M2 LMFI – SOFIX

QUANTIFICATION DU SECOND-ORDRE ET POINTS FIXES EN LOGIQUE

Introduction to system F and weak normalization theorem for F

Alexis Saurin

17th january 2024

Contents

1 Definition of System F

Definition 1.1 (*Types of system* F)

Let us consider an infinte, countable set of type variables (or second-order type variables), $\mathcal{V}_{\mathsf{F}}$. System F types are given by the grammar:

$$T, U, V ::= X \mid U \to V \mid \forall X.T$$

$\forall X.T$ binds X in T, so that types are considered with the expected notions of free and bound (type) variables, capture-free substitition of an F-type for a variable (written $T\{U/X\}$).

1.1 Church-style System F

Church-style System F terms consists in the set of typed terms defined inductively (together with their type) as follows

Definition 1.2 (*Church-style System* F)

One considers, for each type T of F, an infinite countable set of variables for this type, $\mathcal{V}^{T}$.

Church-style System F *terms are the least set such that:*

- *For any variable* x *in* $\mathcal{V}^{\mathcal{T}}$, x^T *is a term of type* T *(with free variables* $\{x\}$*);*
- *For any term* v *of type* V *and any variable* x *in* $\mathcal{V}^{\mathcal{U}}$, $\lambda x^U.v$ *is a term of type* $U \to V$ *(with free variables* $fv(t) \setminus \{x\}$*);*
- *For any terms* t *and* u *of respective types* $U \to T$ *and* U, $(t)u$ *is a term of type* T *(with free variables* $fv(t) \cup fv(u)$*);*
- *For any type variable* X *and any term* t *of type* T, $\Lambda X.t$ *is a term of type* $\forall X.T$ **under the condition that, for any free variable libre** x **of** t**,** X **does not occur free in the type of** x *(with free variables* $fv(t)$*);*
- *For any term* t *of type* $\forall X.T$ *and any type* U *of* F, $(t)U$ *is a term of type* $T\{U/X\}$ *(with free variables* $fv(t)$*).*

Proposition 1.3

If t *is an* F*-term of type* T, x *is a variable of type* U *and* u *is an* F*-term of type* U, *then* $t\{u/x\}$ *is an* F*-term of type* T, *having the free variables as* $(fv(t) \setminus \{x\}) \cup fv(u)$.

Proposition 1.4

If t *is an* F *term of type* T *and if* X *is a type variable that does not occur free in the free variables of* t, *then for any* F*-type* U, $t\{U/X\}$ *is an* F*-term of type* $T\{U/X\}$, *having the same free variables as* t.

The previous propositions ensure that the following definition is meaningful:

Definition 1.5 (*Dynamics of* F)

Church-style F*-terms are equipped with two reduction rules:*

$$\begin{array}{rcl} (\lambda x.t)u & \longrightarrow_\beta & t\{u/x\} \\ (\Lambda X.t)U & \longrightarrow_\forall & t\{U/X\} \end{array}$$

The first rule is called β*-reduction, as usual, while the second is called* **universal reduction**.

1.2 Curry-style System F

As for the simply-typed case, Curry-style system F consists in the pure λ-calculus together with a typing relation given by a type system, inspired by second-order natural deduction. That is simply the type system for simply-typed λ-calculus exgtended with the two following typing rules:

$$\frac{\Gamma \vdash t : T}{\Gamma \vdash t : \forall X.T} \ \forall i \quad (\star)$$

$$\frac{\Gamma \vdash t : \forall X.T}{\Gamma \vdash t : T\{U/X\}} \ \forall e(U)$$

$(\star)$ The inference rule $\forall i$ can only be applied if X does not occur free in the type of (the variables in) Γ.

The judgement that a λ-term t appear in conclusion of a typing derivation with type T under context Γ will be written $\Gamma \vdash_{\mathsf{F}} t : T$.

1.3 Relations between the two presentations

As in the simply-typed case, there are two approaches to typing in System F

As in the simply-typed case, there are two approaches to typing in System F: the style *à la* Church where terms hold typing information which are in correspondence with deductions of implicative second-order natural deduction, and the style *à la* Curry where we use pure λ-terms and define a typing relation $\vdash_{\mathsf{F}}$ relating F typing-contexts, Γ, pure λ-terms, t and F types, T, written $\Gamma \vdash_{\mathsf{F}} t : T$.

One shall now related the two presentations of system F through a forgetful map which sends Church-style system F to Curry-typable terms and one will study the properties of this mapping from the point of view of normalization. This will show the equivalence between weak (resp. strong) normalization in Church-style and Curry-Style system F.

Definition 1.6 (*forgetful map*)

One defines inductively a type-forgetting map from Church-style F*-terms to pure* λ*-terms:*

- $[x^T]^- = x$;
- $[\lambda x^T.\, t]^- = \lambda x.\, t$;
- $[(t)\, u]^- = ([t]^-)\, [u]^-$;
- $[\Lambda X.\, t]^- = [t]^-$;
- $[(t)\, T]^- = [t]^-$.

One remarks that the forgetful operation is compatible with substitution: $[t\,\{u/x\}]^- = [t]^-\,\{[u]^-/x\}$, for any t, u and x of F with compatible types.

One can also show that the forgetful operation sends Church-style F-terms to F-Curry-style typable terms, which have, in particular, the expected type:

Proposition 1.7

Let $t : T$ *be a Church-style* F*-term with free variables among* $(x_i^{T_i})_{1\leq i\leq n}$*. Then* $x_1 : T_1, \ldots x_n : T_n \vdash_{\mathsf{F}} [t]^- : T$ *is derivable in (Curry-style) F.*

Proof: The previous result can be proved by induction on the structure of term t.

□

It is straightforward that, conversely:

Proposition 1.8

A type derivation δ *for a judgment* $\Gamma \vdash_{\mathsf{F}} t : T$ *is isomorphic to a Church-style term* $u : T$ *the free variables of which are among the variables of* Γ *(and typed according to* Γ*. In addition,* $[u]^- = t$*.*

Before relating the reductions in Church-style and Curry-style system F and their normalization properties, it is useful to remark the following fact that is easily proved since each universal reduction step makes the number of constructions of Church-style terms strictly decrease:

Lemma 1.9

Universal reduction is strongly normalizing in (Church-style) system F.

Reductions in Church-style and Curry-style system F can also be compared:

Proposition 1.10

1. *The type-erasure of a (Church-style) normal form is a normal form.*
2. *If* t *reduces to* u *with a universal-step, then* $[t]^- = [u]^-$.

3. *If t reduces to u with a β-step, then $[t]^-$ reduces to $[u]^-$ with a β-step.*

4. *The previous statements show that if $[t]^-$ is normal, reductions from t contain only universal steps.*

5. *If $[t]^-$ reduces in one step to u, then t reduces in at least one step to some v such that $[v]^- = u$.*

Proof: One proves the last two points only.

For (4), let us remark that if a reduction from t contains a β-reduction, the translation of this reduction would contain a reduction from $[t]^-$ containing at least one β-step. Contrapositively, is $[t]^-$ is normal, reductions from t cannot contain β-reduction.

The last point is proved by induction on the structure of t:

Variable case, λ-abstraction, universal abstraction or universal application are easy. Let us focus on the application case and consider $t = (v)\, w$. One reasons as follows: of the redex which is fired is in $[v]^-$ or $[w]^-$, one uses the induction hypothesis to conclude. Otherwise, $[t]^-$ is itself a redex that is $[v]^- = \lambda x.\, [v']^-$ for some v' and $u = [v']^- \left\{[w]^-/x\right\}$. But it is not true that v starts necessarily with a λ-abstraction: it may begins with universal abstractions or applications around its λ-abstraction. Let us consider v'' the normal form of v by universal reductions: v'' cannot begin with a universal abstraction since v'' has the type of v' and v, that is $U \to T$ for some U. In the same way, it cannot begin with universal applications as it would require that some universal abstractions generalise the type of $\lambda x.\, v'$ to introduce the universal quantifier but in that case, the term would not be normal for universal reduction anymore... One concludes that v'' has form $\lambda x^A.\, v_0''$. And finally $t \longrightarrow^\star \mathit{univ}(\lambda x^A.\, v_0'')\, w \longrightarrow_\beta v_0'' \left\{w/x\right\} = t'$ et $[t']^- = [v_0'']^- \left\{[w]^-/x\right\} = u$.

□

Finally, one can state the result we expected:

Theorem 1.11

Weak (resp. strong) normalization of Church-style system F *is equivalent to the weak (resp. strong) normalization of Curry-style system* F.

Proof: For weak normalization, the reasoning is direct and sumple: Assume F is weakly normalizing and let t be a λ-term which is typable in Curry-style F. We know that there exists u such that $[u]^- = t$ and that by hypothesis, u has a normal form v. The previous propositions ensure that $t \longrightarrow^\star [v]^-$ which is normal. In the other direction, if every λ-term which is typable in Curry-style F normalizes, let us consider some term t of Curry-style F. We know that $[t]^-$ normalizes to u and that there exists v in F such that $t \longrightarrow^\star v$ and $[v]^- = u$. Let us remark that v is not necessarily normal but that, by the previous propositions, all its reductions are universal, which we know to be strongly normalizing.

For strong normalization, the reasoning is slightly less immediate. Assume that there exists a term t typable in Curry-style system F from which an infinite reduction sequence can be drawn. We know that there exists some (Church-style) F-term u such that $[u]^- = t$ and the previous propositions ensure that from this term, an infinite reduction sequence can also be build. By contraposition, strong normalization of F ensures that every pure term that is typable in (Curry-style) system F strongly normalizes. Assume now that there exists some F-term t having an infinite reduction ρ. By strong normalization of universal reduction, we know that ρ contains infinitely many β-reductions from which one conclude that $[t]^-$ also has an infinite reduction.

□

2 Weak normalization of system F

In this section, we prove weak normalization of Church-style system F, by adapting the reducibility proof of thesimply typed λ-calculus.

2.1 Introducing reducibility candidates

In order to extend the reducibility technique from simple types to System F, one faces a difficulty. Taking inspiration of the notation of reducibility at arrow types, the natural extension of reducibility would be:

$$\mathsf{RED}^{\mathsf{WN}}(\forall X.U) = \{t : \forall X.U \mid \text{for all type } V, (t)\, V \in \mathsf{RED}^{\mathsf{WN}}(U\,\{V/X\})\}$$

But this would be an ill-formed definition (ill-founded rather) since $U\,\{V/X\}$ may be arbitrarily more complex than $\forall X.U$ as soon as X occurs freely in U.

To solve this problem, the idea will consist in avoiding to define reducible terms for each type but, rather to define an abstract notion of set of reducible terms, from basic properties it should satisfy, and to axiomatise the notion of reducibility in some sense.

From there, one will define a notion of parametric reducibility from valuations which will associate *reducibility candidates* to each type variable.

This will permit, in order to treat second-order quantification, to avoid letting the type vary over all possible instances of types (which would be incompatible with an inductive definition as noted above) but rather to let the valuations vary over all possible assignements of candidates.

To proceed, one shall therefore identify the characteristic of candidates which are given by the properties that we need for the theorem. There are actually two properties which are crucial to prove weak normalization in the simply typed case:

- the sets $\mathsf{RED}^{\mathsf{WN}}(T)$ are ***closed by β-expansion***;
- the sets $\mathsf{RED}^{\mathsf{WN}}(T)$ are ***adapted***.

(Note that the defintion of reducibility depends on the property we want to prove and that, to prove strong normalization, it should be adapted.)

2.2 Notions of reducibility candidates

The notation of neutral term is adapted (in F, a neutral term is a term which does not start with a λ-abstraction, nor a universal abstraction) and we keep unchanged the definitions of $\mathsf{Neut}(T)$ and $\mathsf{Norm}(T)$:

Definition 2.1 ($\mathsf{Neut}(T)$, $\mathsf{Norm}(T)$)

$$\mathsf{Neut}(T) = \{t \in \mathsf{F};\ t \textit{ normal and neutral of type } T\}$$

$$\mathsf{Norm}(T) = \{t \in \mathsf{F};\ t \textit{ normalisable of type } T\}.$$

One can now define the notion of reducibility candidates as:

Definition 2.2 (*Reducibility candidate*)

A **reducibility candidate** *of type T is a set R of λ-terms of type T which satisfies the following two conditions:*

- *(CR1) R is closed by β-expansion;*
- *(CR2)* $\mathsf{Neut}(T) \subseteq R \subseteq \mathsf{Norm}(T)$.

One shall denote by $\mathsf{CR}(T)$ the set of all reducibility candidates of type T. Note that for any T, $\mathsf{Norm}(T) \in \mathsf{CR}(T)$ so $\mathsf{CR}(T)$ is never empty.

If R is a reducibility candidate, there is a unique type T such that $R \in \mathsf{CR}(T)$, this type will be noted $\mathsf{Type}(R)$.

A valuation ρ is a partial function from type variables to reducibility candidates the domain of which, noted $\mathsf{dom}(\rho)$, is finite. One shall write $\rho[X := R]$ for the valuation ρ' of domain $\mathsf{dom}(\rho) \cup \{X\}$ such that:

$$\rho'(X) = R \qquad \rho'(Y) = \rho(Y) \text{ si } Y \neq X.$$

Definition 2.3

One shall say that a valuation ρ covers a type T (resp. a term $t : T$) if its domain contains all the free type variables of T (resp. of T and of the types of the free variables of t).

Definition 2.4 (*Initial valuation,* ρ_Γ)

Let $\Gamma = X_1, \ldots, X_n$, one defines the **initial valuation** *(or default valuation) on Γ, ρ_Γ as the valuation defined on Γ such that $\rho_\Gamma(X_i) = \mathsf{Norm}(X_i)$.*

A valuation induces a type substitution:

Definition 2.5 (T^ρ, t^ρ)

Let ρ be a valuation, one defines the type substitution:

- $X_i^\rho = X_i$ *if* $X \notin \mathsf{dom}(\rho); U$ *if* $X \in \mathsf{dom}(\rho)$ *and* $\rho(X) \in CR(U)$
- $T^\rho = T\{X_i^\rho / X_i, 1 \leq i \leq n\}$ *if the free variables of T are the $X_1, \ldots, X_n$.*

One also defines a substitution on terms:

$$t^\rho = t\{X_i^\rho / X_i, 1 \leq i \leq n\}.$$

Remark 2.6

In particular, if the free type variables of T and t are all in Γ, one has $T^{\rho_\Gamma} = T$ and $t^{\rho_\Gamma} = t$.

One can now define by induction on the structure of types, the sets of reducible terms, given a valuation ρ:

Definition 2.7 ($\mathsf{RED^{WN}}_\rho(T)$)

$\mathsf{RED^{WN}}_\rho(T)$ is defined by induction on T (if ρ covers T):

- $\mathsf{RED^{WN}}_\rho(X) = \rho(X)$;
- $\mathsf{RED^{WN}}_\rho(U \to V) = \{t : (U \to V)^\rho / \forall u \in \mathsf{RED^{WN}}_\rho(U), (t)u \in \mathsf{RED^{WN}}_\rho(V)\}$;
- $\mathsf{RED^{WN}}_\rho(\forall X.U) = \{t : (\forall X.U)^\rho / \forall V \in \mathit{Type}, \forall R \in CR(V), (t)V \in \mathsf{RED^{WN}}_{\rho[X:=R]}(U)\}$

2.3 Proof of the normalization theorem

We shall need the following two easy lemmas:

Lemma 2.8

If $t : \forall X.T$, and U is a type, then $(t)U$ normalizable implies that t is itself normalizable.

Proof: As in the simply typed case with the variable application.

□

Lemma 2.9 (*substitution*)

Let V, W be types and ρ be a valuation covering V, W. One has:

$$\mathsf{RED^{WN}}_\rho(V\{W/Y\}) = \mathsf{RED^{WN}}_{\rho[Y:=\mathsf{RED^{WN}}_\rho(W)]}(V).$$

Proof: By induction on type V.

$\square$

The normalization theorem relies on the following two lemmas:

Lemma 2.10

For any type T and any valuation ρ covering T, $\mathsf{RED^{WN}}_\rho(T) \in \mathsf{CR}(T^\rho)$.

Proof: It is straightforward that terms of $\mathsf{RED^{WN}}_\rho(T)$ are of type T^ρ. One proves by induction on types that those sets are candidates (*ie.* that CR1 and CR2 are satisfied):

- $T = X$: this is trivial by definition of a valuation: $\rho(X) \in CR(X^\rho)$;
- $T = U \to V$: one replays the reasoning done for simple types, ***exactly***.
  - let $t_1 \longrightarrow t \in \mathsf{RED^{WN}}_\rho(T)$ and let $u\mathsf{RED^{WN}}_\rho(U)$. One has $(_1)u \longrightarrow (t)u \in \mathsf{RED^{WN}}_\rho(V)$ so that $(t_1)V \in \mathsf{RED^{WN}}_\rho(V)$ by induction hypothesis; since this holds for any $u\mathsf{RED^{WN}}_\rho(U)$, one has $t_1 \in \mathsf{RED^{WN}}_\rho(T)$ which completes the proof of CR1 in that case.
  - Let $t \in \mathsf{Neut}(T^\rho)$ and let $u \in \mathsf{RED^{WN}}_\rho(U)$. $(t)u$ is neutral and since, by induction hypothesis, u has a normal form v, $(t)u \longrightarrow^\star (t)v$ which is neutral and normal of type V^ρ and therefore in $\mathsf{RED^{WN}}_\rho(V)$ by CR2. By induction hypothesis again, we have closure of reducibility at type V by β-expansion and $t(u) \in \mathsf{RED^{WN}}_\rho(V)$.
  - Let $t \in \mathsf{RED^{WN}}_\rho(T)$ and $x^{U^\rho} \in \mathsf{Neut}(U^\rho) \subseteq \mathsf{RED^{WN}}_\rho(U)$ by induction hypothesis. By CR2, $(t)x^{U^\rho}$ is normalisable so that $(t)x^{U^\rho} \longrightarrow^\star n$ in normal form. By analysing this reduction, we notice that t is itself normalisable to $\lambda x^{U^\rho}.n$
- $T = \forall X.U$: One proceeds as follows:
  - let $t_1 \longrightarrow t \in \mathsf{RED^{WN}}_\rho(T)$ and let V be a type and $R \in CR(V)$. One has $(_1)V \longrightarrow (t)V \in \mathsf{RED^{WN}}_{\rho[X:=R]}(U)$ so that $(t_1)V \in \mathsf{RED^{WN}}_{\rho[X:=R]}(U)$ by induction hypothesis and since this holds for any V and any R, one has $t_1 \in \mathsf{RED^{WN}}_\rho(\forall X.U)$ which completes the proof of CR1 in that case.
  - Let $t \in \mathsf{Neut}(T^\rho)$ and V be a type, $R \in CR(V)$. t is neutral and normal so that $(t)V$ is also neutral and normal of type $U^{\rho[X:=R]}$. Since CR2 holds at U, by induction hypothesis, one has $(t)V \in \mathsf{RED^{WN}}_{\rho[X:=R]}(U)$ and $t \in \mathsf{RED^{WN}}_\rho(\forall X.U)$.
  - Let $t \in \mathsf{RED^{WN}}_\rho(T)$, V be a type and $R \in CR(V)$. One has $(t)V \in \mathsf{RED^{WN}}_{\rho[X:=R]}(U) \subseteq \mathsf{Norm}(U^{\rho[X:=R]})$ by induction hypothesis and the previous lemma ensures that in that case $t \in \mathsf{Norm}(\forall X.U^\rho)$.

$\square$

Lemma 2.11 (*Adequation lemma*)

Let $t : T$ be a Church-style F-term of free variables $(x_i^{U_i})_{1\leq i\leq n}$, then for any valuation ρ covering t and for any $u_i \in \mathsf{RED^{WN}}_\rho(U_i)$, $1 \leq i \leq n$, one has: $t^\rho\,\{u_i/x_i, 1 \leq i \leq n\} \in \mathsf{RED^{WN}}_\rho(T)$.

Theorem 2.12

Every term of (Church-style) System F (weakly) normalises.

Proof of the normalization theorem: The proof is identical to the simply-typed case.

Let $t : T$ be of free variables $(x_i^{T_i})_{1\leq i\leq n}$ and let Γ be such that is contains all the free variables occurring in T and in the T_i. One has $T^{\rho\Gamma} = T$ and $T_i^{\rho\Gamma} = T_i$. Since the $\mathsf{RED^{WN}}_\rho(U)$ are candidates, one has by (CR2) that $1 \leq i \leq n$, $x_i^{T_i} \in \mathsf{RED^{WN}}_\rho(T_i)$ since (term) variables are neutral.

Adequation lemma ensures that $t\left\{x_i^{T_i}/x_i, 1 \leq i \leq n\right\} = t \in \mathsf{RED^{WN}}_\rho(T) \subseteq \mathsf{Norm}(T)$.

$\square$

One can now prove adequation lemma:

Proof of adequation lemma: The lemma is proved by induction on the structure of term, as usual. The case for variable, lambda-abstraction and application are treated as in the simply typed case. One only details the constructions which are specific to F:

- If $t = \Lambda X.u$, $U = \forall X.U'$ and $u : U'$. One can assume that X is not in the domain of the valuation, renaming the bound variable if needed. Let V be a type and $R \in CR(V)$ and let $u' = u^{\rho[X:=R]}\{u_i/x_i\}$. By induction hypothesis, one has $u' \in \mathsf{RED}^{\mathsf{WN}}{}_{\rho[X:=R]}(T)$ for any $R \in CR(V)$ and because $(t^\rho\{u_i/x_i\})V \to u^\rho\{u_i/x_i\}\{V/X\} = u^{\rho[X:=R]}\{u_i/x_i\} = u' \in \mathsf{RED}^{\mathsf{WN}}{}_{\rho[X:=R]}(T)$. Finally, closure of candidates by anti-reduction ensures that $(t^\rho\{u_i/x_i\})V \in \mathsf{RED}^{\mathsf{WN}}{}_{\rho[X:=R]}(T)$.
- If $t = (u)V$, with $u : \forall X.U$. Let us set $U' = U\{V/X\}$ and $u' = u^\rho\{u_i/x_i\}$. We have, by induction hypothesis that $u' \in \mathsf{RED}^{\mathsf{WN}}{}_\rho(\forall X.U)$ so that $(u')V^\rho \in \mathsf{RED}^{\mathsf{WN}}{}_{\rho[X:=R]}(U)$. But we also have $(u')V^\rho = t^\rho\{u_i/x_i\}$ and one can conclude, by using the lemma on substitutions, that $t^\rho\{u_i/x_i\} \in \mathsf{RED}^{\mathsf{WN}}{}_\rho(U')$.

□

2.4 Towards strong normalization

One can adapt the notion of reducibility candidate to obtain a proof of strong normalization.

The definition of candidates shall of course be adapted to the normalization property we want to establish, indeed:

- Contrarily to WN, SN is not stable by β-expansion in general (otherwise SN and WN would be equivalent simply because a normal for is always SN and every (weakly) normalizable term is the β-expanded of a normal form).
- On the other hand, SN is stable by β-reduction (contrarily to WN, in general: think of a non-deterministic (and therfore non-confluent) rewriting system with a rule $a + b \longrightarrow a$ and $a + b \longrightarrow b$ and consider $\Omega + \lambda x.x$ which reduces to $\lambda x.x$ and has therfore a normal form, but which also reduces to Ω which is not WN ...)
- Before, one had property $\mathsf{Neut}(T) \subseteq R \subseteq \mathsf{Norm}(T)$. It will be needed to refine the lower and upper bounds to be more precise about the candidates, for instance asking that $R \subseteq \mathsf{SN}(T)$.