Lambda Calculus: A Case for Inductive Definitions Part II Monotone Inductive Types

Ralph Matthes
Institut für Informatik der Universität München
Oettingenstraße 67, 80538 München
matthes@informatik.uni-muenchen.de

August 11, 2000

Abstract

These lecture notes contain an advanced treatment of inductive types. It is assumed that the reader has access to the first part of the lecture notes which introduces to λ-calculus and system F, and is available at http://www.tcs.informatik.uni-muenchen.de/~matthes/

Contents

1 Introduction

After having used induction on inductively defined sets so successfully in the first part of these notes, the means of induction are added to the lambda calculus itself: By a constructively-minded inspection of Tarski's fixed-point theorem the most general formulation of inductive types is gained (via the Curry-Howard isomorphism). This gives a lot more insight into the capability of system F for modelling abstract data types and into usual formulations of inductive types found in the literature.

Citations are quite rare in these notes. This does not indicate that I consider the results to be original although I hope that several of them are. Credits are given in my research papers. In a future version, I might add more citations to enhance fairness.

2 Monotone Inductive Types

The expressiveness of system F is highlighted by the fact that least pre-fixed-points of monotone operators can be represented—even with respect to reduction behaviour. Its main practical consequence arises in the field of program extraction: The computational content of intuitionistic proofs with inductive definitions consists of terms of system F whose normalization yields the objects whose existence has been proved.[1] Later we will see that one also needs to model fixed-points (not only pre-fixed-points) in order to get primitive recursion (not only iteration), and those fixed-points are not available in system F as is generally believed and greatly supported by [SU99].

2.1 Tarski's Fixed-Point Theorem and System F

By studying the concept of a complete lattice and its representation in system F, we arrive at a representation of iteration on monotone inductive types in system F (for more explanation see [Mat99c]).

Definition 1 *Let $\mathcal{U}$ be a set and $\leq$ be a partial order on $\mathcal{U}$ (i. e., $\leq$ is a binary relation on $\mathcal{U}$ which is reflexive, antisymmetric and transitive) and $\bigwedge : \wp\mathcal{U} \to \mathcal{U}$ (with $\wp\mathcal{U}$ the powerset of $\mathcal{U}$) a function which determines for every $\mathcal{M} \subseteq \mathcal{U}$ the infimum (the greatest lower bound) of $\mathcal{M}$ w. r. t. $\leq$, i. e., $\forall \mathsf{M} \in \mathcal{M}. \bigwedge \mathcal{M} \leq \mathsf{M}$ and*

$$\forall \mathsf{N} \in \mathcal{U}.(\forall \mathsf{M} \in \mathcal{M}.\mathsf{N} \leq \mathsf{M}) \Rightarrow \mathsf{N} \leq \bigwedge \mathcal{M}.$$

Then $(\mathcal{U}, \leq, \bigwedge)$ is called a complete lattice.

Note that in this situation $\bigvee \mathcal{M} := \bigwedge \{\mathsf{N} \in \mathcal{U} \mid \forall \mathsf{M} \in \mathcal{M}.\mathsf{M} \leq \mathsf{N}\}$ (the infimum of the upper bounds of $\mathcal{M}$) gives the supremum (the least upper bound) of $\mathcal{M}$ w. r. t. $\leq$, i. e., $\forall \mathsf{M} \in \mathcal{M}.\mathsf{M} \leq \bigvee \mathcal{M}$ and

$$\forall \mathsf{N} \in \mathcal{U}.(\forall \mathsf{M} \in \mathcal{M}.\mathsf{M} \leq \mathsf{N}) \Rightarrow \bigvee \mathcal{M} \leq \mathsf{N}.$$

[1] Unfortunately, because of lack of space, this claim cannot be substantiated in these notes.

We now show that the infimum of the family $(\rho[\alpha := \sigma])_{\sigma\in\mathcal{T}_u}$ can be represented in system F (compare with the treatment of infimum types in [Mat98, pp.35–38]).

This may be motivated in a complete lattice of sets with $\leq = \subseteq$ and $\bigwedge = \bigcap$. There, we have

$$x \in \bigwedge \mathcal{M} \iff \forall M \in \mathcal{U}.\, M \in \mathcal{M} \Rightarrow x \in M.$$

Now we want to represent $\bigwedge\{\rho[\alpha := \sigma] \mid \sigma \in \mathcal{T}_u\}$ by some type $\mathrm{i}\alpha\rho$. $\mathcal{U}$ will be the set $\mathcal{T}_u$ of types. Instead of $\mathcal{M}$, we consider a type ρ, seen as a function of the type variable α. We once more profit from the λ-notation when saying that $\lambda\alpha\rho$ shall model $\mathcal{M}$. The index set becomes $\mathcal{T}_u$, hence we use σ instead of M. How do we express $\sigma \in \lambda\alpha\rho$? Simply by $\rho[\alpha := \sigma]$. The equivalence above would now read

$$x \in \mathrm{i}\alpha\rho \iff \forall\sigma \in \mathcal{T}_u.\rho[\alpha := \sigma] \Rightarrow x \in \sigma.$$

Note that this is only a formal manipulation since the types are no sets. However, modified realizability (see e. g. [Ber93]) tells us how to interpret this statement: Remove the first-order part (the occurrences of x), internalize the quantification over the types and replace $\Rightarrow$ by $\to$ (the type former pertaining to function spaces). Hence, the left-hand side of the equivalence becomes $\mathrm{i}\alpha\rho$, and the right-hand side $\forall\alpha.\rho \to \alpha$. This justifies why we simply *define* $\mathrm{i}\alpha\rho := \forall\alpha.\rho \to \alpha$.

Does our $\mathrm{i}\alpha\rho$ have the defining properties of the infimum? In the situation of a complete lattice of sets with $\leq = \subseteq$, we have to check

($\bigwedge_E$) If $x \in \bigwedge \mathcal{M}$ and $M \in \mathcal{M}$ then $x \in M$.

($\bigwedge_I$) If $\forall M \in \mathcal{U}.M \in \mathcal{M} \Rightarrow \forall y(y \in N \Rightarrow y \in M)$ and $x \in N$ then $x \in \bigwedge \mathcal{M}$.

After the translation in the style of modified realizability, we have

($\bigwedge_E$) If $\mathrm{i}\alpha\rho$ and $\rho[\alpha := \sigma]$ then σ.

($\bigwedge_I$) If $\forall\alpha.\rho \to \tau \to \alpha$ and τ then $\mathrm{i}\alpha\rho$.

Its only reasonable interpretation can be statements on types of terms. In fact, we have:

(i_E) If $\Gamma \vdash r : \mathrm{i}\alpha\rho$ and $\Gamma \vdash s : \rho[\alpha := \sigma]$ then $\Gamma \vdash r\sigma s : \sigma$.

(i_I) If $\Gamma \vdash \ell : \forall\alpha.\rho \to \tau \to \alpha$ (with $\alpha \notin \mathrm{FV}(\tau)$) and $\Gamma \vdash t : \tau$ then, for $C_{\mathrm{i}\alpha\rho,\tau}\ell t := \Lambda\alpha\lambda x^\rho.\ell\alpha x t$ (with $\alpha \notin \mathrm{FTV}(\ell) \cup \mathrm{FTV}(t)$ and $x \notin \mathrm{FV}(\ell) \cup \mathrm{FV}(t)$), we have $\Gamma \vdash C_{\mathrm{i}\alpha\rho,\tau}\ell t : \mathrm{i}\alpha\rho$.

Moreover, there is a derived β-reduction rule for infima: $(C_{\mathrm{i}\alpha\rho,\tau}\ell t)\sigma s \to^*_{\beta\eta} \ell\sigma s t$.

We now turn to fixed-points.

Theorem 1 (Tarski) *Let $(\mathcal{U}, \leq, \bigwedge)$ be a complete lattice and $\Phi : \mathcal{U} \to \mathcal{U}$ be monotone (i. e., if $M \leq N$ then $\Phi(M) \leq \Phi(N)$). Then*

$$\mu\Phi := \bigwedge\{M \in \mathcal{U} \mid \Phi(M) \leq M\}$$

is the least fixed-point of Φ.

Example 1 *Let* $\mathcal{U} := \wp(\mathbb{R})$, $\leq := \subseteq$, $\bigwedge := \bigcap$ *and* $\Phi(M) := \{0\} \cup \{r+1 \mid r \in M\}$. *Then* $\mu\Phi$ *is the set* $\mathbb{N}$.

Definition 2 $M \in \mathcal{U}$ *is a pre-fixed-point of* Φ *iff* $\Phi(M) \leq M$. $M \in \mathcal{U}$ *is a post-fixed-point of* Φ *iff* $M \leq \Phi(M)$.

Proof Let us prove Tarski's fixed-point theorem. By definition, we have that $\mu\Phi \leq M$ for all pre-fixed-points M of Φ. Show that $\mu\Phi$ is itself a pre-fixed-point of Φ. Let M be a pre-fixed-point of Φ. Then $\mu\Phi \leq M$. Because Φ is monotone, this implies $\Phi(\mu\Phi) \leq \Phi(M)$. Because M is a pre-fixed-point of Φ and $\leq$ is transitive, $\Phi(\mu\Phi) \leq M$ follows. Hence, $\Phi(\mu\Phi) \leq M$ for every pre-fixed-point M of Φ and therefore $\Phi(\mu\Phi) \leq \mu\Phi$ by definition of $\mu\Phi$. We conclude that $\mu\Phi$ is the least pre-fixed-point of Φ.

$\mu\Phi$ is also a post-fixed-point[2] of Φ (and consequently the least fixed-point): We have to show that $\mu\Phi \leq \Phi(\mu\Phi)$. By definition of $\mu\Phi$, it suffices to show that $\Phi(\mu\Phi)$ is a pre-fixed-point of Φ, i.e., $\Phi(\Phi(\mu\Phi)) \leq \Phi(\mu\Phi)$. This follows from the first part of the proof and the monotonicity of Φ. $\square$

We now look more closely at the definition and the first part of the above proof. The definition of $\mu\Phi$ does not need the monotonicity of Φ, but only the completeness of the lattice. Assume again a lattice of sets with $\leq = \subseteq$. By $(\bigwedge_E)$, we have

(μ_E) If $x \in \mu\Phi$ and $\forall y.y \in \Phi(M) \Rightarrow y \in M$ then $x \in M$.

From the proof, we see

(μ_I) If Φ is monotone and $x \in \Phi(\mu\Phi)$ then $x \in \mu\Phi$.

An even more careful proof would be: Let Φ be monotone and $x \in \Phi(\mu\Phi) =: N$. Show $x \in \mu\Phi$ by $(\bigwedge_I)$: Assume $M \in \mathcal{U}$ such that $\Phi(M) \subseteq M$ and $y \in N$. We show that $y \in M$: Let $z \in \mu\Phi$. Because of $(\bigwedge_E)$ and $\Phi(M) \subseteq M$, $z \in M$. Hence, $\mu\Phi \subseteq M$. By monotonicity of Φ, $N \subseteq \Phi(M)$, consequently $y \in \Phi(M)$. Since $\Phi(M) \subseteq M$, we arrive at $y \in M$. Now, $(\bigwedge_I)$ applies due to $x \in N$, and yields $x \in \mu\Phi$.

We are now in the position to model the least pre-fixed-point—called the inductive type $\mu\alpha\rho$—of $\lambda\alpha\rho$, i.e., of the operation $\sigma \mapsto \rho[\alpha := \sigma]$ instead of Φ. (Recall that monotonicity is not needed for the definition.) Since

$$\mu\Phi = \bigwedge\{M \in \mathcal{U} \mid \forall x.x \in \Phi(M) \Rightarrow x \in M\},$$

we clearly have to set $\mu\alpha\rho := \mathrm{i}\alpha.\rho \to \alpha = \forall\alpha.(\rho \to \alpha) \to \alpha$.

The "modified realizability version" of (μ_E) would be:

(μ_E) If $\mu\alpha\rho$ and $\rho[\alpha := \sigma] \to \sigma$ then σ.

And, clearly, if $\Gamma \vdash r : \mu\alpha\rho$ and $\Gamma \vdash s : \rho[\alpha := \sigma] \to \sigma$ then $\Gamma \vdash r\sigma s : \sigma$. Of course, this is no surprise, since (μ_E) is nothing but an application of $(\bigwedge_E)$ which has already been modeled in system F. On the other hand, (μ_I) shrinks to

[2]Unfortunately, this fact cannot be used to embed fixed-point types (see section 2.2.1) into F because of bad reduction behaviour.

(μ_I) If $\forall\alpha\forall\beta.(\alpha \to \beta) \to \rho \to \rho[\alpha := \beta]$ and $\rho[\alpha := \mu\alpha\rho]$ then $\mu\alpha\rho$.

Hence, assume that $\Gamma \vdash m : \forall\alpha\forall\beta.(\alpha \to \beta) \to \rho \to \rho[\alpha := \beta]$ and $\Gamma \vdash t : \rho[\alpha := \mu\alpha\rho]$. We construct a term $C_{\mu\alpha\rho}mt$ such that $\Gamma \vdash C_{\mu\alpha\rho}mt : \mu\alpha\rho$, exactly according to the proof of (μ_I): $C_{\mu\alpha\rho}mt := C_{i\alpha.\rho\to\alpha,\rho[\alpha:=\mu\alpha\rho]}\ell t$ with

$$\ell := \Lambda\alpha\lambda z^{\rho\to\alpha}\lambda y^{\rho[\alpha:=\mu\alpha\rho]}.z\Big(m(\mu\alpha\rho)\alpha(\lambda x^{\mu\alpha\rho}.x\alpha z)y\Big).$$

The term m is called a *monotonicity witness* for $\lambda\alpha\rho$ or even for $\mu\alpha\rho$.

Exercise 1 *Verify step by step that this construction serves its purpose, and that it indeed can be read off the proof of* (μ_I). *(Do not be puzzled with the two completely different meanings of* x, y *and* z.*)*

Obviously, $C_{\mu\alpha\rho}mt \to^*_{\beta\eta} \Lambda\alpha\lambda z^{\rho\to\alpha}.z\Big(m(\mu\alpha\rho)\alpha(\lambda x^{\mu\alpha\rho}.x\alpha z)t\Big)$. Hence,

$$\boxed{(C_{\mu\alpha\rho}mt)\sigma s \to^*_{\beta\eta} s\Big(m(\mu\alpha\rho)\sigma(\lambda x^{\mu\alpha\rho}.x\sigma s)t\Big)}$$

which provides F with *iteration on monotone inductive types*, to be understood as follows: $\Gamma \vdash F := \lambda x^{\mu\alpha\rho}.x\sigma s : \mu\alpha\rho \to \sigma$ if $\Gamma \vdash s : \rho[\alpha := \sigma] \to \sigma$. F represents the function from $\mu\alpha\rho$ to σ defined by iteration on $\mu\alpha\rho$, with step function s, and the characteristic reduction behaviour (note that application associates to the left, and hence F is not applied to t in the reduct shown)

$$F(C_{\mu\alpha\rho}mt) \to^*_{\beta\eta} s(m(\mu\alpha\rho)\sigma F t).$$

Example 2 *The standard representation of the naturals in system* F *is* $\forall\alpha.(\alpha \to \alpha) \to \alpha \to \alpha$. *The aim is to give one by using the concept of inductive types. First recall that* $1 = \forall\alpha.\alpha \to \alpha$, $\rho\times\sigma = \forall\alpha.(\rho \to \sigma \to \alpha) \to \alpha$ *and* $\rho + \sigma = \forall\alpha.(\rho \to \alpha) \to (\sigma \to \alpha) \to \alpha$ *for some* $\alpha \notin FV(\rho) \cup FV(\sigma)$.

Intuitively, $1 \to \alpha$ *is isomorphic to* α, *and* $(\alpha \to \alpha) \to \alpha \to \alpha$ *isomorphic to* $((\alpha \to \alpha) \times \alpha) \to \alpha$, *hence to* $((\alpha \to \alpha) \times (1 \to \alpha)) \to \alpha$ *which in turn is intuitively isomorphic to* $((\alpha + 1) \to \alpha) \to \alpha$, *hence to* $((1 + \alpha) \to \alpha) \to \alpha$. *We now set*

$$\mathsf{nat} := \mu\alpha.1 + \alpha = \forall\alpha.((1 + \alpha) \to \alpha) \to \alpha.$$

A closed monotonicity witness for nat *is given by*

$$m := \Lambda\alpha\Lambda\beta\lambda f^{\alpha\to\beta}\lambda x^{1+\alpha}\Lambda\gamma\lambda u^{1\to\gamma}\lambda v^{\beta\to\gamma}.x\gamma u(\lambda z^{\alpha}.v(fz)).$$

Define $0 := C_{\mathsf{nat}}m(\mathsf{INL}_{1,\mathsf{nat}}\mathsf{IN1})$ *and* $\mathsf{S}t := C_{\mathsf{nat}}m(\mathsf{INR}_{1,\mathsf{nat}}t)$ *(with* $\mathsf{IN1} = \Lambda\alpha\lambda x^{\alpha}x$ *the canonical inhabitant of* 1 *and* $\mathsf{INL}_{\rho,\sigma}r = \Lambda\alpha\lambda x^{\rho\to\alpha}\lambda y^{\sigma\to\alpha}.xr$ *and* $\mathsf{INR}_{\rho,\sigma}r = \Lambda\alpha\lambda x^{\rho\to\alpha}\lambda y^{\sigma\to\alpha}.yr$ *the canonical injections into* $\rho+\sigma$*). Then* $\vdash 0 : \mathsf{nat}$ *and* $\Gamma \vdash t : \mathsf{nat} \Rightarrow \Gamma \vdash \mathsf{S}t : \mathsf{nat}$.

Now assume $\Gamma \vdash a : \sigma$ *and* $\Gamma \vdash b : \sigma \to \sigma$. *Set* $s_{a,b} := \lambda z^{1+\sigma}.z\sigma(\lambda u^1 a)b$. *Setting* $F_{a,b} := \lambda x^{\mathsf{nat}}.x\sigma s_{a,b}$, *we get* $\Gamma \vdash F_{a,b} : \mathsf{nat} \to \sigma$, *and, by some calculation,* $F_{a,b}0 \to^*_{\beta\eta} a$ *and* $F_{a,b}(\mathsf{S}t) \to^*_{\beta\eta} b(F_{a,b}t)$. *Therefore, iteration on natural numbers is a (basic) example of our concept of monotone inductive types.*

Exercise 2 *Let* ρ *be an arbitrary type with* $\alpha \notin FV(\rho)$. *Set*

$$\tau := (((\alpha \to \rho) \to \alpha) \to \alpha) \to \alpha.$$

Show that there is a term m *in system* F *such that*

$$\vdash m : \forall\alpha\forall\beta.(\alpha \to \beta) \to \tau \to \tau[\alpha := \beta].$$

Use m *to construct an inhabitant of* $\mu\alpha.1 + \tau$, *i. e., a term* r *such that* $\vdash r : \mu\alpha.1 + \tau$, *and normalize it. (Remark: The idea to study this type is due to Ulrich Berger.)*

Exercise 3 *Set* $\mathsf{tree}(\rho) := \mu\alpha.1 + (\rho \to \alpha)$ *for some* $\alpha \notin FV(\rho)$. *This type represents the well-founded trees branching over the type* ρ*: Define a monotonicity witness for* $\mathsf{tree}(\rho)$ *and terms* nil *and* $\mathsf{lim}\, t$ *such that* $\vdash \mathsf{nil} : \mathsf{tree}(\rho)$ *and* $\Gamma \vdash t : \rho \to \mathsf{tree}(\rho) \Rightarrow \Gamma \vdash \mathsf{lim}\, t : \mathsf{tree}(\rho)$ *(hence,* $\mathsf{lim}\, t$ *represents the tree consisting of a* ρ*-family of trees) and that iteration on* $\mathsf{tree}(\rho)$ *is recovered in the sense that for terms* a *and* b *with* $\Gamma \vdash a : \sigma$ *and* $\Gamma \vdash b : (\rho \to \sigma) \to \sigma$, *one can find a term* $G_{a,b}$ *with* $\Gamma \vdash G_{a,b} : \mathsf{tree}(\rho) \to \sigma$ *and* $G_{a,b}\mathsf{nil} \to^*_{\beta\eta} a$ *and* $G_{a,b}(\mathsf{lim}\, t) \to^*_{\beta\eta} b(\lambda z^\rho.G_{a,b}(tz))$.

Set $\tau := \mathsf{tree}(\alpha) \to \alpha$ *and define a closed monotonicity witness for* $\lambda\alpha\tau$. *(Remark: This type was brought to my attention by Ulrich Berger, too.)*

Exercise 4 *Show that there is no term* r *in system* F *such that* $\vdash r : \mu\alpha\alpha$. *Hint: Study the shapes of the normal forms and use normalization of system* F.

2.2 Fixed-Point Types

The aim of this section is the study of systems with the (generalized) successor and the (generalized) predecessor in isolation. Hence, we do neither consider means of iteration nor those of primitive recursion although the former are already present in system F and the latter are representable as will be shown in section 2.3.4.

2.2.1 Non-Interleaving Positive Fixed-Point Types

We *extend* system F by types $f\alpha\rho$ which are supposed to describe arbitrary fixed-points of $\lambda\alpha\rho$, i. e., of the operation $\sigma \mapsto \rho[\alpha := \sigma]$. For the time being, we confine ourselves to (non-strictly) positive dependencies which moreover have to be non-interleaved, i. e., $f\alpha\rho$ may only be formed when every occurrence of α in ρ is "to the left of an even number of $\to$" and not free in some subexpression $f\beta\sigma$ of $f\alpha\rho$. The last clause may be rephrased as follows: If fixed-point types $f\beta\sigma$ are formed with a free parameter α then the formation of a fixed-point type $f\alpha\rho$—hence w. r. t. that parameter α—is forbidden.[3] More formally:

[3]Note that otherwise there would be a very high degree of freedom in the interpretation of $f\alpha\rho$ since $f\beta\sigma$ is intended only to model an arbitrary fixed-point.

Definition 3 *We inductively define the set $\mathcal{T}_{\mathrm{npf}}$ of non-interleaving positive fixed-point types and simultaneously for every $\rho \in \mathcal{T}_{\mathrm{npf}}$ the sets $\mathsf{N}_+(\rho)$ and $\mathsf{N}_-(\rho)$ of type variables which occur only positively or occur only negatively, respectively, and moreover do not occur in the scope of a fixed-point type formation (the set $FV(\rho)$ of free type variables is defined as before with the additional $FV(\mathrm{f}\alpha\rho) := FV(\rho) \setminus \{\alpha\}$). Let always range p over the set $\{+,-\}$ of polarities and set $-+ := -$ and $-- := +$.*

- (V) $\alpha \in \mathcal{T}_{\mathrm{npf}}$. $\mathsf{N}_+(\alpha) := \mathcal{V}_\mathsf{T}$. $\mathsf{N}_-(\alpha) := \mathcal{V}_\mathsf{T} \setminus \{\alpha\}$.
- $(\to)$ *If* $\rho, \sigma \in \mathcal{T}_{\mathrm{npf}}$ *then* $\rho \to \sigma \in \mathcal{T}_{\mathrm{npf}}$ *and* $\mathsf{N}_\mathrm{p}(\rho \to \sigma) := \mathsf{N}_{-\mathrm{p}}(\rho) \cap \mathsf{N}_\mathrm{p}(\sigma)$.
- $(\forall)$ *If* $\rho \in \mathcal{T}_{\mathrm{npf}}$ *then* $\forall\alpha\rho \in \mathcal{T}_{\mathrm{npf}}$ *and* $\mathsf{N}_\mathrm{p}(\forall\alpha\rho) := \mathsf{N}_\mathrm{p}(\rho) \cup \{\alpha\}$.
- (f) *If* $\rho \in \mathcal{T}_{\mathrm{npf}}$ *and* $\alpha \in \mathsf{N}_+(\rho)$ *(the only place where the* $\mathsf{N}_\mathrm{p}(\rho)$ *enter the conditions) then* $\mathrm{f}\alpha\rho \in \mathcal{T}_{\mathrm{npf}}$ *and* $\mathsf{N}_\mathrm{p}(\mathrm{f}\alpha\rho) := \mathcal{V}_\mathsf{T} \setminus FV(\mathrm{f}\alpha\rho)$.

Note the change of the polarity in rule $(\to)$ which substantiates the slogan that α's occurrences may only be to the left of an even number of $\to$. In rule (f) we achieve non-interleavedness by removing any free variable of $\mathrm{f}\alpha\rho$.

Examples 3

- $\mathcal{T}_\mathrm{u} \subset \mathcal{T}_{\mathrm{npf}}$.
- $\beta \notin \mathsf{N}_+((\rho \to \beta) \to (\sigma \to \beta) \to \beta)$ *although* $(\rho \to \beta) \to (\sigma \to \beta) \to \beta \in \mathcal{T}_{\mathrm{npf}}$ *if* $\rho, \sigma \in \mathcal{T}_{\mathrm{npf}}$.
- $\alpha, \beta \in \mathsf{N}_+(\alpha + \beta)$.
- $\mathrm{f}\alpha.1 + \alpha \in \mathcal{T}_{\mathrm{npf}}$.
- $\mathrm{f}\beta.1 + (\rho \to \beta) \in \mathcal{T}_{\mathrm{npf}}$ *for* $\rho \in \mathcal{T}_{\mathrm{npf}}$ *and* $\beta \notin FV(\rho)$.
- $(\mathrm{f}\beta.1 + (\alpha \to \beta)) \to \alpha \in \mathcal{T}_{\mathrm{npf}}$ *and* $\alpha \notin \mathsf{N}_+((\mathrm{f}\beta.1 + (\alpha \to \beta)) \to \alpha)$ *although* α *only occurs positively.*

We now define the extension NPF of system F by non-interleaving positive fixed-point types. The set of types is $\mathcal{T}_{\mathrm{npf}}$. The term formation rules are extended, hence the set $\mathcal{T}_\mathsf{NPF}$ of terms has the same defining clauses as $\mathcal{T}_\mathsf{F}$ (but with $\mathcal{T}_\mathsf{F}$ replaced by $\mathcal{T}_\mathsf{NPF}$) and, additionally,

- If $t \in \mathcal{T}_\mathsf{NPF}$ and $\mathrm{f}\alpha\rho \in \mathcal{T}_{\mathrm{npf}}$ then $\mathsf{C}_{\mathrm{f}\alpha\rho} t \in \mathcal{T}_\mathsf{NPF}$.
- If $r \in \mathcal{T}_\mathsf{NPF}$ and $\mathrm{f}\alpha\rho \in \mathcal{T}_{\mathrm{npf}}$[4] then $r\mathsf{E}_{\mathrm{f}\alpha\rho} \in \mathcal{T}_\mathsf{NPF}$.

The definition of the free variables is extended in the obvious way:

- $\mathrm{FV}(\mathsf{C}_{\mathrm{f}\alpha\rho} t) := \mathrm{FV}(t)$.
- $\mathrm{FV}(r\mathsf{E}_{\mathrm{f}\alpha\rho}) := \mathrm{FV}(r)$.

The definition of the free type variables of a term is extended by:

[4]In the sequel, it will be understood that all the mentioned types are taken from $\mathcal{T}_{\mathrm{npf}}$.

- $\mathrm{FTV}(\mathsf{C}_{f\alpha\rho}t) := \mathrm{FV}(f\alpha\rho) \cup \mathrm{FTV}(t)$.
- $\mathrm{FTV}(r\mathsf{E}_{f\alpha\rho}) := \mathrm{FTV}(r) \cup \mathrm{FV}(f\alpha\rho)$.

The substitution $\rho[\alpha := \sigma]$ of σ for the variable α in the type ρ is defined as expected: The binder f is treated like the binder $\forall$. Note that this does not lead out of the set $\mathcal{T}_{npf}$.[5]

The definition of $r[x := s]$ gets the new clauses

- $(\mathsf{C}_{f\alpha\rho}t)[x := s] := \mathsf{C}_{f\alpha\rho}t[x := s]$.
- $(r\mathsf{E}_{f\alpha\rho})[x := s] := r[x := s]\mathsf{E}_{f\alpha\rho}$.

Finally, $r[\alpha := \sigma]$ is extended by

- $(\mathsf{C}_{f\gamma\rho}t)[\alpha := \sigma] := \mathsf{C}_{f\gamma\rho[\alpha:=\sigma]}t[\alpha := \sigma]$.
- $(r\mathsf{E}_{f\alpha\rho})[\alpha := \sigma] := r[\alpha := \sigma]\mathsf{E}_{f\gamma\rho[\alpha:=\sigma]}$.

Note that we may assume in the preceding clauses that $\gamma \notin \{\alpha\} \cup \mathrm{FV}(\sigma)$.

Definition 4 (Typing for system NPF) *The inductive definition of the relation $\Gamma \vdash r : \rho$ for system F is reinterpreted over the larger sets $\mathcal{T}_{npf}$ and $\mathcal{T}_{\mathsf{NPF}}$ of types and terms of NPF, and the following rules are added:*

$$\frac{\Gamma \vdash t : \rho[\alpha := f\alpha\rho]}{\Gamma \vdash \mathsf{C}_{f\alpha\rho}t : f\alpha\rho}(f_I) \qquad \frac{\Gamma \vdash r : f\alpha\rho}{\Gamma \vdash r\mathsf{E}_{f\alpha\rho} : \rho[\alpha := f\alpha\rho]}(f_E)$$

Definition 5 ($\beta\eta$-reduction for NPF) *The relation $\to_{\beta\eta}$ of system F is reinterpreted over the sets $\mathcal{T}_{npf}$ and $\mathcal{T}_{\mathsf{NPF}}$, and the following clauses are added to the inductive definition:*

(β_f) $(\mathsf{C}_{f\alpha\rho}t)\mathsf{E}_{f\alpha\rho} \to_{\beta\eta} t$ *(outer fixed-point β-reduction).*

(η_f) $\mathsf{C}_{f\alpha\rho}(r\mathsf{E}_{f\alpha\rho}) \to_{\beta\eta} r$ *(outer fixed-point η-reduction).*

(C) $t \to_{\beta\eta} t' \Rightarrow \mathsf{C}_{f\alpha\rho}t \to_{\beta\eta} \mathsf{C}_{f\alpha\rho}t'$ *(reduction under $\mathsf{C}_{f\alpha\rho}$).*

(E) $r \to_{\beta\eta} r' \Rightarrow r\mathsf{E}_{f\alpha\rho} \to_{\beta\eta} r'\mathsf{E}_{f\alpha\rho}$ *(reduction under $\mathsf{E}_{f\alpha\rho}$).*

Hence, (β_f) and (η_f) establish the intuitive isomorphism $f\alpha\rho \simeq \rho[\alpha := f\alpha\rho]$. Clearly, subject reduction still holds for NPF.[6] The proof of typed local confluence carries over from system F with the interesting new cases of the rule pairs E/β_f and C/η_f. In the first case, we have

$$\begin{tikzcd}
 & (\mathsf{C}_{f\alpha\rho}(r\mathsf{E}_{f\alpha\rho}))\mathsf{E}_{f\alpha\rho} \arrow[dl, "\beta\eta"'] \arrow[dr, "\beta\eta"] & \\
r\mathsf{E}_{f\alpha\rho} & & r\mathsf{E}_{f\alpha\rho}
\end{tikzcd}$$

[5] A precise proof needs additional statements which are shown in footnote 2 in [Mat99b].

[6] The rule (η_f) requires to attach the type information to E since $f\alpha\rho$ cannot be read off $\rho[\alpha := f\alpha\rho]$: one can always produce the decomposition $\rho[\alpha := f\alpha\rho] = \rho[\alpha := f\alpha\rho][\beta := f\beta.\rho[\alpha := f\alpha\rho]]$ with "fresh" β. (The index to C is even more needed, but has already been extensively used in the constructions of section 2.1.)

The second critical pair is also trivial:

$$\begin{tikzcd}[column sep=small] & \mathsf{C}_{f\alpha\rho}((\mathsf{C}_{f\alpha\rho}t)\mathsf{E}_{f\alpha\rho}) \arrow[dl, "\beta\eta"'] \arrow[dr, "\beta\eta"] & \\ \mathsf{C}_{f\alpha\rho}t & & \mathsf{C}_{f\alpha\rho}t \end{tikzcd}$$

Note that in both situations, the three type indices always have to be equal in order to allow the alternatives (w. r. t. the *rule* applied).

Theorem 2 (Strong normalization of system NPF**)** *If* $\Gamma \vdash r : \rho$ *then* r *is strongly normalizing w. r. t.* $\to_{\beta\eta}$.

Proof By a relatively straightforward extension of the proof for system F in the first part of these notes (p.52 and pp.55–60; this is no surprise since that proof has been designed for the purpose of extensions to inductive types and fixed-point types). $\square$

Exercise 5 *Consider the following simplification of* NPF*: Remove the type information from* $\mathsf{E}_{f\alpha\rho}$*, hence only form* $r\mathsf{E}$ *instead of* $r\mathsf{E}_{f\alpha\rho}$*. Keep the typing rules (which do not exploit this information) and remove* (η_f) *(since subject reduction otherwise fails, cf. footnote 6). Prove strong normalization of the resulting system by merging the proof for* $\mathsf{NPF}_{\mathsf{eta}+}$ *in [Mat99b, pp.305–309] into the one for system* F.[7]
Hints: Give a presentation of the forms of typable terms with vectors $\vec{S}$ *which now shall denote lists of terms, types* and *symbols* E*. Reinterpret the definition of* SN *with these vectors, close* SN *under* $\mathsf{C}_{f\alpha\rho}$ *and under* β_f*-expansion (with vectors added, hence under head* β_f*-expansion). Show that* SN *is contained in the set of strongly normalizing terms. Draw the definition of saturatedness from that of* SN *and adjust the saturated closure* cl*. Check that the constructions of* $\mathcal{M} \to \mathcal{N}$ *and* $\forall\Phi$ *are not affected by the modifications. Define a fixed-point construction on saturated sets: Assume that* Φ *is a monotone function from saturated sets to saturated sets, i. e.,* $\Phi : \mathsf{SAT} \to \mathsf{SAT}$*. Define for* $\mathcal{M} \in \mathsf{SAT}$ *the sets* $\mathsf{I}_\Phi(\mathcal{M}) := \{\mathsf{C}_{f\alpha\rho}t \mid t \in \Phi(\mathcal{M})\}$ *and* $\mathsf{E}_\Phi(\mathcal{M}) := \{r \mid r\mathsf{E} \in \Phi(\mathcal{M})\}$*, and the monotone functions* Φ_I *and* Φ_E *from* SAT *to* SAT *by* $\Phi_I(\mathcal{M}) := \mathsf{cl}(\mathsf{I}_\Phi(\mathcal{M}))$ *and* $\Phi_E(\mathcal{M}) := \mathsf{cl}(\mathsf{E}_\Phi(\mathcal{M}))$*. Since* SAT *is a complete lattice, they have fixed-points* $f_I(\Phi)$ *and* $f_E(\Phi)$*, respectively (we do not constrain the choice among the possible fixed-points except that it has to be a function of* Φ*). Prove that* $I_\Phi(\mathcal{M}) \subseteq \mathsf{SN}$*,* $\mathsf{E}_\Phi(\mathcal{M}) \cap \mathsf{SN} \in \mathsf{SAT}$*, and* $\mathsf{I}_\Phi(\mathcal{M}) \subseteq \mathsf{E}_\Phi(\mathcal{M})$*. Conclude that* $\mathsf{I}_\Phi(\mathcal{M}) \subseteq \Phi_I(\mathcal{M})$*,* $\Phi_E(\mathcal{M}) = \mathsf{E}_\Phi(\mathcal{M}) \cap \mathsf{SN}$*, and that* $f_I(\Phi)$ *is a post-fixed-point of* Φ_E *and* $f_E(\Phi)$ *is a pre-fixed-point of* Φ_I*. Write* $f(\Phi)$ *for both* $f_I(\Phi)$ *and* $f_E(\Phi)$ *and put the facts together to prove:*

[7]For the original NPF, one has to assign types to the saturated sets in the candidate assignments, and therefore has to keep more closely to the proof in [Mat99b] which even considers typed saturated sets (because there fixed types are assumed, in contrast to our type assignment system).

(f_I) *If* $t \in \Phi(f(\Phi))$ *then* $C_{f\alpha\rho}t \in f(\Phi)$.

(f_E) *If* $r \in f(\Phi)$ *then* $rE \in \Phi(f(\Phi))$.

The definition of candidate assignment may remain unchanged. The definition of the saturated set $SC^\rho[\Gamma]$ *of the strongly computable terms w. r. t. the type* ρ *and the candidate assignment* Γ *has to be given simultaneously with the proof that if* $(\alpha : \mathcal{M}) \in \Gamma$ *and* $\alpha \in N_+(\rho)$ *then* $SC^\rho[\Gamma]$ *is an increasing function of* $\mathcal{M}$*, and if* $\alpha \in N_-(\rho)$ *then* $SC^\rho[\Gamma]$ *is a decreasing function of* $\mathcal{M}$*, and if* $\alpha \notin FV(\rho)$ *then* $SC^\rho[\Gamma \setminus \{(\alpha : \mathcal{M})\}] = SC^\rho[\Gamma]$*. We may then define* $SC^{f\alpha\rho}[\Gamma] := f(\Phi)$ *with* $\Phi(\mathcal{M}) := SC^\rho[\Gamma, \alpha : \mathcal{M}]$*, since* Φ *is monotone by the induction hypothesis (*$\alpha \in N_+(\rho)$ *is required for* $f\alpha\rho$ *being a type). Since* $N_p(f\alpha\rho) = \mathcal{V}_T \setminus FV(f\alpha\rho)$*, the additional statements immediately follow from the induction hypothesis.*
The remaining steps are as before: Extend the proof of the substitution lemma and of the fact that typable terms are strongly computable under substitution. Specialization yields the result.

2.2.2 Monotone Fixed-Point Types

We are not confined to non-interleaving positive fixed-point types. Monotonicity suffices. We first give a naive example showing how it should *not* be done. The correct formulation yields a confluent and strongly normalizing system MF. However, since MF and NPF can be embedded into each other even with respect to reduction behaviour, monotonicity is no real extension of non-interleaving positivity.

We turn to the unsuccessful way of introducing monotone fixed-point types. Consider $\rho := f\alpha.\alpha \to 1$ (which is not allowed in NPF). Intuitively, $\rho \simeq \rho \to 1$. A closed monotonicity witness for ρ is given by

$$m := \Lambda\alpha\Lambda\beta\lambda f^{\alpha\to\beta}\lambda x^{\alpha\to 1}\lambda y^{\beta}.\mathsf{IN1},$$

i.e., $\vdash m : \forall\alpha\forall\beta.(\alpha \to \beta) \to (\alpha \to 1) \to \beta \to 1$. We assume that we have C_ρ and E_ρ in the system, with the same typing rules as for NPF. Setting $\omega := \lambda x^\rho.(xE_\rho)x$, we therefore get $\vdash \omega : \rho \to 1$. Hence $\vdash C_\rho\omega : \rho$ and $\vdash \omega(C_\rho\omega) : 1$. However, $\omega(C_\rho\omega) \to_{\beta\eta} ((C_\rho\omega)E_\rho)(C_\rho\omega) \to_{\beta\eta} \omega(C_\rho\omega)$, provided we include $(C_\rho t)E_\rho \to_{\beta\eta} t$ into the definition of $\to_{\beta\eta}$. Consequently, the system is not strongly normalizing (and since there are no other reduction possiblities except those leading to the cycle, it is not even weakly normalizing).

What is the problem with this example? We showed monotonicity, but did not use it. This will be remedied in the following system MF of monotone fixed-point types: It is the extension of system F by arbitrary types $f\alpha\rho$. The new term rules, defining $\mathcal{T}_{MF}$, are

- If $t \in \mathcal{T}_{MF}$ and $m \in \mathcal{T}_{MF}$ then $C_{f\alpha\rho}mt \in \mathcal{T}_{MF}$.
- If $r \in \mathcal{T}_{MF}$ then $rE_{f\alpha\rho} \in \mathcal{T}_{MF}$.

The typing rule (f_I) of NPF is changed to

$$\frac{\Gamma \vdash m : \forall\alpha\forall\beta.(\alpha \to \beta) \to \rho \to \rho[\alpha := \beta] \qquad \Gamma \vdash t : \rho[\alpha := f\alpha\rho]}{C_{f\alpha\rho} m t : f\alpha\rho} (f_I)$$

(f_E) is taken from NPF. Note that we could redo the above example with the only difference that the term m would have to be carried around. But we did not yet specify (β_f) for MF!

Definition 6 ($\beta\eta$-reduction for MF) *The relation $\to_{\beta\eta}$ of system F is extended by the following clauses:*

- (β_f) $(C_{f\alpha\rho} m t) E_{f\alpha\rho} \to_{\beta\eta} m(f\alpha\rho)(f\alpha\rho)(\lambda y^{f\alpha\rho} y)t$ *(outer fixed-point β-reduction).*
- (C) $m \to_{\beta\eta} m' \wedge t \to_{\beta\eta} t' \Rightarrow C_{f\alpha\rho} m t \to_{\beta\eta} C_{f\alpha\rho} m' t'$ *(reduction under $C_{f\alpha\rho}$).*
- (E) $r \to_{\beta\eta} r' \Rightarrow r E_{f\alpha\rho} \to_{\beta\eta} r' E_{f\alpha\rho}$ *(reduction under $E_{f\alpha\rho}$).*

Clearly, we still have subject reduction and typed local confluence. But why is it reasonable to reduce $(C_{f\alpha\rho} m t) E_{f\alpha\rho}$ to $m(f\alpha\rho)(f\alpha\rho)(\lambda y^{f\alpha\rho} y)t$? Because "typical" monotonicity witnesses m have the property that for any ρ and any t, $m\rho\rho(\lambda y^{\rho} y)t \to^*_{\beta\eta} t$. And "bizarre" monotonicity witnesses like in the example above do not lead to non-normalizing terms. This will now be expressed more formally by giving embeddings of NPF (without outer fixed-point η-reduction) into MF and vice versa.

2.2.3 Embedding NPF into MF

We embed NPF, but without outer fixed-point η-reduction, into MF, i. e., we define for every term $r \in \mathcal{T}_{\mathsf{NPF}}$ a term $r' \in \mathcal{T}_{\mathsf{MF}}$ such that $\Gamma \vdash r : \rho \Rightarrow \Gamma \vdash r' : \rho$ and $r \to_{\beta\eta} \hat{r} \Rightarrow r' \to^+_{\beta\eta} \hat{r}'$ (with $\to^+_{\beta\eta}$ the transitive closure of $\to_{\beta\eta}$ which intuitively says that there is at least one $\to_{\beta\eta}$ step). We first define closed monotonicity witnesses m for every $f\alpha\rho \in \mathcal{T}_{npf}$ such that for any τ and any t, $m\tau\tau(\lambda y^{\tau} y)t \to^*_{\beta\eta} t$. Since positivity and negativity are defined simultaneously, we cannot deal with monotonicity witnesses in isolation.

Definition 7 *For every $\rho \in \mathcal{T}_{npf}$ and $\alpha \in N_p(\rho)$ define a term $\mathit{lift}^p_{\lambda\alpha\rho}$ such that $\vdash \mathit{lift}^p_{\lambda\alpha\rho} : \forall\alpha^-\forall\alpha^+.(\alpha^- \to \alpha^+) \to \rho[\alpha := \alpha^{-p}] \to \rho[\alpha := \alpha^p]$ (with α^- and α^+ different type variables not in $FV(\rho)$). This is done by induction on ρ.*

- (triv) *If $\alpha \notin FV(\rho)$ then $\mathit{lift}^p_{\lambda\alpha\rho} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^- \to \alpha^+}\lambda x^{\rho} x$. All the other cases are under the proviso "otherwise".*
- (V) $\mathit{lift}^+_{\lambda\alpha\alpha} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^- \to \alpha^+} f$.
- $(\to)$ $\mathit{lift}^p_{\lambda\alpha.\rho\to\sigma} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^- \to \alpha^+}\lambda x^{\rho[\alpha:=\alpha^{-p}]\to\sigma[\alpha:=\alpha^{-p}]}\lambda y^{\rho[\alpha:=\alpha^{p}]}.$ $\mathit{lift}^p_{\lambda\alpha\sigma}\alpha^-\alpha^+ f\Big(x(\mathit{lift}^{-p}_{\lambda\alpha\rho}\alpha^-\alpha^+ f y)\Big)$.

($\forall$) *$\mathit{lift}^{p}_{\lambda\alpha\forall\gamma\rho} := \Lambda\alpha^{-}\Lambda\alpha^{+}\lambda f^{\alpha^{-}\to\alpha^{+}}\lambda x^{\forall\gamma\rho[\alpha:=\alpha^{-p}]}\Lambda\gamma.\mathit{lift}^{p}_{\lambda\alpha\rho}\alpha^{-}\alpha^{+}f(x\gamma)$ (we assume that $\gamma \notin \{\alpha, \alpha^{-}, \alpha^{+}\}$).*

Note that there is no specific case for (f) *since this is already covered by* (triv)*—thanks to the exclusion of interleaving.*

Lemma 1 *Whenever $\alpha \in N_p(\rho)$ then $\mathit{lift}^{p}_{\lambda\alpha\rho}\tau\tau(\lambda y^{\tau}y)t \to^{*}_{\beta\eta} t$.*

Proof Easy induction on ρ. □

The embedding of NPF into MF is straightforward: Define r' by recursion on r with homomorphic rules except for $(\mathsf{C}_{f\alpha\rho}t)' := \mathsf{C}_{f\alpha\rho}\mathsf{lift}^{+}_{\lambda\alpha\rho}t'$. This is a valid term in $\mathcal{T}_{\mathsf{MF}}$ since no $\mathsf{C}_{f\beta\sigma}$ appears in $\mathsf{lift}^{p}_{\lambda\alpha\rho}$—again thanks to the exclusion of interleaving. Clearly, $\Gamma \vdash r : \rho$ (in NPF) implies $\Gamma \vdash r' : \rho$ (in MF).

Lemma 2 $(r[x := s])' = r'[x := s']$ *and* $(r[\alpha := \sigma])' = r'[\alpha := \sigma]$.

Proof Induction on r. The first result holds since $\mathsf{lift}^{+}_{\lambda\alpha\rho}$ is closed, the second since $(\mathsf{lift}^{p}_{\lambda\alpha\rho}[\beta := \sigma])' = \mathsf{lift}^{p}_{\lambda\alpha.\rho[\beta:=\sigma]}$ for $\alpha \notin \{\beta\} \cup \mathrm{FV}(\sigma)$. □

Lemma 3 *If $r \to_{\beta\eta} \hat{r}$ without the rule of outer fixed-point η-reduction then $r' \to^{+}_{\beta\eta} \hat{r}'$.*

Proof By induction on $\to_{\beta\eta}$.
Case (β): $((\lambda x^{\rho}r)s)' = (\lambda x^{\rho}r')s' \to_{\beta\eta} r'[x := s'] = (r[x := s])'$.
Case (β_{F}): $((\Lambda\alpha r)\sigma)' = (\Lambda\alpha r')\sigma \to_{\beta\eta} r'[\alpha := \sigma] = (r[\alpha := \sigma])'$.
These two cases needed the preceding lemma, while the most interesting case is (β_{f}):

$$((\mathsf{C}_{f\alpha\rho}t)\mathsf{E}_{f\alpha\rho})' = (\mathsf{C}_{f\alpha\rho}\mathsf{lift}^{+}_{\lambda\alpha\rho}t')\mathsf{E}_{f\alpha\rho} \to_{\beta\eta} \mathsf{lift}^{+}_{\lambda\alpha\rho}(f\alpha\rho)(f\alpha\rho)(\lambda y^{f\alpha\rho}y)t' \to^{*}_{\beta\eta} t'$$

by the last but one lemma. □

2.2.4 Embedding MF into NPF

While the last section only justified the formulation of monotone fixed-point types, we will now give an embedding of MF into NPF which allows us to conclude that also MF is strongly normalizing. This time, clearly also the types have to be transformed. Hence, we define a type $\rho' \in \mathcal{T}_{npf}$ for every type of MF, and for every term $r \in \mathcal{T}_{\mathsf{MF}}$ a term $r' \in \mathcal{T}_{\mathsf{NPF}}$ such that whenever $\Gamma \vdash r : \rho$ in MF then $\Gamma' \vdash r' : \rho'$ in NPF (where Γ' is derived from Γ by replacing every type σ occurring in Γ by σ'). Moreover, if $r \to_{\beta\eta} \hat{r}$ in MF then $r' \to^{+}_{\beta\eta} \hat{r}'$ in NPF. Therefore, strong normalization of typable terms of NPF carries over to strong normalization of typable terms of MF since an infinite $\to_{\beta\eta}$-reduction sequence starting in r induces an infinite $\to^{+}_{\beta\eta}$-reduction sequence starting from r', hence also an infinite $\to_{\beta\eta}$-reduction sequence from r' which does not exist. Since our aim is to inherit strong normalization via embeddings, this notion will now be fixed as follows:

Definition 8 (Embedding) *A type-respecting reduction-preserving embedding (embedding for short) of a term rewrite system $\mathcal{S}$ with typing relation $\vdash_{\mathcal{S}}$ into a term rewrite system $\mathcal{S}'$ with typing relation $\vdash_{\mathcal{S}'}$ is a function $-'$ (the $-$ sign represents the indefinite argument of the function $'$) which assigns to every type ρ of $\mathcal{S}$ a type ρ' of $\mathcal{S}'$ and to every term r of $\mathcal{S}$ a term r' of $\mathcal{S}'$ such that the following implications hold: If $\Gamma \vdash_{\mathcal{S}} r : \rho$ then $\Gamma' \vdash_{\mathcal{S}'} r' : \rho$, (where Γ' is Γ with all the types primed), and if $r \to \hat{r}$ in $\mathcal{S}$, then $r' \to^+ \hat{r}'$ in $\mathcal{S}'$. ($\to^+$ denotes the transitive closure of $\to$.)*

Definition 9 *Define $\rho' \in \mathcal{T}_{\mathrm{npf}}$ for every type ρ of* MF *by recursion on ρ as follows:*

(V) $\alpha' := \alpha$.

($\to$) $(\rho \to \sigma)' := \rho' \to \sigma'$.

($\forall$) $(\forall\alpha\rho)' := \forall\alpha\rho'$.

(f) $(\mathsf{f}\alpha\rho)' := \mathsf{f}\alpha\forall\beta.(\alpha \to \beta) \to \rho'[\alpha := \beta]$. *(Note that by induction hypothesis, $\rho' \in \mathcal{T}_{\mathrm{npf}}$, hence $\rho'[\alpha := \beta] \in \mathcal{T}_{\mathrm{npf}}$. Since $\alpha \notin FV(\rho'[\alpha := \beta])$, $\alpha \in N_+(\rho'[\alpha := \beta])$, hence $\alpha \in N_+(\forall\beta.(\alpha \to \beta) \to \rho'[\alpha := \beta])$).*

Obviously, $\mathrm{FV}(\rho') = \mathrm{FV}(\rho)$.

Lemma 4 $(\rho[\alpha := \sigma])' = \rho'[\alpha := \sigma']$.

Proof Induction on ρ. □

Definition 10 *Define $r' \in \mathcal{T}_{\mathsf{NPF}}$ for every $r \in \mathcal{T}_{\mathsf{MF}}$ by recursion on r as follows:*

- $x' := x$.
- $(\lambda x^{\rho} r)' := \lambda x^{\rho'} r'$.
- $(rs)' := r's'$.
- $(\Lambda\alpha r)' := \Lambda\alpha r'$.
- $(r\sigma)' := r'\sigma'$.
- $(\mathsf{C}_{\mathsf{f}\alpha\rho} m t)' := \mathsf{C}_{(\mathsf{f}\alpha\rho)'}\Big(\Lambda\beta\lambda z^{(\mathsf{f}\alpha\rho)' \to \beta}.m'(\mathsf{f}\alpha\rho)'\beta z t'\Big)$.
- $(r\mathsf{E}_{\mathsf{f}\alpha\rho})' := r'\mathsf{E}_{(\mathsf{f}\alpha\rho)'}(\mathsf{f}\alpha\rho)'(\lambda y^{(\mathsf{f}\alpha\rho)'} y)$.

Lemma 5 *If $\Gamma \vdash r : \rho$ then $\Gamma' \vdash r' : \rho'$.*

Proof Induction on $\Gamma \vdash r : \rho$. □

Lemma 6 $(r[x := s])' = r'[x := s']$ *and* $(r[\alpha := \sigma])' = r'[\alpha := \sigma']$.

Proof Induction on r. □

Lemma 7 *If* $r \to_{\beta\eta} \hat{r}$ *then* $r' \to^{+}_{\beta\eta} \hat{r}'$.

Proof By induction on $\to_{\beta\eta}$. The only interesting case is that of an outer fixed-point β-reduction: $(\mathsf{C}_{f\alpha\rho} m t)\mathsf{E}_{f\alpha\rho})' =$

$$= \mathsf{C}_{(f\alpha\rho)'}\Big(\Lambda\beta\lambda z^{(f\alpha\rho)'\to\beta}.m'(f\alpha\rho)'\beta z t'\Big)\mathsf{E}_{(f\alpha\rho)'}(f\alpha\rho)'(\lambda y^{(f\alpha\rho)'}y)$$
$$\to_{\beta\eta} \Big(\Lambda\beta\lambda z^{(f\alpha\rho)'\to\beta}.m'(f\alpha\rho)'\beta z t'\Big)(f\alpha\rho)'(\lambda y^{(f\alpha\rho)'}y)$$
$$\to_{\beta\eta} \Big(\lambda z^{(f\alpha\rho)'\to(f\alpha\rho)'}.m'(f\alpha\rho)'(f\alpha\rho)'z t'\Big)(\lambda y^{(f\alpha\rho)'}y)$$
$$\to_{\beta\eta} m'(f\alpha\rho)'(f\alpha\rho)'(\lambda y^{(f\alpha\rho)'}y)t' = \Big(m(f\alpha\rho)(f\alpha\rho)(\lambda y^{f\alpha\rho}y)t\Big)'. \qquad \square$$

Corollary 8 (Strong normalization of MF) *If* $\Gamma \vdash r : \rho$ *in* MF *then* r *is strongly normalizing w. r. t.* $\to_{\beta\eta}$.

2.3 Positive Inductive Types, Monotone Inductive Types, Primitive Recursion, and the Relation to Fixed-Point Types

2.3.1 Positive Inductive Types

System F is extended by constructions for iteration on types $\mu\alpha\rho$ with α only occurring positively in ρ. In this way, the monotonicity witnesses need not be carried around in the terms, and canonical closed monotoncity witnesses are used, which exist by positivity, and are defined once and for all. The resulting system is called PI.

Definition 11 *Inductively define the set* $\mathcal{T}_{\mathrm{pi}}$ *of positive inductive types and simultaneously for every* $\rho \in \mathcal{T}_{\mathrm{pi}}$ *the sets* $+(\rho)$ *and* $-(\rho)$ *of type variables which occur only positively or occur only negatively in* ρ*, respectively. (Again,* p *will always range over* $\{+,-\}$*, and* $-+ := -$ *and* $-- := +$*.)*

- (V) $\alpha \in \mathcal{T}_{\mathrm{pi}}$. $+(\alpha) := \mathcal{V}_{\mathsf{T}}$. $-(\alpha) := \mathcal{V}_{\mathsf{T}} \setminus \{\alpha\}$.
- $(\to)$ *If* $\rho, \sigma \in \mathcal{T}_{\mathrm{pi}}$ *then* $\rho \to \sigma \in \mathcal{T}_{\mathrm{pi}}$ *and* $p(\rho \to \sigma) := (-p)(\rho) \cap p(\sigma)$.
- $(\forall)$ *If* $\rho \in \mathcal{T}_{\mathrm{pi}}$ *then* $\forall\alpha\rho \in \mathcal{T}_{\mathrm{pi}}$ *and* $p(\forall\alpha\rho) := p(\rho) \cup \{\alpha\}$.
- (μ) *If* $\rho \in \mathcal{T}_{\mathrm{pi}}$ *and* $\alpha \in +(\rho)$ *(the place where the* $p(\rho)$ *enter the conditions) then* $\mu\alpha\rho \in \mathcal{T}_{\mathrm{pi}}$ *and* $p(\mu\alpha\rho) := p(\rho) \cup \{\alpha\}$.

The set $\mathrm{FV}(\rho)$ of free type variables of ρ is defined as expected (with $\mathrm{FV}(\mu\alpha\rho) = \mathrm{FV}(\rho) \setminus \{\alpha\}$).

Examples 4 *For* $\rho \in \mathcal{T}_{\mathrm{pi}}$*, the type of well-founded trees with branching degree* ρ*,* $\mathsf{tree}(\rho) := \mu\alpha.1 + (\rho \to \alpha) \in \mathcal{T}_{\mathrm{pi}}$*, and also the type of "heavily-branching" well-founded trees* $\mathsf{Tree} := \mu\alpha.1 + (\mathsf{tree}(\alpha) \to \alpha) \in \mathcal{T}_{\mathrm{pi}}$ *since* $\alpha \in -(\mathsf{tree}(\alpha))$*. This type exhibits interleaving since the free parameter* α *of* $\mathsf{tree}(\alpha)$ *is bound by the outer* μ*. Note that the branching degree of* Tree *is* $\mathsf{tree}(\mathsf{Tree})$*, hence the well-founded trees over* Tree *about to be defined.*

Note that $\mathcal{T}_{\mathrm{pi}}$ is closed under substitution.

System PI has $\mathcal{T}_{\mathrm{pi}}$ as the set of types, and the term formation rules of F are reinterpreted over $\mathcal{T}_{\mathrm{pi}}$, and extended by the following two clauses to yield the set $\mathcal{T}_{\mathsf{PI}}$ of terms of PI:

- If $t \in \mathcal{T}_{\mathsf{PI}}$ and $\mu\alpha\rho \in \mathcal{T}_{\mathrm{pi}}$ then $\mathsf{C}_{\mu\alpha\rho}t \in \mathcal{T}_{\mathsf{PI}}$.
- If $r \in \mathcal{T}_{\mathsf{PI}}$, $\sigma \in \mathcal{T}_{\mathrm{pi}}$ and $s \in \mathcal{T}_{\mathsf{PI}}$ then $r\mathsf{E}_{\mu}\sigma s \in \mathcal{T}_{\mathsf{PI}}$.

Free type variables $\mathrm{FTV}(r)$, free term variables $\mathrm{FV}(r)$ and $r[x := s]$ and $r[\alpha := \sigma]$ are defined in the obvious way.

The typing rules of F are extended by:

$$\frac{\Gamma \vdash t : \rho[\alpha := \mu\alpha\rho]}{\Gamma \vdash \mathsf{C}_{\mu\alpha\rho}t : \mu\alpha\rho}(\mu_I) \qquad \frac{\Gamma \vdash r : \mu\alpha\rho \qquad \Gamma \vdash s : \rho[\alpha := \sigma] \to \sigma}{\Gamma \vdash r\mathsf{E}_{\mu}\sigma s : \sigma}(\mu_E)$$

Before the β-reduction rule of iteration for PI can be defined, we have to provide the canonical monotonicity witnesses. Because of the possible interleaving of inductive types, we first have to define the height $\mathsf{h}(\lambda\vec{\alpha}\rho)$ of a multiply abstracted type $\lambda\vec{\alpha}\rho$ which is nothing but the type ρ, seen as dependent on the type variables $\vec{\alpha} = \alpha_1, \ldots, \alpha_n$.

Definition 12 *Define the height* $\mathsf{h}(\lambda\vec{\alpha}\rho) \in \mathbb{N}$ *by recursion on* ρ *as follows:*

- *If* $\vec{\alpha} \cap FV(\rho) = \emptyset$ *then* $\mathsf{h}(\lambda\vec{\alpha}\rho) := 0$. Otherwise:
- $\mathsf{h}(\lambda\vec{\alpha}\alpha) := 0$.
- $\mathsf{h}(\lambda\vec{\alpha}.\rho \to \sigma) := 1 + \max(\mathsf{h}(\lambda\vec{\alpha}\rho), \mathsf{h}(\lambda\vec{\alpha}\sigma))$.
- $\mathsf{h}(\lambda\vec{\alpha}\nabla\alpha\rho) := 1 + \mathsf{h}(\lambda\vec{\alpha}\lambda\alpha\rho)$ *for* $\nabla \in \{\forall, \mu\}$.

Define $(\lambda\vec{\alpha}\rho)[\gamma := \sigma] := \lambda\vec{\alpha}.\rho[\gamma := \sigma]$ (we assume that $\vec{\alpha} \cap (\{\gamma\} \cup \mathrm{FV}(\sigma)) = \emptyset$).

Lemma 9 $\mathsf{h}((\lambda\vec{\alpha}\rho)[\gamma := \sigma]) = \mathsf{h}(\lambda\vec{\alpha}\rho)$. *For* $\vec{\alpha}' \subseteq \vec{\alpha}$, $\mathsf{h}(\lambda\vec{\alpha}\rho) \geq \mathsf{h}(\lambda\vec{\alpha}'\rho)$.

Proof Induction on ρ. Unequality may occur when the removal of type variables leads into the initial case of the height definition. □

Corollary 10 *If* $\alpha \in FV(\nabla\gamma\rho)$ *then* $\mathsf{h}(\lambda\alpha\nabla\gamma\rho) > \mathsf{h}(\lambda\gamma.\rho[\alpha := \sigma])$ *(we assume that* $\gamma \notin \{\alpha\} \cup FV(\sigma)$*) and* $\mathsf{h}(\lambda\alpha\nabla\gamma\rho) > \mathsf{h}(\lambda\alpha.\rho[\gamma := \sigma])$ *(we assume that* $\alpha \notin \{\gamma\} \cup FV(\sigma)$*).*

Definition 13 *For every* $\rho \in \mathcal{T}_{\mathrm{pi}}$ *and* $\alpha \in p(\rho)$ *define a term* $\mathit{lift}^{p}_{\lambda\alpha\rho}$ *such that* $\vdash \mathit{lift}^{p}_{\lambda\alpha\rho} : \forall\alpha^-\forall\alpha^+.(\alpha^- \to \alpha^+) \to \rho[\alpha := \alpha^{-p}] \to \rho[\alpha := \alpha^{p}]$ *(with* α^- *and* α^+ *different type variables not in* $FV(\rho)$*). This is done by induction on* $\mathsf{h}(\lambda\alpha\rho)$ *(compare with section 2.2.3).*

(triv) *If* $\alpha \notin FV(\rho)$ *then* $\mathit{lift}^{p}_{\lambda\alpha\rho} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^- \to \alpha^+}\lambda x^{\rho}x$. *All the other cases are under the proviso "otherwise".*

(V) $\mathit{lift}^{+}_{\lambda\alpha\alpha} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^- \to \alpha^+}f$.

($\to$) $\mathit{lift}^p_{\lambda\alpha.\rho\to\sigma} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^-\to\alpha^+}\lambda x^{\rho[\alpha:=\alpha^{-p}]\to\sigma[\alpha:=\alpha^{-p}]}\lambda y^{\rho[\alpha:=\alpha^p]}.$ $\mathit{lift}^p_{\lambda\alpha\sigma}\alpha^-\alpha^+ f\Big(x(\mathit{lift}^{-p}_{\lambda\alpha\rho}\alpha^-\alpha^+ fy)\Big)$.

($\forall$) $\mathit{lift}^p_{\lambda\alpha\forall\gamma\rho} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^-\to\alpha^+}\lambda x^{\forall\gamma\rho[\alpha:=\alpha^{-p}]}\Lambda\gamma.\mathit{lift}^p_{\lambda\alpha\rho}\alpha^-\alpha^+ f(x\gamma)$ *(we assume that* $\gamma\notin\{\alpha,\alpha^-,\alpha^+\}$*).*

(μ) $\mathit{lift}^p_{\lambda\alpha\mu\gamma\rho} := \Lambda\alpha^-\Lambda\alpha^+\lambda f^{\alpha^-\to\alpha^+}\lambda x^{\mu\gamma\rho[\alpha:=\alpha^{-p}]}.x\mathsf{E}_\mu(\mu\gamma.\rho[\alpha:=\alpha^p])$ $\Big(\lambda z^{\rho[\alpha:=\alpha^{-p}][\gamma:=\mu\gamma.\rho[\alpha:=\alpha^p]]}.\mathsf{C}_{\mu\gamma.\rho[\alpha:=\alpha^p]}(\mathit{lift}^p_{\lambda\alpha\rho[\gamma:=\mu\gamma.\rho[\alpha:=\alpha^p]]}\alpha^-\alpha^+ fz)\Big)$. *Note that* $\mathsf{h}(\lambda\alpha\mu\gamma\rho) > \mathsf{h}(\lambda\alpha\rho[\gamma:=\mu\gamma.\rho[\alpha:=\alpha^p]])$ *by the corollary. (We again assumed that* $\gamma\notin\{\alpha,\alpha^-,\alpha^+\}$*.)*

Lemma 11 *Every occurrence of* $\mathsf{C}_{\mu\alpha'\rho'}t$ *in* $\mathit{lift}^p_{\lambda\alpha\rho}$ *has* $\mathsf{h}(\lambda\alpha'\rho') < \mathsf{h}(\lambda\alpha\rho)$.

Proof By induction on ρ, using the corollary. □

Definition 14 ($\beta\eta$-reduction for PI) *The relation* $\to_{\beta\eta}$ *of system* F *is extended by the following clauses:*

(β_μ) $(\mathsf{C}_{\mu\alpha\rho}t)\mathsf{E}_\mu\sigma s \to_{\beta\eta} s\Big(\mathit{lift}^+_{\lambda\alpha\rho}(\mu\alpha\rho)\sigma(\lambda x^{\mu\alpha\rho}.x\mathsf{E}_\mu\sigma s)t\Big)$ *(β-reduction rule of iteration on positive inductive types).*

(C) $t\to_{\beta\eta} t' \Rightarrow \mathsf{C}_{\mu\alpha\rho}t \to_{\beta\eta} \mathsf{C}_{\mu\alpha\rho}t'$ *(reduction under* $\mathsf{C}_{\mu\alpha\rho}$*).*

(E) $r\to_{\beta\eta} r' \wedge s\to_{\beta\eta} s' \Rightarrow r\mathsf{E}_\mu\sigma s \to_{\beta\eta} r'\mathsf{E}_\mu\sigma s'$ *(reduction under* E_μ*).*

Exercise 6 *Set* $\mathsf{cont}(\rho) := \mu\alpha.1 + (\alpha\to\rho)\to\rho$ *for some* $\alpha\notin FV(\rho)$*. Set* $\mathsf{D} := \mathsf{C}_{\mathsf{cont}(\rho)}(\mathsf{INL}_{1,(\mathsf{cont}(\rho)\to\rho)\to\rho}\mathsf{IN}1)$ *and* $\mathsf{C}f := \mathsf{C}_{\mathsf{cont}(\rho)}(\mathsf{INR}_{1,(\mathsf{cont}(\rho)\to\rho)\to\rho}f)$. *Show that* $\vdash \mathsf{D} : \mathsf{cont}(\rho)$ *and that whenever* $\Gamma\vdash f : (\mathsf{cont}(\rho)\to\rho)\to\rho$*, then* $\Gamma\vdash \mathsf{C}f : \mathsf{cont}(\rho)$*. Define a term* e *such that* $\vdash e : \mathsf{cont}(\mathsf{nat})\to\mathsf{nat}$ *and* $e\mathsf{D}\to^+_{\beta\eta} 0$ *and* $e(\mathsf{C}f)\to^+_{\beta\eta} fe$*. (Therefore, definitions of this kind are strongly normalizing. Note that* fe *is by no means* e *applied to an argument smaller than* $\mathsf{C}f$ *in any sense. The idea to study* e *is taken from [Hof95].)*

2.3.2 Monotone Inductive Types

In contrast to PI, we do not specify the monotonicity witnesses in advance but carry them around like in MF. The resulting system will be called MI. It has types $\mu\alpha\rho$ without restriction, and the term rules of F are extended to yield $\mathcal{T}_{\mathsf{MI}}$ as follows:

- If $m\in\mathcal{T}_{\mathsf{MI}}$ and $t\in\mathcal{T}_{\mathsf{MI}}$ then $\mathsf{C}_{\mu\alpha\rho}mt\in\mathcal{T}_{\mathsf{MI}}$
- If $r\in\mathcal{T}_{\mathsf{MI}}$ and $s\in\mathcal{T}_{\mathsf{MI}}$ then $r\mathsf{E}_\mu\sigma s\in\mathcal{T}_{\mathsf{MI}}$.

The typing rules of system F are extended by

$$\frac{\Gamma\vdash m : \forall\alpha\forall\beta.(\alpha\to\beta)\to\rho\to\rho[\alpha:=\beta] \qquad \Gamma\vdash t : \rho[\alpha:=\mu\alpha\rho]}{\Gamma\vdash \mathsf{C}_{\mu\alpha\rho}mt : \mu\alpha\rho}(\mu_I)$$

$$\frac{\Gamma\vdash r : \mu\alpha\rho \qquad \Gamma\vdash s : \rho[\alpha:=\sigma]\to\sigma}{\Gamma\vdash r\mathsf{E}_\mu\sigma s : \sigma}(\mu_E)$$

The reduction relation $\to_{\beta\eta}$ of F is now extended by the β-reduction rule of iteration on monotone inductive types (and the obvious rules of term closure not shown here):

$$(\beta_\mu)\ \ (\mathsf{C}_{\mu\alpha\rho} m t)\mathsf{E}_\mu \sigma s \to_{\beta\eta} s\Big(m(\mu\alpha\rho)\sigma(\lambda x^{\mu\alpha\rho}.x\mathsf{E}_\mu\sigma s)t\Big)$$

We already know this rule from section 2.1 where the encoding of monotone inductive types in F had the property that

$$(\mathsf{C}_{\mu\alpha\rho} m t)\sigma s \to^+_{\beta\eta} s\Big(m(\mu\alpha\rho)\sigma(\lambda x^{\mu\alpha\rho}.x\sigma s)t\Big).$$

(The property was stated with $\to^*_{\beta\eta}$ instead of $\to^+_{\beta\eta}$, but trivially, at least one reduction step was needed.) Therefore, it is obvious that MI embeds into F, hence strong normalization is inherited from F for typable terms. Nevertheless, there are good reasons to extend system F by those constructions explicitly which will become clear in the next section. But before that, it is shown that MI at least covers the positive inductive types, as expressed by an embedding of system PI into MI.

Definition 15 *Define the set* ST *of stratified terms of* PI *inductively:*

- $x \in \mathsf{ST}$.
- *If* $r \in \mathsf{ST}$ *then* $\lambda x^\rho r \in \mathsf{ST}$.
- *If* $r, s \in \mathsf{ST}$ *then* $rs \in \mathsf{ST}$.
- *If* $t \in \mathsf{ST}$, $\mu\alpha\rho \in \mathcal{T}_{\mathrm{pi}}$ *and* $\mathit{lift}^+_{\lambda\alpha\rho} \in \mathsf{ST}$ *then* $\mathsf{C}_{\mu\alpha\rho} t \in \mathsf{ST}$.
- *If* $r, s \in \mathsf{ST}$ *then* $r\mathsf{E}_\mu \sigma s \in \mathsf{ST}$.

Define by recursion on $r \in \mathsf{ST}$ the term $r' \in \mathcal{T}_{\mathsf{MI}}$ such that $\Gamma \vdash r : \rho$ (in PI) implies $\Gamma \vdash r' : \rho$ (in MI): Everything shall be done homomorphically, except for $(\mathsf{C}_{\mu\alpha\rho} t)' := \mathsf{C}_{\mu\alpha\rho}(\mathsf{lift}^+_{\lambda\alpha\rho})' t'$ (for which the definition of ST has been designed).

Lemma 12 *For* $r \in \mathsf{ST}$, $(r[x := s])' = r'[x := s']$ *and* $(r[\alpha := \sigma])' = r'[\alpha := \sigma]$.

Proof Induction on $r \in \mathsf{ST}$. We need the same observations on $\mathsf{lift}^p_{\lambda\alpha\rho}$ as in the proof of Lemma 2. □

Lemma 13 *Every (not necessarily proper) subterm* r *of* $\mathit{lift}^p_{\lambda\alpha\rho}$ *is stratified, i. e.,* $r \in \mathsf{ST}$.

Proof Main induction on $\mathsf{h}(\lambda\alpha\rho)$, side induction on the term r. If r is not of the form $\mathsf{C}_{\mu\alpha'\rho'} t$ then the side induction hypothesis applies. If $r = \mathsf{C}_{\mu\alpha'\rho'} t$ then by Lemma 11, $\mathsf{h}(\lambda\alpha'\rho') < \mathsf{h}(\lambda\alpha\rho)$. By the main induction hypothesis, applied to $\mathsf{lift}^+_{\lambda\alpha'\rho'}$ itself, $\mathsf{lift}^+_{\lambda\alpha'\rho'} \in \mathsf{ST}$. By the side induction hypothesis, $t \in \mathsf{ST}$, hence also $r \in \mathsf{ST}$. □

Corollary 14 $\mathsf{ST} = \mathcal{T}_{\mathsf{PI}}$, *hence every term is stratified.*

Proof Since $\mathsf{lift}^+_{\lambda\alpha\rho} \in \mathsf{ST}$, the definition of ST does not impose any restriction on the terms in $\mathcal{T}_{\mathsf{PI}}$ that enter ST. □

Hence, r' is defined for every $r \in \mathcal{T}_{\mathsf{PI}}$, and it is easy to check that if $r \to_{\beta\eta} \hat{r}$ in PI then $r' \to_{\beta\eta} \hat{r}'$ in MI.

Note that the technical problems only arised since the canonical monotonicity witnesses may contain $\mathsf{C}_{\mu\alpha\rho}t$ due to the allowed interleaving of positive inductive types.

2.3.3 Adding (Full) Primitive Recursion

In the systems PI and MI, we only have modeled iteration on inductive types which is already available in system F. Recall from example 2 that in the case of naturals, this provides us with a term construction $F_{a,b}$ such that whenever $\Gamma \vdash a : \sigma$ and $\Gamma \vdash b : \sigma \to \sigma$ then $\Gamma \vdash F_{a,b} : \mathsf{nat} \to \sigma$ and $F_{a,b}0 \to_{\beta\eta} a$ and $F_{a,b}(\mathsf{S}t) \to^*_{\beta\eta} b(F_{a,b}t)$. But we also want to model primitive recursion. In the case of naturals, this would require a term construction $R_{a,b}$ such that $\Gamma \vdash a : \sigma$ and $\Gamma \vdash b : \mathsf{nat} \to \sigma \to \sigma$ imply $\Gamma \vdash R_{a,b} : \mathsf{nat} \to \sigma$, and $R_{a,b}0 \to_{\beta\eta} a$ and $R_{a,b}(\mathsf{S}t) \to^*_{\beta\eta} bt(F_{a,b}t)$. Obviously, this is a combination of inversion (provided by the systems of fixed-point types) and iteration (allowing to use the function to be defined at the smaller argument t). In the general situation, a slightly different formulation is used: We extend PI to PIR (positive inductive types with iteration and primitive recursion) by keeping the types and extending the set of terms by one rule, leading to the set $\mathcal{T}_{\mathsf{PIR}}$, as follows: If $r, s \in \mathcal{T}_{\mathsf{PIR}}$ then $r\mathsf{E}^+_\mu \sigma s \in \mathcal{T}_{\mathsf{PIR}}$. The additional typing rule is:

$$\frac{\Gamma \vdash r : \mu\alpha\rho \qquad \Gamma \vdash s : \rho[\alpha := \mu\alpha\rho \times \sigma] \to \sigma}{\Gamma \vdash r\mathsf{E}^+_\mu \sigma s : \sigma}(\mu^+_{\mathsf{E}})$$

This gives rise to the additional β-reduction rule (β^+_μ) of primitive recursion on positive inductive types:[8]

$$(\mathsf{C}_{\mu\alpha\rho}t)\mathsf{E}^+_\mu\sigma s \to_{\beta\eta} s\Big(\mathsf{lift}^+_{\lambda\alpha\rho}(\mu\alpha\rho)(\mu\alpha\rho\times\sigma)\Big(\lambda x^{\mu\alpha\rho}.\langle x, (\lambda x^{\mu\alpha\rho}.x\mathsf{E}^+_\mu\sigma s)x\rangle_{\mu\alpha\rho,\sigma}\Big)t\Big).$$

Note that we keep iteration since this is needed for the definition of $\mathsf{lift}^{p}_{\lambda\alpha\rho}$. Clearly, subject reduction still holds.

Exercise 7 *Show that we indeed modeled primitive recursion on naturals in* PIR. *(Of course, this is a very special instance of primitive recursion on arbitrary positive inductive types.)*

System MI may be extended by (full) primitive recursion as well: The system MIR has the same types as MI, but the term system is extended to the set $\mathcal{T}_{\mathsf{MIR}}$ by adding $r\mathsf{E}^+_\mu\sigma s$ and the respective typing rule as for PIR. Accordingly, the new reduction rule (β^+_μ) becomes

$$(\mathsf{C}_{\mu\alpha\rho}mt)\mathsf{E}^+_\mu\sigma s \to_{\beta\eta} s\Big(m(\mu\alpha\rho)(\mu\alpha\rho\times\sigma)\Big(\lambda x^{\mu\alpha\rho}.\langle x, (\lambda x^{\mu\alpha\rho}.x\mathsf{E}^+_\mu\sigma s)x\rangle_{\mu\alpha\rho,\sigma}\Big)t\Big).$$

[8] Recall that $\langle r, s\rangle_{\rho,\sigma} = \Lambda\alpha\lambda z^{\rho\to\sigma\to\alpha}.zrs$ gives the pair, and $rL_{\rho,\sigma} = r\rho(\lambda x^\rho\lambda y^\sigma.x)$ and $rR_{\rho,\sigma} = r\sigma(\lambda x^\rho\lambda y^\sigma.y)$ model the projections.

MIR again enjoys subject reduction. The embedding of PI into MI may be extended in the obvious way to an embedding of PIR into MIR. In [Mat99a, 3.2] it is shown that NPF (without (η_f)) embeds into the non-interleaved fragment (called NPI) of PIR without iteration, hence an embedding of PIR into system F is at least as unlikely to find as one of NPF (without (η_f)) into F.

2.3.4 Embedding Monotone Inductive Types with Primitive Recursion into Non-Interleaving Fixed-Point Types

Although we added a lot of expressivity to system F when defining MIR, we do not get beyond NPF—even w. r. t. reduction behaviour, i. e., there is an embedding of MIR into NPF which is indeed a collapse: Primitive recursion on arbitrary monotone inductive types (with possible interleaving and also possibly free variables in the monotonicity witnesses) is reduced to the folding and unfolding of fixed-points for non-interleaved positive dependencies.

The embedding is given as follows: Define $\rho' \in \mathcal{T}_{\mathrm{npf}}$ by recursion on ρ homomorphically, except for

$$(\mu\alpha\rho)' := \mathsf{f}\beta\forall\gamma.\Big(\Big(\forall\alpha.(\beta\times\gamma\to\alpha)\to\rho'\Big)\to\gamma\Big)\to\gamma.$$

By induction hypothesis, $\rho' \in \mathcal{T}_{\mathrm{npf}}$. Hence, $(\mu\alpha\rho)' \in \mathcal{T}_{\mathrm{npf}}$ since β only occurs positively in

$$\forall\gamma.\Big(\Big(\forall\alpha.(\beta\times\gamma\to\alpha)\to\rho'\Big)\to\gamma\Big)\to\gamma.$$

In fact, the only occurrence of β is 6 times to the left of $\to$ (do not forget that the coding of $\beta\times\gamma$ provides 2 of them).

Define $r' \in \mathcal{T}_{\mathsf{NPF}}$ by recursion on $r \in \mathcal{T}_{\mathsf{MIR}}$ homomorphically, but with

$$(\mathsf{C}_{\mu\alpha\rho}mt)' := \mathsf{C}_{(\mu\alpha\rho)'}\Big(\Lambda\gamma\lambda z^{(\forall\alpha.((\mu\alpha\rho)'\times\gamma\to\alpha)\to\rho')\to\gamma}.z\Big(\Lambda\alpha\lambda u^{(\mu\alpha\rho)'\times\gamma\to\alpha}.$$
$$m'(\mu\alpha\rho)'\alpha\Big(\lambda x^{(\mu\alpha\rho)'}.u\langle x,(\lambda x^{(\mu\alpha\rho)'}.x\mathsf{E}_{(\mu\alpha\rho)'}\gamma z)x\rangle_{(\mu\alpha\rho)',\gamma}\Big)t\Big)\Big)$$

and

$$(r\mathsf{E}^{+}_{\mu}\sigma s)' := r'\mathsf{E}_{(\mu\alpha\rho)'}\sigma'\Big(\lambda z^{\forall\alpha.((\mu\alpha\rho)'\times\sigma'\to\alpha)\to\rho'}.s'\Big(z((\mu\alpha\rho)'\times\sigma')(\lambda x^{(\mu\alpha\rho)'\times\sigma'}x)\Big)\Big)$$

Quite similarly,

$$(r\mathsf{E}_{\mu}\sigma s)' := r'\mathsf{E}_{(\mu\alpha\rho)'}\sigma'\Big(\lambda z^{\forall\alpha.((\mu\alpha\rho)'\times\sigma'\to\alpha)\to\rho'}.s'\Big(z\sigma'(\lambda x^{(\mu\alpha\rho)'\times\sigma'}.x\mathsf{R}_{(\mu\alpha\rho)',\sigma'})\Big)\Big)$$

It is routine to check that $\Gamma \vdash r : \rho$ in MIR implies $\Gamma' \vdash r' : \rho'$ in NPF, and also that reduction is preserved, i. e., that $r \to_{\beta\eta} \hat{r}$ in MIR implies $r' \to^{+}_{\beta\eta} \hat{r}'$ in NPF. Therefore, we have an embedding of MIR into NPF, and consequently, strong normalization of the typable terms of MIR ensues.

References

[Ber93] Ulrich Berger. Program extraction from normalization proofs. In Marc Bezem and J.F. Groote, editors, *Typed Lambda Calculi and Applications*, volume 664 of *Lecture Notes in Computer Science*, pages 91–106. Springer Verlag, 1993.

[Hof95] Martin Hofmann. Approaches to recursive datatypes—a case study. 5 pages. Unpublished, April 1995.

[Mat98] Ralph Matthes. *Extensions of System F by Iteration and Primitive Recursion on Monotone Inductive Types*. Doktorarbeit (PhD thesis), University of Munich, 1998. Available via the homepage `http://www.tcs.informatik.uni-muenchen.de/~matthes/`.

[Mat99a] Ralph Matthes. Monotone (co)inductive types and positive fixed-point types. *Theoretical Informatics and Applications*, 33(4/5):309–328, 1999.

[Mat99b] Ralph Matthes. Monotone fixed-point types and strong normalization. In Georg Gottlob, Etienne Grandjean, and Katrin Seyr, editors, *Computer Science Logic, 12th International Workshop, Brno, Czech Republic, August 24–28, 1998, Proceedings*, volume 1584 of *Lecture Notes in Computer Science*, pages 298–312. Springer Verlag, 1999.

[Mat99c] Ralph Matthes. Tarski's fixed-point theorem and lambda calculi with monotone inductive types. To appear in: Benedikt Löwe and Florian Rudolph, Foundations of the Formal Sciences, Refereed Papers of a Research Colloquium, Humboldt-Universität zu Berlin, May 7-9, 1999.

[SU99] Zdzisław Spławski and Paweł Urzyczyn. Type Fixpoints: Iteration vs. Recursion. *SIGPLAN Notices*, 34(9):102–113, 1999. Proceedings of the 1999 International Conference on Functional Programming (ICFP), Paris, France.