Proof Checker Notes

Yu-Yang Lin

July 17, 2015

1 Syntax Grammar

$$
\begin{array}{rl}
\text{(types)} & \tau ::= \text{bool} \mid \tau \rightarrow \tau \mid \text{nat} \mid \text{list } \tau \\
\text{(hypotheses)} & \text{A , B} ::= \top \mid \bot \mid \text{A} \wedge \text{B} \mid \text{A} \vee \text{B} \mid \text{A} \supset \text{B} \mid \forall\, x : \tau \,.\, \text{A} \mid \exists\, x : \tau \,.\, \text{A} \mid \text{t} = \text{t} : \tau \\
\text{(terms)} & \text{e , t} ::= x \mid \text{t t} \mid \text{true} \mid \text{false} \mid [\,] \mid \text{t} :: \text{t} \mid \text{zero} \mid \text{suc(t)} \\
\text{(term context)} & \psi ::= . \mid \psi, x : \tau
\end{array}
$$

$$
\begin{array}{lll}
\psi & \vdash & \text{t} : \tau \\
\psi & \vdash & \text{A prop}
\end{array}
$$

2 Specification rules for terms and propositional hypotheses

Note: functions are included as term types, but not directly as term constructors. Instead, function terms are added into the term context (ψ) manually. This simplifies the checker since function type inference is not required.

2.1 Terms

Natural Numbers:

$$\frac{}{\psi \vdash \text{zero} : \text{nat}}\ \text{(nat-zero)} \qquad \frac{\psi \vdash \text{t} : \text{nat}}{\psi \vdash \text{suc(t)} : \text{nat}}\ \text{(nat-suc-n)}$$

Booleans:

$$\frac{}{\psi \vdash \text{true} : \text{bool}}\ \text{(bool-true)} \qquad \frac{}{\psi \vdash \text{false} : \text{bool}}\ \text{(bool-false)}$$

Lists:

$$\frac{}{\psi \vdash [\,] : \text{list t}}\ \text{(list-nil)} \qquad \frac{\psi \vdash \text{t}' : \text{t} \qquad \psi \vdash \text{t}'' : \text{list t}}{\psi \vdash \text{t}' :: \text{t}'' : \text{list t}}\ \text{(list-cons)}$$

Variables:

$$\frac{x : \tau \in \psi}{\psi \vdash x : \tau}\ \text{(var)}$$

Application:

$$\frac{\psi \vdash \text{t} : \tau \rightarrow \tau' \qquad \psi \vdash \text{t}' : \tau}{\psi \vdash \text{t t}' : \tau'}\ \text{(app)}$$

2.2 Propositions

Truth and Falsity Propositions:

$$\frac{}{\psi \vdash \top \text{ prop}} \quad (\top\text{-prop}) \qquad\qquad \frac{}{\psi \vdash \bot \text{ prop}} \quad (\bot\text{-prop})$$

Binary Relation Propositions:

$$\frac{\psi \vdash \text{A prop} \qquad \psi \vdash \text{B prop}}{\psi \vdash \text{A} \wedge \text{B prop}} \quad (\wedge\text{-prop})$$

$$\frac{\psi \vdash \text{A prop} \qquad \psi \vdash \text{B prop}}{\psi \vdash \text{A} \vee \text{B prop}} \quad (\vee\text{-prop})$$

$$\frac{\psi \vdash \text{A prop} \qquad \psi \vdash \text{B prop}}{\psi \vdash \text{A} \supset \text{B prop}} \quad (\supset\text{-prop})$$

$$\frac{\psi \vdash \text{t}\colon \tau \qquad \psi \vdash \text{t}'\colon \tau}{\psi \vdash (\text{t} = \text{t}' : \tau) \text{ prop}} \quad (\text{eq-prop})$$

Quantifier Propositions:

$$\frac{\psi, x : \tau \vdash \text{A prop}}{\psi \vdash \forall x : \tau.\ \text{A prop}} \quad (\forall\text{-prop})$$

$$\frac{\psi, x : \tau \vdash \text{A prop}}{\psi \vdash \exists x : \tau.\ \text{A prop}} \quad (\exists\text{-prop})$$

3 Implementation rules for type inference and checking

3.1 Syntax grammar

$$\begin{array}{rl} \text{(infer)} & \text{e ::= x} \mid \text{e v} \mid \text{true} \mid \text{false} \mid \text{zero} \mid \text{suc(e)} \\ \text{(check)} & \text{v ::= v :: v} \mid \text{nil} \mid \text{e} \end{array}$$

Type Inferece Rule:

$$\overset{-}{\psi} \vdash \overset{-}{t} \Rightarrow \overset{+}{\tau}$$

Type Checking Rule:

$$\overset{-}{\psi} \vdash \overset{-}{t} \Leftarrow \overset{-}{\tau}$$

3.2 Term type inference

Variables:

$$\frac{x : \tau \in \psi}{\psi \vdash x \Rightarrow \tau} \quad (\text{var})$$

Application:

$$\frac{\psi \vdash \text{t} \Rightarrow \tau \to \tau' \qquad \psi \vdash \text{t}' \Leftarrow \tau}{\psi \vdash \text{t t}' \Rightarrow \tau'} \quad (\text{app})$$

Natural Numbers:

$$\frac{}{\psi \vdash \text{zero} \Rightarrow \text{nat}} \text{ (nat-zero)} \qquad \frac{\psi \vdash \text{t} \Leftarrow \text{nat}}{\psi \vdash \text{suc(t)} \Rightarrow \text{nat}} \text{ (nat-suc-n)}$$

Booleans:

$$\frac{}{\psi \vdash \text{true} \Rightarrow \text{bool}} \text{ (bool-true)} \qquad \frac{}{\psi \vdash \text{false} \Rightarrow \text{bool}} \text{ (bool-false)}$$

3.3 Term type checking

Lists:

$$\frac{}{\psi \vdash [\,] \Leftarrow \text{list t}} \text{ (list-nil)} \qquad \frac{\psi \vdash \text{t}' \Leftarrow \text{t} \qquad \psi \vdash \text{t}'' \Leftarrow \text{list t}}{\psi \vdash \text{t}' :: \text{t}'' \Leftarrow \text{list t}} \text{ (list-cons)}$$

Inference Case:

$$\frac{\psi \vdash \text{t} \Rightarrow \tau' \qquad \tau = \tau'}{\psi \vdash \text{t} \Leftarrow \tau} \text{ (app)}$$

3.4 Propositions type checking

Truth and Falsity Propositions:

$$\frac{}{\psi \vdash \top \text{ prop}} \text{ ($\top$-prop)} \qquad \frac{}{\psi \vdash \bot \text{ prop}} \text{ ($\bot$-prop)}$$

Binary Relation Propositions:

$$\frac{\psi \vdash \text{A prop} \qquad \psi \vdash \text{B prop}}{\psi \vdash \text{A} \wedge \text{B prop}} \text{ ($\wedge$-prop)}$$

$$\frac{\psi \vdash \text{A prop} \qquad \psi \vdash \text{B prop}}{\psi \vdash \text{A} \vee \text{B prop}} \text{ ($\vee$-prop)}$$

$$\frac{\psi \vdash \text{A prop} \qquad \psi \vdash \text{B prop}}{\psi \vdash \text{A} \supset \text{B prop}} \text{ ($\supset$-prop)}$$

$$\frac{\psi \vdash \text{t} \Leftarrow \tau \qquad \psi \vdash \text{t}' \Leftarrow \tau}{\psi \vdash (\text{t} = \text{t}' \Leftarrow \tau) \text{ prop}} \text{ (eq-prop)}$$

Quantifier Propositions:

$$\frac{\psi, x \Leftarrow \tau \vdash \text{A prop}}{\psi \vdash \forall x \Leftarrow \tau.\ \text{A prop}} \text{ ($\forall$-prop)}$$

$$\frac{\psi, x \Leftarrow \tau \vdash \text{A prop}}{\psi \vdash \exists x \Leftarrow \tau.\ \text{A prop}} \text{ ($\exists$-prop)}$$

3.5 Function signatures

$$\begin{array}{lcl} \texttt{infer\_term} & : & \psi \rightarrow \texttt{t} \rightarrow \tau\ \texttt{option} \\ \texttt{check\_term} & : & \psi \rightarrow \texttt{t} \rightarrow \tau \rightarrow \texttt{unit option} \\ \texttt{check\_prop} & : & \psi \rightarrow \texttt{A} \rightarrow \texttt{unit option} \end{array}$$

```
val infer_term  :  ctx -> term -> tp option
val check_term  :  ctx -> term -> tp -> unit option
val check_prop  :  ctx -> prop -> unit option
```

4 Well-formedness of proofs

4.1 Syntax grammar

$$
\begin{array}{rrcl}
\text{(proofs)} & p\,,q & ::= & \text{by H} \\
 & & | & (p\,,q) \\
 & & | & \texttt{let}\ (\text{H}',\text{H}'') = \text{H}\ \texttt{in}\ p \\
 & & | & (p\,,q\,)\ \texttt{either} \\
 & & | & \texttt{match [H] : (A} \vee \texttt{B) with (} \\
 & & & \qquad \texttt{| A [H'] : p -> C} \\
 & & & \qquad \texttt{| B [H''] : q -> C)} \\
\text{(hypotheses context)} & \Gamma & ::= & \cdot \\
 & & | & \Gamma\,, \text{H} : \text{A} \\
 & & | & \text{Assume A [H] , p}
\end{array}
$$

$$
\begin{array}{rl}
\psi;\Gamma & \vdash p : A \\
\psi & \vdash \Gamma
\end{array}
$$

4.2 Rules

Truth and Falsity:

$$
\frac{}{\psi;\Gamma \vdash \top : C}\ (\top R) \qquad \frac{}{\psi;\Gamma, H : \bot \vdash \text{Absurd } H : C}\ (\bot L)
$$

Conjunction:

$$
\frac{\psi;\Gamma, H : A \wedge B\,, H' : A\,, H'' : B \vdash p : C}{\psi;\Gamma, H : A \wedge B \vdash \texttt{let}\ (H', H'')\ =\ H\ \texttt{in}\ p : C}\ (\wedge L)
$$

$$
\frac{\psi;\Gamma \vdash p : A \qquad \psi;\Gamma \vdash q : B}{\psi;\Gamma \vdash (p\,,q) : A \wedge B}\ (\wedge R)
$$

Disjunction:

$$
\frac{\psi;\Gamma\,, H : A \vee B\,, H' : A \vdash p : C \qquad \psi;\Gamma\,, H : A \vee B\,, H'' : B \vdash q : C}{\psi;\Gamma, H : A \vee B \vdash \texttt{match [H] with(}\ A\ \texttt{[H'] :}\ p\ |\ B\ \texttt{[H''] :}\ q\ \texttt{)} : C}\ (\vee L)
$$

$$
\frac{\psi;\Gamma \vdash p : A}{\psi;\Gamma \vdash \texttt{Left}\ p : A \vee B}\ (\vee R_1)
$$

$$
\frac{\psi;\Gamma \vdash q : B}{\psi;\Gamma \vdash \texttt{Right}\ q : A \vee B}\ (\vee R_2)
$$

Implication:

$$
\frac{\psi;\Gamma, H : A \supset B \vdash p : A \qquad \psi;\Gamma, H : A \supset B\,, H' : B \vdash q : C}{\psi;\Gamma, H : A \supset B \vdash (p, B\ [\ H'\ ]\ \text{via}\ H, q) : C}\ (\supset L)
$$

$$
\frac{\psi;\Gamma\,, H : A \vdash p : B}{\psi;\Gamma \vdash (\ \text{Assume}\ A\ [\ H\ ]\,, p\ ) : A \supset B}\ (\supset R)
$$

Using hypotheses:

$$
\frac{}{\psi;\Gamma, [\ H\ ] : A \vdash \text{by } H : A}\ (\text{by})
$$

$$\frac{\psi;\Gamma \vdash \mathrm{p} : \mathrm{A}}{\psi;\Gamma \vdash \mathrm{p}\ \text{Therefore A} : \mathrm{A}} \quad \text{(therefore)}$$

4.3 Function signature

$$\texttt{check\_proof} \ : \ \psi \rightarrow \Gamma \rightarrow \texttt{P} \rightarrow \texttt{A} \rightarrow \texttt{unit option}$$

5 Quantifiers in proofs

5.1 Rules

Existentials:

$$\frac{\psi;\Gamma \vdash \mathrm{t} : \tau \qquad \psi;\Gamma \vdash \mathrm{p} : [\mathrm{x} \mapsto \mathrm{t}]\ \mathrm{A}}{\psi;\Gamma \vdash \text{Choose t ; p} : \exists x : \tau.\mathrm{A}} \quad (\exists\ \mathrm{R})$$

$$\frac{\psi, y : \tau;\Gamma, \mathrm{H} : \exists x : \tau.A, \mathrm{H'} \text{:}\ [x \mapsto y]A \vdash \mathrm{p} : \mathrm{C}}{\psi;\Gamma, \mathrm{H} : \exists x : \tau.\ \mathrm{A} \vdash \texttt{let}\ (y, \mathrm{H'}) = \mathrm{H}\ \texttt{in}\ \mathrm{p} : \mathrm{C}} \quad (\exists\ \mathrm{L})$$

Universals:

$$\frac{\psi, y : \tau;\Gamma, \vdash \mathrm{p} : [x \mapsto y]\ \mathrm{A}}{\psi;\Gamma \vdash \text{Assume } y : \tau\ .\ \mathrm{p} : \forall x : \tau.A} \quad (\forall\ \mathrm{R})$$

$$\frac{\psi;\Gamma \vdash \mathrm{t} : \tau \qquad \psi;\Gamma\ , \mathrm{H} : \forall x : \tau\ .\ \mathrm{A}\ , \mathrm{H'} : [\mathrm{x} \mapsto \mathrm{t}]\ \mathrm{A} \vdash \mathrm{p} : \mathrm{C}}{\psi;\Gamma, \mathrm{H} : \forall x : \tau.\ \mathrm{A} \vdash \texttt{let}\ \mathrm{H'} = \mathrm{H}\ \text{with t}\ \texttt{in}\ \mathrm{p} : \mathrm{C}} \quad (\forall\ \mathrm{L})$$

5.2 Substituting terms into variables

$$\texttt{subs} \ = \ [x \mapsto z]$$

$$\begin{array}{lcl} \texttt{subs\_term} & : & x \rightarrow \texttt{t} \rightarrow \texttt{t} \rightarrow \texttt{t} \\ \texttt{subs\_prop} & : & x \rightarrow \texttt{t} \rightarrow \texttt{A} \rightarrow [x] \rightarrow \texttt{A} \end{array}$$

```
val subs_term  :  var -> term -> term -> term
val subs_prop  :  var -> term -> prop -> var list -> prop
```

6 α-equivalence

6.1 Terms

Variables:

$$\frac{}{x \overset{\alpha}{=} x} \quad (\text{var}\overset{\alpha}{=})$$

Booleans:

$$\frac{}{\text{true} \overset{\alpha}{=} \text{true}} \quad (\text{bool-true}\overset{\alpha}{=}) \qquad \frac{}{\text{false} \overset{\alpha}{=} \text{false}} \quad (\text{bool-false}\overset{\alpha}{=})$$

Natural Numbers:

$$\frac{}{\mathrm{zero} \overset{\alpha}{=} \mathrm{zero}} \;(\text{nat-zero}\overset{\alpha}{=}) \qquad \frac{\mathrm{t} \overset{\alpha}{=} \mathrm{t}'}{\mathrm{suc(\,t\,)} \overset{\alpha}{=} \mathrm{suc(\,t'\,)}} \;(\text{nat-suc-n}\overset{\alpha}{=})$$

Lists:

$$\frac{}{[] \overset{\alpha}{=} []} \;(\text{list-nil}\overset{\alpha}{=}) \qquad \frac{\mathrm{e} \overset{\alpha}{=} \mathrm{e}' \qquad \mathrm{v} \overset{\alpha}{=} \mathrm{v}'}{\mathrm{e::v} \overset{\alpha}{=} \mathrm{e'::v'}} \;(\text{list-cons}\overset{\alpha}{=})$$

Application:

$$\frac{\mathrm{e} \overset{\alpha}{=} \mathrm{e}' \qquad \mathrm{v} \overset{\alpha}{=} \mathrm{v}'}{\mathrm{e\ v} \overset{\alpha}{=} \mathrm{e'\ v'}} \;(\text{var}\overset{\alpha}{=})$$

6.2 Propositions

Truth and Falsity:

$$\frac{}{\top \overset{\alpha}{=} \top} \;(\top \overset{\alpha}{=}) \qquad \frac{}{\bot \overset{\alpha}{=} \bot} \;(\bot \overset{\alpha}{=})$$

Binary Relations:

$$\frac{\mathrm{A} \overset{\alpha}{=} \mathrm{A}' \qquad \mathrm{B} \overset{\alpha}{=} \mathrm{B}'}{\mathrm{A} \wedge \mathrm{B} \overset{\alpha}{=} \mathrm{A}' \wedge \mathrm{B}'} \;(\wedge \overset{\alpha}{=})$$

$$\frac{\mathrm{A} \overset{\alpha}{=} \mathrm{A}' \qquad \mathrm{B} \overset{\alpha}{=} \mathrm{B}'}{\mathrm{A} \vee \mathrm{B} \overset{\alpha}{=} \mathrm{A}' \vee \mathrm{B}'} \;(\vee \overset{\alpha}{=})$$

$$\frac{\mathrm{A} \overset{\alpha}{=} \mathrm{A}' \qquad \mathrm{B} \overset{\alpha}{=} \mathrm{B}'}{\mathrm{A} \supset \mathrm{B} \overset{\alpha}{=} \mathrm{A}' \supset \mathrm{B}'} \;(\supset \overset{\alpha}{=})$$

Equality:

$$\frac{\mathrm{t}_1 \overset{\alpha}{=} \mathrm{t}_1' \qquad \mathrm{t}_2 \overset{\alpha}{=} \mathrm{t}_2' \qquad \tau \overset{\alpha}{=} \tau'}{(\mathrm{t}_1 = \mathrm{t}_2 : \tau) \overset{\alpha}{=} (\mathrm{t}_1{}' = \mathrm{t}_2{}' : \tau')} \;(= \overset{\alpha}{=})$$

Quantifiers:

$$\frac{\text{И}\, z\,.\ (x\ z)\ \mathrm{B} \overset{\alpha}{=} (x\ z)\ \mathrm{B}' \qquad \tau \overset{\alpha}{=} \tau'}{\exists x : \tau.\ \mathrm{B} \overset{\alpha}{=} \exists y : \tau'.\ \mathrm{B}'} \;(\exists \overset{\alpha}{=})$$

$$\frac{\text{И}\, z\,.\ (x\ z)\ \mathrm{B} \overset{\alpha}{=} (x\ z)\ \mathrm{B}' \qquad \tau \overset{\alpha}{=} \tau'}{\forall x : \tau.\ \mathrm{B} \overset{\alpha}{=} \forall y : \tau'.\ \mathrm{B}'} \;(\forall \overset{\alpha}{=})$$

6.3 Swapping variable names

$$\texttt{swap} = (x\ \ z)$$

$$\begin{aligned} \texttt{swap\_term} &: x \to z \to \texttt{t} \to \texttt{t} \\ \texttt{swap\_prop} &: x \to z \to \texttt{A} \to \texttt{A} \end{aligned}$$

```
val swap_term  :  var -> var -> term -> term
val swap_prop  :  var -> var -> prop -> prop
```

7 Induction in proofs

7.1 Rules through predicates

Natural Numbers:

$$\frac{\psi;\Gamma \vdash \mathrm{p} : \mathrm{P(zero)} \qquad \psi\ ,\, n : \mathrm{nat}\ ;\Gamma\ ,\mathrm{H} : \mathrm{P}(n) \vdash \mathrm{q} : \mathrm{P(suc}(n))}{\psi;\Gamma \vdash (\text{Induction on nat: case zero : p ; case suc}(n) : [\mathrm{H}]\ ,\mathrm{q}) : (\forall\ m : \mathrm{nat}\ .\ \mathrm{P}(m))} \quad \text{(induction-nat)}$$

Lists:

$$\frac{\psi;\Gamma \vdash \mathrm{p} : \mathrm{P(\ [\,]\ )} \qquad \psi\ ,\, x : \tau\ ,\, xs : \mathrm{list}\ \tau\ ;\Gamma\ ,\mathrm{H} : \mathrm{P}(xs) \vdash \mathrm{q} : \mathrm{P}(x :: xs)}{\psi;\Gamma \vdash (\text{Induction on list: case [] : p ; case } (x :: xs) : [\mathrm{H}]\ ,\mathrm{q}) : (\forall\ ys : \mathrm{list}\ \tau\ .\ \mathrm{P}(ys))} \quad \text{(induction-list)}$$

Booleans:

$$\frac{\psi;\Gamma \vdash \mathrm{p} : \mathrm{P(\ true\ )} \qquad \psi;\Gamma \vdash \mathrm{q} : \mathrm{P(\ false\ )}}{\psi;\Gamma \vdash (\text{Induction on bool: case true : p ; case false : q) : } (\forall\ b : \mathrm{bool}\ .\ \mathrm{P(}\ b\ ))} \quad \text{(induction-bool)}$$

7.2 Rules through substitution

Natural Numbers:

$$\frac{\psi;\Gamma \vdash \mathrm{p} : [\mathrm{m} \mapsto \mathrm{zero}]\ \mathrm{C} \qquad \psi\ ,\, n : \mathrm{nat}\ ;\Gamma\ ,\mathrm{H} : [\mathrm{m} \mapsto \mathrm{zero}]\ \mathrm{C} \vdash \mathrm{q} : [\mathrm{m} \mapsto \mathrm{suc}(n)]\ \mathrm{C}}{\psi;\Gamma \vdash (\text{Ind-Nat: zero : p ; suc}(n) : [\mathrm{H}]\ ,\mathrm{q}) : (\forall\ m : \mathrm{nat}\ .\ \mathrm{C})} \quad \text{(induction-nat)}$$

Lists:

$$\frac{\psi;\Gamma \vdash \mathrm{p} : [ys \mapsto [\,]\ ]\ \mathrm{C} \qquad \psi\ ,\, x : \tau\ ,\, xs : \mathrm{list}\ \tau\ ;\Gamma\ ,\mathrm{H} : [ys \mapsto xs\ ]\ \mathrm{C} \vdash \mathrm{q} : [ys \mapsto x :: xs\ ]\ \mathrm{C}}{\psi;\Gamma \vdash (\text{Ind-List: [] : p ; } (x :: xs) : [\mathrm{H}]\ ,\mathrm{q}) : (\forall\ ys : \mathrm{list}\ \tau\ .\ \mathrm{C})} \quad \text{(induction-list)}$$

Booleans:

$$\frac{\psi;\Gamma \vdash \mathrm{p} : [b \mapsto \mathrm{true}\ ]\ \mathrm{C} \qquad \psi;\Gamma \vdash \mathrm{q} : [b \mapsto \mathrm{false}\ ]\ \mathrm{C}}{\psi;\Gamma \vdash (\text{Ind-Bool: true : p ; false : q) : } (\forall\ b : \mathrm{bool}\ .\ \mathrm{C})} \quad \text{(induction-bool)}$$

8 Equality in proofs

8.1 Abstract congruence closure [1, p. 4–7]

8.1.1 Definition

Rewrite-Rules:

D-rule :	$f(c_0, ...c_k) \rightarrow c$	*where f is a term constructor and c_i are constants in K*
C-rule :	$c \rightarrow d$	*where c and d are constants in K*

Sets:

$$D : \{D\text{-}rule\}$$
$$C : \{C\text{-}rule\}$$

$$E : \{(t = t) : \tau\}$$
$$K : \{x | x \notin E\}$$
$$R : D \cup C$$

Closure Construction:

$$
\begin{aligned}
\mathit{state} &: (K, E, R) \\
\texttt{state\_transition} &: \mathit{state} \rightarrow \mathit{state}\ \texttt{option} \\
\texttt{construct\_closure}\ \sigma &= \texttt{construct\_closure}(\texttt{state\_transition}\ \sigma) \\
[\![\texttt{construct\_closure}]\!] &: (\emptyset, E, \emptyset) \mapsto (K, \emptyset, R) \\
&\quad \textit{where } R \textit{ is the (abstract) congruence closure for } E
\end{aligned}
$$

8.1.2 Sate transition rules

$$\text{Extension:} \quad \frac{(K, E[t], R) \qquad t = f(c_0, ..., c_k)}{(K \cup \{c\}, E[c], R \cup \{t \rightarrow c\ : D\})} \quad \text{(Ext)}$$

$$\text{Simplification:} \quad \frac{(K, E[t], R \cup \{t \rightarrow c : D\})}{(K, E[c], R \cup \{t \rightarrow c : D\})} \quad \text{(Sim)}$$

$$\text{Orientation:} \quad \frac{(K \cup \{c\}, E \cup \{t = c\}, R\})}{(K \cup \{c\}, E, R \cup \{t \rightarrow c : D\})} \quad \text{(Ori)}$$

$$\text{Deletion:} \quad \frac{(K, E \cup \{t = t\}, R\})}{(K, E, R\})} \quad \text{(Del)}$$

$$\text{Deduction:} \quad \frac{(K, E, R \cup \{t \rightarrow c : D, t \rightarrow d : D\})}{(K, E \cup \{c = d\}, R \cup \{t \rightarrow d : D\})} \quad \text{(Ded)}$$

$$\text{Collapse:} \quad \frac{(K, E, R \cup \{s[c] \rightarrow c' : D, c \rightarrow d : C\})}{(K, E, R \cup \{s[d] \rightarrow c' : D, c \rightarrow d : C\})} \quad \text{(Col)}$$

$$\text{Composition:} \quad \frac{(K, E, R \cup \{t \rightarrow c : D, c \rightarrow d : C\})}{(K, E, R \cup \{t \rightarrow d : D, c \rightarrow d : C\})} \quad \text{(Com)}$$

References

[1] Leo Bachmair, Ashish Tiwari, and Laurent Vigneron. Abstract congruence closure. *J. Autom. Reasoning*, 31(2):129–168, 2003.