

Termination of R and Its Certification

```

Section TerminationCriterion.
  Variable R : rules.

  Lemma m4_termination :
    let R_gt := partition part_
    let R_ge := partition part_
    monotone I succ ->
    and R_ge = nil ->
    WF (red (and R_gt)) ->
    WF (red R).

writing
ion

End TerminationCriterion.

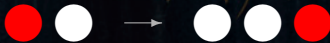
```

25 September 2008

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ▶ ↺ 🔍 ↻

Termination of Rewriting and Its Certification

Rewriting: what is it all about...



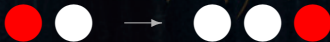
```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  intros. apply WF_incl with (ma  
    apply ma_relative_termination  
    simpl. apply WF_incl with (red  
    apply red_mod_empty_incl_red.  
  Qed.  
  
End TerminationCriterion.
```

Theorem 2.5

Let R, R', S, S' be TRS over Σ .
Let $(A, [], \succ, \succeq)$ be an ext. monotone
1) $[L, \alpha] \succeq [r, \alpha]$ for every rule $L \rightarrow r$ in R
2) $[L, \alpha] \succ [r, \alpha]$ for every rule $L \rightarrow r$ in R'
Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies $SN(\rightarrow_{R'} / \rightarrow_S)$

Termination of Rewriting and Its Certification

Rewriting: what is it all about...



```

Section TerminationCriterion.
Variable R : rules.

Lemma ma_termination :
  let R_gt := partition part_succ
  let R_ge := partition part_succ
  monotone 1 succ ->
  and R_ge = nil ->
  WP (red (end R_gt)) ->
  WP (red R)..

Proof.
  intros. apply WP_incl with (ma_
  apply ma_relative_termination
  simpl. apply WP_incl with (red
  apply red_mod_empty_incl_red.
Qed.

End TerminationCriterion.
  
```

End TerminationCriterion.

Theorem 2.5

Let R, R', S, S' be TRS over

Let $(A, [], >, \geq)$ be an ext. monoton

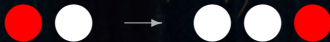
1) $[l, \alpha] \geq [r, \alpha]$ for every rule

2) $[l, \alpha] > [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies SN

Termination of Rewriting and Its Certification

Rewriting: what is it all about...



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  intros. apply WF_incl with (ma  
  apply ma_relative_termination  
  simpl. apply WF_incl with (red  
  apply red_mod_empty_incl_red.  
Qed.  
  
End TerminationCriterion.
```

Theorem 2.5

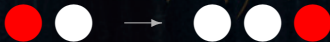
Let R, R', S, S' be TRS over Σ .
Let $(A, [], \succ, \succeq)$ be an ext. monotone

- 1) $[l, \alpha] \succeq [r, \alpha]$ for every rule
- 2) $[l, \alpha] \succ [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies $SN(\rightarrow_{R'})$

Termination of Rewriting and Its Certification

Rewriting: what is it all about...



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  intros. apply WF_incl with (ma_termination).  
  apply ma_relative_termination.  
  simpl. apply WF_incl with (red_mod_empty_incl_red).  
  Qed.  
End TerminationCriterion.
```

End TerminationCriterion.

Theorem 2.5

Let R, R', S, S' be TRS over

Let $(A, [], >, \geq)$ be an ext. monoton

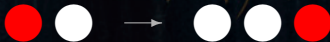
1) $[l, \alpha] \geq [r, \alpha]$ for every rule

2) $[l, \alpha] > [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies $SN(\rightarrow_{R'})$

Termination of Rewriting and Its Certification

Rewriting: what is it all about...



```

Section TerminationCriterion.
Variable R : rules.

Lemma ma_termination :
  let R_gt := partition part_succ
  let R_ge := partition part_succ
  monotone 1 succ ->
  and R_ge = nil ->
  WF (red (end R_gt)) ->
  WF (red R)..

Proof.
  intros. apply WF_incl with (ma_
  apply ma_relative_termination
  simpl. apply WF_incl with (red
  apply red_mod_empty_incl_red.
Qed.

End TerminationCriterion.
  
```

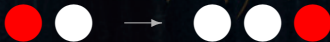
End TerminationCriterion.

Theorem 2.5

Let R, R', S, S' be TRS over Σ .
 Let $(A, [], >, \geq)$ be an ext. monoton.
 1) $[L, \alpha] \geq [R, \alpha]$ for every rule $L \rightarrow R$ in R
 2) $[L, \alpha] > [R, \alpha]$ for every rule $L \rightarrow R$ in R'
 Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies $SN(\rightarrow_{R'})$

Termination of Rewriting and Its Certification

Rewriting: what is it all about...



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  intros. apply WF_incl with (ma  
    apply ma_relative_termination  
    simpl. apply WF_incl with (red  
      apply red_mod_empty_incl_red.  
Qed.  
  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  admit.  
End TerminationCriterion.
```

Theorem 2.5

Let R, R', S, S' be TRS over

Let $(A, [], \succ, \succeq)$ be an ext. mono

1) $[L, \alpha] \succeq [r, \alpha]$ for every rule

2) $[L, \alpha] \succ [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies SN

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned}0 + y &= y \\ s(x) + y &= s(x + y)\end{aligned}$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
Proof.  
  apply ma_termination with (R_gt := R_ge).  
  apply ma_termination with (R_ge := empty_incl_red).  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned}0 + y &\rightarrow y \\ s(x) + y &\rightarrow s(x + y)\end{aligned}$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  apply ma_termination with (R_gt := R_ge).  
  apply ma_termination.  
  apply WF_incl with (red := red).  
  apply empty_incl_red.  
  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned}0 + y &\rightarrow y \\ s(x) + y &\rightarrow s(x + y)\end{aligned}$$

Now let us do some math... how about $2 + 2$?

$$s(s(0)) + s(s(0)) \rightarrow$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
Proof.  
  apply ma_relative_termination.  
  simpl. apply WF_incl with (re  
  empty_incl_red.  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned}0 + y &\rightarrow y \\ s(x) + y &\rightarrow s(x + y)\end{aligned}$$

Now let us do some math... how about $2 + 2$?

$$s(s(0)) + s(s(0)) \rightarrow$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
Proof.  
  apply ma_relative_termination.  
  simpl. apply WF_incl with (red  
  empty_incl_red.  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned}0 + y &\rightarrow y \\ s(x) + y &\rightarrow s(x + y)\end{aligned}$$

Now let us do some math... how about $2 + 2$?

$$s(s(0)) + s(s(0)) \rightarrow s(s(0) + s(s(0))) \rightarrow$$

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned} 0 + y &\rightarrow y \\ s(x) + y &\rightarrow s(x + y) \end{aligned}$$

Now let us do some math... how about $2 + 2$?

$$\begin{aligned} s(s(0)) + s(s(0)) &\rightarrow s(s(0) + s(s(0))) \rightarrow \\ s(s(0 + s(s(0)))) &\rightarrow \end{aligned}$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  apply ma_relative_termination.  
  simpl. apply WF_incl with (red  
  empty_incl_red.  
  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$$\begin{aligned}0 + y &\rightarrow y \\ s(x) + y &\rightarrow s(x + y)\end{aligned}$$

Now let us do some math... how about $2 + 2$?

$$\begin{aligned}s(s(0)) + s(s(0)) &\rightarrow s(s(0) + s(s(0))) \rightarrow \\ s(s(0 + s(s(0)))) &\rightarrow s(s(s(s(0))))\end{aligned}$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
Proof.  
  apply ma_relative_termination.  
  simpl. apply WF_incl with (red  
  empty_incl_red.  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

Term rewriting is a specialization of rewriting where the objects under consideration are terms.

Example

$0 + y \rightarrow y$
 $s(x) + y \rightarrow s(x + y)$

Term rewriting is a model of computation.

Now let us do some math... how about $0 + 2$?

$$\begin{aligned} s(s(0)) + s(s(0)) &\rightarrow s(s(0) + s(s(0))) \rightarrow s(s(0 + s(s(0)))) \rightarrow s(s(0 + s(s(0 + s(s(0)))))) \end{aligned}$$

Termination of Rewriting and Its Certification

How difficult is it to prove termination?

```

Section TerminationCriterion.
  Variable R : rules.

  Lemma ma_termination :
    let R_gt := partition par
    let R_ge := partition par
    monotone 1 succ ->
    and R_ge = nil ->
    WF (red (and R_gt)) ->
    WF (red R).

  Proof.
    apply WF_incl with
    apply ma_relative_termina
    simpl. apply WF_incl with
    apply red_mod_empty_incl
    Qed.

End TerminationCriterion.

```

How difficult is it to prove term

Termination of Rewriting and Its Certification

How difficult is it to prove termination?

Example

$$\begin{aligned} h\ 1\ 1 &\rightarrow 1\ h \\ 1\ 1\ h\ b &\rightarrow 1\ 1\ s\ b \\ 1\ s &\rightarrow s\ 1 \\ b\ s &\rightarrow b\ h \\ h\ 1\ b &\rightarrow t\ 1\ 1\ b \\ 1\ t &\rightarrow t\ 1\ 1\ 1 \\ b\ t &\rightarrow b\ h \end{aligned}$$

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
Qed.  
  
End TerminationCriterion.
```


Termination of Rewriting and Its Certification

How difficult is it to prove termination?

Example

Collatz conjecture:

The function $f(n) = \begin{cases} n/2 & \text{if } n \text{ even} \\ 3n + 1 & \text{if } n \text{ odd} \end{cases}$

converges to 1.

$h \ 1 \ b \rightarrow t \ 1 \ 1 \ b$

$1 \ t \rightarrow t \ 1 \ 1 \ 1$

$b \ t \rightarrow b \ h$

Section TerminationCriterion.

Variable R : rules.

```
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).
```

```
Proof. apply WF_incl with (s  
  apply ma_relative_termination  
  simpl. apply WF_incl with (re  
  apply red_mod_empty_incl_red.
```

Criterion.

Termination of Rewriting and Its Certification

How difficult is it to prove termination?

Example

Collatz conjecture:

The function $f(n) = \begin{cases} n/2 & \text{if } n \text{ even} \\ 3n + 1 & \text{if } n \text{ odd} \end{cases}$

converges to 1.

Open problem!

Termination of Rewriting and Its Certification

How difficult is it to prove termination?

Example

Collatz conjecture:

The function $f(n) = \begin{cases} n/2 & \text{if } n \text{ even} \\ 3n + 1 & \text{if } n \text{ odd} \end{cases}$

converges to 1.

Open problem!

Termination is undecidable

Termination of Rewriting and Its Certification

1979 Polynomial interpretations

1982 Recursive path order

1995 Semantic labeling

2000 Dependency pairs

2006 Matrix interpretations



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  intros. apply WF_incl with (ma  
  apply ma_relative_termination  
  simpl. apply WF_incl with (re  
  apply red_mod_empty_incl_red.  
Qed.  
  
End TerminationCriterion.
```

Termination of Rewriting and Its Certification

1979 Polynomial interpretations

1982 Recursive path order

1995 Semantic labeling

2000 Dependency pairs

2006 Matrix interpretations



Termination of Rewriting and Its Certification

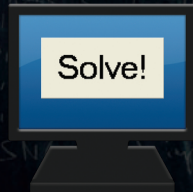
1979 Polynomial interpretations

1982 Recursive path order

1995 Semantic labeling

2000 Dependency pairs

2006 Matrix interpretations



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_gt  
  let R_ge := partition part_ge  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WP (red (end R_gt)) ->  
  WP (red R).  
Proof.  
  intro.  
  apply ma_termination.  
  with (re  
  with (re  
  incl_red.  
end ma_termination.
```

Termination of Rewriting and Its Certification

1979 Polynomial interpretations

1982 Recursive path order

1995 Semantic labeling

2000 Dependency pairs

2006 Matrix interpretations



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WP (red (end R_gt)) ->  
  WP (red R).  
  
Proof.  
  intro.  
  apply (ma_termination with (ma_termination with (red_incl_red).  
OnCriterion.
```

Termination of Rewriting and Its Certification



Section TerminationCriterion.

Variable R : rules.

```
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).
```

Proof.

```
  intros. apply WF_incl with (ma_termination  
    apply ma_relative_termination  
    simpl. apply WF_incl with (red  
    apply red_mod_empty_incl_red.  
  Qed.
```

End TerminationCriterion.

Theorem 2.5.

Let R, R', S, S' be TRS over

Let $(A, [], \succ, \succeq)$ be an ext. monoton

1) $[L, \alpha] \succeq [r, \alpha]$ for every rule

2) $[L, \alpha] \succ [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies SN

Termination of Rewriting and Its Certification



Section TerminationCriterion.

Variable R : rules.

```
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WP (red (end R_gt)) ->  
  WP (red R).
```

Proof.

```
intros. apply WP_incl with (ma_termination).  
apply ma_relative_termination.  
simpl. apply WP_incl with (red).  
apply red_mod_empty_incl_red.  
Qed.
```

End TerminationCriterion.

Theorem 2.5.

Let R, R', S, S' be TRS over

Let $(A, [], \succ, \succeq)$ be an ext. monoton

1) $[L, \alpha] \succeq [r, \alpha]$ for every rule

2) $[L, \alpha] \succ [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies SN

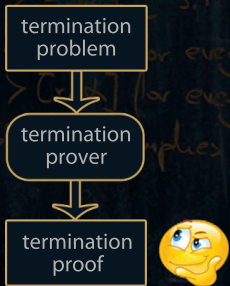
Termination of Rewriting and Its Certification



How can we ensure reliability?

```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WF (red (end R_gt)) ->  
  WF (red R).  
  
Proof.  
  intros. apply WF_incl with (ma_termination).  
  apply ma_relative_termination.  
  simpl. apply WF_incl with (red).  
  apply red_mod_empty_incl_red.  
  Qed.  
End TerminationCriterion.
```


Termination of Rewriting and Its Certification



termination
certified termination



Section TerminationCriterion.

Variable R : rules.

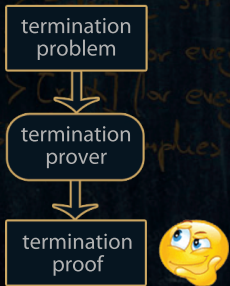
```
Lemma ma_termination :  
  let R_gt := partition part_s  
  let R_ge := partition part_s  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WP (red (end R_gt)) ->  
  WP (red R).
```

Proof.

WP incl with (a
ive_termination
WP incl with (re
empty_incl_red.

End TerminationCriterion.

Termination of Rewriting and Its Certification



termination
certified termination



```
Section TerminationCriterion.  
Variable R : rules.  
  
Lemma ma_termination :  
  let R_gt := partition part_succ  
  let R_ge := partition part_succ  
  monotone 1 succ ->  
  and R_ge = nil ->  
  WP (red (end R_gt)) ->  
  WP (red R).  
  
Proof.  
  WP_incl with (ma_termination)  
  WP_incl with (red_empty_incl_red).  
End TerminationCriterion.
```

▶ New methods and refinement methods for proving termination

- ▶ New methods and refinements to existing methods for proving termination of rewriting.

existing

Contributions

Let R, R' be TRSs over a signature Σ
Let $(A, [], \succ, \succeq)$ be an extended, weakly
monotone Σ -algebra s.t.

$\forall \alpha \ [l, \alpha] \succeq [r, \alpha]$ for every rule $l \rightarrow r$ in R
 $\forall \alpha \ [l, \alpha] \succeq [r, \alpha]$ for every rule $l \rightarrow r$ in R'

- ▶ New methods and refinements to existing methods for proving termination of rewriting.
- ▶ Contributions adding to the progress in the area of certification of termination proofs.

Section TerminationCriterion.

Variable R : rules.

Lemma ma_termination :

```
let R_gt := partition part_succ  
let R_ge := partition part_succ  
monotone 1 succ ->  
and R_ge = nil ->  
WF (red (end R_gt)) ->  
WF (red R).
```

Proof.

```
intros. apply WF_incl with (ma  
apply ma_relative_termination  
simpl. apply WF_incl with (red  
apply red_mod_empty_incl_red.  
Qed.
```

Qed.

Section TerminationCriterion.

Let R, R', S, S' be TRSs over

Let $(A, [], \succ, \succeq)$ be an ext. monotone

1) $[l, \alpha] \succeq [r, \alpha]$ for every rule

2) $[l, \alpha] \succeq [r, \alpha]$ for every rule

Then $SN(\rightarrow_R / \rightarrow_{S'})$ implies SN

