

Rewriting

Part 5. Confluence of Term Rewriting Systems

Temur Kutsia

RISC, JKU Linz



Confluence Is Undecidable

The following problem is undecidable:

Given: A finite TRS R .

Question: Is R confluent or not?



Confluence Is Undecidable

The following problem is undecidable:

Given: A finite TRS R .

Question: Is R confluent or not?

Proof.

Idea:

- ▶ Given a set of identities E such that $\mathcal{V}ar(l) \approx \mathcal{V}ar(r)$ for all $l \approx r \in E$, l and r not being variables.
- ▶ Construct a TRS whose confluence problem is equivalent to the ground word problem for E .
- ▶ Undecidability of the ground word problem for E (see e.g. Example 4.1.4 from the book of Baader and Nipkow) will imply undecidability of the confluence problem.



Confluence Is Undecidable

The following problem is undecidable:

Given: A finite TRS R .

Question: Is R confluent or not?

Proof.

Construction of a TRS:

1. $R := E \cup E^{-1}$ is a confluent TRS.
2. $R_{st} := R \cup \{a \rightarrow s, a \rightarrow t\}$, where s and t are given ground terms and a is a new constant.
3. R_{st} is confluent iff $s \approx_E t$.

Hence, the ground word problem for E reduces to the confluence problem for $R_{s,t}$. □



A Decidable Subcase

Theorem 5.1

For terminating TRSs, confluence is decidable.

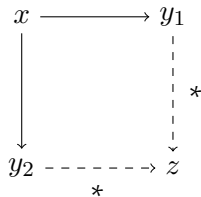
Proof idea:

- ▶ By Newman's lemma, if a TRS is terminating and locally confluent, then it is confluent.
- ▶ To prove the theorem, we need to prove that local confluence is decidable for terminating TRSs.



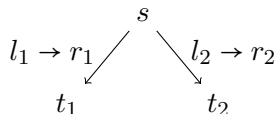
How to Test Local Confluence?

Local confluence:



How to Test Local Confluence?

To test for local confluence of \rightarrow_R , consider reductions:



That means, there are rules $l_1 \rightarrow r_1, l_2 \rightarrow r_2 \in R$, positions $p_1, p_2 \in \mathcal{Pos}(s)$, and substitutions σ_1, σ_2 such that

- ▶ $s|_{p_1} = \sigma_1(l_1)$ and $t_1 = s[\sigma_1(r_1)]_{p_1}$.
- ▶ $s|_{p_2} = \sigma_2(l_2)$ and $t_2 = s[\sigma_2(r_2)]_{p_2}$.

Consider several cases, depending on the relative positions of p_1 and p_2 .

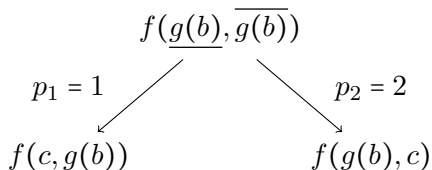


How to Test Local Confluence?

Case 1: p_1 and p_2 are parallel positions.

Example: $R := \{f(a, g(x)) \rightarrow f(x, x), g(b) \rightarrow c\}$

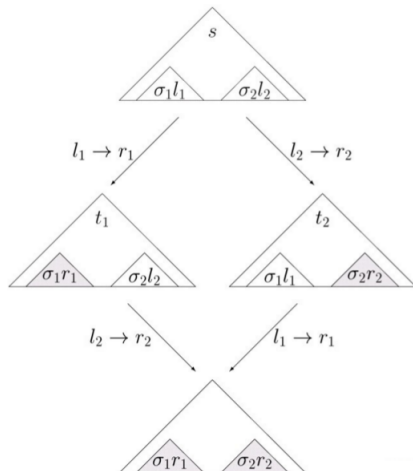
Peak:



How to Test Local Confluence?

Case 1: p_1 and p_2 are parallel positions.

Outcome: The reducts are joinable.



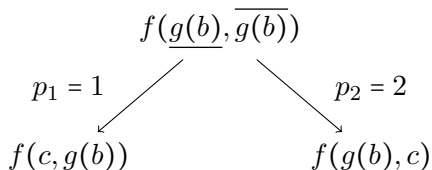
How to Test Local Confluence?

Case 1: p_1 and p_2 are parallel positions.

Outcome: The reducts are joinable.

Example: $R := \{f(a, g(x)) \rightarrow f(x, x), g(b) \rightarrow c\}$

Peak:



Joinability:

$$f(c, \underline{g(b)}) \rightarrow f(c, c)$$

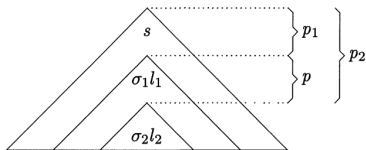
$$f(\underline{g(b)}, c) \rightarrow f(c, c)$$



How to Test Local Confluence?

Case 2: One position is a prefix of another.

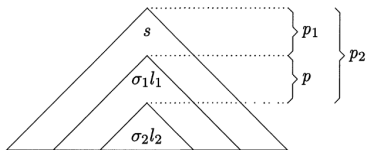
Say, p_1 is a prefix of p_2 : $p_2 = p_1p$ for some p .



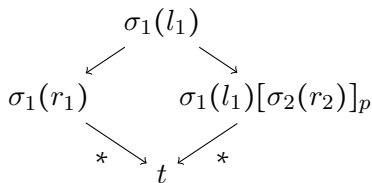
How to Test Local Confluence?

Case 2: One position is a prefix of another.

Say, p_1 is a prefix of p_2 : $p_2 = p_1p$ for some p .



We restrict our attention to $\sigma_1(l_1)$, because



implies $s[\sigma_1(r_1)]_{p_1} \xrightarrow{*} s[t] \xleftarrow{*} s[\sigma_1(l_1)[\sigma_2(r_2)]_p]_{p_1} = s[\sigma_2(r_2)]_{p_2}$.

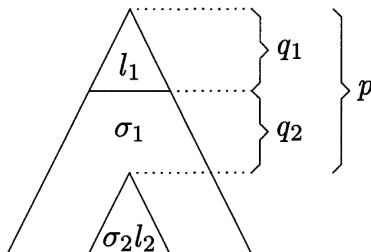


How to Test Local Confluence?

Case 2.1: The redex $\sigma_2(l_2)$ does not overlap with l_1 itself, but is contained in σ_1 .

$p = q_1q_2$ such that q_1 is a variable position in l_1 .

$\sigma_1(l_1)$ has the form:



Non-critical overlap.



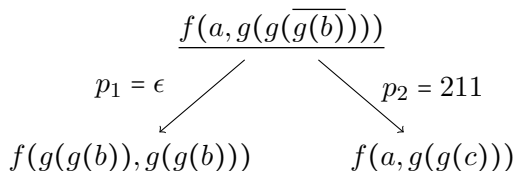
How to Test Local Confluence?

Case 2.1: The redex $\sigma_2(l_2)$ does not overlap with l_1 itself, but is contained in σ_1 .

$p = q_1q_2$ such that q_1 is a variable position in l_1 .

Example: $R := \{f(a, g(x)) \rightarrow f(x, x), g(b) \rightarrow c\}$

Peak:



$l_1 = f(a, g(x))$, $\sigma_1 = \{x \mapsto g(g(b))\}$, $l_2 = g(b)$,
 $\sigma_2 = \varepsilon$.

$p = 211$, $q_1 = 21$, $q_2 = 1$.



How to Test Local Confluence?

Case 2.1: The redex $\sigma_2(l_2)$ does not overlap with l_1 itself, but is contained in σ_1 .

$p = q_1q_2$ such that q_1 is a variable position in l_1 .

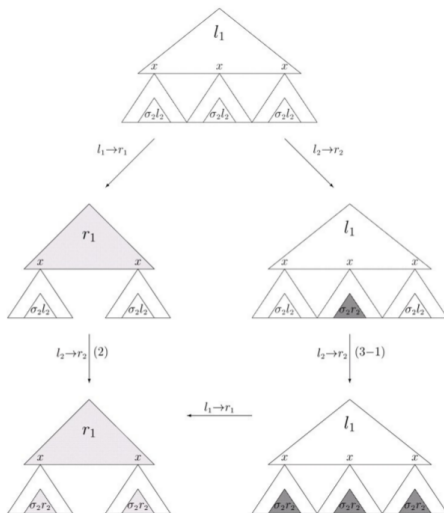
Outcome: The reducts are joinable.

The analysis is complicated by the fact that $x = l_1|_{q_1}$ may occur repeatedly both in l_1 and r_1 .



How to Test Local Confluence?

Case 2.1: Instance: x appears three times in l_1 and twice in r_1 .



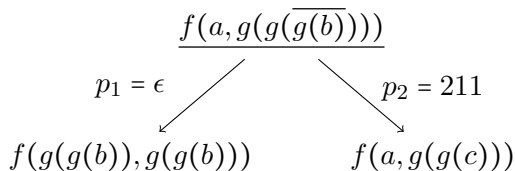
How to Test Local Confluence?

Case 2.1: The redex $\sigma_2(l_2)$ does not overlap with l_1 itself, but is contained in σ_1 .

$p = q_1 q_2$ such that q_1 is a variable position in l_1 .

Example: $R := \{f(a, g(x)) \rightarrow f(x, x), g(b) \rightarrow c\}$

Peak:



The reducts are joinable.

$f(g(g(b)), g(g(b))) \xrightarrow{2} f(g(c), g(c))$.

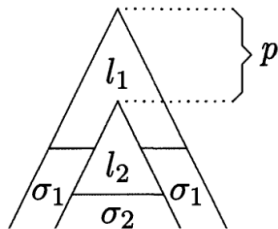
$f(a, g(g(c))) \rightarrow f(g(c), g(c))$.



How to Test Local Confluence?

Case 2.2: Two left-hand sides l_1 and l_2 overlap.
 $p \in \mathcal{Pos}(l_1)$, $l_1|_p$ is not a variable, and
 $\sigma_1(l_1|_p) = \sigma_2(l_2)$.

$\sigma_1(l_1)$ has the form:



Critical overlap.



How to Test Local Confluence?

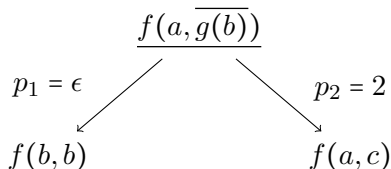
Case 2.2: Two left-hand sides l_1 and l_2 overlap.

$p \in \mathcal{Pos}(l_1)$, $l_1|_p$ is not a variable, and

$\sigma_1(l_1|_p) = \sigma_2(l_2)$.

In the case of critical overlap, local confluence need not hold.

Example: $R := \{f(a, g(x)) \rightarrow f(x, x), g(b) \rightarrow c\}$



$l_1 = f(a, g(x))$, $\sigma_1 = \{x \mapsto b\}$, $l_2 = g(b)$, $\sigma_2 = \varepsilon$.

$p = 2$.

How to Test Local Confluence?

Case 2.2: Two left-hand sides l_1 and l_2 overlap.
 $p \in \mathcal{Pos}(l_1)$, $l_1|_p$ is not a variable, and
 $\sigma_1(l_1|_p) = \sigma_2(l_2)$.

Problem: Critical overlaps must be checked for local confluence. How to do that?

Answer: It is enough to check **finitely many critical pairs**.



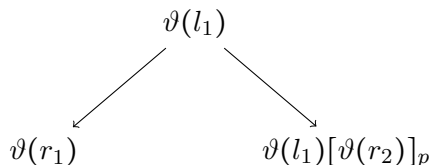
How to Test Local Confluence?

Definition 5.1

Let

- ▶ $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ be two rules which do not share variables,
- ▶ $p \in \mathcal{Pos}(l_1)$ be a position such that $l_1|_p$ is not a variable, and
- ▶ ϑ be an mgu of $l_1|_p$ and l_2

Then the pair $\langle \vartheta(r_1), \vartheta(l_1)[\vartheta(r_2)]_p \rangle$ is called a **critical pair**.



How to Test Local Confluence?

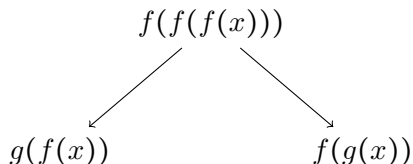
- ▶ The critical pairs of a TRS R are the critical pairs between any of two of its renamed rules and are denoted by $CP(R)$.
- ▶ Includes overlaps of a rule with a renamed copy of itself.



How to Test Local Confluence?

Example 5.1

- ▶ Let $R := \{f(f(x)) \rightarrow g(x)\}$.
- ▶ Take a critical pair between the rule and its renamed copy, $f(f(x)) \rightarrow g(x)$ and $f(f(y)) \rightarrow g(y)$



- ▶ The terms in the critical pair, $g(f(x))$ and $f(g(x))$, are not joinable.
- ▶ R is not locally confluent.



How to Test Local Confluence?

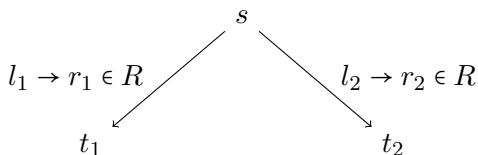
- ▶ Hence, local confluence test reduces to checking joinability of critical pairs.
- ▶ The analysis of the cases on the previous slides leads to the **Critical Pair Lemma**.



How to Test Local Confluence?

Lemma 5.1 (Critical Pair Lemma)

If R is a TRS and



then $t_1 \downarrow_R t_2$, or $t_1 = s[u_1]_{p_1}$ and $t_2 = s[u_2]_{p_2}$ for some p_1, p_2 , where $\langle u_1, u_2 \rangle$ or $\langle u_2, u_1 \rangle$ is an instance of a critical pair of R .

Proof.

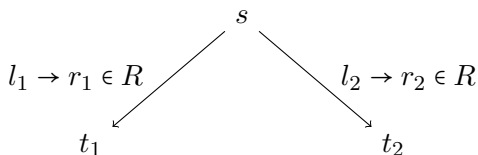
- ▶ When there is no overlap or a non-critical overlap, then $t_1 \downarrow_R t_2$.
- ▶ When there is a critical overlap, then $s|_{p_1} = \sigma(l_1)$ and $\sigma(l_1|_p) = \sigma(l_2)$.



How to Test Local Confluence?

Lemma 5.1 (Critical Pair Lemma)

If R is a TRS and



then $t_1 \downarrow_R t_2$, or $t_1 = s[u_1]_{p_1}$ and $t_2 = s[u_2]_{p_2}$ for some p_1, p_2 , where $\langle u_1, u_2 \rangle$ or $\langle u_2, u_1 \rangle$ is an instance of a critical pair of R .

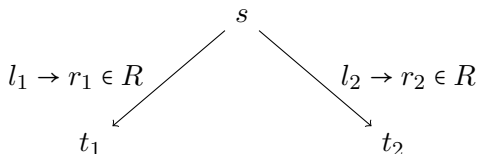
Proof (cont.)

- ▶ Hence, σ unifies $l_1|_p$ and l_2 and, therefore, is an instance of their mgu ϑ .
- ▶ Therefore, $\langle \sigma(r_1), \sigma(l_1)[\sigma(r_2)]_p \rangle$ is an instance of the critical pair $\langle \vartheta(r_1), \vartheta(l_1)[\vartheta(r_2)]_p \rangle$

How to Test Local Confluence?

Lemma 5.1 (Critical Pair Lemma)

If R is a TRS and



then $t_1 \downarrow_R t_2$, or $t_1 = s[u_1]_{p_1}$ and $t_2 = s[u_2]_{p_2}$ for some p_1, p_2 , where $\langle u_1, u_2 \rangle$ or $\langle u_2, u_1 \rangle$ is an instance of a critical pair of R .

Proof (cont.)

- ▶ $t_1 = s[\sigma(r_1)]_{p_1}$, $t_2 = s[\sigma(l_1)[\sigma(r_2)]]_{p_1}$, $p_2 = p_1 p$.



How to Test Local Confluence?

Theorem 5.2 (Critical Pair Theorem)

A TRS is locally confluent iff all its critical pairs are joinable.

Proof.

(\Leftarrow) Using the Critical Pair Lemma: Given $t_i = s[u_i]_p$, $i = 1, 2$, where $\langle u_1, u_2 \rangle$ (wlog) is an instance of some critical pair $\langle v_1, v_2 \rangle$ under a substitution φ , then $v_i \xrightarrow{*} t$ for some term t implies $u_i \xrightarrow{*} \varphi(t)$ and, hence, $t_i \xrightarrow{*} s[\varphi(t)]_p$, $i = 1, 2$.



How to Test Local Confluence?

Theorem 5.2 (Critical Pair Theorem)

A TRS is locally confluent iff all its critical pairs are joinable.

Proof.

- (\Leftarrow) Using the Critical Pair Lemma: Given $t_i = s[u_i]_p$, $i = 1, 2$, where $\langle u_1, u_2 \rangle$ (wlog) is an instance of some critical pair $\langle v_1, v_2 \rangle$ under a substitution φ , then $v_i \xrightarrow{*} t$ for some term t implies $u_i \xrightarrow{*} \varphi(t)$ and, hence, $t_i \xrightarrow{*} s[\varphi(t)]_p$, $i = 1, 2$.
- (\Rightarrow) Every critical pair is the product of a fork $\vartheta(r_1) \leftarrow \vartheta(l_1) \rightarrow \vartheta(l_1)[\vartheta(r_2)]_p$. Joinability follows from local confluence.



How to Test Local Confluence?

Theorem 5.2 (Critical Pair Theorem)

A TRS is locally confluent iff all its critical pairs are joinable.

Corollary 5.1

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



How to Test Local Confluence?

- ▶ The problem of testing local confluence reduces to critical pair joinability test.
- ▶ For **terminating** TRSs, the problem whether two terms are joinable can be decided.
- ▶ For **finite** TRSs, the number of critical pairs is finite.
- ▶ Hence, for terminating and finite TRSs **local confluence is decidable**.
- ▶ Therefore, for terminating and finite TRSs **confluence is decidable**.



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:

- ▶ For each pair of rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ (there are $|R|^2$ of them) and for every $p \in \mathcal{Pos}(l_1)$ with a nonvariable $l_1|_p$ (there are at most $|l_1|$ of them) try to generate critical pairs.



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:

- ▶ For each pair of rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ (there are $|R|^2$ of them) and for every $p \in \mathcal{Pos}(l_1)$ with a nonvariable $l_1|_p$ (there are at most $|l_1|$ of them) try to generate critical pairs.
- ▶ It involves unification of $l_1|_p$ and l_2 (decidable, unitary).



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:

- ▶ For each pair of rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ (there are $|R|^2$ of them) and for every $p \in \mathcal{Pos}(l_1)$ with a nonvariable $l_1|_p$ (there are at most $|l_1|$ of them) try to generate critical pairs.
- ▶ It involves unification of $l_1|_p$ and l_2 (decidable, unitary).
- ▶ Result: finitely many critical pairs.



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:

- ▶ For each pair of rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ (there are $|R|^2$ of them) and for every $p \in \mathcal{Pos}(l_1)$ with a nonvariable $l_1|_p$ (there are at most $|l_1|$ of them) try to generate critical pairs.
- ▶ It involves unification of $l_1|_p$ and l_2 (decidable, unitary).
- ▶ Result: finitely many critical pairs.
- ▶ For each critical pair $\langle u_1, u_2 \rangle$, reduce u_i , to some R -normal form \hat{u}_i , $i = 1, 2$.



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:

- ▶ For each pair of rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ (there are $|R|^2$ of them) and for every $p \in \mathcal{Pos}(l_1)$ with a nonvariable $l_1|_p$ (there are at most $|l_1|$ of them) try to generate critical pairs.
- ▶ It involves unification of $l_1|_p$ and l_2 (decidable, unitary).
- ▶ Result: finitely many critical pairs.
- ▶ For each critical pair $\langle u_1, u_2 \rangle$, reduce u_i , to some R -normal form \hat{u}_i , $i = 1, 2$.
- ▶ If $\hat{u}_1 = \hat{u}_2$ for all such pairs, R is confluent (Corollary 5.1).



Deciding (Local) Confluence for Terminating Finite TRSs

Let R be a terminating finite TRS.

Decision procedure:

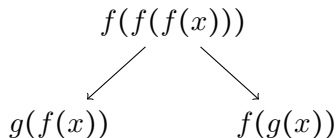
- ▶ For each pair of rules $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ (there are $|R|^2$ of them) and for every $p \in \mathcal{Pos}(l_1)$ with a nonvariable $l_1|_p$ (there are at most $|l_1|$ of them) try to generate critical pairs.
- ▶ It involves unification of $l_1|_p$ and l_2 (decidable, unitary).
- ▶ Result: finitely many critical pairs.
- ▶ For each critical pair $\langle u_1, u_2 \rangle$, reduce u_i , to some R -normal form \hat{u}_i , $i = 1, 2$.
- ▶ If $\hat{u}_1 = \hat{u}_2$ for all such pairs, R is confluent (Corollary 5.1).
- ▶ If $\hat{u}_1 \neq \hat{u}_2$ for such a pair, we have a non-confluent situation:
$$\hat{u}_1 \xleftarrow{*} u_1 \leftarrow u \rightarrow u_2 \xrightarrow{*} \hat{u}_2.$$



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.2

Recall the TRS $\{f(f(x)) \rightarrow g(x)\}$, which is not locally confluent. The only critical pair $\langle g(f(x)), f(g(x)) \rangle$ is not joinable.



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.2

Recall the TRS $\{f(f(x)) \rightarrow g(x)\}$, which is not locally confluent.
The only critical pair $\langle g(f(x)), f(g(x)) \rangle$ is not joinable.



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.2

Recall the TRS $\{f(f(x)) \rightarrow g(x)\}$, which is not locally confluent.
The only critical pair $\langle g(f(x)), f(g(x)) \rangle$ is not joinable.

This example illustrates that the two conditions in the definition of the critical pairs are necessary:



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.2

Recall the TRS $\{f(f(x)) \rightarrow g(x)\}$, which is not locally confluent. The only critical pair $\langle g(f(x)), f(g(x)) \rangle$ is not joinable.

This example illustrates that the two conditions in the definition of the critical pairs are necessary:

- ▶ Rules are to be renamed. Otherwise $f(f(x))$ and $f(x)$ are not unifiable.



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.2

Recall the TRS $\{f(f(x)) \rightarrow g(x)\}$, which is not locally confluent. The only critical pair $\langle g(f(x)), f(g(x)) \rangle$ is not joinable.

This example illustrates that the two conditions in the definition of the critical pairs are necessary:

- ▶ Rules are to be renamed. Otherwise $f(f(x))$ and $f(x)$ are not unifiable.
- ▶ The critical pair of a rule and (a renamed copy of) itself has to be taken into account. Otherwise all one-rule systems would appear to be locally-confluent.



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.2

Recall the TRS $\{f(f(x)) \rightarrow g(x)\}$, which is not locally confluent. The only critical pair $\langle g(f(x)), f(g(x)) \rangle$ is not joinable.

This example illustrates that the two conditions in the definition of the critical pairs are necessary:

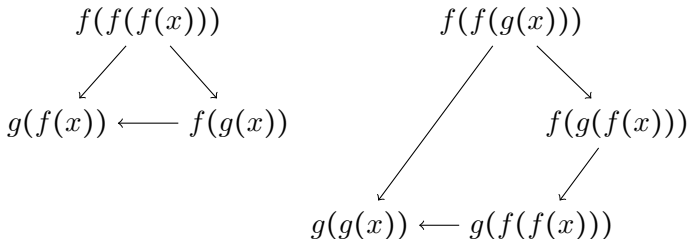
- ▶ Rules are to be renamed. Otherwise $f(f(x))$ and $f(x)$ are not unifiable.
- ▶ The critical pair of a rule and (a renamed copy of) itself has to be taken into account. Otherwise all one-rule systems would appear to be locally-confluent.
- ▶ Critical pairs can be helpful lemmas: $g(f(x)) \approx_R f(g(x))$ is an interesting consequence of $f(f(x)) \rightarrow_R g(x)$ which may not be apparent at first sight.



Deciding (Local) Confluence for Terminating Finite TRSs

Example 5.3

The TRS $\{f(f(x)) \rightarrow g(x), f(g(x)) \rightarrow g(f(x))\}$ is locally confluent. Both critical pairs are joinable:



Since the TRS is also terminating (use LPO with $f > g$), it is also confluent.



Deciding (Local) Confluence for Terminating Finite TRSs

- ▶ Because critical pairs are equational consequences, adding a critical pair as a new rewrite rule does not change the induced equality.
- ▶ If R is a TRS and R' is obtained from R by adding a critical pair as a new rule, then $\approx_R = \approx_{R'}$.
- ▶ The idea of adding a critical pair as a new rule is called “completion”.

