

Preserve "computational equivalence": [GmeinerGramlich08]

Refinement of the transformation of [ABH03]

- Encode conditional arguments in (potential) overlaps.
- Rewrite steps have local impact.

Example

$$\mathcal{R} = \left\{ \begin{array}{l} f(g(x)) \to x \Leftarrow x \to 0 \\ g(s(0)) \to g(0) \end{array} \right\}$$

$$\mathbb{T}_{gg}(\mathcal{R}) = \left\{ \begin{array}{r} f(g'(x,\bot)) \to f(g'(x,\langle x \rangle)) \\ f(g'(x,\langle 0 \rangle)) \to x \\ g'(s(0),z) \to g'(0,\bot) \end{array} \right\}$$

$$f(g'(s(0),\bot)) \to f(g'(s(0),\langle s(0) \rangle)) \to f(g'(0,\bot)) \to f(g'(0,\langle 0 \rangle)) \to 0$$

A sorted-list-datastructure

Example

$$\mathcal{R}_f = \{\, f(x, f(y, ys)) \to f(y, f(x, ys)) \Leftarrow x > y \to^* \mathit{true} \quad \}$$

Conditional argument is added to every overlapping term:

- Overlap 1: Term $f(y, ys)$: Add conditional argument 1 (yielding $f''(y, ys, \bot))$
- Overlap 2: Root position: Add conditional argument 2 (yielding $f'(x, f'(y, ys, z_1, z_2), z_3, \bot))$

$$\mathbb{T}_{gg}(\mathcal{R}_f) = \left\{ \begin{array}{l} f'(x, f'(y, ys, \bot, z_2), z_3, z_4) \to f'(x, f'(y, ys, \langle x > y \rangle, z_2), z_3, \langle x > y \rangle) \\ f'(x, f'(y, ys, z_1, z_2), z_3, \bot) \to f'(x, f'(y, ys, \langle x > y \rangle, z_2), z_3, \langle x > y \rangle) \\ f'(x, f'(y, ys, \langle \mathit{true} \rangle, z_2), z_3, \langle \mathit{true} \rangle) \to f'(y, f'(x, ys, \bot, \bot), \bot, \bot) \end{array} \right\}$$

Comparision to $\mathbb{T}_{sr}$

- $\mathbb{T}_{sr}$ resets too often (local changes reset conditional arguments in outer term positions):
  $f'(0, f'(1, f'(\cdots f'(n, f'(n-1, nil, \bot), \langle\{true\}\rangle \cdots), \langle\{false\}\rangle), \langle\{false\}\rangle) \to^*$
  $f'(0, f'(1, f'(\cdots \{f'(n-1, f'(n, nil, \bot), \bot)\} \cdots), \langle\{false\}\rangle), \langle\{false\}\rangle) \to^*$
  $\{f'(0, f'(1, f'(\cdots f'(n-1, f'(n, nil, \bot), \bot) \cdots), \bot), \bot)\}$
- $\mathbb{T}_{gg}$ requires multiple evaluations of conditional arguments

Sorting $f(s^{n-1}(0), f(s^{n-2}(0), \ldots)$ using innermost rewriting ($n = 55$)

- $\mathbb{T}_{sr}$: 62863 (62863)
- $\mathbb{T}_{gg}$: 115335 (59895)

Sorting $f(s^{n-1}(0), f(s^{n-2}(0), \ldots)$ using outermost rewriting ($n = 55$)

- $\mathbb{T}_{sr}$: 166319/820763 (140084/820763)
- $\mathbb{T}_{gg}$: 144538/196955 (35144/76537)

When $\mathbb{T}_{gg}$ does not work

Collapsing systems

- Applying collapsing CTRSs, no conditional argument is set to $\bot$
- Might block other conditional arguments
- Solution: Transform into non-collapsing CTRS.

Non-left-linear systems

- Non-left-linar rule must overlap with conditional rule
- Equal original terms may not be equal in transformed system because of conditional arguments
- Solution: Add an additional non-left-linear rule (?).

Are $\mathbb{T}_{gg}/\mathbb{T}_{abh}/\mathbb{T}_{sr}$ sound whenever U_{seq} is sound?

Example

$$\mathcal{R} = \left\{ \begin{array}{cc} a \to c & g(x,x) \to h(x,x) \\ \nwarrow\!\!\!\!\!\times\!\!\!\!\!\nearrow & \\ b \to d & h(f(d),d) \to A \\ f(x) \to x \Leftarrow x \to^* c & \end{array} \right\}$$

$$g(f(a),f(b)) \not\to^* h(f(d),d) \not\to^* A$$

Are $\mathbb{T}_{gg}/\mathbb{T}_{abh}/\mathbb{T}_{sr}$ sound whenever U_{seq} is sound?

Example

$$\mathcal{R} = \left\{ \begin{array}{cc} a \to c & g(x,x) \to h(x,x) \\ \searrow\!\!\!\!\!\nearrow & \\ b \to d & h(f(d),d) \to A \\ f(x) \to x \Leftarrow x \to^{*} c & \end{array} \right\}$$

$$\mathbb{U}_{seq}(\mathcal{R}) = \left\{ \begin{array}{cc} a \to c & g(x,x) \to h(x,x) \\ \searrow\!\!\!\!\!\nearrow & \\ b \to d & h(f(d),d) \to A \\ f(x) \to U(x,x) & U(c,x) \to x \end{array} \right\}$$

$$\begin{aligned} g(f(a),f(b)) &\to g(U(a,a),U(b,b)) \to^{*} g(U(c,d),U(c,d)) \\ &\to h(U(c,d),U(c,d)) \to h(U(c,d),d) \end{aligned}$$

$\mathbb{U}_{seq}$ is sound because $\mathcal{R}$ is $\mathbb{U}_{seq}$-non-erasing.

Are $\mathbb{T}_{gg}/\mathbb{T}_{abh}/\mathbb{T}_{sr}$ sound whenever U_{seq} is sound?

Example

$$\mathcal{R} = \left\{ \begin{array}{cc} \begin{array}{c} a \to c \\ \nwarrow\!\!\!\!\nearrow \\ b \to d \end{array} & \begin{array}{r} g(x,x) \to h(x,x) \\ \\ h(f(d),d) \to A \end{array} \\ f(x) \to x \Leftarrow x \to^* c & \end{array} \right\}$$

$$\mathbb{T}_{gg}(\mathcal{R}) = \left\{ \begin{array}{cc} \begin{array}{c} a \to c \\ \nwarrow\!\!\!\!\nearrow \\ b \to d \end{array} & \begin{array}{r} g(x,x) \to h(x,x) \\ \\ h(f'(d,z),d) \to A \end{array} \\ f'(x,\bot) \to f'(x,\langle x \rangle) & f'(x,\langle c \rangle) \to x \end{array} \right\}$$

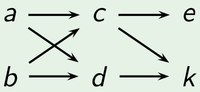
$$\begin{aligned} g(f(a),f(b)) &\to g(f'(a,\langle a \rangle),f'(b,\langle b \rangle)) \to^* g(f'(d,\langle c \rangle),f'(d,\langle c \rangle)) \\ &\to h(f'(d,\langle c \rangle),f'(d,\langle c \rangle)) \to h(f'(d,\langle c \rangle),d) \to A \end{aligned}$$

Summary

- Sequential unraveling and optimized sequential unraveling.
- Optimized unraveling is not sound for joinability.
- Other transformations preserve normal forms
- Soundness of $\mathbb{U}_{seq}$ does not imply soundness of other transformations.

Appendix: Unsoundness Example [Marchiori96]

Example



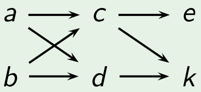
$$f(x) \rightarrow x \Leftarrow x \rightarrow^* e$$

$$h(x,x) \rightarrow g(x,x,f(k))$$

$$g(d,x,x) \rightarrow A$$

Appendix: Unsoundness Example [Marchiori96]

Example



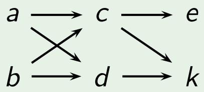
$$h(x,x) \rightarrow g(x,x,f(k))$$

$$g(d,x,x) \rightarrow A$$

$$f(x) \rightarrow x \Leftarrow x \rightarrow^{*} e \quad \Longrightarrow \quad \begin{cases} f(x) \rightarrow U(x,x) \\ U(e,x) \rightarrow x \end{cases}$$

Appendix: Unsoundness Example [Marchiori96]

Example



$$f(x) \to x \Leftarrow x \to^* e \quad \Longrightarrow \quad \begin{cases} f(x) \to U(x,x) \\ U(e,x) \to x \end{cases}$$

$$h(x,x) \to g(x,x,f(k))$$

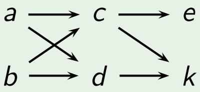
$$g(d,x,x) \to A$$

$$h(f(a),f(b)) \to^* h(U(a,a),U(b,b)) \to^* h(U(c,d),U(c,d))$$

$$\to g(U(c,d),U(c,d),f(k)) \to^* g(d,U(k,k),U(k,k)) \to A$$

Appendix: Unsoundness Example [Marchiori96]

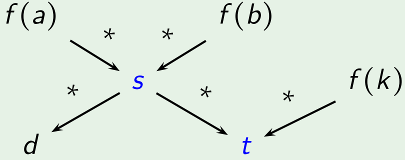
Example



$$h(x,x) \to g(x,x,f(k))$$
$$g(d,x,x) \to A$$

$$f(x) \to x \Leftarrow x \to^* e \quad \Longrightarrow \quad \begin{cases} f(x) \to U(x,x) \\ U(e,x) \to x \end{cases}$$

For the following diagram, there are terms s, t in $\mathbb{U}(\mathcal{R})$ but not in $\mathcal{R}$



- In $\mathbb{U}(\mathcal{R})$, $s = U(c,d), t = U(k,k)$.
- In $\mathcal{R}$, $t = f(k)$, therefore $s \in \{f(c), f(d), f(k)\}$, yet $s \not\to^*_{\mathcal{R}} d$.

Inconsistency in collapsing CTRSs

Example

$$\mathcal{R} = \left\{ \begin{array}{r} i(a,a) \to a \\ g(f(x,b)) \to x \\ f(g(x),y) \to h(x) \Leftarrow i(x,y) \to^* a \end{array} \right\}$$

$$\mathcal{R}' = \left\{ \begin{array}{rl} i(a,a) & \to a \\ g'(f'(x,b,z_1),z_2) & \to x \\ f'(g'(x,\bot),y,z_2) & \to f'(g'(x,\langle i(x,y)\rangle),y,\langle i(x,y)\rangle) \\ f'(g'(x,z_1),y,\bot) & \to f'(g'(x,\langle i(x,y)\rangle),y,\langle i(x,y)\rangle) \\ f'(g'(x,\langle a\rangle),y,\langle a\rangle) & \to h(x) \end{array} \right\}$$

Solution: Wrap terms in $\mathcal{R}$: $\{g(\{f(\{x\},\{b\}))\} \to \{x\}$.

Inconsistency in non-LL CTRSs

Example

$$\mathcal{R} = \left\{ \begin{array}{c} g(0) \to 0 \\ f(x,x) \to A \\ f(g(x),y) \to A \Leftarrow x \to^* 0, y \to^* 0 \end{array} \right\}$$

$$\mathbb{T}_{gg}(\mathcal{R}) = \left\{ \begin{array}{c} g'(0,z) \to 0 \\ f'(x,x,z) \to A \\ f'(g'(x,\bot),y,z_2) \to f'(g'(x,\langle x,y\rangle),y,\langle x,y\rangle) \\ f'(g'(x,z_1),y,\bot) \to f'(g'(x,\langle x,y\rangle),y,\langle x,y\rangle) \\ f'(g'(x,\langle 0,0\rangle),y,\langle 0,0\rangle) \to A \end{array} \right\}$$

Solution (?): Add rule $f(g(x),g(x)) \to A$ to $\mathcal{R}$.