Reducing Rewrite Properties to Properties on Ground Terms*

Alexander Lochmann

August 7, 2022

Abstract

This AFP entry relates important rewriting properties between the set of terms and the set of ground terms induced by a given signature. The properties considered are confluence, strong/local confluence, the normal form property, unique normal forms with respect to reduction and conversion, commutation, conversion equivalence, and normalization equivalence.

Contents

1 Introduction **3**

2 Preliminaries **4**
- 2.1 Additional operations on terms and positions 4
  - 2.1.1 Linearity 4
  - 2.1.2 Positions induced by contexts, by variables and by given subterms 4
  - 2.1.3 Replacing functions symbols that aren't specified in the signature by variables 4
  - 2.1.4 Replace term at a given position in contexts 5
  - 2.1.5 Multihole context closure of a term relation as inductive set 5
- 2.2 Destruction and introduction of *all-ctxt-closed* 5
- 2.3 Lemmas for *poss* and ordering of positions 5
- 2.4 Lemmas for (|-) and *replace-term-at* 6
- 2.5 *term-to-sig* invariants and distributions 8
- 2.6 Misc 8

*Supported by FWF (Austrian Science Fund) projects P30301.

3 Rewriting 10
3.1 Basic rewrite definitions 10
3.1.1 Rewrite steps with implicit signature declaration (encoded in the type) 10
3.1.2 Restrict relations to terms induced by a given signature 11
3.1.3 Rewriting under a given signature/restricted to ground terms 11
3.1.4 Rewriting sequences involving a root step 11
3.2 Monotonicity laws 11
3.3 Introduction, elimination, and destruction rules for *sig-step*, *rstep*, *rrstep*, *srrstep*, and *srstep* 12
3.3.1 Transitive and relfexive closure distribution over *sig-step* 13
3.3.2 Distributivity laws 14
3.4 Substitution closure of *srstep* 15
3.5 Context closure of *srstep* 16
3.6 Removing/Replacing constants in a rewrite sequence that do not appear in the rewrite system 17
3.6.1 Removal lemma applied to various rewrite relations . 18

4 Confluence related rewriting properties 20
4.1 Confluence related ARS properties 20
4.2 Signature closure of relation to model multihole context closure 20
4.3 Specific results about rewriting under a linear variable-separated system 26
4.4 Specific results about rewriting under a ground system 31
4.5 funas 32

5 Reducing Rewrite Properties to Properties on Ground Terms over Left-Linear Right-Ground Systems 33

6 LLRG results 33
6.1 Specialized for monadic signature 34

7 Reducing Rewrite Properties to Properties on Ground Terms over Ground Systems 35

8 Reducing Rewrite Properties to Properties on Ground Terms over Linear Variable-Separated Systems 37

9 Linear-variable separated results 38
9.1 Specialized for monadic signature 41

1 Introduction

Rewriting is an abstract model of computation. Among other things, it studies important properties including the following:

$$\text{CR}: \quad \forall s \forall t \forall u \, (s \to^* t \wedge s \to^* u \implies t \downarrow u) \qquad \text{confluence}$$

$$\text{SCR}: \quad \forall s \forall t \forall u \, (s \to t \wedge s \to u \implies \exists v \, (t \to^= v \wedge u \to^* v)) \qquad \text{strong confluence}$$

$$\text{WCR}: \quad \forall s \forall t \forall u \, (s \to t \wedge s \to u \implies t \downarrow u) \qquad \text{local confluence}$$

$$\text{NFP}: \quad \forall s \forall t \forall u \, (s \to^* t \wedge s \to^! u \implies t \to^! u) \qquad \text{normal form property}$$

$$\text{UNR}: \quad \forall s \forall t \forall u \, (s \to^! t \wedge s \to^! u \implies t = u) \qquad \text{unique normal forms with respect to reduction}$$

$$\text{UNC}: \quad \forall t \forall u \, (t \leftrightarrow^* u \wedge \text{NF}(t) \wedge \text{NF}(u) \implies t = u) \qquad \text{unique normal forms with respect to conversion}$$

We also consider the following properties involving two TRSs $\mathcal{R}$ and $\mathcal{S}$:

$$\text{COM}: \quad \forall s \forall t \forall u \, (s \to_{\mathcal{R}}^* t \wedge s \to_{\mathcal{S}}^* u \implies \exists v \, (t \to_{\mathcal{S}}^* v \wedge u \to_{\mathcal{R}}^* v)) \qquad \text{commutation}$$

$$\text{CE}: \quad \forall s \forall t \, (s \leftrightarrow_{\mathcal{R}}^* t \iff s \leftrightarrow_{\mathcal{S}}^* t) \qquad \text{conversion equivalence}$$

$$\text{NE}: \quad \forall s \forall t \, (s \to_{\mathcal{R}}^! t \iff s \to_{\mathcal{S}}^! t) \qquad \text{normalization equivalence}$$

An interesting observation is that for each of these properties there exists a rewrite system that satisfies the property when restricted to ground terms but not when arbitrary terms are allowed. Consider the left-linear right-ground TRS $\mathcal{R}$ consisting of the rules

$$\mathrm{a} \to \mathrm{b} \qquad \mathrm{f}(\mathrm{a}, x) \to \mathrm{b} \qquad \mathrm{f}(\mathrm{b}, \mathrm{b}) \to \mathrm{b}$$

over the signature $\mathcal{F} = \{\mathrm{a}, \mathrm{b}, \mathrm{f}\}$. It is ground-confluent because every ground term in $\mathcal{T}(\mathcal{F})$ rewrites to b. Confluence does not hold; the term $\mathrm{f}(\mathrm{a}, x)$ rewrites to the different normal forms b and $\mathrm{f}(\mathrm{b}, x)$.

In this AFP entry, properties on arbitrary terms are reduced to the corresponding properties on ground terms, for left-linear right-ground rewrite systems and for linear variable-separated systems. To do this, I formalized fundamental term rewriting operations that include the root step and the one step rewriting relations. Also, I added definitions for conversion equivalence, normalization equivalence, strong confluence and the normal from property extending the list of important rewriting properties of the AFP entry “Abstract Rewriting” [1].

Rewrite sequences that contain a root step play an important role in the formalization. The table of contents should give the reader a good overview of the content of this entry.

2 Preliminaries

theory *Terms-Positions*
imports *Regular-Tree-Relations.Ground-Terms*
begin

2.1 Additional operations on terms and positions

2.1.1 Linearity

fun *linear-term* :: $('f, 'v)$ *term* $\Rightarrow$ *bool* **where**
linear-term (*Var* -) = *True* |
linear-term (*Fun* - *ts*) = (*is-partition* (*map vars-term ts*) $\wedge$ ($\forall$ *t*$\in$*set ts*. *linear-term t*))
abbreviation *linear-sys* $\mathcal{R} \equiv \forall\ (l, r) \in \mathcal{R}$. *linear-term l* $\wedge$ *linear-term r*

2.1.2 Positions induced by contexts, by variables and by given subterms

definition *possc* $C = \{p \mid p\ t.\ p \in poss\ C\langle t\rangle\}$
definition *varposs* $s = \{p \mid p.\ p \in poss\ s \wedge is\text{-}Var\ (s\ |\text{-}\ p)\}$
definition *poss-of-term* $u\ t = \{p.\ p \in poss\ t \wedge t\ |\text{-}\ p = u\}$

2.1.3 Replacing functions symbols that aren't specified in the signature by variables

definition *funas-rel* $\mathcal{R} = (\bigcup\ (l, r) \in \mathcal{R}.\ funas\text{-}term\ l \cup funas\text{-}term\ r)$

fun *term-to-sig* **where**
term-to-sig $\mathcal{F}$ *v* (*Var x*) = *Var x*
| *term-to-sig* $\mathcal{F}$ *v* (*Fun f ts*) =
(*if* (*f*, *length ts*) $\in \mathcal{F}$ *then Fun f* (*map* (*term-to-sig* $\mathcal{F}$ *v*) *ts*) *else Var v*)

fun *ctxt-well-def-hole-path* **where**
ctxt-well-def-hole-path $\mathcal{F}$ *Hole* $\longleftrightarrow$ *True*
| *ctxt-well-def-hole-path* $\mathcal{F}$ (*More f ss C ts*) $\longleftrightarrow$ (*f*, *Suc* (*length ss* + *length ts*)) $\in$ $\mathcal{F}$ $\wedge$ *ctxt-well-def-hole-path* $\mathcal{F}$ *C*

fun *inv-const-ctxt* **where**
inv-const-ctxt $\mathcal{F}$ *v Hole* = *Hole*
| *inv-const-ctxt* $\mathcal{F}$ *v* ((*More f ss C ts*))
= (*More f* (*map* (*term-to-sig* $\mathcal{F}$ *v*) *ss*) (*inv-const-ctxt* $\mathcal{F}$ *v C*) (*map* (*term-to-sig* $\mathcal{F}$ *v*) *ts*))

fun *inv-const-ctxt'* **where**
inv-const-ctxt' $\mathcal{F}$ *v Hole* = *Var v*
| *inv-const-ctxt'* $\mathcal{F}$ *v* ((*More f ss C ts*))
= (*if* (*f*, *Suc* (*length ss* + *length ts*)) $\in \mathcal{F}$ *then Fun f* (*map* (*term-to-sig* $\mathcal{F}$ *v*) *ss* @ *inv-const-ctxt'* $\mathcal{F}$ *v C* # *map* (*term-to-sig* $\mathcal{F}$ *v*) *ts*) *else Var v*)

2.1.4 Replace term at a given position in contexts

fun *replace-term-context-at* :: $('f, 'v)$ *ctxt* $\Rightarrow$ *pos* $\Rightarrow$ $('f, 'v)$ *term* $\Rightarrow$ $('f, 'v)$ *ctxt*
(-[- $\leftarrow$ -]$_C$ [*1000*, *0*] *1000*) **where**
replace-term-context-at $\Box$ *p u* = $\Box$
| *replace-term-context-at* (*More f ss C ts*) (*i* # *ps*) *u* =
(*if i* < *length ss then More f* (*ss*[*i* := (*ss* ! *i*)[*ps* $\leftarrow$ *u*]]) *C ts*
else if i = *length ss then More f ss* (*replace-term-context-at C ps u*) *ts*
else More f ss C (*ts*[(*i* − *Suc* (*length ss*)) := (*ts* ! (*i* − *Suc* (*length ss*)))[*ps* $\leftarrow$ *u*]]))

abbreviation *constT c* $\equiv$ *Fun c* []

2.1.5 Multihole context closure of a term relation as inductive set

definition *all-ctxt-closed* **where**
all-ctxt-closed F r $\longleftrightarrow$ ($\forall$ *f ts ss*. (*f*, *length ss*) $\in$ *F* $\longrightarrow$ *length ts* = *length ss* $\longrightarrow$
($\forall$ *i*. *i* < *length ts* $\longrightarrow$ (*ts* ! *i*, *ss* ! *i*) $\in$ *r*) $\longrightarrow$
($\forall$ *i*. *i* < *length ts* $\longrightarrow$ *funas-term* (*ts* ! *i*) $\cup$ *funas-term* (*ss* ! *i*) $\subseteq$ *F*) $\longrightarrow$ (*Fun f ts*, *Fun f ss*) $\in$ *r*) $\wedge$
($\forall$ *x*. (*Var x*, *Var x*) $\in$ *r*)

2.2 Destruction and introduction of *all-ctxt-closed*

lemma *all-ctxt-closedD*: *all-ctxt-closed F r* $\Longrightarrow$ (*f*,*length ss*) $\in$ *F* $\Longrightarrow$ *length ts* = *length ss*
$\Longrightarrow$ $[\![\bigwedge$ *i*. *i* < *length ts* $\Longrightarrow$ (*ts* ! *i*, *ss* ! *i*) $\in$ *r* $]\!]$
$\Longrightarrow$ $[\![\bigwedge$ *i*. *i* < *length ts* $\Longrightarrow$ *funas-term* (*ts* ! *i*) $\subseteq$ *F* $]\!]$
$\Longrightarrow$ $[\![\bigwedge$ *i*. *i* < *length ts* $\Longrightarrow$ *funas-term* (*ss* ! *i*) $\subseteq$ *F* $]\!]$
$\Longrightarrow$ (*Fun f ts*, *Fun f ss*) $\in$ *r*
⟨*proof*⟩

lemma *trans-ctxt-sig-imp-all-ctxt-closed*: **assumes** *tran*: *trans r*
and *refl*: $\bigwedge$ *t*. *funas-term t* $\subseteq$ *F* $\Longrightarrow$ (*t*,*t*) $\in$ *r*
and *ctxt*: $\bigwedge$ *C s t*. *funas-ctxt C* $\subseteq$ *F* $\Longrightarrow$ *funas-term s* $\subseteq$ *F* $\Longrightarrow$ *funas-term t* $\subseteq$ *F* $\Longrightarrow$ (*s*,*t*) $\in$ *r* $\Longrightarrow$ (*C* ⟨ *s* ⟩, *C* ⟨ *t* ⟩) $\in$ *r*
shows *all-ctxt-closed F r* ⟨*proof*⟩

2.3 Lemmas for *poss* and ordering of positions

lemma *subst-poss-mono*: *poss s* $\subseteq$ *poss* ($s \cdot \sigma$)
⟨*proof*⟩

lemma *par-pos-prefix* [*simp*]:
(*i* # *p*) $\perp$ (*i* # *q*) $\Longrightarrow$ *p* $\perp$ *q*
⟨*proof*⟩

lemma *pos-diff-itself* [*simp*]: $p -_p p$ = []
⟨*proof*⟩

lemma *pos-les-eq-append-diff* [*simp*]:
$p \le_p q \Longrightarrow p \mathbin{@} (q -_p p) = q$
⟨*proof*⟩

lemma *pos-diff-append-itself* [*simp*]: $(p \mathbin{@} q) -_p p = q$
⟨*proof*⟩

lemma *poss-pos-diffI*:
$p \le_p q \Longrightarrow q \in poss\ s \Longrightarrow q -_p p \in poss\ (s \mid\!\text{-}\ p)$
⟨*proof*⟩

lemma *less-eq-poss-append-itself* [*simp*]: $p \le_p (p \mathbin{@} q)$
⟨*proof*⟩

lemma *poss-ctxt-apply* [*simp*]:
$hole\text{-}pos\ C \mathbin{@} p \in poss\ C\langle s\rangle \longleftrightarrow p \in poss\ s$
⟨*proof*⟩

lemma *pos-replace-at-pres*:
$p \in poss\ s \Longrightarrow p \in poss\ s[p \leftarrow t]$
⟨*proof*⟩

lemma *par-pos-replace-pres*:
$p \in poss\ s \Longrightarrow p \perp q \Longrightarrow p \in poss\ s[q \leftarrow t]$
⟨*proof*⟩

lemma *poss-of-termE* [*elim*]:
assumes $p \in poss\text{-}of\text{-}term\ u\ s$
and $p \in poss\ s \Longrightarrow s \mid\!\text{-}\ p = u \Longrightarrow P$
shows P ⟨*proof*⟩

lemma *poss-of-term-Cons*:
$i \mathbin{\#} p \in poss\text{-}of\text{-}term\ u\ (Fun\ f\ ts) \Longrightarrow p \in poss\text{-}of\text{-}term\ u\ (ts\ !\ i)$
⟨*proof*⟩

lemma *poss-of-term-const-ctxt-apply*:
assumes $p \in poss\text{-}of\text{-}term\ (constT\ c)\ C\langle s\rangle$
shows $p \perp (hole\text{-}pos\ C) \lor (hole\text{-}pos\ C) \le_p p$ ⟨*proof*⟩

2.4 Lemmas for (|-) and *replace-term-at*

lemma *subt-at-append-dist*:
$p \mathbin{@} q \in poss\ s \Longrightarrow s \mid\!\text{-}\ (p \mathbin{@} q) = (s \mid\!\text{-}\ p) \mid\!\text{-}\ q$
⟨*proof*⟩

lemma *ctxt-apply-term-subt-at-hole-pos* [*simp*]:
$C\langle s\rangle \mid\!\text{-}\ (hole\text{-}pos\ C \mathbin{@} q) = s \mid\!\text{-}\ q$
⟨*proof*⟩

lemma *subst-subt-at-dist*:
$p \in poss\ s \Longrightarrow s \cdot \sigma\ |\text{-}\ p = s\ |\text{-}\ p \cdot \sigma$
⟨*proof*⟩

lemma *replace-term-at-subt-at-id* [*simp*]: $s[p \leftarrow (s\ |\text{-}\ p)] = s$
⟨*proof*⟩

lemma *replace-term-at-same-pos* [*simp*]:
$s[p \leftarrow u][p \leftarrow t] = s[p \leftarrow t]$
⟨*proof*⟩
lemma *subt-at-vars-term*:
$p \in poss\ s \Longrightarrow s\ |\text{-}\ p = Var\ x \Longrightarrow x \in vars\text{-}term\ s$
⟨*proof*⟩

lemma *linear-term-varposs-subst-replace-term*:
$linear\text{-}term\ s \Longrightarrow p \in varposs\ s \Longrightarrow p \leq_p q \Longrightarrow$
$(s \cdot \sigma)[q \leftarrow u] = s \cdot (\lambda\ x.\ if\ Var\ x = s\ |\text{-}\ p\ then\ (\sigma\ x)[q -_p p \leftarrow u]\ else\ (\sigma\ x))$
⟨*proof*⟩
lemma *par-hole-pos-replace-term-context-at*:
$p \perp hole\text{-}pos\ C \Longrightarrow C\langle s\rangle[p \leftarrow u] = (C[p \leftarrow u]_C)\langle s\rangle$
⟨*proof*⟩

lemma *par-pos-replace-term-at*:
$p \in poss\ s \Longrightarrow p \perp q \Longrightarrow s[q \leftarrow t]\ |\text{-}\ p = s\ |\text{-}\ p$
⟨*proof*⟩

lemma *less-eq-subt-at-replace*:
$p \in poss\ s \Longrightarrow p \leq_p q \Longrightarrow s[q \leftarrow t]\ |\text{-}\ p = (s\ |\text{-}\ p)[q -_p p \leftarrow t]$
⟨*proof*⟩

lemma *greater-eq-subt-at-replace*:
$p \in poss\ s \Longrightarrow q \leq_p p \Longrightarrow s[q \leftarrow t]\ |\text{-}\ p = t\ |\text{-}\ (p -_p q)$
⟨*proof*⟩

lemma *replace-subterm-at-itself* [*simp*]:
$s[p \leftarrow (s\ |\text{-}\ p)[q \leftarrow t]] = s[p\ @\ q \leftarrow t]$
⟨*proof*⟩

lemma *hole-pos-replace-term-at* [*simp*]:
$hole\text{-}pos\ C \leq_p p \Longrightarrow C\langle s\rangle[p \leftarrow u] = C\langle s[p -_p hole\text{-}pos\ C \leftarrow u]\rangle$
⟨*proof*⟩

lemma *ctxt-of-pos-term-apply-replace-at-ident*:
assumes $p \in poss\ s$

shows $(\textit{ctxt-at-pos}\ s\ p)\langle t\rangle = s[p \leftarrow t]$
⟨proof⟩

lemma *ctxt-apply-term-replace-term-hole-pos* [*simp*]:
$C\langle s\rangle[\textit{hole-pos}\ C\ @\ q\ \leftarrow u] = C\langle s[q\ \leftarrow u]\rangle$
⟨proof⟩

lemma *ctxt-apply-subt-at-hole-pos* [*simp*]: $C\langle s\rangle$ |- *hole-pos* $C = s$
⟨proof⟩

lemma *subt-at-imp-supteq'*:
assumes $p \in \textit{poss}\ s$ **and** $s|\text{-}p = t$ **shows** $s \trianglerighteq t$ ⟨proof⟩

lemma *subt-at-imp-supteq*:
assumes $p \in \textit{poss}\ s$ **shows** $s \trianglerighteq s|\text{-}p$
⟨proof⟩

2.5 *term-to-sig* invariants and distributions

lemma *fuans-term-term-to-sig* [*simp*]: $\textit{funas-term}\ (\textit{term-to-sig}\ \mathcal{F}\ v\ t) \subseteq \mathcal{F}$
⟨proof⟩

lemma *term-to-sig-id* [*simp*]:
$\textit{funas-term}\ t \subseteq \mathcal{F} \Longrightarrow \textit{term-to-sig}\ \mathcal{F}\ v\ t = t$
⟨proof⟩

lemma *term-to-sig-subst-sig* [*simp*]:
$\textit{funas-term}\ t \subseteq \mathcal{F} \Longrightarrow \textit{term-to-sig}\ \mathcal{F}\ v\ (t \cdot \sigma) = t \cdot (\lambda\ x.\ \ \textit{term-to-sig}\ \mathcal{F}\ v\ (\sigma\ x))$
⟨proof⟩

lemma *funas-ctxt-ctxt-inv-const-ctxt-ind* [*simp*]:
$\textit{funas-ctxt}\ C \subseteq \mathcal{F} \Longrightarrow \textit{inv-const-ctxt}\ \mathcal{F}\ v\ C = C$
⟨proof⟩

lemma *term-to-sig-ctxt-apply* [*simp*]:
$\textit{ctxt-well-def-hole-path}\ \mathcal{F}\ C \Longrightarrow \textit{term-to-sig}\ \mathcal{F}\ v\ C\langle s\rangle = (\textit{inv-const-ctxt}\ \mathcal{F}\ v\ C)\langle \textit{term-to-sig}\ \mathcal{F}\ v\ s\rangle$
⟨proof⟩

lemma *term-to-sig-ctxt-apply'* [*simp*]:
$\neg\ \textit{ctxt-well-def-hole-path}\ \mathcal{F}\ C \Longrightarrow \textit{term-to-sig}\ \mathcal{F}\ v\ C\langle s\rangle = \textit{inv-const-ctxt}'\ \mathcal{F}\ v\ C$
⟨proof⟩

lemma *funas-ctxt-ctxt-well-def-hole-path*:
$\textit{funas-ctxt}\ C \subseteq \mathcal{F} \Longrightarrow \textit{ctxt-well-def-hole-path}\ \mathcal{F}\ C$
⟨proof⟩

2.6 Misc

lemma *funas-term-subt-at*:

$(f,\ n) \in$ *funas-term* $t \Longrightarrow (\exists\ p\ ts.\ p \in$ *poss* $t \wedge t$ |- $p =$ *Fun f ts* $\wedge$ *length ts* $= n)$
⟨*proof*⟩

lemma *finite-poss*: *finite* (*poss s*)
⟨*proof*⟩

lemma *finite-varposs*: *finite* (*varposs s*)
⟨*proof*⟩

lemma *gound-linear* [*simp*]: *ground t* $\Longrightarrow$ *linear-term t*
⟨*proof*⟩

declare *ground-substI*[*intro*, *simp*]
lemma *ground-ctxt-substI*:
$(\bigwedge x.\ x \in$ *vars-ctxt* $C \Longrightarrow$ *ground* $(\sigma\ x)) \Longrightarrow$ *ground-ctxt* $(C \cdot_c \sigma)$
⟨*proof*⟩

lemma *funas-ctxt-subst-apply-ctxt*:
funas-ctxt $(C \cdot_c \sigma) =$ *funas-ctxt* $C \cup (\bigcup$ (*funas-term* ‘ σ ‘ *vars-ctxt* C))
⟨*proof*⟩

lemma *varposs-Var*[*simp*]:
varposs (*Var x*) = {[]}
⟨*proof*⟩

lemma *varposs-Fun*[*simp*]:
varposs (*Fun f ts*) = { i # p| i p. $i <$ *length ts* $\wedge\ p \in$ *varposs* (*ts* ! i)}
⟨*proof*⟩

lemma *vars-term-varposs-iff*:
$x \in$ *vars-term* $s \longleftrightarrow (\exists\ p \in$ *varposs* $s.\ s$ |- $p =$ *Var* x)
⟨*proof*⟩

lemma *vars-term-empty-ground*:
vars-term $s = \{\} \Longrightarrow$ *ground s*
⟨*proof*⟩

lemma *ground-subst-apply*: *ground t* $\Longrightarrow t \cdot \sigma = t$
⟨*proof*⟩

lemma *varposs-imp-poss*:
$p \in$ *varposs* $s \Longrightarrow p \in$ *poss* s ⟨*proof*⟩

lemma *varposs-empty-gound*:
varposs $s = \{\} \longleftrightarrow$ *ground s*
⟨*proof*⟩

lemma *funas-term-subterm-atI* [*intro*]:

$p \in poss\ s \Longrightarrow funas\text{-}term\ s \subseteq \mathcal{F} \Longrightarrow funas\text{-}term\ (s\ |\text{-}\ p) \subseteq \mathcal{F}$
⟨proof⟩

lemma *varposs-ground-replace-at*:
$p \in varposs\ s \Longrightarrow ground\ u \Longrightarrow varposs\ s[p \leftarrow u] = varposs\ s - \{p\}$
⟨proof⟩

lemma *funas-term-replace-at-upper*:
$funas\text{-}term\ s[p \leftarrow t] \subseteq funas\text{-}term\ s \cup funas\text{-}term\ t$
⟨proof⟩

lemma *funas-term-replace-at-lower*:
$p \in poss\ s \Longrightarrow funas\text{-}term\ t \subseteq funas\text{-}term\ (s[p \leftarrow t])$
⟨proof⟩

lemma *poss-of-term-possI* [*intro*!]:
$p \in poss\ s \Longrightarrow s\ |\text{-}\ p = u \Longrightarrow p \in poss\text{-}of\text{-}term\ u\ s$
⟨proof⟩

lemma *poss-of-term-replace-term-at*:
$p \in poss\ s \Longrightarrow p \in poss\text{-}of\text{-}term\ u\ s[p \leftarrow u]$
⟨proof⟩

lemma *constT-nfunas-term-poss-of-term-empty*:
$(c,\ 0) \notin funas\text{-}term\ t \longleftrightarrow poss\text{-}of\text{-}term\ (constT\ c)\ t = \{\}$
⟨proof⟩

lemma *poss-of-term-poss-emptyD*:
assumes $poss\text{-}of\text{-}term\ u\ s = \{\}$
shows $p \in poss\ s \Longrightarrow s\ |\text{-}\ p \neq u$ ⟨proof⟩

lemma *possc-subt-at-ctxt-apply*:
$p \in possc\ C \Longrightarrow p \perp hole\text{-}pos\ C \Longrightarrow C\langle s\rangle\ |\text{-}\ p = C\langle t\rangle\ |\text{-}\ p$
⟨proof⟩

end

3 Rewriting

theory *Rewriting*
imports *Terms-Positions*
begin

3.1 Basic rewrite definitions

3.1.1 Rewrite steps with implicit signature declaration (encoded in the type)

inductive-set *rrstep* :: $('f,\ 'v)\ term\ rel \Rightarrow ('f,\ 'v)\ term\ rel$ **for** $\mathcal{R}$ **where**

[*intro*]: $(l,\ r) \in \mathcal{R} \Longrightarrow (l \cdot \sigma,\ r \cdot \sigma) \in$ *rrstep* $\mathcal{R}$

inductive-set *rstep* :: $('f,\ 'v)$ *term rel* $\Rightarrow$ $('f,\ 'v)$ *term rel* **for** $\mathcal{R}$ **where**
$(s,\ t) \in$ *rrstep* $\mathcal{R} \Longrightarrow (C\langle s\rangle,\ C\langle t\rangle) \in$ *rstep* $\mathcal{R}$

3.1.2 Restrict relations to terms induced by a given signature

definition *sig-step* $\mathcal{F}$ $\mathcal{R}$ = *Restr* $\mathcal{R}$ (*Collect* (λ *s*. *funas-term s* $\subseteq \mathcal{F}$))

3.1.3 Rewriting under a given signature/restricted to ground terms

abbreviation *srrstep* $\mathcal{F}$ $\mathcal{R}$ $\equiv$ *sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$)
abbreviation *srstep* $\mathcal{F}$ $\mathcal{R}$ $\equiv$ *sig-step* $\mathcal{F}$ (*rstep* $\mathcal{R}$)
abbreviation *gsrstep* $\mathcal{F}$ $\mathcal{R}$ $\equiv$ *Restr* (*sig-step* $\mathcal{F}$ (*rstep* $\mathcal{R}$)) (*Collect ground*)

3.1.4 Rewriting sequences involving a root step

abbreviation (*input*) *relto* :: *'a rel* $\Rightarrow$ *'a rel* $\Rightarrow$ *'a rel* **where**
relto R S $\equiv$ $S\widehat{\ }*$ *O R O* $S\widehat{\ }*$
definition *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ $\equiv$ *relto* (*sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$)) (*srstep* $\mathcal{F}$ $\mathcal{R}$)

3.2 Monotonicity laws

lemma *Restr-mono*: *Restr r A* $\subseteq$ *r* ⟨*proof*⟩

lemma *Restr-trancl-mono-set*: $(Restr\ r\ A)^+ \subseteq A \times A$
⟨*proof*⟩

lemma *rrstep-rstep-mono*: *rrstep* $\mathcal{R} \subseteq$ *rstep* $\mathcal{R}$
⟨*proof*⟩

lemma *sig-step-mono*:
$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow$ *sig-step* $\mathcal{F}$ $\mathcal{R} \subseteq$ *sig-step* $\mathcal{G}$ $\mathcal{R}$
⟨*proof*⟩

lemma *sig-step-mono2*:
$\mathcal{R} \subseteq \mathcal{L} \Longrightarrow$ *sig-step* $\mathcal{F}$ $\mathcal{R} \subseteq$ *sig-step* $\mathcal{F}$ $\mathcal{L}$
⟨*proof*⟩

lemma *srrstep-monp*:
$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow$ *srrstep* $\mathcal{F}$ $\mathcal{R} \subseteq$ *srrstep* $\mathcal{G}$ $\mathcal{R}$
⟨*proof*⟩

lemma *srstep-monp*:
$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow$ *srstep* $\mathcal{F}$ $\mathcal{R} \subseteq$ *srstep* $\mathcal{G}$ $\mathcal{R}$
⟨*proof*⟩

lemma *srsteps-monp*:

$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow (srstep\ \mathcal{F}\ \mathcal{R})^+ \subseteq (srstep\ \mathcal{G}\ \mathcal{R})^+$
⟨*proof*⟩

lemma *srsteps-eq-monp*:
$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow (srstep\ \mathcal{F}\ \mathcal{R})^* \subseteq (srstep\ \mathcal{G}\ \mathcal{R})^*$
⟨*proof*⟩

lemma *srsteps-with-root-step-sig-mono*:
$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow$ *srsteps-with-root-step* $\mathcal{F}\ \mathcal{R} \subseteq$ *srsteps-with-root-step* $\mathcal{G}\ \mathcal{R}$
⟨*proof*⟩

3.3 Introduction, elimination, and destruction rules for *sig-step*, *rstep*, *rrstep*, *srrstep*, and *srstep*

lemma *sig-stepE* [*elim*, *consumes 1*]:
$(s,\ t) \in$ *sig-step* $\mathcal{F}\ \mathcal{R} \Longrightarrow [\![(s,\ t) \in \mathcal{R} \Longrightarrow$ *funas-term* $s \subseteq \mathcal{F} \Longrightarrow$ *funas-term* $t \subseteq \mathcal{F} \Longrightarrow P]\!] \Longrightarrow P$
⟨*proof*⟩

lemma *sig-stepI* [*intro*]:
funas-term $s \subseteq \mathcal{F} \Longrightarrow$ *funas-term* $t \subseteq \mathcal{F} \Longrightarrow (s,\ t) \in \mathcal{R} \Longrightarrow (s,\ t) \in$ *sig-step* $\mathcal{F}\ \mathcal{R}$
⟨*proof*⟩

lemma *rrstep-subst* [*elim*, *consumes 1*]:
assumes $(s,\ t) \in$ *rrstep* $\mathcal{R}$
obtains $l\ r\ \sigma$ **where** $(l,\ r) \in \mathcal{R}\ s = l \cdot \sigma\ t = r \cdot \sigma$ ⟨*proof*⟩

lemma *rstep-imp-C-s-r*:
assumes $(s,\ t) \in$ *rstep* $\mathcal{R}$
shows $\exists\, C\ \sigma\ l\ r.\ (l,r) \in \mathcal{R} \wedge s = C\langle l\cdot\sigma\rangle \wedge t = C\langle r\cdot\sigma\rangle$⟨*proof*⟩

lemma *rstep-imp-C-s-r′* [*elim*, *consumes 1*]:
assumes $(s,\ t) \in$ *rstep* $\mathcal{R}$
obtains $C\ l\ r\ \sigma$ **where** $(l,r) \in \mathcal{R}\ s = C\langle l\cdot\sigma\rangle\ t = C\langle r\cdot\sigma\rangle$ ⟨*proof*⟩

lemma *rrstep-basicI* [*intro*]:
$(l,\ r) \in \mathcal{R} \Longrightarrow (l,\ r) \in$ *rrstep* $\mathcal{R}$
⟨*proof*⟩

lemma *rstep-ruleI* [*intro*]:
$(l,\ r) \in \mathcal{R} \Longrightarrow (l,\ r) \in$ *rstep* $\mathcal{R}$
⟨*proof*⟩

lemma *rstepI* [*intro*]:
$(l,\ r) \in \mathcal{R} \Longrightarrow s = C\langle l \cdot \sigma\rangle \Longrightarrow t = C\langle r \cdot \sigma\rangle \Longrightarrow (s,\ t) \in$ *rstep* $\mathcal{R}$
⟨*proof*⟩

lemma *rstep-substI* [*intro*]:
$(s, t) \in rstep\ \mathcal{R} \Longrightarrow (s \cdot \sigma, t \cdot \sigma) \in rstep\ \mathcal{R}$
⟨*proof*⟩

lemma *rstep-ctxtI* [*intro*]:
$(s, t) \in rstep\ \mathcal{R} \Longrightarrow (C\langle s\rangle, C\langle t\rangle) \in rstep\ \mathcal{R}$
⟨*proof*⟩

lemma *srrstepD*:
$(s, t) \in srrstep\ \mathcal{F}\ \mathcal{R} \Longrightarrow (s, t) \in rrstep\ \mathcal{R} \wedge funas\text{-}term\ s \subseteq \mathcal{F} \wedge funas\text{-}term\ t \subseteq \mathcal{F}$
⟨*proof*⟩

lemma *srstepD*:
$(s, t) \in (srstep\ \mathcal{F}\ \mathcal{R}) \Longrightarrow (s, t) \in rstep\ \mathcal{R} \wedge funas\text{-}term\ s \subseteq \mathcal{F} \wedge funas\text{-}term\ t \subseteq \mathcal{F}$
⟨*proof*⟩

lemma *srstepsD*:
$(s, t) \in (srstep\ \mathcal{F}\ \mathcal{R})^+ \Longrightarrow (s, t) \in (rstep\ \mathcal{R})^+ \wedge funas\text{-}term\ s \subseteq \mathcal{F} \wedge funas\text{-}term\ t \subseteq \mathcal{F}$
⟨*proof*⟩

3.3.1 Transitive and relfexive closure distribution over *sig-step*

lemma *funas-rel-converse*:
$funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F} \Longrightarrow funas\text{-}rel\ (\mathcal{R}^{-1}) \subseteq \mathcal{F}$ ⟨*proof*⟩

lemma *rstep-term-to-sig-r*:
assumes $(s, t) \in rstep\ \mathcal{R}$ **and** $funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F}$ **and** $funas\text{-}term\ s \subseteq \mathcal{F}$
shows $(s, term\text{-}to\text{-}sig\ \mathcal{F}\ v\ t) \in rstep\ \mathcal{R}$
⟨*proof*⟩

lemma *rstep-term-to-sig-l*:
assumes $(s, t) \in rstep\ \mathcal{R}$ **and** $funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F}$ **and** $funas\text{-}term\ t \subseteq \mathcal{F}$
shows $(term\text{-}to\text{-}sig\ \mathcal{F}\ v\ s, t) \in rstep\ \mathcal{R}$
⟨*proof*⟩

lemma *rstep-trancl-sig-step-r*:
assumes $(s, t) \in (rstep\ \mathcal{R})^+$ **and** $funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F}$ **and** $funas\text{-}term\ s \subseteq \mathcal{F}$
shows $(s, term\text{-}to\text{-}sig\ \mathcal{F}\ v\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^+$ ⟨*proof*⟩

lemma *rstep-trancl-sig-step-l*:
assumes $(s, t) \in (rstep\ \mathcal{R})^+$ **and** $funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F}$ **and** $funas\text{-}term\ t \subseteq \mathcal{F}$
shows $(term\text{-}to\text{-}sig\ \mathcal{F}\ v\ s, t) \in (srstep\ \mathcal{F}\ \mathcal{R})^+$ ⟨*proof*⟩

lemma *rstep-srstepI* [*intro*]:
$funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F} \Longrightarrow funas\text{-}term\ s \subseteq \mathcal{F} \Longrightarrow funas\text{-}term\ t \subseteq \mathcal{F} \Longrightarrow (s, t) \in rstep\ \mathcal{R} \Longrightarrow (s, t) \in srstep\ \mathcal{F}\ \mathcal{R}$

⟨*proof*⟩

lemma *rsteps-srstepsI* [*intro*]:
funas-rel $\mathcal{R} \subseteq \mathcal{F} \Longrightarrow$ *funas-term* $s \subseteq \mathcal{F} \Longrightarrow$ *funas-term* $t \subseteq \mathcal{F} \Longrightarrow (s,\ t) \in (rstep\ \mathcal{R})^{+} \Longrightarrow (s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^{+}$
⟨*proof*⟩

lemma *rsteps-eq-srsteps-eqI* [*intro*]:
funas-rel $\mathcal{R} \subseteq \mathcal{F} \Longrightarrow$ *funas-term* $s \subseteq \mathcal{F} \Longrightarrow$ *funas-term* $t \subseteq \mathcal{F} \Longrightarrow (s,\ t) \in (rstep\ \mathcal{R})^{*} \Longrightarrow (s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^{*}$
⟨*proof*⟩

lemma *rsteps-eq-relcomp-srsteps-eq-relcompI* [*intro*]:
assumes *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *funas*: *funas-term* $s \subseteq \mathcal{F}$ *funas-term* $t \subseteq \mathcal{F}$
and *steps*: $(s,\ t) \in (rstep\ \mathcal{R})^{*}\ O\ (rstep\ \mathcal{S})^{*}$
shows $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^{*}\ O\ (srstep\ \mathcal{F}\ \mathcal{S})^{*}$
⟨*proof*⟩

3.3.2 Distributivity laws

lemma *rstep-smycl-dist*:
$(rstep\ \mathcal{R})^{\leftrightarrow} = rstep\ (\mathcal{R}^{\leftrightarrow})$
⟨*proof*⟩

lemma *sig-step-symcl-dist*:
$(sig\text{-}step\ \mathcal{F}\ \mathcal{R})^{\leftrightarrow} = sig\text{-}step\ \mathcal{F}\ (\mathcal{R}^{\leftrightarrow})$
⟨*proof*⟩

lemma *srstep-symcl-dist*:
$(srstep\ \mathcal{F}\ \mathcal{R})^{\leftrightarrow} = srstep\ \mathcal{F}\ (\mathcal{R}^{\leftrightarrow})$
⟨*proof*⟩

lemma *Restr-smycl-dist*:
$(Restr\ \mathcal{R}\ \mathcal{A})^{\leftrightarrow} = Restr\ (\mathcal{R}^{\leftrightarrow})\ \mathcal{A}$
⟨*proof*⟩

lemmas *rew-symcl-inwards* = *rstep-smycl-dist sig-step-symcl-dist srstep-symcl-dist Restr-smycl-dist*
lemmas *rew-symcl-outwards* = *rew-symcl-inwards*[*symmetric*]

lemma *rstep-converse-dist*:
$(rstep\ \mathcal{R})^{-1} = rstep\ (\mathcal{R}^{-1})$
⟨*proof*⟩

lemma *srrstep-converse-dist*:
$(srrstep\ \mathcal{F}\ \mathcal{R})^{-1} = srrstep\ \mathcal{F}\ (\mathcal{R}^{-1})$
⟨*proof*⟩

lemma *sig-step-converse-rstep*:
$(srstep\ \mathcal{F}\ \mathcal{R})^{-1} = sig\text{-}step\ \mathcal{F}\ ((rstep\ \mathcal{R})^{-1})$
⟨*proof*⟩

lemma *srstep-converse-dist*:
$(srstep\ \mathcal{F}\ \mathcal{R})^{-1} = srstep\ \mathcal{F}\ (\mathcal{R}^{-1})$
⟨*proof*⟩

lemma *Restr-converse*: $(Restr\ \mathcal{R}\ A)^{-1} = Restr\ (\mathcal{R}^{-1})\ A$
⟨*proof*⟩

lemmas *rew-converse-inwards* = *rstep-converse-dist srrstep-converse-dist sig-step-converse-rstep srstep-converse-dist Restr-converse trancl-converse*[*symmetric*] *rtrancl-converse*[*symmetric*]
lemmas *rew-converse-outwards* = *rew-converse-inwards*[*symmetric*]

lemma *sig-step-rsteps-dist*:
$funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F} \Longrightarrow sig\text{-}step\ \mathcal{F}\ ((rstep\ \mathcal{R})^{+}) = (srstep\ \mathcal{F}\ \mathcal{R})^{+}$
⟨*proof*⟩

lemma *sig-step-rsteps-eq-dist*:
$funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F} \Longrightarrow sig\text{-}step\ \mathcal{F}\ ((rstep\ \mathcal{R})^{+}) \cup Id = (srstep\ \mathcal{F}\ \mathcal{R})^{*}$
⟨*proof*⟩

lemma *sig-step-conversion-dist*:
$(srstep\ \mathcal{F}\ \mathcal{R})^{\leftrightarrow *} = (srstep\ \mathcal{F}\ (\mathcal{R}^{\leftrightarrow}))^{*}$
⟨*proof*⟩

lemma *gsrstep-conversion-dist*:
$(gsrstep\ \mathcal{F}\ \mathcal{R})^{\leftrightarrow *} = (gsrstep\ \mathcal{F}\ (\mathcal{R}^{\leftrightarrow}))^{*}$
⟨*proof*⟩

lemma *sig-step-grstep-dist*:
$gsrstep\ \mathcal{F}\ \mathcal{R} = sig\text{-}step\ \mathcal{F}\ (Restr\ (rstep\ \mathcal{R})\ (Collect\ ground))$
⟨*proof*⟩

3.4 Substitution closure of *srstep*

lemma *srstep-subst-closed*:
assumes $(s,\ t) \in srstep\ \mathcal{F}\ \mathcal{R}\ \bigwedge x.\ funas\text{-}term\ (\sigma\ x) \subseteq \mathcal{F}$
shows $(s \cdot \sigma,\ t \cdot \sigma) \in srstep\ \mathcal{F}\ \mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-subst-closed*:
assumes $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^{+}\ \bigwedge x.\ funas\text{-}term\ (\sigma\ x) \subseteq \mathcal{F}$
shows $(s \cdot \sigma,\ t \cdot \sigma) \in (srstep\ \mathcal{F}\ \mathcal{R})^{+}$ ⟨*proof*⟩

lemma *srsteps-eq-subst-closed*:
assumes $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^{*}\ \bigwedge x.\ funas\text{-}term\ (\sigma\ x) \subseteq \mathcal{F}$
shows $(s \cdot \sigma,\ t \cdot \sigma) \in (srstep\ \mathcal{F}\ \mathcal{R})^{*}$ ⟨*proof*⟩

lemma *srsteps-eq-subst-relcomp-closed*:
assumes $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^*\ O\ (srstep\ \mathcal{F}\ \mathcal{S})^*\ \bigwedge x.\ funas\text{-}term\ (\sigma\ x) \subseteq \mathcal{F}$
shows $(s \cdot \sigma,\ t \cdot \sigma) \in (srstep\ \mathcal{F}\ \mathcal{R})^*\ O\ (srstep\ \mathcal{F}\ \mathcal{S})^*$
⟨*proof*⟩

3.5 Context closure of *srstep*

lemma *srstep-ctxt-closed*:
assumes $funas\text{-}ctxt\ C \subseteq \mathcal{F}$ **and** $(s,\ t) \in srstep\ \mathcal{F}\ \mathcal{R}$
shows $(C\langle s\rangle,\ C\langle t\rangle) \in srstep\ \mathcal{F}\ \mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-ctxt-closed*:
assumes $funas\text{-}ctxt\ C \subseteq \mathcal{F}$ **and** $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^+$
shows $(C\langle s\rangle,\ C\langle t\rangle) \in (srstep\ \mathcal{F}\ \mathcal{R})^+$ ⟨*proof*⟩

lemma *srsteps-eq-ctxt-closed*:
assumes $funas\text{-}ctxt\ C \subseteq \mathcal{F}$ **and** $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^*$
shows $(C\langle s\rangle,\ C\langle t\rangle) \in (srstep\ \mathcal{F}\ \mathcal{R})^*$ ⟨*proof*⟩

lemma *sig-steps-join-ctxt-closed*:
assumes $funas\text{-}ctxt\ C \subseteq \mathcal{F}$ **and** $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^\downarrow$
shows $(C\langle s\rangle,\ C\langle t\rangle) \in (srstep\ \mathcal{F}\ \mathcal{R})^\downarrow$ ⟨*proof*⟩

The following lemma shows that every rewrite sequence either contains a root step or is root stable

lemma *nsrsteps-with-root-step-step-on-args*:
assumes $(s,\ t) \in (srstep\ \mathcal{F}\ \mathcal{R})^+\ (s,\ t) \notin srsteps\text{-}with\text{-}root\text{-}step\ \mathcal{F}\ \mathcal{R}$
shows $\exists\ f\ ss\ ts.\ s = Fun\ f\ ss \wedge t = Fun\ f\ ts \wedge length\ ss = length\ ts \wedge$
$(\forall\ i < length\ ts.\ (ss\ !\ i,\ ts\ !\ i) \in (srstep\ \mathcal{F}\ \mathcal{R})^*)$ ⟨*proof*⟩

lemma *rstep-to-pos-replace*:
assumes $(s,\ t) \in rstep\ \mathcal{R}$
shows $\exists\ p\ l\ r\ \sigma.\ p \in poss\ s \wedge (l,\ r) \in \mathcal{R} \wedge s\ |\text{-}\ p = l \cdot \sigma \wedge t = s[p \leftarrow r \cdot \sigma]$
⟨*proof*⟩

lemma *pos-replace-to-rstep*:
assumes $p \in poss\ s\ (l,\ r) \in \mathcal{R}$
and $s\ |\text{-}\ p = l \cdot \sigma\ t = s[p \leftarrow r \cdot \sigma]$
shows $(s,\ t) \in rstep\ \mathcal{R}$
⟨*proof*⟩

end
theory *Replace-Constant*
imports *Rewriting*
begin

3.6 Removing/Replacing constants in a rewrite sequence that do not appear in the rewrite system

lemma *funas-term-const-subst-conv*:
$(c,\ 0) \notin$ *funas-term* $l \longleftrightarrow \neg\ (l \trianglerighteq$ *constT* $c)$
⟨*proof*⟩

lemma *fresh-const-single-step-replace*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *occ*: $p \in$ *poss-of-term* (*constT* c) s **and** *step*: $(s,\ t) \in$ *rstep* $\mathcal{R}$
shows $(s[p \leftarrow u],\ t) \in$ *rstep* $\mathcal{R}\ \vee$
$(\exists\ q.\ q \in$ *poss-of-term* (*constT* c) $t \wedge (s[p \leftarrow u],\ t[q \leftarrow u]) \in$ *rstep* $\mathcal{R})$
⟨*proof*⟩

lemma *fresh-const-steps-replace*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *occ*: $p \in$ *poss-of-term* (*constT* c) s **and** *steps*: $(s,\ t) \in (rstep\ \mathcal{R})^+$
shows $(s[p \leftarrow u],\ t) \in (rstep\ \mathcal{R})^+\ \vee$
$(\exists\ q.\ q \in$ *poss-of-term* (*constT* c) $t \wedge (s[p \leftarrow u],\ t[q \leftarrow u]) \in (rstep\ \mathcal{R})^+)$
⟨*proof*⟩

lemma *remove-const-lhs-steps*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *const*: $(c,\ 0) \notin$ *funas-term* t
and *pos*: $p \in$ *poss-of-term* (*constT* c) s
and *steps*: $(s,\ t) \in (rstep\ \mathcal{R})^+$
shows $(s[p \leftarrow u],\ t) \in (rstep\ \mathcal{R})^+$ ⟨*proof*⟩

Now we can show that we may remove a constant substitution

definition *const-replace-closed* **where**
const-replace-closed $c\ U = (\forall\ s\ t\ u\ p.$
$p \in$ *poss-of-term* (*constT* c) $s \longrightarrow (s,\ t) \in U \longrightarrow$
$(\exists\ q.\ q \in$ *poss-of-term* (*constT* c) $t \wedge (s[p \leftarrow u],\ t[q \leftarrow u]) \in U) \vee (s[p \leftarrow u],$
$t) \in U)$

lemma *const-replace-closedD*:
assumes *const-replace-closed* $c\ U\ p \in$ *poss-of-term* (*constT* c) $s\ (s,\ t) \in U$
shows $(s[p \leftarrow u],\ t) \in U \vee (\exists\ q.\ q \in$ *poss-of-term* (*constT* c) $t \wedge (s[p \leftarrow u],\ t[q$
$\leftarrow u]) \in U)$ ⟨*proof*⟩

lemma *const-replace-closedI*:
assumes $\bigwedge\ s\ t\ u\ p.\ p \in$ *poss-of-term* (*constT* c) $s \Longrightarrow (s,\ t) \in U \Longrightarrow$
$(\exists\ q.\ q \in$ *poss-of-term* (*constT* c) $t \wedge (s[p \leftarrow u],\ t[q \leftarrow u]) \in U) \vee (s[p \leftarrow$
$u],\ t) \in U$
shows *const-replace-closed* $c\ U$ ⟨*proof*⟩

abbreviation *const-subst* :: $'f \Rightarrow\ 'v \Rightarrow (\'f,\ 'v)$ *Term.term* **where**
const-subst $c \equiv (\lambda\ x.$ *Fun* c [])

lemma *lin-fresh-rstep-const-replace-closed*:

linear-sys $\mathcal{R} \Longrightarrow (c,\ 0) \notin$ *funas-rel* $\mathcal{R} \Longrightarrow$ *const-replace-closed c* (*rstep* $\mathcal{R}$)
⟨*proof*⟩

lemma *const-replace-closed-symcl*:
const-replace-closed c U $\Longrightarrow$ *const-replace-closed c* ($U^=$)
⟨*proof*⟩

lemma *const-replace-closed-trancl*:
const-replace-closed c U $\Longrightarrow$ *const-replace-closed c* (U^+)
⟨*proof*⟩

lemma *const-replace-closed-rtrancl*:
const-replace-closed c U $\Longrightarrow$ *const-replace-closed c* (U^*)
⟨*proof*⟩

lemma *const-replace-closed-relcomp*:
const-replace-closed c U $\Longrightarrow$ *const-replace-closed c V* $\Longrightarrow$ *const-replace-closed c* ($U\ O\ V$)
⟨*proof*⟩

const-replace-closed allow the removal of a fresh constant substitution

lemma *const-replace-closed-remove-subst-lhs*:
assumes *repcl*: *const-replace-closed c U*
and *const*: $(c,\ 0) \notin$ *funas-term t*
and *steps*: $(s \cdot$ *const-subst c*, $t) \in U$
shows $(s,\ t) \in U$ ⟨*proof*⟩

3.6.1 Removal lemma applied to various rewrite relations

lemma *remove-const-subst-step-lhs*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *const*: $(c,\ 0) \notin$ *funas-term t*
and *step*: $(s \cdot$ *const-subst c*, $t) \in$ (*rstep* $\mathcal{R}$)
shows $(s,\ t) \in$ (*rstep* $\mathcal{R}$)
⟨*proof*⟩

lemma *remove-const-subst-steps-lhs*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *const*: $(c,\ 0) \notin$ *funas-term t*
and *steps*: $(s \cdot$ *const-subst c*, $t) \in$ (*rstep* $\mathcal{R})^+$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^+$
⟨*proof*⟩

lemma *remove-const-subst-steps-eq-lhs*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *const*: $(c,\ 0) \notin$ *funas-term t*
and *steps*: $(s \cdot$ *const-subst c*, $t) \in$ (*rstep* $\mathcal{R})^*$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^*$ ⟨*proof*⟩

lemma *remove-const-subst-steps-rhs*:

assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *const*: $(c,\ 0) \notin$ *funas-term* s
and *steps*: $(s,\ t \cdot$ *const-subst* $c) \in$ (*rstep* $\mathcal{R})^+$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^+$
⟨*proof*⟩

lemma *remove-const-subst-steps-eq-rhs*:
assumes *lin*: *linear-sys* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *const*: $(c,\ 0) \notin$ *funas-term* s
and *steps*: $(s,\ t \cdot$ *const-subst* $c) \in$ (*rstep* $\mathcal{R})^*$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^*$
⟨*proof*⟩

Main lemmas

lemma *const-subst-eq-ground-eq*:
assumes $s \cdot$ *const-subst* $c = t \cdot$ *const-subst* d $c \neq d$
and $(c,\ 0) \notin$ *funas-term* t $(d,\ 0) \notin$ *funas-term* s
shows $s = t$ ⟨*proof*⟩

lemma *remove-const-subst-steps*:
assumes *linear-sys* $\mathcal{R}$ **and** $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$ **and** $(d,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and $c \neq d$ $(c,\ 0) \notin$ *funas-term* t $(d,\ 0) \notin$ *funas-term* s
and $(s \cdot$ *const-subst* $c,\ t \cdot$ *const-subst* $d) \in$ (*rstep* $\mathcal{R})^*$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^*$
⟨*proof*⟩

lemma *remove-const-subst-relcomp-lhs*:
assumes *sys*: *linear-sys* $\mathcal{R}$ *linear-sys* $\mathcal{S}$
and *fr*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$ **and** *fs*:$(c,\ 0) \notin$ *funas-rel* $\mathcal{S}$
and *funas*: $(c,\ 0) \notin$ *funas-term* t
and *seq*: $(s \cdot$ *const-subst* $c,\ t) \in$ (*rstep* $\mathcal{R})^*$ *O* (*rstep* $\mathcal{S})^*$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^*$ *O* (*rstep* $\mathcal{S})^*$ ⟨*proof*⟩

lemma *remove-const-subst-relcomp-rhs*:
assumes *sys*: *linear-sys* $\mathcal{R}$ *linear-sys* $\mathcal{S}$
and *fr*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$ **and** *fs*:$(c,\ 0) \notin$ *funas-rel* $\mathcal{S}$
and *funas*: $(c,\ 0) \notin$ *funas-term* s
and *seq*: $(s,\ t \cdot$ *const-subst* $c) \in$ (*rstep* $\mathcal{R})^*$ *O* (*rstep* $\mathcal{S})^*$
shows $(s,\ t) \in$ (*rstep* $\mathcal{R})^*$ *O* (*rstep* $\mathcal{S})^*$
⟨*proof*⟩

lemma *remove-const-subst-relcomp*:
assumes *sys*: *linear-sys* $\mathcal{R}$ *linear-sys* $\mathcal{S}$
and *fr*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$ $(d,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *fs*:$(c,\ 0) \notin$ *funas-rel* $\mathcal{S}$ $(d,\ 0) \notin$ *funas-rel* $\mathcal{S}$
and *diff*: $c \neq d$ **and** *funas*: $(c,\ 0) \notin$ *funas-term* t $(d,\ 0) \notin$ *funas-term* s
and *seq*: $(s \cdot$ *const-subst* $c,\ t \cdot$ *const-subst* $d) \in$ (*rstep* $\mathcal{R})^*$ *O* (*rstep* $\mathcal{S})^*$

shows $(s,\ t) \in (\mathit{rstep}\ \mathcal{R})^* \ O\ (\mathit{rstep}\ \mathcal{S})^*$
⟨*proof*⟩

end

4 Confluence related rewriting properties

theory *Rewriting-Properties*
imports *Rewriting*
Abstract−Rewriting.Abstract-Rewriting
begin

4.1 Confluence related ARS properties

definition $\mathit{SCR\text{-}on}\ r\ A \equiv (\forall\, a \in A.\ \forall\ b\ c.\ (a,\ b) \in r \wedge (a,\ c) \in r \longrightarrow$
$(\exists\ d.\ (b,\ d) \in r^{=} \wedge\ (c,\ d) \in r^*))$

abbreviation $\mathit{SCR} :: {'a}\ \mathit{rel} \Rightarrow \mathit{bool}$ **where** $\mathit{SCR}\ r \equiv \mathit{SCR\text{-}on}\ r\ \mathit{UNIV}$

definition $\mathit{NFP\text{-}on} :: {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{set} \Rightarrow \mathit{bool}$ **where**
$\mathit{NFP\text{-}on}\ r\ A \longleftrightarrow (\forall\, a{\in}A.\ \forall\, b\ c.\ (a,\ b) \in r^* \wedge (a,\ c) \in r^! \longrightarrow (b,\ c) \in r^*)$

abbreviation $\mathit{NFP} :: {'a}\ \mathit{rel} \Rightarrow \mathit{bool}$ **where** $\mathit{NFP}\ r \equiv \mathit{NFP\text{-}on}\ r\ \mathit{UNIV}$

definition $\mathit{CE\text{-}on} :: {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{set} \Rightarrow \mathit{bool}$ **where**
$\mathit{CE\text{-}on}\ r\ s\ A \longleftrightarrow (\forall\, a{\in}A.\ \forall\, b.\ (a,\ b) \in r^{\leftrightarrow *} \longleftrightarrow (a,\ b) \in s^{\leftrightarrow *})$

abbreviation $\mathit{CE} :: {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{rel} \Rightarrow \mathit{bool}$ **where** $\mathit{CE}\ r\ s \equiv \mathit{CE\text{-}on}\ r\ s\ \mathit{UNIV}$

definition $\mathit{NE\text{-}on} :: {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{set} \Rightarrow \mathit{bool}$ **where**
$\mathit{NE\text{-}on}\ r\ s\ A \longleftrightarrow (\forall\, a{\in}A.\ \forall\, b.\ (a,\ b) \in r^! \longleftrightarrow (a,\ b) \in s^!)$

abbreviation $\mathit{NE} :: {'a}\ \mathit{rel} \Rightarrow {'a}\ \mathit{rel} \Rightarrow \mathit{bool}$ **where** $\mathit{NE}\ r\ s \equiv \mathit{NE\text{-}on}\ r\ s\ \mathit{UNIV}$

4.2 Signature closure of relation to model multihole context closure

lemma *all-ctxt-closed-sig-rsteps* [*intro*]:
fixes $\mathcal{R} :: ({'f},{'v})\ \mathit{term}\ \mathit{rel}$
shows $\mathit{all\text{-}ctxt\text{-}closed}\ \mathcal{F}\ ((\mathit{srstep}\ \mathcal{F}\ \mathcal{R})^*)$ (**is** *all-ctxt-closed* - ($?R^*$))
⟨*proof*⟩

lemma *sigstep-trancl-funas*:
$(s,\ t) \in (\mathit{srstep}\ \mathcal{F}\ \mathcal{S})^* \Longrightarrow s \neq t \Longrightarrow \mathit{funas\text{-}term}\ s \subseteq \mathcal{F}$
$(s,\ t) \in (\mathit{srstep}\ \mathcal{F}\ \mathcal{S})^* \Longrightarrow s \neq t \Longrightarrow \mathit{funas\text{-}term}\ t \subseteq \mathcal{F}$
⟨*proof*⟩

lemma *srrstep-to-srestep*:
$(s,\ t) \in \mathit{srrstep}\ \mathcal{F}\ \mathcal{R} \Longrightarrow (s,\ t) \in \mathit{srstep}\ \mathcal{F}\ \mathcal{R}$

⟨*proof*⟩

lemma *srsteps-with-root-step-srstepsD*:
$(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R} \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
⟨*proof*⟩

lemma *srsteps-with-root-step-sresteps-eqD*:
$(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R} \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^*$
⟨*proof*⟩

lemma *symcl-srstep-conversion*:
$(s, t) \in$ *srstep* $\mathcal{F}$ $(\mathcal{R}^{\leftrightarrow}) \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^{\leftrightarrow*}$
⟨*proof*⟩

lemma *symcl-srsteps-conversion*:
$(s, t) \in$ (*srstep* $\mathcal{F}$ $(\mathcal{R}^{\leftrightarrow}))^* \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^{\leftrightarrow*}$
⟨*proof*⟩

lemma *NF-srstep-args*:
assumes *Fun f ss* $\in$ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) *funas-term* (*Fun f ss*) $\subseteq \mathcal{F}$ $i <$ *length ss*
shows *ss* ! $i \in$ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *all-ctxt-closed-srstep-conversions* [*simp*]:
all-ctxt-closed $\mathcal{F}$ ((*srstep* $\mathcal{F}$ $\mathcal{R})^{\leftrightarrow*}$)
⟨*proof*⟩

lemma *NFP-stepD*:
NFP $r \Longrightarrow (a, b) \in r^* \Longrightarrow (a, c) \in r^* \Longrightarrow c \in$ *NF* $r \Longrightarrow (b, c) \in r^*$
⟨*proof*⟩

lemma *NE-symmetric*: *NE r s* $\Longrightarrow$ *NE s r*
⟨*proof*⟩

lemma *CE-symmetric*: *CE r s* $\Longrightarrow$ *CE s r*
⟨*proof*⟩

Reducing the quantification over rewrite sequences for properties *CR* ... to rewrite sequences containing at least one root step

lemma *all-ctxt-closed-sig-reflE*:
all-ctxt-closed $\mathcal{F}$ $\mathcal{R} \Longrightarrow$ *funas-term* $t \subseteq \mathcal{F} \Longrightarrow (t, t) \in \mathcal{R}$
⟨*proof*⟩

lemma *all-ctxt-closed-relcomp* [*intro*]:

$(\bigwedge s\ t.\ (s,\ t) \in \mathcal{R} \Longrightarrow s \neq t \Longrightarrow \textit{funas-term}\ s \subseteq \mathcal{F} \wedge \textit{funas-term}\ t \subseteq \mathcal{F}) \Longrightarrow$
$(\bigwedge s\ t.\ (s,\ t) \in \mathcal{S} \Longrightarrow s \neq t \Longrightarrow \textit{funas-term}\ s \subseteq \mathcal{F} \wedge \textit{funas-term}\ t \subseteq \mathcal{F}) \Longrightarrow$
$\textit{all-ctxt-closed}\ \mathcal{F}\ \mathcal{R} \Longrightarrow \textit{all-ctxt-closed}\ \mathcal{F}\ \mathcal{S} \Longrightarrow \textit{all-ctxt-closed}\ \mathcal{F}\ (\mathcal{R}\ O\ \mathcal{S})$
$\langle \textit{proof} \rangle$

abbreviation $\textit{prop-to-rel}\ P \equiv \{(s,\ t)|\ s\ t.\ P\ s\ t\}$

abbreviation $\textit{prop-mctxt-cl}\ \mathcal{F}\ P \equiv \textit{all-ctxt-closed}\ \mathcal{F}\ (\textit{prop-to-rel}\ P)$

lemma *prop-mctxt-cl-Var*:
$\textit{prop-mctxt-cl}\ \mathcal{F}\ P \Longrightarrow P\ (\textit{Var}\ x)\ (\textit{Var}\ x)$
$\langle \textit{proof} \rangle$

lemma *prop-mctxt-cl-refl-on*:
$\textit{prop-mctxt-cl}\ \mathcal{F}\ P \Longrightarrow \textit{funas-term}\ t \subseteq \mathcal{F} \Longrightarrow P\ t\ t$
$\langle \textit{proof} \rangle$

lemma *prop-mctxt-cl-reflcl-on*:
$\textit{prop-mctxt-cl}\ \mathcal{F}\ P \Longrightarrow \textit{funas-term}\ s \subseteq \mathcal{F} \Longrightarrow P\ s\ s$
$\langle \textit{proof} \rangle$

lemma *reduction-relations-to-root-step*:
assumes $\bigwedge s\ t.\ (s,\ t) \in \textit{srsteps-with-root-step}\ \mathcal{F}\ \mathcal{R} \Longrightarrow P\ s\ t$
and *cl*: $\textit{prop-mctxt-cl}\ \mathcal{F}\ P$
and *well*: $\textit{funas-term}\ s \subseteq \mathcal{F}\ \textit{funas-term}\ t \subseteq \mathcal{F}$
and *steps*: $(s,\ t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^*$
shows $P\ s\ t\ \langle \textit{proof} \rangle$

abbreviation $\textit{comp-rrstep-rel}\ \mathcal{F}\ \mathcal{R}\ \mathcal{S} \equiv \textit{srsteps-with-root-step}\ \mathcal{F}\ \mathcal{R}\ O\ (\textit{srstep}\ \mathcal{F}\ \mathcal{S})^* \cup$
$(\textit{srstep}\ \mathcal{F}\ \mathcal{R})^*\ O\ \textit{srsteps-with-root-step}\ \mathcal{F}\ \mathcal{S}$

abbreviation $\textit{comp-rrstep-rel}'\ \mathcal{F}\ \mathcal{R}\ \mathcal{S} \equiv \textit{srsteps-with-root-step}\ \mathcal{F}\ \mathcal{R}\ O\ (\textit{srstep}\ \mathcal{F}\ \mathcal{S})^+ \cup$
$(\textit{srstep}\ \mathcal{F}\ \mathcal{R})^+\ O\ \textit{srsteps-with-root-step}\ \mathcal{F}\ \mathcal{S}$

lemma *reduction-join-relations-to-root-step*:
assumes $\bigwedge s\ t.\ (s,\ t) \in \textit{comp-rrstep-rel}\ \mathcal{F}\ \mathcal{R}\ \mathcal{S} \Longrightarrow P\ s\ t$
and *cl*: $\textit{prop-mctxt-cl}\ \mathcal{F}\ P$
and *well*: $\textit{funas-term}\ s \subseteq \mathcal{F}\ \textit{funas-term}\ t \subseteq \mathcal{F}$
and *steps*: $(s,\ t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^*\ O\ (\textit{srstep}\ \mathcal{F}\ \mathcal{S})^*$
shows $P\ s\ t\ \langle \textit{proof} \rangle$

definition $\textit{commute-redp}\ \mathcal{F}\ \mathcal{R}\ \mathcal{S}\ s\ t \longleftrightarrow (s,\ t) \in ((\textit{srstep}\ \mathcal{F}\ \mathcal{S})^*\ O\ ((\textit{srstep}\ \mathcal{F}\ \mathcal{R})^{-1})^*)$

declare *subsetI*[*rule del*]
lemma *commute-redp-mctxt-cl*:
prop-mctxt-cl $\mathcal{F}$ (*commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$)
⟨*proof*⟩
declare *subsetI*[*intro*!]

lemma *commute-rrstep-intro*:
assumes $\bigwedge$ *s t*. (*s*, *t*) ∈ *comp-rrstep-rel′* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{S}$ $\Longrightarrow$ *commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t*
shows *commute* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *commute-to-rrstep*:
assumes *commute* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *comp-rrstep-rel* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{S}$ $\Longrightarrow$ *commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t*
⟨*proof*⟩

lemma *CR-Aux*:
assumes $\bigwedge$ *s t*. (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ ($\mathcal{R}^{-1}$))* *O srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ $\Longrightarrow$ *commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{R}$ *s t*
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *comp-rrstep-rel* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{R}$ $\Longrightarrow$ *commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{R}$ *s t*
⟨*proof*⟩

lemma *CR-rrstep-intro*:
assumes $\bigwedge$ *s t*. (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ ($\mathcal{R}^{-1}$))$^+$ *O srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ $\Longrightarrow$ *commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{R}$ *s t*
shows *CR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *CR-to-rrstep*:
assumes *CR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *comp-rrstep-rel* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{R}$ $\Longrightarrow$ *commute-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{R}$ *s t*
⟨*proof*⟩

definition *NFP-redp* **where**
NFP-redp $\mathcal{F}$ $\mathcal{R}$ *s t* $\longleftrightarrow$ *t* ∈ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) $\longrightarrow$ (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{R}$)*

lemma *prop-mctxt-cl-NFP-redp*:
prop-mctxt-cl $\mathcal{F}$ (*NFP-redp* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *NFP-rrstep-intro*:
assumes $\bigwedge$ *s t*. (*s*, *t*) ∈ *comp-rrstep-rel′* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{R}$ $\Longrightarrow$ *NFP-redp* $\mathcal{F}$ $\mathcal{R}$ *s t*
shows *NFP* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *NFP-lift-to-conversion*:
assumes *NFP r* (*s*, *t*) ∈ ($r^{\leftrightarrow}$)* **and** *t* ∈ *NF r*
shows (*s*, *t*) ∈ r^* ⟨*proof*⟩

lemma *NFP-to-rrstep*:
assumes *NFP* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{R}^{\leftrightarrow}$) ⟹ *NFP-redp* $\mathcal{F}$ $\mathcal{R}$ *s t*
⟨*proof*⟩

definition *UN-redp* $\mathcal{F}$ $\mathcal{R}$ *s t* ⟷ *s* ∈ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) ∧ *t* ∈ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) ⟶ *s* = *t*

lemma *prop-mctxt-cl-UN-redp*:
prop-mctxt-cl $\mathcal{F}$ (*UN-redp* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *UNC-rrstep-intro*:
assumes$\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{R}^{\leftrightarrow}$) ⟹ *UN-redp* $\mathcal{F}$ $\mathcal{R}$ *s t*
shows *UNC* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *UNC-to-rrstep*:
assumes *UNC* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{R}^{\leftrightarrow}$) ⟹ *UN-redp* $\mathcal{F}$ $\mathcal{R}$ *s t*
⟨*proof*⟩

lemma *UNF-rrstep-intro*:
assumes $\bigwedge$ *t u*. (*t*, *u*) ∈ *comp-rrstep-rel′* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{R}$ ⟹ *UN-redp* $\mathcal{F}$ $\mathcal{R}$ *t u*
shows *UNF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *UNF-to-rrstep*:
assumes *UNF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *comp-rrstep-rel* $\mathcal{F}$ ($\mathcal{R}^{-1}$) $\mathcal{R}$ ⟹ *UN-redp* $\mathcal{F}$ $\mathcal{R}$ *s t*
⟨*proof*⟩

lemma *CE-rrstep-intro*:
assumes $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{R}^{\leftrightarrow}$) ⟹ (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{S}$)$^{\leftrightarrow *}$
and $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{S}^{\leftrightarrow}$) ⟹ (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{R}$)$^{\leftrightarrow *}$
shows *CE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *CE-to-rrstep*:
assumes *CE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{R}^{\leftrightarrow}$) ⟹ (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{S}$)$^{\leftrightarrow *}$
$\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ ($\mathcal{S}^{\leftrightarrow}$) ⟹ (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{R}$)$^{\leftrightarrow *}$
⟨*proof*⟩

definition *NE-redp* **where**
NE-redp $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t* ⟷ *t* ∈ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) ⟶ *t* ∈ *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) ⟶ (*s*, *t*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{S}$)*

lemma *prop-mctxt-cl-NE-redp*:
prop-mctxt-cl $\mathcal{F}$ (*NE-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *NE-rrstep-intro*:
assumes $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ ⟹ *NE-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t*
and $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{S}$ ⟹ *NE-redp* $\mathcal{F}$ $\mathcal{S}$ $\mathcal{R}$ *s t*
and *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) = *NF* (*srstep* $\mathcal{F}$ $\mathcal{S}$)
shows *NE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *NE-to-rrstep*:
assumes *NE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
shows $\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ ⟹ *NE-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t*
$\bigwedge$ *s t*. (*s*, *t*) ∈ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{S}$ ⟹ *NE-redp* $\mathcal{F}$ $\mathcal{S}$ $\mathcal{R}$ *s t*
⟨*proof*⟩

lemma *NE-NF-eq*:
NE $\mathcal{R}$ $\mathcal{S}$ ⟹ *NF* $\mathcal{R}$ = *NF* $\mathcal{S}$
⟨*proof*⟩

abbreviation *SCRp* $\mathcal{F}$ $\mathcal{R}$ *t u* ≡ ∃ *v*. (*t*, *v*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{R}$)$^=$ ∧ (*u*, *v*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{R}$)*
lemma *SCR-rrstep-intro*:
assumes $\bigwedge$ *s t u*. (*s*, *t*) ∈ *sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$) ⟹ (*s*, *u*) ∈ *srstep* $\mathcal{F}$ $\mathcal{R}$ ⟹ *SCRp* $\mathcal{F}$ $\mathcal{R}$ *t u*
and $\bigwedge$ *s t u*. (*s*, *t*) ∈ *srstep* $\mathcal{F}$ $\mathcal{R}$ ⟹ (*s*, *u*) ∈ *sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$) ⟹ *SCRp* $\mathcal{F}$ $\mathcal{R}$ *t u*
shows *SCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *SCE-to-rrstep*:
assumes *SCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $\bigwedge$ *s t u*. (*s*, *t*) ∈ *sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$) ⟹ (*s*, *u*) ∈ *srstep* $\mathcal{F}$ $\mathcal{R}$ ⟹ *SCRp* $\mathcal{F}$ $\mathcal{R}$ *t u*
$\bigwedge$ *s t u*. (*s*, *t*) ∈ *srstep* $\mathcal{F}$ $\mathcal{R}$ ⟹ (*s*, *u*) ∈ *sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$) ⟹ *SCRp* $\mathcal{F}$ $\mathcal{R}$ *t u*
⟨*proof*⟩

lemma *WCR-rrstep-intro*:
assumes $\bigwedge$ *s t u*. (*s*, *t*) ∈ *sig-step* $\mathcal{F}$ (*rrstep* $\mathcal{R}$) ⟹ (*s*, *u*) ∈ *srstep* $\mathcal{F}$ $\mathcal{R}$ ⟹ (*t*, *u*) ∈ (*srstep* $\mathcal{F}$ $\mathcal{R}$)$^\downarrow$
shows *WCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

end

theory *Rewriting-LLRG-LV-Mondaic*
imports *Rewriting*
Replace-Constant
begin

4.3 Specific results about rewriting under a linear variable-separated system

lemma *card-varposs-ground*:
$card\ (varposs\ s) = 0 \longleftrightarrow ground\ s$
⟨*proof*⟩

lemma *poss-of-term-subst-apply-varposs*:
assumes $p \in poss\text{-}of\text{-}term\ (constT\ c)\ (s \cdot \sigma)\ (c,\ 0) \notin funas\text{-}term\ s$
shows $\exists\ q.\ q \in varposs\ s \wedge q \leq_p p$ ⟨*proof*⟩

lemma *poss-of-term-hole-poss*:
assumes $p \in poss\text{-}of\text{-}term\ t\ C\langle s\rangle$ **and** $hole\text{-}pos\ C \leq_p p$
shows $p -_p hole\text{-}pos\ C \in poss\text{-}of\text{-}term\ t\ s$ ⟨*proof*⟩

lemma *remove-const-subst-from-match*:
assumes $s \cdot const\text{-}subst\ c = C\langle l \cdot \sigma\rangle\ (c,\ 0) \notin funas\text{-}term\ l\ linear\text{-}term\ l$
shows $\exists\ D\ \tau.\ s = D\langle l \cdot \tau\rangle$ ⟨*proof*⟩

definition $llrg\ \mathcal{R} \longleftrightarrow (\forall\ (l,\ r) \in \mathcal{R}.\ linear\text{-}term\ l \wedge ground\ r)$

definition $lv\ \mathcal{R} \longleftrightarrow (\forall\ (l,\ r) \in \mathcal{R}.\ linear\text{-}term\ l \wedge linear\text{-}term\ r \wedge vars\text{-}term\ l \cap vars\text{-}term\ r = \{\})$

definition $monadic\ \mathcal{F} \longleftrightarrow (\forall\ (f,\ n) \in \mathcal{F}.\ n \leq Suc\ 0)$

— NF of ground terms

lemma *ground-NF-srstep-gsrstep*:
$ground\ s \Longrightarrow s \in NF\ (srstep\ \mathcal{F}\ \mathcal{R}) \Longrightarrow s \in NF\ (gsrstep\ \mathcal{F}\ \mathcal{R})$
⟨*proof*⟩

lemma *NF-to-fresh-const-subst-NF*:
assumes *lin*: $linear\text{-}sys\ \mathcal{R}$ **and** *fresh-const*: $(c,\ 0) \notin funas\text{-}rel\ \mathcal{R}\ funas\text{-}rel\ \mathcal{R} \subseteq \mathcal{F}$
and *nf-f*: $funas\text{-}term\ s \subseteq \mathcal{F}\ s \in NF\ (srstep\ \mathcal{F}\ \mathcal{R})$
shows $s \cdot const\text{-}subst\ c \in NF\ (gsrstep\ \mathcal{H}\ \mathcal{R})$
⟨*proof*⟩

lemma *fresh-const-subst-NF-pres*:
assumes *fresh-const*: $(c, 0) \notin$ *funas-rel* $\mathcal{R}$ *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$
and *nf-f*: *funas-term* $s \subseteq \mathcal{F}$ $\mathcal{F} \subseteq \mathcal{H}$ $(c, 0) \in \mathcal{H}$ $s \cdot$ *const-subst* $c \in NF$ (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows $s \in NF$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *linear-sys-gNF-eq-NF-eq*:
assumes *lin*: *linear-sys* $\mathcal{R}$ *linear-sys* $\mathcal{S}$
and *well*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *fresh*: $(c, 0) \notin$ *funas-rel* $\mathcal{R}$ $(c, 0) \notin$ *funas-rel* $\mathcal{S}$
and *lift*: $\mathcal{F} \subseteq \mathcal{H}$ $(c, 0) \in \mathcal{H}$
and *nf*: NF (*gsrstep* $\mathcal{H}$ $\mathcal{R}$) $= NF$ (*gsrstep* $\mathcal{H}$ $\mathcal{S}$)
shows NF (*srstep* $\mathcal{F}$ $\mathcal{R}$) $= NF$ (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩
lemma *gsrsteps-to-srsteps*:
$(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^+ \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
⟨*proof*⟩

lemma *gsrsteps-eq-to-srsteps-eq*:
$(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^* \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^*$
⟨*proof*⟩

lemma *gsrsteps-to-rsteps*:
$(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^+ \Longrightarrow (s, t) \in$ (*rstep* $\mathcal{R})^+$
⟨*proof*⟩

lemma *gsrsteps-eq-to-rsteps-eq*:
$(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^* \Longrightarrow (s, t) \in$ (*rstep* $\mathcal{R})^*$
⟨*proof*⟩

lemma *gsrsteps-eq-relcomp-srsteps-relcompD*:
$(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^*$ O (*gsrstep* $\mathcal{F}$ $\mathcal{S})^* \Longrightarrow (s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^*$ O (*srstep* $\mathcal{F}$ $\mathcal{S})^*$
⟨*proof*⟩

lemma *gsrsteps-eq-relcomp-to-rsteps-relcomp*:
$(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^*$ O (*gsrstep* $\mathcal{F}$ $\mathcal{S})^* \Longrightarrow (s, t) \in$ (*rstep* $\mathcal{R})^*$ O (*rstep* $\mathcal{S})^*$
⟨*proof*⟩

lemma *ground-srsteps-gsrsteps*:
assumes *ground* s *ground* t
and $(s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
shows $(s, t) \in$ (*gsrstep* $\mathcal{F}$ $\mathcal{R})^+$
⟨*proof*⟩

lemma *ground-srsteps-eq-gsrsteps-eq*:
assumes *ground s ground t*
and $(s, t) \in (srstep\ \mathcal{F}\ \mathcal{R})^*$
shows $(s, t) \in (gsrstep\ \mathcal{F}\ \mathcal{R})^*$
⟨*proof*⟩

lemma *srsteps-eq-relcomp-gsrsteps-relcomp*:
assumes $(s, t) \in (srstep\ \mathcal{F}\ \mathcal{R})^*\ O\ (srstep\ \mathcal{F}\ \mathcal{S})^*$
and *ground s ground t*
shows $(s, t) \in (gsrstep\ \mathcal{F}\ \mathcal{R})^*\ O\ (gsrstep\ \mathcal{F}\ \mathcal{S})^*$
⟨*proof*⟩

lemma *llrg-ground-rhs*:
$llrg\ \mathcal{R} \Longrightarrow (l, r) \in \mathcal{R} \Longrightarrow ground\ r$
⟨*proof*⟩

lemma *llrg-rrsteps-groundness*:
assumes *llrg* $\mathcal{R}$ **and** $(s, t) \in (srrstep\ \mathcal{F}\ \mathcal{R})$
shows *ground t* ⟨*proof*⟩

lemma *llrg-rsteps-pres-groundness*:
assumes *llrg* $\mathcal{R}$ *ground s*
and $(s, t) \in (srstep\ \mathcal{F}\ \mathcal{R})^*$
shows *ground t* ⟨*proof*⟩

lemma *llrg-srsteps-with-root-step-ground*:
assumes *llrg* $\mathcal{R}$ **and** $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}\ \mathcal{R}$
shows *ground t* ⟨*proof*⟩

lemma *llrg-srsteps-with-root-step-inv-ground*:
assumes *llrg* $\mathcal{R}$ **and** $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}\ (\mathcal{R}^{-1})$
shows *ground s* ⟨*proof*⟩

lemma *llrg-funas-term-step-pres*:
assumes *llrg* $\mathcal{R}$ **and** $(s, t) \in (rstep\ \mathcal{R})$
shows *funas-term t* $\subseteq$ *funas-rel* $\mathcal{R} \cup$ *funas-term s*
⟨*proof*⟩

lemma *llrg-funas-term-steps-pres*:
assumes *llrg* $\mathcal{R}$ **and** $(s, t) \in (rstep\ \mathcal{R})^*$
shows *funas-term t* $\subseteq$ *funas-rel* $\mathcal{R} \cup$ *funas-term s*
⟨*proof*⟩

lemma *monadic-ground-ctxt-apply*:
monadic $\mathcal{F} \Longrightarrow$ *funas-ctxt* $C \subseteq \mathcal{F} \Longrightarrow ground\ r \Longrightarrow ground\ C\langle r\rangle$
⟨*proof*⟩

lemma *llrg-monadic-rstep-pres-groundness*:
assumes *llrg* $\mathcal{R}$ *monadic* $\mathcal{F}$
and $(s, t) \in$ *srstep* $\mathcal{F}$ $\mathcal{R}$
shows *ground t* ⟨*proof*⟩

lemma *llrg-monadic-rsteps-groundness*:
assumes *llrg* $\mathcal{R}$ *monadic* $\mathcal{F}$
and $(s, t) \in (srstep\ \mathcal{F}\ \mathcal{R})^+$
shows *ground t* ⟨*proof*⟩

fun *monadic-term* **where**
monadic-term (*Var x*) = *True*
| *monadic-term* (*Fun f* []) = *True*
| *monadic-term* (*Fun f ts*) = (*length ts* = *Suc 0* ∧ *monadic-term* (*hd ts*))

fun *monadic-get-leave* **where**
monadic-get-leave (*Var x*) = (*Var x*)
| *monadic-get-leave* (*Fun f* []) = *Fun f* []
| *monadic-get-leave* (*Fun f ts*) = *monadic-get-leave* (*hd ts*)

fun *monadic-replace-leave* **where**
monadic-replace-leave t (*Var x*) = *t*
| *monadic-replace-leave t* (*Fun f* []) = *t*
| *monadic-replace-leave t* (*Fun f ts*) = *Fun f* [*monadic-replace-leave t* (*hd ts*)]

lemma *monadic-replace-leave-undo-const-subst*:
assumes *monadic-term s*
shows *monadic-replace-leave* (*monadic-get-leave s*) ($s \cdot$ *const-subst c*) = *s* ⟨*proof*⟩

lemma *monadic-replace-leave-context*:
assumes *monadic-term* $C\langle s\rangle$
shows *monadic-replace-leave t* $C\langle s\rangle = C\langle$*monadic-replace-leave t s*$\rangle$ ⟨*proof*⟩

lemma *monadic-replace-leave-subst*:
assumes *monadic-term* $(s \cdot \sigma)$ ¬ *ground s*
shows *monadic-replace-leave t* $(s \cdot \sigma) = s \cdot (\lambda\ x.$ *monadic-replace-leave t* $(\sigma\ x))$
⟨*proof*⟩

lemma *monadic-sig*:
monadic $\mathcal{F} \Longrightarrow (f,$ *length ts*$) \in \mathcal{F} \Longrightarrow$ *length ts* ≤ *Suc 0*
⟨*proof*⟩

lemma *monadic-sig-funas-term-mt*:
monadic $\mathcal{F} \Longrightarrow$ *funas-term s* $\subseteq \mathcal{F} \Longrightarrow$ *monadic-term s*
⟨*proof*⟩

lemma *monadic-term-const-pres* [*intro*]:
monadic-term s ⟹ *monadic-term* ($s \cdot$ *const-subst c*)

⟨*proof*⟩

lemma *remove-const-lv-mondaic-step-lhs*:
assumes *lv*: *lv* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *mon*: *monadic* $\mathcal{F}$
and *step*: $(s \cdot$ *const-subst* $c,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $(s,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *remove-const-lv-mondaic-step-rhs*:
assumes *lv*: *lv* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *mon*: *monadic* $\mathcal{F}$
and *step*: $(s,\ t \cdot$ *const-subst* $c) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)
shows $(s,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *remove-const-lv-mondaic-steps-lhs*:
assumes *lv*: *lv* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *mon*: *monadic* $\mathcal{F}$
and *steps*: $(s \cdot$ *const-subst* $c,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
shows $(s,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
⟨*proof*⟩

lemma *remove-const-lv-mondaic-steps-rhs*:
assumes *lv*: *lv* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *mon*: *monadic* $\mathcal{F}$
and *steps*: $(s,\ t \cdot$ *const-subst* $c) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
shows $(s,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
⟨*proof*⟩

lemma *remove-const-lv-mondaic-steps*:
assumes *lv*: *lv* $\mathcal{R}$ **and** *fresh*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$
and *mon*: *monadic* $\mathcal{F}$
and *steps*: $(s \cdot$ *const-subst* $c,\ t \cdot$ *const-subst* $c) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
shows $(s,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{R})^+$
⟨*proof*⟩

lemma *lv-root-step-idep-subst*:
assumes *lv* $\mathcal{R}$
and $(s,\ t) \in$ *srrstep* $\mathcal{F}$ $\mathcal{R}$
and *well*: $\bigwedge x.$ *funas-term* $(\sigma\ x) \subseteq \mathcal{F}$ $\bigwedge x.$ *funas-term* $(\tau\ x) \subseteq \mathcal{F}$
shows $(s \cdot \sigma,\ t \cdot \tau) \in$ *srrstep* $\mathcal{F}$ $\mathcal{R}$
⟨*proof*⟩

lemma *lv-srsteps-with-root-step-idep-subst*:
assumes *lv* $\mathcal{R}$
and $(s,\ t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$

and *well*: $\bigwedge$ *x*. *funas-term* (σ *x*) $\subseteq \mathcal{F}$ $\bigwedge$ *x*. *funas-term* (τ *x*) $\subseteq \mathcal{F}$
shows ($s \cdot \sigma$, $t \cdot \tau$) $\in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ ⟨*proof*⟩

end
theory *Rewriting-GTRS*
imports *Rewriting*
Replace-Constant
begin

4.4 Specific results about rewriting under a ground system

abbreviation *ground-sys* $\mathcal{R} \equiv (\forall\ (s, t) \in \mathcal{R}.\ \textit{ground}\ s \wedge \textit{ground}\ t)$

lemma *srrstep-ground*:
assumes *ground-sys* $\mathcal{R}$
and (s, t) $\in$ *srrstep* $\mathcal{F}$ $\mathcal{R}$
shows *ground s ground t* ⟨*proof*⟩

lemma *srstep-pres-ground-l*:
assumes *ground-sys* $\mathcal{R}$ *ground s*
and (s, t) $\in$ *srstep* $\mathcal{F}$ $\mathcal{R}$
shows *ground t* ⟨*proof*⟩

lemma *srstep-pres-ground-r*:
assumes *ground-sys* $\mathcal{R}$ *ground t*
and (s, t) $\in$ *srstep* $\mathcal{F}$ $\mathcal{R}$
shows *ground s* ⟨*proof*⟩

lemma *srsteps-pres-ground-l*:
assumes *ground-sys* $\mathcal{R}$ *ground s*
and (s, t) $\in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)$^+$
shows *ground t* ⟨*proof*⟩

lemma *srsteps-pres-ground-r*:
assumes *ground-sys* $\mathcal{R}$ *ground t*
and (s, t) $\in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)$^+$
shows *ground s* ⟨*proof*⟩

lemma *srsteps-eq-pres-ground-l*:
assumes *ground-sys* $\mathcal{R}$ *ground s*
and (s, t) $\in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)*
shows *ground t* ⟨*proof*⟩

lemma *srsteps-eq-pres-ground-r*:
assumes *ground-sys* $\mathcal{R}$ *ground t*
and (s, t) $\in$ (*srstep* $\mathcal{F}$ $\mathcal{R}$)*
shows *ground s* ⟨*proof*⟩

lemma *srsteps-with-root-step-ground*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$
shows *ground s ground t* ⟨*proof*⟩

4.5 funas

lemma *srrstep-funas*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in$ *srrstep* $\mathcal{F}$ $\mathcal{R}$
shows *funas-term s* $\subseteq$ *funas-rel* $\mathcal{R}$ *funas-term t* $\subseteq$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srstep-funas-l*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in$ *srstep* $\mathcal{F}$ $\mathcal{R}$
shows *funas-term t* $\subseteq$ *funas-term s* $\cup$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srstep-funas-r*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in$ *srstep* $\mathcal{F}$ $\mathcal{R}$
shows *funas-term s* $\subseteq$ *funas-term t* $\cup$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-funas-l*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^+$
shows *funas-term t* $\subseteq$ *funas-term s* $\cup$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-funas-r*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^+$
shows *funas-term s* $\subseteq$ *funas-term t* $\cup$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-eq-funas-l*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^*$
shows *funas-term t* $\subseteq$ *funas-term s* $\cup$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-eq-funas-r*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^*$
shows *funas-term s* $\subseteq$ *funas-term t* $\cup$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩

lemma *srsteps-with-root-step-funas*:
assumes *ground-sys* $\mathcal{R}$
and $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$
shows *funas-term s* $\subseteq$ *funas-rel* $\mathcal{R}$ *funas-term t* $\subseteq$ *funas-rel* $\mathcal{R}$
⟨*proof*⟩

end

5 Reducing Rewrite Properties to Properties on Ground Terms over Left-Linear Right-Ground Systems

theory *Ground-Reduction-on-LLRG*
imports
Rewriting-Properties
Rewriting-LLRG-LV-Mondaic
begin

lemma *llrg-linear-sys*:
llrg $\mathcal{R} \Longrightarrow$ *linear-sys* $\mathcal{R}$
⟨*proof*⟩

6 LLRG results

lemma *llrg-commute*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *fresh*: $(c,\ 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$ *llrg* $\mathcal{S}$
and *com*: *commute* (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{S}$)
shows *commute* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *llrg-CR*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$
and *fresh*: $(c,\ 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$
and *com*: *CR* (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{R}$)
shows *CR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *llrg-SCR*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ **and** *fresh*: $(c,\ 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$
and *scr*: *SCR* (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{R}$)
shows *SCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *llrg-WCR*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ **and** *fresh*: $(c,\ 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$
and *wcr*: *WCR* (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{R}$)
shows *WCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *llrg-UNF*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ **and** *fresh*: $(c, 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$
and *unf*: *UNF* (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{R}$)
shows *UNF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *llrg-NFP*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ **and** *fresh*: $(c, 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$
and *nfp*: *NFP* (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{R}$)
shows *NFP* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *llrg-NE-aux*:
assumes $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$
and *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$ **and** *fresh*: $(c, 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$ *llrg* $\mathcal{S}$
and *ne*: *NE* (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{S}$)
shows *NE-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t*
⟨*proof*⟩

lemma *llrg-NE*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$ **and** *fresh*: $(c, 0) \notin \mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$ *llrg* $\mathcal{S}$
and *ne*: *NE* (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{S}$)
shows *NE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

6.1 Specialized for monadic signature

lemma *monadic-commute*:
assumes *llrg* $\mathcal{R}$ *llrg* $\mathcal{S}$ *monadic* $\mathcal{F}$
and *com*: *commute* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows *commute* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *monadic-CR*:
assumes *llrg* $\mathcal{R}$ *monadic* $\mathcal{F}$
and *CR* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *CR* (*srstep* $\mathcal{F}$ $\mathcal{R}$) ⟨*proof*⟩

lemma *monadic-SCR*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *monadic* $\mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$

and *scr*: *SCR* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *SCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *monadic-WCR*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *monadic* $\mathcal{F}$
and *llrg*: *llrg* $\mathcal{R}$
and *wcr*: *WCR* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *WCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *monadic-UNF*:
assumes *llrg* $\mathcal{R}$ *monadic* $\mathcal{F}$
and *unf*: *UNF* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *UNF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *monadic-UNC*:
assumes *llrg* $\mathcal{R}$ *monadic* $\mathcal{F}$
and *well*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$
and *unc*: *UNC* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *UNC* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *monadic-NFP*:
assumes *llrg* $\mathcal{R}$ *monadic* $\mathcal{F}$
and *nfp*: *NFP* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *NFP* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

end

7 Reducing Rewrite Properties to Properties on Ground Terms over Ground Systems

theory *Ground-Reduction-on-GTRS*
imports
Rewriting-Properties
Rewriting-GTRS
Rewriting-LLRG-LV-Mondaic
begin

lemma *ground-sys-nf-eq-lift*:
fixes $\mathcal{R}$ :: ('*f*, '*v*) *term rel*
assumes *gtrs*: *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$

and *nf*: *NF* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) = *NF* (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows *NF* (*srstep* $\mathcal{F}$ $\mathcal{R}$) = *NF* (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *ground-sys-inv*:
ground-sys $\mathcal{R}$ $\Longrightarrow$ *ground-sys* ($\mathcal{R}^{-1}$) ⟨*proof*⟩

lemma *ground-sys-symcl*:
ground-sys $\mathcal{R}$ $\Longrightarrow$ *ground-sys* ($\mathcal{R}^{\leftrightarrow}$) ⟨*proof*⟩

lemma *ground-sys-comp-rrstep-rel′-ground*:
assumes *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$
and (*s*, *t*) ∈ *comp-rrstep-rel′* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$
shows *ground s ground t*
⟨*proof*⟩

lemma *GTRS-commute*:
assumes *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$
and *com*: *commute* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows *commute* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *GTRS-CR*:
assumes *ground-sys* $\mathcal{R}$
and *CR* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *CR* (*srstep* $\mathcal{F}$ $\mathcal{R}$) ⟨*proof*⟩

lemma *GTRS-SCR*:
assumes *gtrs*: *ground-sys* $\mathcal{R}$
and *scr*: *SCR* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *SCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *GTRS-WCR*:
assumes *gtrs*: *ground-sys* $\mathcal{R}$
and *wcr*: *WCR* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *WCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *GTRS-UNF*:
assumes *gtrs*: *ground-sys* $\mathcal{R}$
and *unf*: *UNF* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *UNF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *GTRS-UNC*:
assumes *gtrs*: *ground-sys* $\mathcal{R}$

and *unc*: *UNC* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *UNC* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *GTRS-NFP*:
assumes *ground-sys* $\mathcal{R}$
and *nfp*: *NFP* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$)
shows *NFP* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *GTRS-NE-aux*:
assumes $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$
and *gtrs*: *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$
and *ne*: *NE* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows *NE-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ s t
⟨*proof*⟩

lemma *GTRS-NE*:
assumes *gtrs*: *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$
and *ne*: *NE* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows *NE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *gtrs-CE-aux*:
assumes *step*:$(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $(\mathcal{R}^{\leftrightarrow})$
and *gtrs*: *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$
and *ce*: *CE* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows $(s, t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{S})^{\leftrightarrow *}$
⟨*proof*⟩

lemma *gtrs-CE*:
assumes *gtrs*: *ground-sys* $\mathcal{R}$ *ground-sys* $\mathcal{S}$
and *ce*: *CE* (*gsrstep* $\mathcal{F}$ $\mathcal{R}$) (*gsrstep* $\mathcal{F}$ $\mathcal{S}$)
shows *CE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

end

8 Reducing Rewrite Properties to Properties on Ground Terms over Linear Variable-Separated Systems

theory *Ground-Reduction-on-LV*
imports

Rewriting-Properties
Rewriting-LLRG-LV-Mondaic
begin

lemma *lv-linear-sys*: *lv* $\mathcal{R} \Longrightarrow$ *linear-sys* $\mathcal{R}$
⟨*proof*⟩

lemma *comp-rrstep-rel′-sig-mono*:
$\mathcal{F} \subseteq \mathcal{G} \Longrightarrow$ *comp-rrstep-rel′* $\mathcal{F}\ \mathcal{R}\ \mathcal{S} \subseteq$ *comp-rrstep-rel′* $\mathcal{G}\ \mathcal{R}\ \mathcal{S}$
⟨*proof*⟩

lemma *srsteps-eqD*: $(s,\ t) \in$ (*srstep* $\mathcal{F}\ \mathcal{R})^* \Longrightarrow (s,\ t) \in$ (*rstep* $\mathcal{R})^*$
⟨*proof*⟩

9 Linear-variable separated results

locale *open-terms-two-const-lv* =
fixes $\mathcal{R}$ :: ($'f$, $'v$) *term rel* **and** $\mathcal{F}$ *c d*
assumes *lv*: *lv* $\mathcal{R}$ **and** *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$
and *fresh*: $(c,\ 0) \notin \mathcal{F}\ (d,\ 0) \notin \mathcal{F}$
and *diff*: $c \neq d$
begin

abbreviation $\mathcal{H} \equiv$ *insert* $(c,\ 0)$ (*insert* $(d,0)\ \mathcal{F})$
abbreviation $\sigma_c \equiv$ *const-subst c*
abbreviation $\sigma_d \equiv$ *const-subst d*

lemma *sig-mono*: $\mathcal{F} \subseteq \mathcal{H}$ ⟨*proof*⟩
lemma *fresh-sym-c*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩
lemma *fresh-sym-d*: $(d,\ 0) \notin$ *funas-rel* $\mathcal{R}$ ⟨*proof*⟩
lemma *fresh-sym-c-inv*: $(c,\ 0) \notin$ *funas-rel* $(\mathcal{R}^{-1})$ ⟨*proof*⟩
lemma *fresh-sym-d-inv*: $(d,\ 0) \notin$ *funas-rel* $(\mathcal{R}^{-1})$ ⟨*proof*⟩

lemmas *all-fresh* = *fresh-sym-c fresh-sym-d fresh-sym-c-inv fresh-sym-d-inv*

lemma *sig-inv*: *funas-rel* $(\mathcal{R}^{-1}) \subseteq \mathcal{F}$ ⟨*proof*⟩
lemma *lv-inv*: *lv* $(\mathcal{R}^{-1})$ ⟨*proof*⟩
lemma *well-subst*:
$\bigwedge x.$ *funas-term* ((*const-subst c*) *x*) $\subseteq \mathcal{H}$
$\bigwedge x.$ *funas-term* ((*const-subst d*) *x*) $\subseteq \mathcal{H}$
⟨*proof*⟩

lemma *srsteps-with-root-step-to-grsteps*:
assumes $(s,\ t) \in$ *srsteps-with-root-step* $\mathcal{F}\ \mathcal{R}$
shows $(s \cdot \sigma_c,\ t \cdot \sigma_d) \in$ (*gsrstep* $\mathcal{H}\ \mathcal{R})^*$
⟨*proof*⟩

lemma *comp-rrstep-rel′-to-grsteps*:

assumes $(s,\ t) \in$ *comp-rrstep-rel′* $\mathcal{F}$ $(\mathcal{R}^{-1})$ $\mathcal{R}$
shows $(s \cdot \sigma_c,\ t \cdot \sigma_d) \in (\textit{gsrstep}\ \mathcal{H}\ (\mathcal{R}^{-1}))^*\ O\ (\textit{gsrstep}\ \mathcal{H}\ \mathcal{R})^*$
⟨*proof*⟩

lemma *gsrsteps-eq-to-srsteps*:
assumes $(s \cdot \sigma_c,\ t \cdot \sigma_d) \in (\textit{gsrstep}\ \mathcal{H}\ \mathcal{R})^*$
and *funas*: *funas-term* $s \subseteq \mathcal{F}$ *funas-term* $t \subseteq \mathcal{F}$
shows $(s,\ t) \in (\textit{srstep}\ \mathcal{F}\ \mathcal{R})^*$
⟨*proof*⟩

lemma *convert-NF-to-GNF*:
funas-term $t \subseteq \mathcal{F} \Longrightarrow t \in \textit{NF}\ (\textit{srstep}\ \mathcal{F}\ \mathcal{R}) \Longrightarrow t \cdot \sigma_c \in \textit{NF}\ (\textit{gsrstep}\ \mathcal{H}\ \mathcal{R})$
funas-term $t \subseteq \mathcal{F} \Longrightarrow t \in \textit{NF}\ (\textit{srstep}\ \mathcal{F}\ \mathcal{R}) \Longrightarrow t \cdot \sigma_d \in \textit{NF}\ (\textit{gsrstep}\ \mathcal{H}\ \mathcal{R})$
⟨*proof*⟩

lemma *convert-GNF-to-NF*:
funas-term $t \subseteq \mathcal{F} \Longrightarrow t \cdot \sigma_c \in \textit{NF}\ (\textit{gsrstep}\ \mathcal{H}\ \mathcal{R}) \Longrightarrow t \in \textit{NF}\ (\textit{srstep}\ \mathcal{F}\ \mathcal{R})$
funas-term $t \subseteq \mathcal{F} \Longrightarrow t \cdot \sigma_d \in \textit{NF}\ (\textit{gsrstep}\ \mathcal{H}\ \mathcal{R}) \Longrightarrow t \in \textit{NF}\ (\textit{srstep}\ \mathcal{F}\ \mathcal{R})$
⟨*proof*⟩

lemma *lv-CR*:
assumes *cr*: *CR* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows *CR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *lv-WCR*:
assumes *wcr*: *WCR* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows *WCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *lv-NFP*:
assumes *nfp*: *NFP* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows *NFP* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *lv-UNF*:
assumes *unf*: *UNF* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows *UNF* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *lv-UNC*:
assumes *unc*: *UNC* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows *UNC* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

lemma *lv-SCR*:

assumes *scr*: *SCR* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$)
shows *SCR* (*srstep* $\mathcal{F}$ $\mathcal{R}$)
⟨*proof*⟩

end

locale *open-terms-two-const-lv-two-sys* =
open-terms-two-const-lv $\mathcal{R}$
for $\mathcal{R}$ :: (*'f*, *'v*) *term rel* +
fixes $\mathcal{S}$ :: (*'f*, *'v*) *term rel*
assumes *lv-S*: *lv* $\mathcal{S}$ **and** *sig-S*: *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
begin

lemma *fresh-sym-c-S*: $(c,\ 0) \notin$ *funas-rel* $\mathcal{S}$ ⟨*proof*⟩
lemma *fresh-sym-d-S*: $(d,\ 0) \notin$ *funas-rel* $\mathcal{S}$ ⟨*proof*⟩

lemma *lv-commute*:
assumes *com*: *commute* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$) (*gsrstep* $\mathcal{H}$ $\mathcal{S}$)
shows *commute* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

lemma *lv-NE*:
assumes *ne*: *NE* (*gsrstep* $\mathcal{H}$ $\mathcal{R}$) (*gsrstep* $\mathcal{H}$ $\mathcal{S}$)
shows *NE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

end

— CE is special as it only needs one additional constant therefore not included in the locale
lemma *lv-CE-aux*:
assumes $(s,\ t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $(\mathcal{R}^{\leftrightarrow})$
and *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *fresh*: $(c,\ 0) \notin \mathcal{F}$ **and** *const*: $(a,\ 0) \in \mathcal{F}$
and *lv*: *lv* $\mathcal{R}$ *lv* $\mathcal{S}$
and *ce*: *CE* (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{S}$)
shows $(s,\ t) \in$ (*srstep* $\mathcal{F}$ $\mathcal{S})^{\leftrightarrow *}$
⟨*proof*⟩

lemma *lv-CE*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *fresh*: $(c,\ 0) \notin \mathcal{F}$ **and** *const*: $(a,\ 0) \in \mathcal{F}$
and *lv*: *lv* $\mathcal{R}$ *lv* $\mathcal{S}$
and *ce*: *CE* (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c,\ 0)$ $\mathcal{F}$) $\mathcal{S}$)

shows *CE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

9.1 Specialized for monadic signature

lemma *lv-NE-aux*:
assumes $(s, t) \in$ *srsteps-with-root-step* $\mathcal{F}$ $\mathcal{R}$ **and** *fresh*: $(c, 0) \notin \mathcal{F}$
and *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *lv*: *lv* $\mathcal{R}$ *lv* $\mathcal{S}$
and *mon*: *monadic* $\mathcal{F}$
and *ne*: *NE* (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{S}$)
shows *NE-redp* $\mathcal{F}$ $\mathcal{R}$ $\mathcal{S}$ *s t*
⟨*proof*⟩

lemma *lv-NE*:
assumes *sig*: *funas-rel* $\mathcal{R} \subseteq \mathcal{F}$ *funas-rel* $\mathcal{S} \subseteq \mathcal{F}$
and *mon*: *monadic* $\mathcal{F}$ **and** *fresh*: $(c, 0) \notin \mathcal{F}$
and *lv*: *lv* $\mathcal{R}$ *lv* $\mathcal{S}$
and *ne*: *NE* (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{R}$) (*gsrstep* (*insert* $(c, 0)$ $\mathcal{F}$) $\mathcal{S}$)
shows *NE* (*srstep* $\mathcal{F}$ $\mathcal{R}$) (*srstep* $\mathcal{F}$ $\mathcal{S}$)
⟨*proof*⟩

end

References

[1] C. Sternagel and R. Thiemann. Abstract rewriting. *Archive of Formal Proofs*, June 2010. https://isa-afp.org/entries/Abstract-Rewriting.html, Formal proof development.