The Reducibility Method for Call-By-Value System $F$

Reorganized by Peng Fu

Last revised: Jun 20, 2010

1 Descriptions

1.1 Types

$T ::= X \mid T_1 \rightarrow T_2 \mid \Pi X. T$

1.2 Terms

$t ::= x \mid (t_1 \, t_2) \mid \lambda x. t \mid c$

1.3 Well-formed Context $\Gamma$

Let $FV(T)$ denote all the free variables of the type $T$, $dom(\Gamma)$ denote the type and term variables in the context $\Gamma$.

$\Gamma ::= \cdot \mid \Gamma, X : type \mid \Gamma, x : T.$

$\emptyset \text{ OK}$

$\Gamma \, OK \quad \Gamma, X : type \, OK$

$\Gamma \, OK \quad FV(T) \subseteq dom(\Gamma)$

$\Gamma, x : T \text{ OK}$

A context $\Gamma$ is well-formed iff $\Gamma \, OK$. We assume all contexts in this note are well-formed.

1.4 Type assignment rules.

$\Gamma(x) = T$

$\Gamma \vdash x : T \quad T_{Var}$

$\Gamma \vdash t_1 : T_2 \rightarrow T_1 \quad \Gamma \vdash t_2 : T_2$

$\Gamma \vdash t_1 \, t_2 : T_1 \quad T_{App}$

$\Gamma, x : T_1 \vdash t : T_2$

$\Gamma \vdash \lambda x. t : T_1 \rightarrow T_2 \quad T_{Lam}$

$\Gamma, X : type \vdash t : T$

$\Gamma \vdash t : \Pi X. T \quad Univ_{Abs}$

$\Gamma \vdash t : \Pi X. T$

$\Gamma \vdash t : [U/X]T \quad Univ_{App}$
1.5 Reduction rules

Left-to-right, call-by-value reduction.

Contexts

\[ C ::= \ast | v C | C t \]

Values

\[ v ::= \lambda x.t | i \]

Inactive terms

\[ i ::= c | (i v) \]

Reductions

\[ C[(\lambda x.t) v] \leadsto C[[v/x]t] \]

2 Reducibility

2.1 Reducibility Candidates

Let \( N \) be the set of terms which have a normal form under our reduction setting. Let \( I \) be the set of all inactive terms.

**Definition** A reducibility candidate \( R \) is a set of terms that satisfies the following conditions:

**CR 1** If \( t \in R \), then \( t \in N \) and closed.

**CR 2** If \( t \in R \) and \( t \leadsto t' \), then \( t' \in R \).

**CR 3** If \( t \) is a closed term, \( t \leadsto t' \) and \( t' \in R \), then \( t \in R \).

**CR 4** \( I \subseteq R \).

**Fact** Let \( \mathcal{R} \) be the set of all reducibility candidates. \( \mathcal{R} \) is a non-empty set.

To show \( \mathcal{R} \) is non-empty, we will just show all the closed term \( t \in N \) is a reducibility candidate, which is obvious from the definition of reducibility candidate.

2.2 Reducibility Sets

As Girard said: Among all the candidates, the true reducibility candidate for \( T \) is to be found. Here we use reducibility set to find the right candidate.

**Definition** Let \( \phi \) be a finite function: \( FV(T) \to \mathcal{R} \). Eg. \( \phi(X) \in \mathcal{R} \). If \( \text{dom}(\phi) = \{X_1, X_2, \ldots X_n\} \), then we usually write \( \phi \) as \([R_1/X_1, \ldots R_n/X_n]\).

The reducibility set \( RED_T \phi \) is defined inductively as follows.
Assume \( t \in RED_{T_1 \rightarrow T_2} \) if \( t \in \phi(X) \).

Assume \( t \in RED_{T_1 \rightarrow T_2} \) if \( \forall u \in RED_{T_1} \Rightarrow (t u) \in RED_{T_2} \).

Assume \( t \in RED_{T_1 \rightarrow T_2} \) if \( \forall R \in \mathcal{R}, t \in RED_R[\phi(R)] \).

2.3 Reducibility Sets as Reducibility Candidates

Now we can show that our reducibility sets are indeed reducibility candidates.

**CR 1** If \( t \in RED_T \), then \( t \in N \) and closed.

**CR 2** If \( t \in RED_T \) and \( t \sim t' \), then \( t' \in RED_T \).

**CR 3** If \( t \) is a closed term, \( t \sim t' \) and \( t' \in RED_T \), then \( t \in RED_T \).

**CR 4** \( I \subseteq RED_T \).

**Proof** By induction on the structure of \( T \).

**Base Case:** \( T = X \)

**CR 1–CR 4** Obvious from the definition.

**Step Case:** \( T = T_1 \rightarrow T_2 \)

**CR 1** Assume \( t \in RED_{T_1 \rightarrow T_2} \). By IH(CR 4), \( RED_{T_1} \) is non-empty. So we can take arbitrary \( u \in RED_{T_1} \). By definition, \( (t u) \in RED_{T_2} \). By IH(CR 1), \( (t u) \in N, u \in N \). So \( t \in N \).

**CR 2** Assume \( t \in RED_{T_1 \rightarrow T_2} \) and \( t \sim t' \). Take arbitrary \( u \in RED_{T_1} \). By definition, we know \( (t u) \in RED_{T_2} \). With our reduction strategy, \( (t u) \sim (t' u) \). By IH(CR 2), \( (t' u) \in RED_{T_2} \). So by definition of \( RED_{T_1 \rightarrow T_2} \), \( t' \in RED_{T_1 \rightarrow T_2} \).

**CR 3** Assume \( t \) is closed, \( t \sim t' \) and \( t' \in RED_{T_1 \rightarrow T_2} \). Take arbitrary \( u \in RED_{T_1} \). By definition, we know \( (t u) \in RED_{T_2} \). With our reduction strategy, \( (t u) \sim (t' u) \). By IH(CR 1), \( u \) is closed, thus we know \( (t u) \) is closed. By IH(CR 3), \( (t u) \in RED_{T_2} \). So by definition of \( RED_{T_1 \rightarrow T_2} \), \( t \in RED_{T_1 \rightarrow T_2} \).

**CR 4** To show \( I \subseteq RED_{T_1 \rightarrow T_2} \), we need to show for arbitrary \( i \in I, i \in RED_{T_1 \rightarrow T_2} \). By definition of inactive terms, \( i \) is already in normal form. Take arbitrary \( u \in RED_{T_1} \). By IH(CR 1), \( u \in N \) and closed. So \( (i u) \sim (i u') \), where \( u' \) is the normal form of \( u \). Thus by definition of inactive terms and IH(CR 4), \( (i u') \in I \subseteq RED_{T_2} \). \( (i u) \) is closed, so by IH(CR 3), \( (i u) \in RED_{T_2} \). So by definition of \( RED_{T_1 \rightarrow T_2} \), \( i \in RED_{T_1 \rightarrow T_2} \). So \( I \subseteq RED_{T_1 \rightarrow T_2} \).

**Step Case:** \( T = \Pi.X.T \)

**CR 1** Assume \( t \in RED_{\Pi.X.T} \). By the fact in section 2.1, \( \mathcal{R} \) is non-empty. Take a arbitrary reducibility candidate \( R \). By definition, \( t \in RED_T[\phi(R/X)] \). By IH(CR 1), \( t \in N \) and closed.

**CR 2** Assume \( t \in RED_{\Pi.X.T} \) and \( t \sim t' \). Consider arbitrary reducibility candidate \( R \). By definition, \( t \in RED_T[\phi(R/X)] \). By IH(CR 2), \( t' \in RED_T[\phi(R/X)] \). So by definition of \( RED_{\Pi.X.T}[\phi(R/X)] \), \( t' \in RED_{\Pi.X.T} \).
CR 3 Assume $t$ is closed, $t \sim t'$ and $t' \in RED_{T\phi}$. Take arbitrary reducibility candidate $R$. By definition, $t' \in RED_T\phi[R/X]$. We know that $t \sim t'$ and $t$ is closed. So by IH(CR 3), $t \in RED_T\phi[R/X]$. So by definition, $t \in RED_{T\phi}$. 

CR 4 We need to show that for arbitrary $i \in I, i \in RED_{T\phi}$. By definition, we actually need to show for arbitrary $R \in \Re, i \in RED_T\phi[R/X]$. By IH(CR 4), $I \subseteq RED_T\phi[R/X]$. So $i \in RED_T\phi[R/X]$. So it’s the case.

3 Substitution Lemma

Substitution Lemma $RED_{[V/X]\phi} = RED_T\phi[RED_V\phi/X]$.

Proof By induction on the structure of $T$.

Base Case: If $T = X$. We need to show $RED_V\phi = RED_X\phi[RED_V\phi/X]$. By definition, $RED_X\phi[RED_V\phi/X] = \phi[RED_V\phi/X](X) = RED_V\phi$. So it is the case.

Step Case: If $T = \Pi.Y.W$. Then we need to show $RED_{\Pi.Y.W}\phi = RED_{\Pi.Y.W}\phi[RED_V\phi/X]$. Take arbitrary $R \in \Re$ and arbitrary $t \in RED_{\Pi.Y.W}\phi$. By definition, $t \in RED_{\Pi.Y.W}\phi[R/Y]$. By IH, $RED_{\Pi.Y.W}\phi[R/Y] = RED_W\phi[R/Y, RED_V\phi/X] = RED_W\phi[R/Y, RED_V\phi/X]$.

So $t \in RED_W\phi[R/Y, RED_V\phi/X]$. By definition, $t \in RED_{\Pi.Y.W}\phi[RED_V\phi/X]$. 

Now let’s prove the other direction. Take arbitrary $t \in RED_{\Pi.Y.W}\phi[RED_V\phi/X]$ and arbitrary $R \in \Re$. By definition, $t \in RED_W\phi[RED_V\phi/X, R/Y]$. By IH, $RED_W\phi[RED_V\phi/X, R/Y] = RED_{\Pi.Y.W}\phi[R/Y]$. So $t \in RED_{\Pi.Y.W}\phi[R/Y]$. By definition, $t \in RED_{\Pi.Y.W}\phi[RED_V\phi/X]$. So it is the case.

Step Case: If $T = T_1 \rightarrow T_2$. Then we need to show $RED_{(\Pi.Y.W)\phi} = RED_{T_1 \rightarrow T_2}\phi[RED_V\phi/X]$. Take arbitrary $u \in RED_{(\Pi.Y.W)\phi}$ and $t \in RED_{(\Pi.Y.W)\phi[R/Y]}$. By definition, $(t u) \in RED_{(\Pi.Y.W)\phi}$. By IH, $RED_{(\Pi.Y.W)\phi} = RED_{T_1\phi[RED_V\phi/X]}$ and $RED_{(\Pi.Y.W)\phi} = RED_{T_2\phi[RED_V\phi/X]}$. So $t \in RED_{T_1\phi[RED_V\phi/X]}$. The other direction is similar.

4 Reducibility Sets and Type assignment

Definition We define the set $[\Gamma]$ of well-typed substitutions $(\sigma, \delta)$ as follows:

\[
(\emptyset, \emptyset) \in []
\]

\[
(\sigma, \delta) \in [\Gamma] \quad R \in \Re \quad (\sigma, \delta) \cup \{(X, R)\}) \in [T, X : type]
\]

\[
(\sigma, \delta) \in [\Gamma] \quad FV(T) \subseteq dom(\Gamma) \quad t \in RED_T\delta
\]

\[
(\sigma \cup \{(x, t)\}, \delta) \in [\Gamma, x : T]
\]

Theorem If $\Gamma \vdash t : T$, then $\forall (\sigma, \delta) \in [\Gamma], (\sigma t) \in RED_T\delta$.

Proof By induction on the typing derivation of $\Gamma \vdash t : T$.

Base Case: The typing derivation looks like:
\[ \Gamma(x) = T \]
\[ \Gamma \vdash x : T \]

Because \( \Gamma(x) = T \) and context is well-formed, \( FV(T) \subseteq dom(\Gamma) \). By definition of \( (\sigma, \delta) \in [\Gamma] \), we have \((x, t) \in \sigma\), where \( t \in RED_T \delta \). So \((\sigma x) = t \in RED_T \delta \).

### Application Case
The typing derivation looks like:
\[
\Gamma \vdash t_1 : (T_2 \rightarrow T_1) \quad \Gamma \vdash t_2 : T_2
\]
\[
\Gamma \vdash t_1 \ t_2 : T_1
\]

We need to show that \( \sigma(t_1 \ t_2) \in RED_{T_1} \delta \). By IH, for any \((\sigma, \delta) \in [\Gamma], (\sigma \ t_1) \in RED_{(T_2 \rightarrow T_1)} \delta \) and \((\sigma \ t_2) \in RED_{T_2} \delta \). By the definition of \( RED_{(T_2 \rightarrow T_1)} \delta \), we have \((\sigma(t_1)(\sigma(t_2))) = (\sigma(t_1) \ t_2) \in RED_{T_1} \delta \).

### Lambda abstract Case
The typing derivation looks like:
\[
\Gamma, x : T \vdash t : T_2
\]
\[\Gamma \vdash \lambda x . t : (T_1 \rightarrow T_2)\]

We need to prove that \( \sigma(\lambda x . t) \in RED_{\lambda x . t} \delta \). Since \( \lambda x . (\sigma t) \in N \) and closed. By definition of \( RED_{\lambda x . t} \delta \), we still need to show for arbitrary \( u \in RED_{T_1} \delta \), \((\lambda x . (\sigma t) \ u) \in RED_{T_2} \delta \). Since \( u \) is closed by CR 1, the normal form of \( u \) must be a value, which means \( u \sim v \). So we have \((\lambda x . (\sigma t)) \ u \sim (\lambda x . (\sigma t)) \ v \), and by CR 2, \( v \in RED_{T_1} \delta \). By definition of call-by-value reduction, \( (\lambda x . (\sigma t)) \ v \sim \sigma[v/x] t \). Since \( v \in RED_{T_1} \delta \), and \( FV(T_1) \subseteq dom(\Gamma) \), we have \((\sigma \cup \{(x, v)\}, \delta) \in [\Gamma, x : T_1] \). By IH, \((\sigma[v/x] t) \in RED_{T_2} \delta \). Since \((\lambda x . (\sigma t) \ u) \) is closed, by CR 3, \((\lambda x . (\sigma t) \ u) \in RED_{T_2} \delta \). So by definition of \( RED_{(T_1 \rightarrow T_2)} \delta \), \( (\sigma(\lambda x . t) \ = \lambda x . (\sigma t) \in RED_{(T_1 \rightarrow T_2)} \delta \).

### Universal abstract Case
The typing derivation looks like:
\[\Gamma, X : type \vdash t : T\]
\[\Gamma \vdash \Pi X . t : T\]

We need to show \( \sigma(t) \in RED_{\Pi X . t} \delta \). By definition of \( RED_{\Pi X . t} \delta \), we just need to show for arbitrary \( R \in \Re, \sigma(t) \in RED_T \delta[R/X] \). By IH, for any \((\sigma, \delta \cup \{(X, R)\}) \in [\Gamma, X : type], \sigma(t) \in RED_T \delta[R/X] \). So it is the case.

### Universal application Case
The typing derivation looks like:
\[\Gamma \vdash t : \Pi X . T\]
\[\Gamma \vdash t : [(U/X)]T\]

We need to show \( \sigma(t) \in RED_{[(U/X)]T} \delta \). By substitution lemma, \( RED_{[(U/X)]T} \delta = RED_T \delta[RED_U \delta/X] \). By IH, we know that \( \sigma(t) \in RED_{(\Pi X . T)} \delta \). By definition, for arbitrary \( R \in \Re, \sigma(t) \in RED_T \delta[R/X] \). We let \( R = RED_U \delta \). So it is the case.

### 5 Conclusion

So for any closed term \( t \), if \( \Gamma \vdash t : T \), where \( dom(\Gamma) \) only contains type variables, then \( t \in RED_T \delta \), and by CR 1, \( t \in N \).