

# Proof Checker Notes

Yu-Yang Lin

July 24, 2015

## 1 Syntax Grammar

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

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

## 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 ( $\psi$ ) manually. This simplifies the checker since function type inference is not required.

### 2.1 Terms

Natural Numbers:

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

Booleans:

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

Lists:

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

Variables:

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

Application:

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

## 2.2 Propositions

Truth and Falsity Propositions:

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

Binary Relation Propositions:

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

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

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

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

Quantifier Propositions:

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

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

## 3 Implementation rules for type inference and checking

### 3.1 Syntax grammar

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

Type Inference Rule:

$$\bar{\psi} \vdash \bar{t} \Rightarrow \bar{\tau}$$

Type Checking Rule:

$$\bar{\psi} \vdash \bar{t} \Leftarrow \bar{\tau}$$

### 3.2 Term type inference

Variables:

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

Application:

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

Natural Numbers:

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

Booleans:

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

### 3.3 Term type checking

Lists:

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

Inference Case:

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

### 3.4 Propositions type checking

Truth and Falsity Propositions:

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

Binary Relation Propositions:

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

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

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

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

Quantifier Propositions:

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

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

### 3.5 Function signatures

```
infer_term  :  ψ → t → τ option
check_term  :  ψ → t → τ → unit option
check_prop  :  ψ → A → unit option

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

$$\begin{array}{ll}
 \text{(proofs)} & p, q ::= \text{by } H \\
 & \quad | (p, q) \\
 & \quad | \text{let } (H', H'') = H \text{ in } p \\
 & \quad | (p, q) \text{ either} \\
 & \quad | \text{match } [H] : (A \vee B) \text{ with } ( \\
 & \quad \quad | A [H'] : p \rightarrow C \\
 & \quad \quad | B [H''] : q \rightarrow C ) \\
 \text{(hypotheses context)} & \Gamma ::= . \\
 & \quad | \Gamma, H : A \\
 & \quad | \text{Assume } A [H], p \\
 & \psi; \Gamma \vdash p : A \\
 & \psi \vdash \Gamma
 \end{array}$$

### 4.2 Rules

Truth and Falsity:

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

Conjunction:

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

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

Disjunction:

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

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

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

Implication:

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

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

Using hypotheses:

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

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

### 4.3 Function signature

`check_proof :  $\psi \rightarrow \Gamma \rightarrow P \rightarrow A \rightarrow \text{unit option}$`

## 5 Quantifiers in proofs

### 5.1 Rules

Existentials:

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

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

Universals:

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

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

### 5.2 Substituting terms into variables

`subs =  $[x \mapsto z]$`

`subs_term :  $x \rightarrow t \rightarrow t \rightarrow t$`   
`subs_prop :  $x \rightarrow t \rightarrow A \rightarrow [x] \rightarrow A$`

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

## 6 $\alpha$ -equivalence

### 6.1 Terms

Variables:

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

Booleans:

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

Natural Numbers:

$$\frac{}{\text{zero} \stackrel{\alpha}{=} \text{zero}} \quad (\text{nat-zero} \stackrel{\alpha}{=}) \quad \frac{t \stackrel{\alpha}{=} t'}{\text{suc}(t) \stackrel{\alpha}{=} \text{suc}(t')} \quad (\text{nat-suc-n} \stackrel{\alpha}{=})$$

Lists:

$$\frac{}{[] \stackrel{\alpha}{=} []} \quad (\text{list-nil} \stackrel{\alpha}{=}) \quad \frac{e \stackrel{\alpha}{=} e' \quad v \stackrel{\alpha}{=} v'}{e::v \stackrel{\alpha}{=} e'::v'} \quad (\text{list-cons} \stackrel{\alpha}{=})$$

Application:

$$\frac{e \stackrel{\alpha}{=} e' \quad v \stackrel{\alpha}{=} v'}{e \ v \stackrel{\alpha}{=} e' \ v'} \quad (\text{var} \stackrel{\alpha}{=})$$

## 6.2 Propositions

Truth and Falsity:

$$\frac{}{\top \stackrel{\alpha}{=} \top} \quad (\top \stackrel{\alpha}{=}) \quad \frac{}{\perp \stackrel{\alpha}{=} \perp} \quad (\perp \stackrel{\alpha}{=})$$

Binary Relations:

$$\frac{A \stackrel{\alpha}{=} A' \quad B \stackrel{\alpha}{=} B'}{A \wedge B \stackrel{\alpha}{=} A' \wedge B'} \quad (\wedge \stackrel{\alpha}{=})$$

$$\frac{A \stackrel{\alpha}{=} A' \quad B \stackrel{\alpha}{=} B'}{A \vee B \stackrel{\alpha}{=} A' \vee B'} \quad (\vee \stackrel{\alpha}{=})$$

$$\frac{A \stackrel{\alpha}{=} A' \quad B \stackrel{\alpha}{=} B'}{A \supset B \stackrel{\alpha}{=} A' \supset B'} \quad (\supset \stackrel{\alpha}{=})$$

Equality:

$$\frac{t_1 \stackrel{\alpha}{=} t'_1 \quad t_2 \stackrel{\alpha}{=} t'_2 \quad \tau \stackrel{\alpha}{=} \tau'}{(t_1 = t_2 : \tau) \stackrel{\alpha}{=} (t'_1 = t'_2 : \tau')} \quad (= \stackrel{\alpha}{=})$$

Quantifiers:

$$\frac{\forall z. (x \ z) \ B \stackrel{\alpha}{=} (x \ z) \ B' \quad \tau \stackrel{\alpha}{=} \tau'}{\exists x : \tau. B \stackrel{\alpha}{=} \exists y : \tau'. B'} \quad (\exists \stackrel{\alpha}{=})$$

$$\frac{\forall z. (x \ z) \ B \stackrel{\alpha}{=} (x \ z) \ B' \quad \tau \stackrel{\alpha}{=} \tau'}{\forall x : \tau. B \stackrel{\alpha}{=} \forall y : \tau'. B'} \quad (\forall \stackrel{\alpha}{=})$$

## 6.3 Swapping variable names

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

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

$$\begin{aligned} \text{val swap\_term} &: \text{var} \rightarrow \text{var} \rightarrow \text{term} \rightarrow \text{term} \\ \text{val swap\_prop} &: \text{var} \rightarrow \text{var} \rightarrow \text{prop} \rightarrow \text{prop} \end{aligned}$$

## 7 Induction in proofs

### 7.1 Rules through predicates

Natural Numbers:

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

Lists:

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

Booleans:

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

### 7.2 Rules through substitution

Natural Numbers:

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

Lists:

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

Booleans:

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

## 8 Equality in proofs

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

#### 8.1.1 Definition

Rewrite-Rules:

$$\begin{array}{lll} D\text{-rule :} & f(c_0, \dots, c_k) \rightarrow c & \text{where } f \text{ is a term constructor and } c_i \text{ are constants in } K \\ C\text{-rule :} & c \rightarrow d & \text{where } c \text{ and } d \text{ are constants in } K \end{array}$$

Sets:

$$\begin{aligned} D &: \{D\text{-rule}\} \\ C &: \{C\text{-rule}\} \\ E &: \{(t = t) : \tau\} \\ K &: \{x \mid x \notin E\} \\ R &: D \cup C \end{aligned}$$

Closure Construction:

$$\text{build\_acc: } \frac{(\emptyset, E, \emptyset)}{(K, \emptyset, R)} \quad \text{state : } (K, E, R) \quad \text{where } R \text{ is the abstract congruence closure (ACC) of } E$$

### 8.1.2 State transition rules

$$\text{Extension: } \frac{(K, E[t], R) \quad t = f(c_0, \dots, c_k) \quad c_i \in K \wedge c \notin K}{(K \cup \{c\}, E[c], R \cup \{t \rightarrow c : D\})} \quad (\text{Ext})$$

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

$$\text{Orientation1: } \frac{(K \cup \{c\}, E \cup \{t = c\}, R)}{(K \cup \{c\}, E, R \cup \{t \rightarrow c : D\})} \quad (\text{Ori1})$$

$$\text{Orientation2: } \frac{(K \cup \{c, d\}, E \cup \{c = d\}, R) \quad c < d}{(K \cup \{c, d\}, E, R \cup \{c \rightarrow d : C\})} \quad (\text{Ori2})$$

$$\text{Orientation3: } \frac{(K \cup \{c, d\}, E \cup \{c = d\}, R) \quad d < c}{(K \cup \{c, d\}, E, R \cup \{d \rightarrow c : C\})} \quad (\text{Ori2})$$

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

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

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

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

## 8.2 Equality through congruence

$$\frac{\psi ; \Gamma \vdash H_i : (t_i = t'_i) : \tau_i \quad \forall i \in \{1 \dots n\} . t_i = t'_i \xrightarrow{\quad} t = t'}{\psi ; \Gamma \vdash \text{By Equality } (H_0, \dots, H_n) : (t = t' : \tau)} \quad (\text{eq})$$

## References

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