Non-Strictly Positive Fixed-Points for Classical Natural Deduction

Ralph Matthes

Institut für Informatik der Universität München
Oettingenstraße 67, D-80687 München, Germany

Abstract

Termination for classical natural deduction is difficult in the presence of commuting/permutative conversions for disjunction. An approach based on reducibility candidates is presented that uses non-strictly positive inductive definitions.

It covers second-order universal quantification and also the extension of the logic by fixed-points of non-strictly positive operators, which appears to be a new result.

Finally, the relation to Parigot's strictly-positive inductive definition of his set of reducibility candidates and to his notion of generalized reducibility candidates is explained.

Key words: proof by contradiction, natural deduction, second-order logic, system F, disjunction, commuting/permutative conversion, fixed-point type, strong normalization, reducibility candidate, saturated set, $\lambda\mu$-calculus

For Helmut Schwichtenberg on the occasion of his sixtieth birthday.

Helmut Schwichtenberg has been my academic teacher and doctoral advisor who initiated my fascination for each of the three major aspects of the present article: Through his lectures on type theory [1] he opened my mind for system F, through his work with Jaco van de Pol [2] and his lecture notes on proof theory [3] he challenged me to conquer permutative conversions as well, and his account of control operators and classical logic [4] convinced me that an intuitionist may delve into classical logic without sacrificing his conviction.

Email address: `matthes@informatik.uni-muenchen.de` (Ralph Matthes).
URL: `http://www.tcs.informatik.uni-muenchen.de/~matthes/` (Ralph Matthes).

1 Introduction and Overview

Natural deduction is intuitionistic in nature. Classical logic can be recovered by adding, e. g., the law of excluded middle. A fertile field of study is the extension by a rule corresponding to proof by contradiction. Natural deduction can be formulated as a lambda-calculus with types. The classical feature gives rise to interesting additional reduction rules that correspond to proof transformations. A central question is whether these reductions always terminate for typable terms, in other words, for the computational counterpart of the classical proofs. For many formulations and extensions, this can be answered in the affirmative. Although many proofs based on various principles such as the semantical method of candidates of reducibility or the proof-theoretic method of a Kolmogorov translation (corresponding to a translation in continuation-passing style) exist (for the second kind, see [5,6]), the author claims that there is no proof that combines second-order classical logic (with proof by contradiction) in the most natural formulation and disjunction with permutative/commuting conversions. Termination for such a calculus cannot yield the subformula property due to the second-order quantification. But it does provide logical consistency. Moreover, one can think of the calculus as an untyped programming language with classical features. Datatypes are present by second-order encodings. Termination is independent of any reduction strategy, and permutative conversions simplify case analyses also in call-by-name evaluation.

Two proofs of strong normalization are given and carefully compared. Both use logical predicates, one is inspired by Tait [7], the other by Parigot [5]. Strong normalization is captured by an inductive definition of a set SN in the style of van Raamsdonk and Severi [8][9, p.183]. The essential new ingredient is a notion of saturated sets that relies on a clause that is positive but not strictly positive and thus far from being a Horn clause.

It is shown that the proof method also works for the addition of non-strictly positive fixed-points for which no classical system seems to have been analyzed in the literature. Thus, the title about classical natural deduction and non-strictly positive fixed-points can be understood twofold:

- Non-strictly positive fixed-points are used to get saturated sets for the interpretation of types.
- Non-strictly positive fixed-points can be added to the calculus while preserving strong normalization.

The article is organized as follows: The next section fixes the notation for our variety F^+ of second-order lambda-calculus with sums and permutative conversions. Section 3 extends it by the principle RAA, which is another name for proof by contradiction. The resulting system is called $\mathsf{F}^{+\neg}$ and its strong normalization proven by the first method in section 4. This includes the definition of SN and the proof that terms in SN are strongly normalizing. Section 5 covers the extension $\mathsf{PF}^{+\neg}$ by non-strictly positive fixed-points. The second proof method is shown in section 6, first for the intuitionistic case, then for the full system. Finally, the two methods are compared in depth, and extensions are discussed. Throughout, references to the literature are given.

Acknowledgements: I am very thankful for plenty of discussions with René David and Karim Nour during a visit of four weeks to the University of Chambéry in early 2003. Andreas Abel and Markus Latte kindly studied preliminary versions of the text and gave helpful comments for improvement. Finally, I would like to express sincere thanks to the anonymous referee.

2 Definition of System F^+

We present polymorphic lambda-calculus, extended by sums with permutative conversions, also called commuting conversions.

Types. (Denoted by uppercase letters.) Metavariable X ranges over an infinite set TV of type variables.

$$A, B, C, D ::= X \mid \bot \mid A \to B \mid A + B \mid \forall X.A$$

The occurrences of X in A are bound in $\forall X.A$. We identify types whose de Bruijn representations coincide. Let $\mathsf{FV}(A)$ be the set of free type variables in A. The type $\bot$ just serves as a constant in F^+.[1] It is used to define $\neg A := A \to \bot$. The type $A + B$ represents the disjoint sum of A and B. As usual, we also speak of $A + B$ as being the disjunction of A and B.

Terms. (Denoted by lowercase letters.) The metavariable x ranges over an infinite set of term variables.

$$r, s, t ::= x \mid \lambda x.t \mid r\, s \mid \mathsf{inj}_0\, t \mid \mathsf{inj}_1\, t \mid \mathsf{case}\,(r,\, x.\, s,\, y.\, t)$$

[1] Intuitionistic falsum is already present in the pure system: $\forall X.X$ has "ex falsum quodlibet" by trivial universal elimination. By normalization and analysis of normal forms, one shows that $\forall X.X$ is not inhabited, as defined below.

The occurrences of x in t are bound in $\lambda x.t$, and similarly those of x in s and of y in t in the term $\mathsf{case}\,(r,\, x.\, s,\, y.\, t)$. We identify terms with the same de Bruijn representation. (Capture-avoiding) Substitution of a term s for a variable x in term r is denoted by $r[x := s]$. Analogously, we define type substitution $A[X := B]$. We think of terms as parse trees according to the above grammar rules and insert parentheses in case of ambiguities, especially around the application $r\,s$. But we will omit them as much as possible and assume that application associates to the left. Hence, we write $rs_1 \ldots s_n$ for the parenthesized term $(\ldots(rs_1)\ldots s_n)$.

Contexts. They are finite sets of pairs $(x : A)$ of term variables and types and will be denoted by Γ. Term variables in a context Γ are assumed to be distinct, and the notation $\Gamma, x : A$ shall indicate $\Gamma \cup \{(x : A)\}$ where x does not occur in Γ. Write $\mathsf{FV}(\Gamma) := \bigcup_{(x:A)\in\Gamma} \mathsf{FV}(A)$.

Well-typed terms. $\Gamma \vdash t : A$ is inductively defined by

$$\frac{(x{:}A) \in \Gamma}{\Gamma \vdash x : A} \qquad \frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B} \qquad \frac{\Gamma \vdash r : A \to B \qquad \Gamma \vdash s : A}{\Gamma \vdash r\,s : B}$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall X.A}\ \text{if } X \notin \mathsf{FV}(\Gamma) \qquad \frac{\Gamma \vdash t : \forall X.A}{\Gamma \vdash t : A[X := B]}$$

$$\frac{\Gamma \vdash t : A_i}{\Gamma \vdash \mathsf{inj}_i\, t : A_0 + A_1}\ \text{for } i \in \{0,1\}$$

$$\frac{\Gamma \vdash r : A_0 + A_1 \qquad \Gamma, x_i{:}A_i \vdash t_i : C \text{ for each } i \in \{0,1\}}{\Gamma \vdash \mathsf{case}\,(r,\, x_0.\, t_0,\, x_1.\, t_1) : C}$$

A term t is *typable* if there are Γ and A such that $\Gamma \vdash t : A$. Write $\vdash t : A$ for $\emptyset \vdash t : A$. If there is a term t with $\vdash t : A$, the type A is *inhabited*. Evidently, if $\Gamma \vdash t : A$ and $\Gamma \subseteq \Gamma'$ for the context Γ', then $\Gamma' \vdash t : A$ as well, i. e., *weakening* is a derived typing rule. It is also easy to see that the following *cut* rule is a derived typing rule:

$$\frac{\Gamma, x : A \vdash r : B \qquad \Gamma \vdash s : A}{\Gamma \vdash r[x := s] : B}$$

From now on always assume $i \in \{0, 1\}$.

Eliminations. Let $\star$ be a new symbol (hence not a term variable). An elimination is an expression of one of the following shapes:

$$\star\, s \qquad \text{or} \qquad \mathsf{case}\,(\star,\, x_0.\, t_0,\, x_1.\, t_1)$$

Let e always denote an elimination, and write $e[r]$ for the term $e[\star := r]$, where the substitution treats $\star$ as if it were a term variable.[2]

Reduction. The one-step reduction relation $t \longrightarrow t'$ between terms t and t' is defined as the closure of the following axioms under all term constructors.

$$\begin{array}{lll}
(\lambda x.t)\, s & \longrightarrow_\beta & t[x := s] \\
\mathsf{case}\,(\mathsf{inj}_i\, t,\, x_0.\, t_0,\, x_1.\, t_1) & \longrightarrow_\beta & t_i[x_i := t] \\
e[\mathsf{case}\,(r,\, x_0.\, t_0,\, x_1.\, t_1)] & \longrightarrow_\pi & \mathsf{case}\,(r,\, x_0.\, e[t_0],\, x_1.\, e[t_1])
\end{array}$$

In the last clause (of permutation), we assume that x_0, x_1 are not free in e. We denote the transitive closure of $\longrightarrow$ by $\longrightarrow^+$ and the reflexive-transitive closure by $\longrightarrow^*$. It is clear that $r \longrightarrow r'$ implies $r[x := s] \longrightarrow r'[x := s]$ and $s[x := r] \longrightarrow^* s[x := r']$.

Lemma 1 (Subject Reduction) *If* $\Gamma \vdash t : A$ *and* $t \longrightarrow t'$ *then* $\Gamma \vdash t' : A$.

PROOF. This is not trivial since F^+ is formulated in Curry-style. Nevertheless, Krivine's method [10] is very well-suited to cover also the permutative conversions. We omit the lengthy but boring adaptation of Krivine's proof. Note that the proof would be obvious in the absence of the rules for the universal quantifier. E. g., for $\longrightarrow_\pi$, one would know that always t_0, t_1 and $\mathsf{case}\,(r,\, x_0.\, t_0,\, x_1.\, t_1)$ have the same type (in a given typing derivation). Therefore, if e "accepts" a term of that type to yield type A, one can as well apply e to t_0 and t_1 and form a case construct of type A again. $\square$

3 Second-Order Classical Natural Deduction with Sums

We extend F^+ by "reductio ad absurdum" (RAA) which corresponds to stability ($\neg\neg A \to A$, also called "duplex negatio affirmat"). This is done by way of a further term formation rule and by rewrite rules for the simplification of the application of elimination rules to RAA. In the case of modus ponens, they correspond to the fact that stability of $A \to B$ is derivable from that

[2] Textual substitution would suffice for eliminations since $\star$ is never below a binder.

of B. Because of this fact, Prawitz restricts RAA to atoms A with $A \neq \perp$ in his article [11, p. 242]. Our concern is precisely to understand the operational behaviour of RAA for composite types.

System $\mathsf{F}^{+\neg}$ is defined as an extension of F^{+} by adding a rule for RAA. In the term syntax, this is

$$r, s, t ::= \ldots \mid \mu x.r$$

As regards variable binding, $\mu x.r$ is the same as $\lambda x.r$. The choice of the letter μ for this binder is adopted from Parigot's $\lambda\mu$-calculus [12] while Rehof and Sørensen [13] use the letter Δ instead. RAA is modeled by one additional typing rule:

$$\frac{\Gamma, x : \neg A \vdash r : \perp}{\Gamma \vdash \mu x.r : A}$$

Evidently, this breaks the *introduction/elimination dichotomy* of natural deduction, since inhabitants of arbitrary composite types A can be constructed by help of $\mu x.r$, and not only by their respective introduction rules. This is another argument for Prawitz to restrict RAA to atoms A [11, p. 248].

The notion of eliminations is unchanged, and there is a new axiom scheme of μ-reductions:

$$e[\mu x.r] \longrightarrow_{\mu} \mu y.\, r[x := \lambda z.y(e[z])]$$

Here, it is assumed that y is neither free in e nor in $\mu x.r$ and that $z \neq y$ and z not free in e. The notions $\longrightarrow$ (and, consequently, $\longrightarrow^{+}$ and $\longrightarrow^{*}$) shall now refer to this extended set of axiom schemes for reduction.

Lemma 2 *Subject reduction (lemma 1) also holds for* $\mathsf{F}^{+\neg}$.

PROOF. As before, the proof is too unsurprising to be reported here. Again, we show the most important case in the absence of universal quantification. In a given typing derivation of $e[\mu x.r]$, the term r gets type $\perp$ in a context containing $x : \neg A$. Therefore, in e, the symbol $\star$ gets replaced by terms of type A in order to yield terms of type B. Under these circumstances, the right-hand side $\mu y.r[x := \lambda z.y(e[z])]$ of the μ-reduction rule can also be given the type B: The variable y has to be typed with $\neg B$, and x has to be replaced by $\lambda z.y(e[z])$ of the same type, i. e., by $\neg A$. Hence, z has to have type A and $y(e[z])$ type $\perp$ which clearly works well. $\square$

For $e = \star s$, the μ-reduction rule is a term notation for one of Prawitz' transformations [14, p. 40], with the difference that his reduction is in a sense η-expanded and would read as $\mu x.r \longrightarrow_\mu \lambda u.\mu y.r[x := \lambda z.y(zu)]$. In his book, Prawitz deliberately leaves out disjunction in his discussion of classical logic, also for second order [14, p. 69]. Although stability of $A+B$ cannot be reduced to that of A and B, the μ-reduction rule with e a case analysis does no harm to strong normalization. This rule "reduces" stability of $A + B$ to stability of arbitrary types C which is certainly no well-founded process. Therefore, there is no hope that Joly's embedding [15] of classical natural deduction into system F can be extended to sums. But one might argue that one could get sums just by an impredicative encoding. However, one would not get permutative conversions for them. Moreover, section 5 will show that we can also overcome the problem of "reduction" to stability for more complex types for fixed-point types where permutative conversions are of no concern and where an impredicative encoding certainly does not yield acceptable reduction behaviour. Thus, it seems unavoidable to depart from the view that μ-reduction is reduction of stability to "smaller" types.

Note that Stålmarck defines another μ-reduction rule for disjunction [16, p. 132], which in our notation would read

$$\mathsf{case}\,(\mu x.r,\ x_0.\,t_0,\ x_1.\,t_1) \longrightarrow_\mu \mu y.r[x := \lambda z.\mathsf{case}\,(z,\ x_0.\,yt_0,\ x_1.\,yt_1)].$$

The variable y moved from outside the case construct into the two cases, thereby assuring that for typed terms, the case construct in the right-hand side is typed with $\perp$. In that article, strong normalization is shown with a restriction [16, p. 136] of the permutative conversion rule to normal terms r in the reducible expression $e[\mathsf{case}\,(r,\ x_0.\,t_0,\ x_1.\,t_1)]$.

Rehof and Sørensen [13] have precisely our μ-reduction rules, but they prove strong normalization only for a first-order system without disjunction. [3]

Ritter, Pym and Wallen [17] present a radically different formulation of disjunction that uses an additional binder ν and is based on the classical equivalence of $\neg B \to A$ and $A + B$. Its purpose is the analysis of multiple-conclusioned sequent calculus for first-order propositional logic.

Lemma 3 (Local Confluence) $\mathsf{F}^{+\neg}$ *is locally confluent, i. e., if* $r \longrightarrow r_1, r_2$ *then there is a term* t *such that* $r_1, r_2 \longrightarrow^* t$.

PROOF. The critical pairs only come from permutative conversions. Over-

[3] Their *defined* sums do not simulate the μ-reduction rule for disjunction. Addition of the rule $\mu x^{\neg\perp}.r \longrightarrow_\mu r[x := \lambda y^{\perp}.y]$ which only makes sense with Church-style syntax, would justify Stålmarck's rule in the sense that the defined left-hand side and the defined right-hand side of that rule would reduce to a common term.

lappings of rules occur with three kinds of terms:

- $e[\mathsf{case}\,(\mu x.r,\ x_0.\,t_0,\ x_1.\,t_1)]$ overlaps $\longrightarrow_\pi$ with $\longrightarrow_\mu$,
- $e[\mathsf{case}\,(\mathsf{inj}_i\,t,\ x_0.\,t_0,\ x_1.\,t_1)]$ overlaps $\longrightarrow_\pi$ with $\longrightarrow_\beta$,
- $e[\mathsf{case}\,(\mathsf{case}\,(r,\ y_0.\,s_0,\ y_1.\,s_1),\ x_0.\,t_0,\ x_1.\,t_1)]$ overlaps $\longrightarrow_\pi$ with itself.

It is easy to see that the corresponding critical pairs are joinable. $\square$

Remark 4 *We will show that $\mathsf{F}^{+\neg}$ is strongly normalizing, see below, hence Newman's lemma gives confluence for typable terms. We conjecture that every term is confluent, to be provable by the method of complete developments by Takahashi [18]. For a variant of λμ-calculus, this has been done by Andou [19].*

Remark 5 *In classical natural deduction in Curry-style, the simplification rules of Parigot [5] would be*

$$\begin{aligned} &x(\mu y.r) \longrightarrow_\mu r[y := x] \\ &\mu x.xr \quad\ \longrightarrow_\mu r \quad \textit{for } x \textit{ not free in } r \end{aligned}$$

While the first rule does not make sense in this general format (x does not just receive a type $\neg A$), we could very well include the second one in our system. It is a special case of removal of "redundant applications" of RAA in [11, p. 249], written in term notation. We do not include it just for sake of simplicity.

4 Strong Normalization of System $\mathsf{F}^{+\neg}$

Every typable term in $\mathsf{F}^{+\neg}$ is shown to be strongly normalizing by a variant of Tait's saturated sets approach [7] that can also handle $\longrightarrow_\pi$ and $\longrightarrow_\mu$.

Strongly Normalizing Terms. A term t is strongly normalizing iff there is no infinite reduction sequence $t \longrightarrow t_1 \longrightarrow t_2 \longrightarrow \ldots$ Equivalently, the set sn of strongly normalizing terms can be defined inductively by: If for all t' with $t \longrightarrow t'$ one has $t' \in \mathsf{sn}$, then also $t \in \mathsf{sn}$.

We aim at a syntax-directed characterization of typable strongly normalizing terms in the spirit of van Raamsdonk and Severi [8][9, p.183] that modularizes the normalization proof. The proposed solution extends the work on permutative conversions in [20] to classical natural deduction.

Multiple Eliminations. Inductively define multiple eliminations as a set of expressions as follows:

$$\frac{}{\star \text{ is a multiple elim.}} \qquad \frac{e \text{ is an elimination} \quad E \text{ is a multiple elim.}}{e[\star := E] \text{ is a multiple elim.}}$$

The substitution shall again treat $\star$ as a term variable.[4] Every multiple elimination has exactly one occurrence of $\star$. We always denote multiple eliminations by the letter E (and decorated forms of E), and write $e[E]$ for $e[\star := E]$ and $E[r]$ for $E[\star := r]$. Evidently, $e[E[r]] = (e[E])[r]$, and every $E \neq \star$ is of the form $E'[e]$. The *length* of $\star$ is defined to be 0, the length of $e[E]$ is set to the length of E plus 1. Trivially, $r \longrightarrow r'$ implies $E[r] \longrightarrow E[r']$.

We also need special forms of (multiple) eliminations: Let always f denote an elimination of the form $\star s$, and define *special multiple eliminations* as multiple eliminations where in the inductive clause, always an f instead of an e is used. Using always the letter F (or a decoration of F) for such a term, we have that F is either $\star$ or of the form $f[F']$. Therefore, every F has the form $\star s_1 \dots s_n$ with terms $s_1, \dots, s_n$.[5]

Lemma 6 *Every typable term is of exactly one of the following forms:*

- $E[x]$
- $\lambda x.t, \mathsf{inj}_i\, t, \mu x.r$
- $E[(\lambda x.t)s], E[\mathsf{case}\,(\mathsf{inj}_i\, t,\ x_0.\, t_0,\ x_1.\, t_1)], E[e[\mu x.r]]$

The form $E[x]$ is further divided into the mutually exclusive

- $x, F[x]s$ *(these two can be represented as $F[x]$)*
- $\mathsf{case}\,(F[x],\ x_0.\, t_0,\ x_1.\, t_1)$
- $E\Big[e[\mathsf{case}\,(F[x],\ x_0.\, t_0,\ x_1.\, t_1)]\Big]$

PROOF. Existence of the decomposition is proven by induction on terms, uniqueness is obvious. $\square$

Remark 7 *The lemma is certainly not the only way to analyze terms. David and Nour [21] put all the terms r with a permutative redex visible in a decomposition $E[t]$ of r into one category. With our notation, this can be described as follows: Let F_+ always denote a multiple elimination which is a special multiple elimination or of the form* $\mathsf{case}\,(F,\ x_0.\, t_0,\ x_1.\, t_1)$. *(David and Nour would*

[4] Again, textual substitution would suffice.

[5] The more abstract definition of F will be used in the extension by fixed-points.

call those F_+ nice.) Then every typable term is of exactly one of the following forms:

- $F_+[x]$
- $\lambda x.t, \mathsf{inj}_i\, t, \mu x.r$
- $F_+[(\lambda x.t)s], \mathsf{case}\,(\mathsf{inj}_i\, t,\ x_0.\, t_0,\ x_1.\, t_1)$
- $F_+[\mu x.r]$ *with* $F_+ \neq \star$
- $F_+[\mathsf{case}\,(r,\ x_0.\, t_0,\ x_1.\, t_1)]$ *with* $F_+ \neq \star$

One can show that the totality of those terms yields exactly those in the lemma above. For this, it suffices to show that the last form decomposes exactly into the following possibilities:

- $E\Big[e[\mathsf{case}\,(F[x],\ x_0.\, t_0,\ x_1.\, t_1)]\Big]$
- $E[(\lambda x.t)s]$ *with* $E \neq F_+$
- $E[\mathsf{case}\,(\mathsf{inj}_i\, t,\ x_0.\, t_0,\ x_1.\, t_1)]$ *with* $E \neq \star$
- $E[\mu x.r]$ *with* $E \neq F_+$

The following definition of the set SN of terms will be directed by the syntax analysis of the previous lemma: Each of the ten forms can only be derived by one of the rules, and the rule has a general representative of this form in the conclusion. The set SN will contain every strongly normalizing term we are interested in. This is even true in a very strong sense: Apart from the proof of lemma 10 where we show strong normalization for the terms in SN, we will never use any closure property of strongly normalizing terms other than the defining rules of SN. This also excludes the use of derived closure rules of SN. In other words, we do not (except for remarks) even use that SN is the least set closed under its defining rules.

Definition 8 *Inductively define the set* SN *of terms as follows:*

$$\frac{}{x \in \mathsf{SN}} \qquad \frac{F[x], s \in \mathsf{SN}}{F[x]s \in \mathsf{SN}} \qquad \frac{F[x], t_0, t_1 \in \mathsf{SN}}{\mathsf{case}\,(F[x],\ x_0.\, t_0,\ x_1.\, t_1) \in \mathsf{SN}}$$

$$\frac{E\Big[\mathsf{case}\,(F[x],\ x_0.\, e[t_0],\ x_1.\, e[t_1])\Big] \in \mathsf{SN}}{E\Big[e[\mathsf{case}\,(F[x],\ x_0.\, t_0,\ x_1.\, t_1)]\Big] \in \mathsf{SN}}$$

$$\frac{t \in \mathsf{SN}}{\lambda x.t \in \mathsf{SN}} \qquad \frac{t \in \mathsf{SN}}{\mathsf{inj}_i\, t \in \mathsf{SN}} \qquad \frac{r \in \mathsf{SN}}{\mu x.r \in \mathsf{SN}}$$

$$\frac{E[t[x := s]], s \in \mathsf{SN}}{E[(\lambda x.t)s] \in \mathsf{SN}} \qquad \frac{E[t_i[x_i := t]], E[t_{1-i}], t \in \mathsf{SN}}{E[\mathsf{case}\,(\mathsf{inj}_i\, t,\ x_0.\, t_0,\ x_1.\, t_1)] \in \mathsf{SN}}$$

$$\frac{E\Big[\mu y.r[x := \lambda z.y(e[z])]\Big] \in \mathsf{SN} \qquad E[e] \in \mathsf{SN}}{E[e[\mu x.r]] \in \mathsf{SN}}$$

For the last rule, we define $E' \in \mathsf{SN}$ as an abbreviation for "$E'[x] \in \mathsf{SN}$ for some variable x".

Note that the first two rules just say that the term $xs_1 \dots s_n$ is in SN iff the terms $s_1, \dots, s_n$ are in SN. The rule for deriving $E[(\lambda x.t)s] \in \mathsf{SN}$ has been an essential ingredient of proofs of strong normalization since Tait's article [7]. The rules embody a weak head reduction strategy, the side hypotheses of the last three rules ensure even $\mathsf{SN} \subseteq \mathsf{sn}$. Note also that the rule in the second line is the only one where the presence of permutative rules is visible, and it only deals with terms of the form $E[x]$.

Remark 9 *The set of multiple eliminations E with $E \in \mathsf{SN}$ is characterized by the set $\mathsf{SN}_\star$, inductively defined as follows:*

$$\frac{}{\star \in \mathsf{SN}_\star} \qquad \frac{F \in \mathsf{SN}_\star \qquad s \in \mathsf{SN}}{Fs \in \mathsf{SN}_\star} \qquad \frac{F \in \mathsf{SN}_\star \qquad t_0, t_1 \in \mathsf{SN}}{\mathsf{case}\,(F,\, x_0.\, t_0,\, x_1.\, t_1) \in \mathsf{SN}_\star}$$

$$\frac{E\Big[\mathsf{case}\,(F,\, x_0.\, e[t_0],\, x_1.\, e[t_1])\Big] \in \mathsf{SN}_\star}{E\Big[e[\mathsf{case}\,(F,\, x_0.\, t_0,\, x_1.\, t_1)]\Big] \in \mathsf{SN}_\star}$$

PROOF. Obviously, the rules are correct. Completeness is shown by induction on $E[x] \in \mathsf{SN}$. $\Box$

Lemma 10 $\mathsf{SN} \subseteq \mathsf{sn}$, *i. e., the rules of* SN *are sound in the sense that they are closure properties of the set* sn *of strongly normalizing terms.*

PROOF. This is nearly the same as in [20, p.73]. We only consider the essentially new last rule: Let $E[e] \in \mathsf{SN}$ due to $E[e[x']] \in \mathsf{SN}$ for some variable x'. Prove that $E[e[\mu x.r]] \in \mathsf{sn}$ by main induction on the length of E, side induction on $E[e[x']] \in \mathsf{sn}$ and a second side induction on $E\Big[\mu y.r[x := \lambda z.y(e[z])]\Big] \in \mathsf{sn}$. Assume $E[e[\mu x.r]] \longrightarrow t$ and show that $t \in \mathsf{sn}$. If t is derived by $\longrightarrow_\mu$, applied to $e[\mu x.r]$, this is trivial by hypothesis. If t stems from reduction in r, e or E, this is covered by the induction hypotheses. The only other case is with $e = \mathsf{case}\,(\star,\, x_0.\, t_0,\, x_1.\, t_1)$ and $E = E'[e']$ and $t = E'\Big[\mathsf{case}\,(\mu x.r,\, x_0.\, e'[t_0],\, x_1.\, e'[t_1])\Big]$ (by a permutative conversion). Setting $e'' := \mathsf{case}\,(\star,\, x_0.\, e'[t_0],\, x_1.\, e'[t_1])$, this becomes $t = E'[e''[\mu x.r]]$. Then,

$$E'\Big[e'\Big[\mu y.r[x := \lambda z.y(e[z])]\Big]\Big] \longrightarrow E'\Big[\mu y_1.r[x := \lambda z.y(e[z])][y := \lambda z_1.y_1(e'[z_1])]\Big].$$

By finitely many reductions (the treatment of the so-called administrative redices), one arrives at $E'\Big[\mu y_1.r[x := \lambda z.y_1(e'[e[z]])]\Big]$. Again finitely many permutative reduction steps (since the reduction takes place in the term we substitute for x) lead to $E'\Big[\mu y_1.r[x := \lambda z.y_1(e''[z])]\Big]$, which is in sn since we started with a strongly normalizing term (by hypothesis). Clearly, $E[e[x']] = E'[e'[e[x']]] \longrightarrow E'[e''[x']]$, hence this latter term is also in sn. Therefore, we can apply the main induction hypothesis with E' instead of E and e'' instead of e and get $t \in \mathsf{sn}$. $\square$

Remark 11 *The soundness proof for the last rule of* SN *would not work if we replaced the second premise by* $e[x'] \in \mathsf{SN}$ *for some variable* x'. *This proposal intuitively looks reasonable since the only part of the conclusion that does not necessarily appear in the first premise is* e *(in case* x *does not occur free in* r*). But in the proof above, we would have no way to argue that* $e''[x'] \in \mathsf{SN}$ *because this would require* $e'[t_i] \in \mathsf{SN}$ *with* e' *a constituent of* E *and* t_i *subterms of* e. *Moreover, it is not only the proof that would not work: Soundness fails for* $r := e'[e[\mu x.z]]$ *with* $e := \mathsf{case}\,(\star,\ x_0.\,\lambda y.yy,\ x_1.\,x_1)$ *and* $e' := \star(\lambda y.yy)$. *The modified* SN *would contain* r, *but* $r \longrightarrow_\pi \mathsf{case}\,(\mu x.z,\ x_0.\,(\lambda y.yy)(\lambda y.yy),\ x_1.\,x_1(\lambda y.yy))$.

Remark 12 *Every typable and strongly normalizing term is in* SN.

PROOF. This can be proven as in [20, p.74]. $\square$

Remark 13 $\mathsf{SN} \neq \mathsf{sn}$.

PROOF. There are untypable terms in sn which are not in SN, e. g., the term $(\mathsf{inj}_0\, x)y$. (In [22], a more general definition of SN is given which captures this kind of "junk terms".) $\square$

The proof will now follow the lines of usual proofs by the candidate method by Girard [23]. The candidates will be subsets of SN with additional closure properties, modelled after the definition of SN. However, for our notion of saturated sets, in order to treat $\mu x.r$, we use a closure property that is not a Horn clause but amounts to a non-strictly positive operator.

The notion of saturated sets will allow to model the types of $\mathsf{F}^{+\neg}$. The respective constructions for $\rightarrow$ and $+$ are studied beforehand, and their soundness for natural deduction is proven.

Remark 14 *Immediately preceding the definition of* SN, *a promise was made not to use derived closure rules of* SN. *For proofs not involving saturated sets or other logical predicates such as candidates of reducibility or notions of validity or strong computability, it seems unavoidable to use some kind of permutative* *expansion lemma. E. g., in [20, p. 74], one needs (in our notation) the derived rule*

$$\frac{E[\mathsf{case}\,(r,\,x_0.\,e[t_0],\,x_1.\,e[t_1])] \in \mathsf{SN}}{E[e[\mathsf{case}\,(r,\,x_0.\,t_0,\,x_1.\,t_1)]] \in \mathsf{SN}}$$

The definition only decrees it for $r = F[x]$. In the presence of RAA, it seems very hard to prove this rule. In the proofs by David and Nour [21] and by Nour and Saber [24] for $\lambda\mu$-calculus, a slightly simpler closure rule of the set sn *of strongly normalizing terms plays an essential role (for F_+, see remark 7):*

$$\frac{\mathsf{case}\,(r,\,x_0.\,F_+[t_0],\,x_1.\,F_+[t_1]) \in \mathsf{sn}}{F_+[\mathsf{case}\,(r,\,x_0.\,t_0,\,x_1.\,t_1)] \in \mathsf{sn}}$$

This is already difficult for $F_+ = \star s$ and $r = \mu x.t$ and has a very technical proof in [21] (for $\lambda\mu$-calculus). The notion of saturated sets to come is complex but avoids these problems.

Definition 15 *A set $\mathcal{M}$ of terms is* saturated *iff $\mathcal{M} \subseteq \mathsf{SN}$ and*

$$\frac{F[x] \in \mathsf{SN}}{F[x] \in \mathcal{M}} \qquad \frac{F[x] \in \mathsf{SN} \quad t_0, t_1 \in \mathcal{M}}{\mathsf{case}\,(F[x],\,x_0.\,t_0,\,x_1.\,t_1) \in \mathcal{M}}$$

$$\frac{E\Big[\mathsf{case}\,(F[x],\,x_0.\,e[t_0],\,x_1.\,e[t_1])\Big] \in \mathcal{M}}{E\Big[e[\mathsf{case}\,(F[x],\,x_0.\,t_0,\,x_1.\,t_1)]\Big] \in \mathcal{M}}$$

$$\frac{E[t[x := s]] \in \mathcal{M} \quad s \in \mathsf{SN}}{E[(\lambda x.t)s] \in \mathcal{M}} \qquad \frac{E[t_i[x_i := t]], E[t_{1-i}] \in \mathcal{M} \quad t \in \mathsf{SN}}{E[\mathsf{case}\,(\mathsf{inj}_i\,t,\,x_0.\,t_0,\,x_1.\,t_1)] \in \mathcal{M}}$$

$$\frac{E\Big[\mu y.r[x := \lambda z.y(e[z])]\Big], E[e[x']] \in \mathcal{M} \text{ for some variable } x'}{E[e[\mu x.r]] \in \mathcal{M}}$$

$$\frac{r \in \mathsf{SN} \quad \forall z \forall s.\,(\forall t \in \mathcal{M}.s[z := t] \in \mathsf{SN}) \Rightarrow r[x := \lambda z.s] \in \mathsf{SN}}{\mu x.r \in \mathcal{M}}$$

The very last closure condition forms the core idea of the whole article. In the premise, $\mathcal{M}$ occurs non-strictly positively: Written out in even more detail, its most important part reads

$$\Big(\forall t.t \in \mathcal{M} \Rightarrow s[z := t] \in \mathsf{SN}\Big) \Rightarrow r[x := \lambda z.s] \in \mathsf{SN}.$$

Here, $\mathcal{M}$ occurs twice to the left of the implication $\Rightarrow$. The main justification of this rule will be lemma 22 which in turn allows to carry out the normalization proof where the type $\bot$ is interpreted by the largest saturated set, namely SN (see the next lemma).

It is also important to notice that the rule requires a special treatment of $\mu x.r$ already in the notion of saturated sets and not just in a later concrete definition of some saturated set in definition 23. Clearly, this again reflects the breaking of the introduction/elimination dichotomy, mentioned immediately after the typing rule for $\mu x.r$. In section 6, the present approach is compared with that of Parigot [5] who does not mention RAA at all in his definitions of candidates.

Let SAT be the set of all saturated sets, and assume that always saturated sets are denoted by $\mathcal{M}$ and $\mathcal{N}$ (possibly with indices or other decoration).

Lemma 16 $\mathsf{SN} \in \mathsf{SAT}$ *and if* $U \subseteq \mathsf{SAT}$*, then* $\bigcap U \in \mathsf{SAT}$ *(we set* $\bigcap \emptyset := \mathsf{SN}$*).*

PROOF. For $\mathsf{SN} \in \mathsf{SAT}$, the conditions with exception of the last one are just part of the definition of SN, the last one is even fulfilled without use of the second premise. For the closure under intersection, note that the conditions on saturatedness with exception of $\mathcal{M} \subseteq \mathsf{SN}$ constitute a monotone operator. $\square$

Corollary 17 *For every set M of terms, there exists the least saturated superset of $M \cap \mathsf{SN}$, denoted as $\mathsf{clo}(M)$.*

PROOF. $\mathsf{clo}(M) := \bigcap\{\mathcal{M} \mid M \cap \mathsf{SN} \subseteq \mathcal{M}\}$. A slightly more concrete definition of $\mathsf{clo}(M)$ is the set generated inductively from $M \cap \mathsf{SN}$ by the closure conditions of saturatedness—with exception of $\mathcal{M} \subseteq \mathsf{SN}$. $\square$

Remark 18 *Inspection of the rules shows that $\lambda x.r \in \mathsf{clo}(M)$ implies that $\lambda x.r \in M$: There is no rule that forces lambda abstractions into saturated sets. Otherwise, one would later have lambda abstractions in the interpretation of sum types!*

Remark 19 *Notice that we do not suppose that $M \subseteq \mathsf{SN}$. We can only ensure $\mathsf{clo}(M) \supseteq M$ if this is indeed the case.*

Constructions on Saturated Sets. For every variable x set $\mathsf{S}_x(\mathcal{M}, \mathcal{N}) := \{t \mid \forall s \in \mathcal{M}. t[x := s] \in \mathcal{N}\}$. Since $x \in \mathcal{M}$ we have $\mathsf{S}_x(\mathcal{M}, \mathcal{N}) \subseteq \mathcal{N} \subseteq \mathsf{SN}$. With this definition, the last defining rule of saturated sets can more elegantly be expressed by

$$\frac{r \in \mathsf{SN} \quad \forall z \forall s.\, s \in \mathsf{S}_z(\mathcal{M}, \mathsf{SN}) \Rightarrow r[x := \lambda z.s] \in \mathsf{SN}}{\mu x.r \in \mathcal{M}}$$

Define

$$\mathcal{M} \to \mathcal{N} := \mathsf{clo}(\{\lambda x.t \mid t \in \mathsf{S}_x(\mathcal{M},\mathcal{N})\})$$
$$\mathcal{M}_0 + \mathcal{M}_1 := \mathsf{clo}(\{\mathsf{inj}_0\, t \mid t \in \mathcal{M}_0\} \cup \{\mathsf{inj}_1\, t \mid t \in \mathcal{M}_1\})$$

Remark 20 $\mathsf{S}_x(\mathcal{M},\mathcal{N})$ *need not be saturated: For* $\mathcal{M} := \mathsf{clo}(\{\lambda y.yy\}) \ni \lambda y.yy$ *and* $\mathcal{N} := \mathsf{SN}$*, one has* $x(\lambda y.yy) \notin \mathsf{S}_x(\mathcal{M},\mathcal{N})$*. Otherwise,*

$$x(\lambda y.yy)[x := \lambda y.yy] = (\lambda y.yy)(\lambda y.yy) \in \mathcal{N} = \mathsf{SN},$$

which is not the case since this term is not (even weakly) normalizing.

Lemma 21
$$\frac{t \in \mathsf{S}_x(\mathcal{M},\mathcal{N})}{\lambda x.t \in \mathcal{M} \to \mathcal{N}} \qquad \frac{r \in \mathcal{M} \to \mathcal{N} \quad s \in \mathcal{M}}{rs \in \mathcal{N}}$$

$$\frac{t \in \mathcal{M}_i}{\mathsf{inj}_i\, t \in \mathcal{M}_0 + \mathcal{M}_1} \qquad \frac{r \in \mathcal{M}_0 + \mathcal{M}_1 \quad t_0 \in \mathsf{S}_{x_0}(\mathcal{M}_0,\mathcal{N}) \quad t_1 \in \mathsf{S}_{x_1}(\mathcal{M}_1,\mathcal{N})}{\mathsf{case}\,(r,\, x_0.\, t_0,\, x_1.\, t_1) \in \mathcal{N}}$$

PROOF. For the two rules about $\mathcal{M} \to \mathcal{N}$, let $I := \{\lambda x.t \mid t \in \mathsf{S}_x(\mathcal{M},\mathcal{N})\}$ and $\mathcal{E} := \{r \in \mathsf{SN} \mid \forall s \in \mathcal{M}.rs \in \mathcal{N}\}$. Evidently, $I \subseteq \mathsf{SN}$, hence $\mathcal{M} \to \mathcal{N} \supseteq I$ (see remark 19), which yields the first rule. For the second rule, we show that $I \subseteq \mathcal{E}$ and that $\mathcal{E} \in \mathsf{SAT}$, hence that $\mathcal{M} \to \mathcal{N} \subseteq \mathcal{E}$ by minimality.[6] $I \subseteq \mathcal{E}$ comes from $\mathcal{M} \subseteq \mathsf{SN}$ and saturatedness of $\mathcal{N}$ (the respective rule is used with $E := \star$). For $\mathcal{E} \in \mathsf{SAT}$, we have to show all of the closure conditions of saturatedness:

- For $F[x] \in \mathsf{SN}$ show $F[x] \in \mathcal{E}$. Let $s \in \mathcal{M} \subseteq \mathsf{SN}$. Show that $F[x]s \in \mathcal{N}$. This is clear from $F[x]s = (Fs)[x]$, $F[x]s \in \mathsf{SN}$ and Fs being a special multiple elimination.
- If $F[x] \in \mathsf{SN}$ and $t_0, t_1 \in \mathcal{E}$ then show $\mathsf{case}\,(F[x],\, x_0.\, t_0,\, x_1.\, t_1) \in \mathcal{E}$. The term is in SN by definition since $\mathcal{E} \subseteq \mathsf{SN}$. Let $s \in \mathcal{M}$ and set $e := \star s$. Show $e[\mathsf{case}\,(F[x],\, x_0.\, t_0,\, x_1.\, t_1)] \in \mathcal{N}$. Since $\mathcal{N} \in \mathsf{SAT}$, it suffices to show $\mathsf{case}\,(F[x],\, x_0.\, e[t_0],\, x_1.\, e[t_1]) \in \mathcal{N}$. By definition of $\mathcal{E}$, $e[t_0], e[t_1] \in \mathcal{N}$, hence we are done. Notice that we used a permutation of an application into the case construct.
- The four rules where $E[\ldots] \in \mathcal{E}$ is derived: These rules are all closure properties of SN, hence we only have to show that the term, applied to $s \in \mathcal{M}$, is in $\mathcal{N}$. Set $e' := \star s$ for such an s. One just uses the same rule of saturatedness of $\mathcal{N}$ with $e'[E]$ instead of E.

[6] $\mathcal{M} \to \mathcal{N}$ and $\mathcal{E}$ need not be equal: Set $\mathcal{M} := \mathsf{SN}$ and $\mathcal{N} := \mathsf{clo}(J)$ with $J := \{(\lambda x.x)s \mid s \in \mathsf{SN}\} \subseteq \mathsf{SN}$, hence $\mathcal{N} \supseteq J$. Therefore, by definition, $\lambda x.x \in \mathcal{E}$. If $\lambda x.x$ were in $\mathcal{M} \to \mathcal{N}$, then by remark 18, $\lambda x.x \in I$, i. e., $x \in \mathsf{S}_x(\mathcal{M},\mathcal{N})$. Especially for $\lambda y.y \in \mathcal{M}$, we would have $\lambda y.y \in \mathcal{N}$. Again by that remark, $\lambda y.y \in J$, which is not the case.

- The final rule for $\mu x.r$ will be treated jointly with $\mathcal{M}_0 + \mathcal{M}_1$, see below.

For the rules concerning $\mathcal{M}_0+\mathcal{M}_1$, let $I := \{\mathsf{inj}_0\, t \mid t \in \mathcal{M}_0\} \cup \{\mathsf{inj}_1\, t \mid t \in \mathcal{M}_1\}$ and

$$\mathcal{E} := \{r \in \mathsf{SN} \mid \forall \mathcal{N} \forall x_0 \forall t_0 \in \mathsf{S}_{x_0}(\mathcal{M}_0, \mathcal{N}) \forall x_1 \forall t_1 \in \mathsf{S}_{x_1}(\mathcal{M}_1, \mathcal{N}).\ \mathsf{case}\,(r,\ x_0.\, t_0,\ x_1.\, t_1) \in \mathcal{N}\}.$$

Obviously, $I \subseteq \mathsf{SN}$, hence $\mathcal{M}_0 + \mathcal{M}_1 \supseteq I$, i. e., the third rule. Now, we show $I \subseteq \mathcal{E}$ and $\mathcal{E} \in \mathsf{SAT}$, hence $\mathcal{M}_0 + \mathcal{M}_1 \subseteq \mathcal{E}$. For $I \subseteq \mathcal{E}$, assume $t \in \mathcal{M}_i$. Then, $\mathsf{inj}_i\, t \in \mathsf{SN}$. Assume $\mathcal{N}, x_0, t_0, x_1, t_1$ as in the definition of $\mathcal{E}$. Show $\mathsf{case}\,(\mathsf{inj}_i\, t,\ x_0.\, t_0,\ x_1.\, t_1) \in \mathcal{N}$. Since $\mathcal{N} \in \mathsf{SAT}$, it suffices to show $t_i[x_i := t], t_{1-i} \in \mathcal{N}$ and $t \in \mathsf{SN}$. This is immediate from the requirements on t_0, t_1.

For $\mathcal{E} \in \mathsf{SAT}$, we have to consider every closure condition:

- If $F[x] \in \mathsf{SN}$, then $F[x] \in \mathcal{E}$: Assume $\mathcal{N}, x_0, t_0, x_1, t_1$ as usual. Then, $t_0, t_1 \in \mathcal{N}$, hence $\mathsf{case}\,(F[x],\ x_0.\, t_0,\ x_1.\, t_1) \in \mathcal{N}$.
- If $F[x] \in \mathsf{SN}$ and $r_0, r_1 \in \mathcal{E}$ then show $\mathsf{case}\,(F[x],\ y_0.\, r_0,\ y_1.\, r_1) \in \mathcal{E}$. Assume $\mathcal{N}, x_0, t_0, x_1, t_1$ from the definition of $\mathcal{E}$, and set $e := \mathsf{case}\,(\star,\ x_0.\, t_0,\ x_1.\, t_1)$. Show $e[\mathsf{case}\,(F[x],\ y_0.\, r_0,\ y_1.\, r_1)] \in \mathcal{N}$. Since $\mathcal{N} \in \mathsf{SAT}$, if suffices to show $\mathsf{case}\,(F[x],\ y_0.\, e[r_0],\ y_1.\, e[r_1]) \in \mathcal{N}$. By definition of $\mathcal{E}$, $e[r_0], e[r_1] \in \mathcal{N}$, which concludes the proof. Notice that we used a permutation of a case construct into another one.
- The four rules with $E[\ldots]$ in the conclusion are treated with an abstract argument analogous to that for $\mathcal{M} \to \mathcal{N}$.
- The final rule is that for $\mu x.r$. We treat it in parallel with the same rule for $\mathcal{M} \to \mathcal{N}$. Assume $r \in \mathsf{SN}$ and

$$\forall z \forall s.\, (\forall t \in \mathcal{E}. s[z := t] \in \mathsf{SN}) \Rightarrow r[x := \lambda z.s] \in \mathsf{SN}.$$

  We have to show that $\mu x.r \in \mathcal{E}$. $\mu x.r \in \mathsf{SN}$ follows from the first assumption. Let $e := \star s$ for an $s \in \mathcal{M}$ for the treatment of $\mathcal{M} \to \mathcal{N}$ or $e := \mathsf{case}\,(\star,\ x_0.\, t_0,\ x_1.\, t_1)$ with $\mathcal{N}, x_0, t_0, x_1, t_1$ as usual for $\mathcal{M}_0 + \mathcal{M}_1$. We have to show $e[\mu x.r] \in \mathcal{N}$. Because $\mathcal{N}$ is saturated, we only need to show $\mu y.r[x := \lambda z.y(e[z])] \in \mathcal{N}$ and $e[x'] \in \mathsf{SN}$ for some x'. The second part is clear since $s, t_0, t_1 \in \mathsf{SN}$. For the first part, again by saturatedness of $\mathcal{N}$, it suffices to show

(1) $r[x := \lambda z.y(e[z])] \in \mathsf{SN}$ and

(2) $\forall z_1 \forall s_1 \in \mathsf{S}_{z_1}(\mathcal{N}, \mathsf{SN}).\, r\Big[x := \lambda z.y(e[z])\Big][y := \lambda z_1.s_1] \in \mathsf{SN}$.

  The first statement comes directly from our second assumption: We only need to show that for every $t \in \mathcal{E}$, we have $y(e[z])[z := t] \in \mathsf{SN}$. By our choice of e, $e[t] \in \mathcal{N} \subseteq \mathsf{SN}$, hence $y(e[t]) \in \mathsf{SN}$.
  For the second statement, assume $s_1 \in \mathsf{S}_{z_1}(\mathcal{N}, \mathsf{SN})$ ("the third assumption"). We have to show that $r[x := \lambda z.(\lambda z_1.s_1)(e[z])] \in \mathsf{SN}$. By the second assumption, it suffices to show $\forall t \in \mathcal{E}.(\lambda z_1.s_1)(e[t]) \in \mathsf{SN}$. Now, for $t \in \mathcal{E}$,

$e[t] \in \mathcal{N} \subseteq \mathsf{SN}$, hence by the third assumption $s_1[z_1 := e[t]] \in \mathsf{SN}$. We are done by definition of SN. □

The following lemma shows the intended use of the rule for $\mu x.r$ in the definition of saturated sets. It is exactly what we need in the case of the typing rule for $\mu x.r$ in the proof of lemma 25.

Lemma 22 *For every $\mathcal{M} \in \mathsf{SAT}$, $r \in \mathsf{S}_x(\mathcal{M} \to \mathsf{SN}, \mathsf{SN})$ implies $\mu x.r \in \mathcal{M}$.*

PROOF. $\mathsf{S}_x(\mathcal{M} \to \mathsf{SN}, \mathsf{SN}) \subseteq \mathsf{SN}$, hence $r \in \mathsf{SN}$. Also the second premise of the last rule of saturated sets is fulfilled: Assume $s \in \mathsf{S}_z(\mathcal{M}, \mathsf{SN})$. We have to show that $r[x := \lambda z.s] \in \mathsf{SN}$. This is trivial from the assumption and the first rule in the previous lemma which yields $\lambda z.s \in \mathcal{M} \to \mathsf{SN}$. □

The remainder of the proof of strong normalization follows the usual pattern: the types are interpreted as saturated sets, and every typable term lies in the interpretation of its type, and hence is in SN, which implies strong normalization.

Definition 23 *Finite sets of pairs $(X : \mathcal{M})$ of type variables and saturated sets will be denoted by Δ and called a* candidate assignment*. Type variables in type assignments are assumed to be distinct, and the notation $\Delta, X : \mathcal{M}$ shall indicate $\Delta \cup \{(X : \mathcal{M})\}$ and that X does not occur free in Δ. If $(X : \mathcal{M}) \in \Delta$, then $\Delta(X) := \mathcal{M}$, otherwise set $\Delta(X) := \mathsf{SN}$. By recursion on the type A define the set $\mathsf{SC}^A[\Delta] \in \mathsf{SAT}$ of* strongly computable *terms of type A w. r. t. candidate assignment Δ:*

- $\mathsf{SC}^X[\Delta] := \Delta(X)$.
- $\mathsf{SC}^\perp[\Delta] := \mathsf{SN}$.
- $\mathsf{SC}^{A \to B}[\Delta] := \mathsf{SC}^A[\Delta] \to \mathsf{SC}^B[\Delta]$.
- $\mathsf{SC}^{A+B}[\Delta] := \mathsf{SC}^A[\Delta] + \mathsf{SC}^B[\Delta]$.
- $\mathsf{SC}^{\forall X.A}[\Delta] := \bigcap_{\mathcal{M}} \mathsf{SC}^A[\Delta, X : \mathcal{M}]$.

Lemma 24 (Coincidence and Substitution) *The following properties hold:*

- *If for all $X \in \mathsf{FV}(A)$, one has $\Delta(X) = \Delta'(X)$, then $\mathsf{SC}^A[\Delta] = \mathsf{SC}^A[\Delta']$.*
- $\mathsf{SC}^{A[X:=B]}[\Delta] = \mathsf{SC}^A[\Delta, X : \mathsf{SC}^B[\Delta]]$.

PROOF. Induction on A. □

Lemma 25 *If* $x_1 : A_1, \ldots, x_n : A_n \vdash r : A$ *and* $s_i \in \mathsf{SC}^{A_i}[\Delta]$ *for* $i = 1, \ldots, n$, *then* $r[x_1 := s_1, \ldots, x_n := s_n] \in \mathsf{SC}^A[\Delta]$, *where* $r[x_1 := s_1, \ldots, x_n := s_n]$ *denotes simultaneous substitution of* s_i *for* x_i *(*$i = 1, \ldots, n$*) in* r*.*

PROOF. By induction on typing derivations: The variable rule is trivially dealt with, the rules for the universal quantifier need the previous lemma, the rules for $\to$ and $+$ come from lemma 21, the rule RAA from lemma 22. We only show that last argument: Assume that $x_1 : A_1, \ldots, x_n : A_n \vdash \mu x.r : A$ has been derived from $x_1 : A_1, \ldots, x_n : A_n, x : \neg A \vdash r : \bot$. Let $s_1, \ldots, s_n$ be as in the statement of the lemma. Abbreviate $r' := r[x_1 := s_1, \ldots, x_n := s_n]$. We have to show $\mu x.r' \in \mathsf{SC}^A[\Delta]$. Note that $\mathsf{SC}^{\neg A}[\Delta] = \mathsf{SC}^A[\Delta] \to \mathsf{SN}$. Therefore, by lemma 22, it suffices to show $r' \in \mathsf{S}_x(\mathsf{SC}^{\neg A}[\Delta], \mathsf{SN})$. To that end, let $s \in \mathsf{SC}^{\neg A}[\Delta]$. Show that $r'[x := s] \in \mathsf{SN} = \mathsf{SC}^{\bot}[\Delta]$. We are done by the induction hypothesis since $r'[x := s] = r[x_1 := s_1, \ldots, x_n := s_n, x := s]$ under the reasonable assumption that $x \notin \{x_1, \ldots, x_n\}$ and not free in any of the s_i. $\square$

Corollary 26 *System* $\mathsf{F}^{+\neg}$ *is strongly normalizing, i. e., whenever* $\Gamma \vdash r : A$, *then* $r \in \mathsf{sn}$*.*

PROOF. Use the lemma with $\Delta := \emptyset$ and $s_i := x_i \in \mathsf{SC}^{A_i}[\Delta]$. Therefore, $r \in \mathsf{SC}^A[\Delta] \subseteq \mathsf{SN} \subseteq \mathsf{sn}$. $\square$

5 Extension by Fixed-Points

We extend $\mathsf{F}^{+\neg}$ by positive fixed-points, and get the system $\mathsf{PF}^{+\neg}$: The inductive definition of the set of types is extended by the clause $\mathsf{fix}X.A$, with the proviso that X occurs only positively in A. More precisely, we define the set TP of types and for every type $A \in \mathsf{TP}$ the sets $\mathsf{TV}_+(A)$ and $\mathsf{TV}_-(A)$ of type variables that occur only positively in A or only negatively in A, respectively. Let always range p (polarity) over $\{-, +\}$ and set $-- := +$ and $-+ := -$.

- $X \in \mathsf{TP}$, $\mathsf{TV}_-(X) := \mathsf{TV} \setminus \{X\}$, $\mathsf{TV}_+(X) := \mathsf{TV}$.
- $\bot \in \mathsf{TP}$, $\mathsf{TV}_p(\bot) := \mathsf{TV}$.
- If $A, B \in \mathsf{TP}$, then $A \to B \in \mathsf{TP}$ and $\mathsf{TV}_p(A \to B) := \mathsf{TV}_{-p}(A) \cap \mathsf{TV}_p(B)$.
- If $A, B \in \mathsf{TP}$, then $A + B \in \mathsf{TP}$ and $\mathsf{TV}_p(A + B) := \mathsf{TV}_p(A) \cap \mathsf{TV}_p(B)$.
- If $A \in \mathsf{TP}$, then $\forall X.A \in \mathsf{TP}$ and $\mathsf{TV}_p(\forall X.A) := \mathsf{TV}_p(A) \cup \{X\}$.
- If $A \in \mathsf{TP}$ and $X \in \mathsf{TV}_+(A)$ then $\mathsf{fix}X.A \in \mathsf{TP}$ (only here is a positivity condition) and $\mathsf{TV}_p(\mathsf{fix}X.A) := \mathsf{TV}_p(A) \cup \{X\}$.

This definition allows interleaving of fixed-points, e. g., $\mathsf{fix}X.\mathsf{fix}Y.X + Y \in \mathsf{TP}$ since $X \in \mathsf{TV}_+(\mathsf{fix}Y.X + Y)$.[7] It also allows non-strict positivity, e. g., $\mathsf{fix}\,X.A + \neg\neg X$ for $X \notin \mathsf{FV}(A)$ since $X \in \mathsf{TV}_+(\neg\neg X)$ which comes from $X \in \mathsf{TV}_-(\neg X)$ (which in turn rests on $X \in \mathsf{TV}_+(X)$). Note that $X \notin \mathsf{FV}(A)$ implies $X \in \mathsf{TV}_-(A) \cap \mathsf{TV}_+(A)$. In general, and loosely speaking, a non-strictly positive occurrence is to the left of an even number of $\to$. A strictly-positive occurrence is never to the left of $\to$.

The term system is extended by fixed-point folding and unfolding, which yields the following term grammar for $\mathsf{PF}^{+\neg}$:

$$r, s, t ::= \ldots \mid \mathsf{in}\, t \mid \mathsf{out}\, r.$$

The definition of $\Gamma \vdash t : A$ is extended by the two clauses

$$\frac{\Gamma \vdash t : A[X := \mathsf{fix}X.A]}{\Gamma \vdash \mathsf{in}\, t : \mathsf{fix}X.A} \qquad \frac{\Gamma \vdash r : \mathsf{fix}X.A}{\Gamma \vdash \mathsf{out}\, r : A[X := \mathsf{fix}X.A]}$$

and there is a new axiom scheme of reduction

$$\mathsf{out}(\mathsf{in}\, t) \longrightarrow_{\mathsf{fix}} t$$

Moreover, permutation and μ-rule are extended in that eliminations e may now also be of the form $\mathsf{out}\,\star$. The notion of multiple elimination E shall in the sequel be based on this extended set of admissible e's.

Note that the μ-reduction rule for $e = \mathsf{out}\,\star$ corresponds for typed terms to a "reduction" of stability of $\mathsf{fix}X.A$ to that of $A[X := \mathsf{fix}X.A]$. As for the case of disjunction, the latter type can be more complex than the former which rules out the method by Joly [15].

As usual, we now assume that $\longrightarrow$ and the related notions refer to the extended set of axiom schemes.

System F with *non-interleaving* positive fixed-points essentially has been studied by Geuvers [25] under the name F_{ret}, and strong normalization has been shown by him through an embedding into Mendler's system [26]. A direct proof of strong normalization by saturated sets has been given by the author [27] under the name NPF. No embedding into system F exists [28]. One expects that this negative result also holds for any reasonable extension of the two systems by some η-rules.

[7] This is more than just nesting of fixed-points: The outer fix binds the parameter X of the inner fixed-point.

Here, we show that strong normalization also holds with permutative conversions, the reductions for classical logic and arbitrary positive fixed-points. This is done by a quite simple extension of the proof in the previous section. In a nutshell, the proof in [27] for NPF extends to the new reductions.

Let now f stand for either $\star s$ with a term s or $\mathsf{out}\,\star$. The notion of special multiple elimination F is extended accordingly. Now, the form of F can no longer be described as easily as before by $\star s_1 \dots s_n$, but we may have, e. g., $F = \mathsf{out}(\mathsf{out}(\mathsf{out}\,\star s_1)s_2 s_3)s_4$.

Certainly, all the statements have to be reinterpreted over the new syntactic categories (terms, eliminations, multiple eliminations, special multiple eliminations). Moreover, in lemma 6, we now get the two new possible outcomes

- $\mathsf{in}\,t$
- $E[\mathsf{out}(\mathsf{in}\,t)]$

The definition of the set SN has the new rules

$$\frac{F[x] \in \mathsf{SN}}{\mathsf{out}\,F[x] \in \mathsf{SN}} \qquad \frac{t \in \mathsf{SN}}{\mathsf{in}\,t \in \mathsf{SN}} \qquad \frac{E[t] \in \mathsf{SN}}{E[\mathsf{out}(\mathsf{in}\,t)] \in \mathsf{SN}}$$

Establishing lemma 10 again does not require any new argument. The definition of $\mathcal{M} \in \mathsf{SAT}$ gets the only new condition

$$\frac{E[t] \in \mathcal{M}}{E[\mathsf{out}(\mathsf{in}\,t)] \in \mathcal{M}}$$

Clearly, lemma 16 and its corollary still hold. Therefore, $\mathcal{M} \to \mathcal{N}$ and $\mathcal{M}_0 + \mathcal{M}_1$ are still defined. Moreover, lemma 21 is still true, and its proof is unchanged (since the new condition for saturated sets is uniformly treated with the other rules deriving $E[\dots] \in \mathcal{E}$). There are also no changes for lemma 22. Definition 23 first needs a fixed-point construction for saturated sets:

Definition 27 *Let Φ be a monotonically increasing mapping from SAT to SAT, i. e., if $\mathcal{M} \subseteq \mathcal{N}$ then $\Phi(\mathcal{M}) \subseteq \Phi(\mathcal{N})$. Define for $\mathcal{M} \in \mathsf{SAT}$ the set $\Psi(\mathcal{M}) \in \mathsf{SAT}$ by $\Psi(\mathcal{M}) := \mathsf{clo}(\{\mathsf{in}\,t \mid t \in \Phi(\mathcal{M})\})$. Since clo is monotonically increasing, Ψ also is. By lemma 16, SAT is a complete lattice, hence there exists the least fixed-point $\mathsf{fix}\,\Phi \in \mathsf{SAT}$ of Ψ.*

Note that the least fixed-point is just taken for definiteness. We could have taken the greatest fixed-point as well. It is just necessary to ensure that $\mathsf{fix}\,\Phi$ is monotonically increasing in Φ (with the pointwise order on functions Φ).

Lemma 28 *For* $\Phi : \mathsf{SAT} \to \mathsf{SAT}$ *monotonically increasing, one has*

$$\frac{t \in \Phi(\mathsf{fix}\,\Phi)}{\mathsf{in}\,t \in \mathsf{fix}\,\Phi} \qquad \frac{r \in \mathsf{fix}\,\Phi}{\mathsf{out}\,r \in \Phi(\mathsf{fix}\,\Phi)}$$

PROOF. For every $\mathcal{M}$ define

$$I(\mathcal{M}) := \{\mathsf{in}\,t \mid t \in \Phi(\mathcal{M})\} \quad \text{and} \quad \mathcal{E}(\mathcal{M}) := \{r \in \mathsf{SN} \mid \mathsf{out}\,r \in \Phi(\mathcal{M})\}.$$

We proceed as in the proof of lemma 21: Obviously, $I(\mathcal{M}) \subseteq \mathsf{SN}$, hence $\Psi(\mathcal{M}) \supseteq I(\mathcal{M})$. Next, show $I(\mathcal{M}) \subseteq \mathcal{E}(\mathcal{M})$: Let $t \in \Phi(\mathcal{M})$. Show $\mathsf{out}(\mathsf{in}\,t) \in \Phi(\mathcal{M})$. Since $\Phi(\mathcal{M}) \in \mathsf{SAT}$, if suffices to show $t \in \Phi(\mathcal{M})$, which is given. Finally, we have to show that $\mathcal{E}(\mathcal{M})$ is saturated. If this were done, we could infer $\Psi(\mathcal{M}) \subseteq \mathcal{E}(\mathcal{M})$ by minimality of $\Psi(\mathcal{M})$. This would yield the two rules: Assume $t \in \Phi(\mathsf{fix}\,\Phi)$. Then $\mathsf{in}\,t \in I(\mathsf{fix}\,\Phi) \subseteq \Psi(\mathsf{fix}\,\Phi) = \mathsf{fix}\,\Phi$. Assume $r \in \mathsf{fix}\,\Phi = \Psi(\mathsf{fix}\,\Phi) \subseteq \mathcal{E}(\mathsf{fix}\,\Phi)$. Then, $\mathsf{out}\,r \in \Phi(\mathsf{fix}\,\Phi)$.

For $\mathcal{E} := \mathcal{E}(\mathcal{M}) \in \mathsf{SAT}$, we only have to consider the closure conditions of saturated sets. We set $\mathcal{N} := \Phi(\mathcal{M})$.

- If $F[x] \in \mathsf{SN}$, then $F[x] \in \mathcal{E}$: Show $\mathsf{out}(F[x]) = (\mathsf{out}\,F)[x] \in \mathcal{N}$. This is clear since $\mathsf{out}\,F[x] \in \mathsf{SN}$ and $\mathsf{out}\,F$ is a special multiple elimination.
- The proof of $\mathsf{case}\,(F[x],\, x_0.\,t_0,\, x_1.\,t_1) \in \mathcal{E}$ for $F[x] \in \mathsf{SN}$ and $t_0, t_1 \in \mathcal{E} \subseteq \mathsf{SN}$ is done as in the proof for $\mathcal{M} \to \mathcal{N}$ in lemma 21, here with $e := \mathsf{out}\,\star$. This case exhibits a permutation of unfolding into the case construct.
- The now five rules that derive $E[\ldots] \in \mathcal{E}$ are treated as in lemma 21, now with $e' := \mathsf{out}\,\star$.
- The rule for $\mu x.r$ can again be treated in parallel with $\mathcal{M} \to \mathcal{N}$ and $\mathcal{M}_0 + \mathcal{M}_1$ in lemma 21. Not surprisingly, we set $e := \mathsf{out}\,\star$ in that proof part. $\square$

We are now in the position to extend definition 23 by

- $\mathsf{SC}^{\mathsf{fix}\,X.A}[\Delta] := \mathsf{fix}\,\Phi$ with $\Phi : \mathsf{SAT} \to \mathsf{SAT}$, namely $\Phi(\mathcal{M}) := \mathsf{SC}^{A}[\Delta, X : \mathcal{M}]$.

In order to guarantee that Φ, defined above, is monotonically increasing, one has to prove simultaneously with the definition that $\mathsf{SC}^{A}[\Delta, X : \mathcal{M}]$ is monotonically increasing in $\mathcal{M}$ for $X \in \mathsf{TV}_{+}(A)$ and monotonically decreasing for $X \in \mathsf{TV}_{-}(A)$ and not depending on $\mathcal{M}$ if $X \notin \mathsf{FV}(A)$.

Obviously, lemma 24 also holds for $\mathsf{PF}^{+\neg}$. The main lemma 25 about soundness of our interpretation of types easily extends to the two new typing rules: We show the statement for $x_1 : A_1, \ldots, x_n : A_n \vdash \mathsf{in}\,t : \mathsf{fix}\,X.A$ if it already holds for $x_1 : A_1, \ldots, x_n : A_n \vdash t : A[X := \mathsf{fix}\,X.A]$. Write t' for the result of the substitution, applied to t. We have to show that $\mathsf{in}\,t' \in \mathsf{SC}^{\mathsf{fix}\,X.A}[\Delta] = \mathsf{fix}\,\Phi$

for $\Phi(\mathcal{M}) := \mathsf{SC}^A[\Delta, X : \mathcal{M}]$. By hypothesis, $t' \in \mathsf{SC}^{A[X:=\mathsf{fix}\,X.A]}[\Delta]$. By the substitution lemma, this set is $\mathsf{SC}^A[\Delta, X : \mathsf{SC}^{\mathsf{fix}\,X.A}[\Delta]] = \Phi(\mathsf{fix}\,\Phi)$. We are done by lemma 28. The other typing rule is treated analogously.

Corollary 29 *System* $\mathsf{PF}^{+\neg}$ *is strongly normalizing.* □

6 Comparison with Parigot's Generalized Reducibility Candidates

Parigot gave two proofs of strong normalization of Curry-style $\lambda\mu$-calculus by the reducibility method [5]. We first concentrate on the one by *generalized reducibility candidates* and recast it in our framework. Unlike section 4, the concerns for the intuitionistic system and the addition of RAA can be separated, which explains the division into two sections.

6.1 The Intuitionistic Case

The ambient system of our discussion shall here be F^+. Consequently, SN will be meant to be defined without the two rules mentioning $\mu x.r$ in definition 8. Furthermore, the notion of saturatedness SAT shall refer to this reduced SN and not have the two last defining closure rules in definition 15.

The following definition extends Parigot's notion of generalized reducibility candidate [5, p. 1471] to disjunctions with permutative conversions.

Definition 30 (Saturated à la Parigot) *A set $\mathcal{M}$ of terms is P-saturated if $F[x] \in \mathcal{M}$ for every $F[x] \in \mathsf{SN}$ and if there is a set $\mathcal{E}$ of multiple eliminations with $\star \in \mathcal{E}$ such that*

$$\mathcal{M} = \mathcal{E} \to \mathsf{SN} := \{r \mid \forall E \in \mathcal{E}.E[r] \in \mathsf{SN}\}.$$

Let PSAT *be the set of P-saturated sets.*

Immediate consequences of $\mathcal{M} \in \mathsf{PSAT}$ are

- $\mathcal{M} \subseteq \mathsf{SN}$ since $\star \in \mathcal{E}$ and $\star[r] = r$.
- $x \in \mathcal{M}$ by the first requirement, hence for all $E \in \mathcal{E}$, one has $E[x] \in \mathsf{SN}$.

For $\mathcal{M}, \mathcal{N} \in \mathsf{PSAT}$ and every x set $\mathsf{S}_x(\mathcal{M}, \mathcal{N}) := \{t \mid \forall s \in \mathcal{M}.t[x := s] \in \mathcal{N}\}$. Since $x \in \mathcal{M}$ we have $\mathsf{S}_x(\mathcal{M}, \mathcal{N}) \subseteq \mathcal{N} \subseteq \mathsf{SN}$ as before for SAT.

Lemma 31 *The following closure properties hold of* PSAT*:*

- SN *is P-saturated.*

- *If* $U \subseteq \mathsf{PSAT}$*, then* $\bigcap U \in \mathsf{PSAT}$ *(we may set* $\bigcap \emptyset := \mathsf{SN}$*).*
- *If* $\mathcal{M}, \mathcal{N} \in \mathsf{PSAT}$*, then* $\mathcal{M} \to \mathcal{N} := \{r \in \mathsf{SN} \mid \forall s \in \mathcal{M}. rs \in \mathcal{N}\} \in \mathsf{PSAT}$*.*
- *If* $\mathcal{M}_0, \mathcal{M}_1 \in \mathsf{PSAT}$*, then* $\mathcal{M}_0 + \mathcal{M}_1 \in \mathsf{PSAT}$ *where*

  $$\mathcal{M}_0 + \mathcal{M}_1 := \{r \in \mathsf{SN} \mid \forall \mathcal{N} \in \mathsf{PSAT} \forall x_0 \forall t_0 \in \mathsf{S}_{x_0}(\mathcal{M}_0, \mathcal{N}) \\ \forall x_1 \forall t_1 \in \mathsf{S}_{x_1}(\mathcal{M}_1, \mathcal{N}). \mathsf{case}\,(r,\, x_0.\, t_0,\, x_1.\, t_1) \in \mathcal{N}\}.$$

PROOF. We only give the respective sets $\mathcal{E}$:

- $\mathsf{SN} = \{\star\} \to \mathsf{SN}$.
- For every $\mathcal{M} \in U \neq \emptyset$ choose $\mathcal{E}_{\mathcal{M}}$ such that $\star \in \mathcal{E}_{\mathcal{M}}$ and $\mathcal{M} = \mathcal{E}_{\mathcal{M}} \to \mathsf{SN}$. Then $\bigcap U = (\bigcup_{\mathcal{M} \in U} \mathcal{E}_{\mathcal{M}}) \to \mathsf{SN}$.
- If $\mathcal{N} = \mathcal{E}' \to \mathsf{SN}$ with $\star \in \mathcal{E}'$, then $\mathcal{M} \to \mathcal{N} = \mathcal{E} \to \mathsf{SN}$ with

  $$\mathcal{E} := \{\star\} \cup \{E[\star\, s] \mid E \in \mathcal{E}', s \in \mathcal{M}\}.$$

- $\mathcal{M}_0 + \mathcal{M}_1 = \mathcal{E} \to \mathsf{SN}$ with

  $$\mathcal{E} := \{\star\} \cup \{E[\mathsf{case}\,(\star,\, x_0.\, t_0,\, x_1.\, t_1)] \mid \exists \mathcal{E}'. \star, E \in \mathcal{E}', \mathcal{E}' \to \mathsf{SN} \in \mathsf{PSAT}, \\ t_i \in \mathsf{S}_{x_i}(\mathcal{M}_i, \mathcal{E}' \to \mathsf{SN}), i = 0, 1\}.$$

In all cases, one also has to check the first condition of PSAT. For $\mathcal{M}_0 + \mathcal{M}_1$, this requires to show the following simple closure property of SN by induction on the length of E:

$$\frac{\mathsf{case}\,(F[x],\, x_0.\, E[t_0],\, x_1.\, E[t_1]) \in \mathsf{SN}}{E[\mathsf{case}\,(F[x],\, x_0.\, t_0,\, x_1.\, t_1)] \in \mathsf{SN}} \qquad \square$$

With these findings, we can interpret $\bot, \forall, \to, +$ in PSAT. Trivially, we also have the statements of lemma 21 concerning rs and the case construct. For the other two—for $\lambda x.t$ and $\mathsf{inj}_i\, t$—we first show that P-saturated sets are saturated.

Lemma 32 (for F^+) $\mathsf{PSAT} \subseteq \mathsf{SAT}$*.*

PROOF. The first closure rule of SAT is repeated in the definition of PSAT. We have to check the four other rules.

- If $F[x] \in \mathsf{SN}$ and $t_0, t_1 \in \mathcal{M} = \mathcal{E} \to \mathsf{SN}$, show $\mathsf{case}\,(F[x],\, x_0.\, t_0,\, x_1.\, t_1) \in \mathcal{M}$. Let $E \in \mathcal{E}$. Show that $E[\mathsf{case}\,(F[x],\, x_0.\, t_0,\, x_1.\, t_1)] \in \mathsf{SN}$. By the derived closure property in the proof above, it suffices to show $E[t_i] \in \mathsf{SN}$ which follows from $\mathcal{M} = \mathcal{E} \to \mathsf{SN}$.
- If $E[t[x := s]] \in \mathcal{M} = \mathcal{E} \to \mathsf{SN}$ and $s \in \mathsf{SN}$ show $E[(\lambda x.t)s] \in \mathcal{M}$. Let $E' \in \mathcal{E}$. Show $E'[E[(\lambda x.t)s]] \in \mathsf{SN}$. This holds by definition of SN since $E'[E]$ is a multiple elimination.

- The other two rules can be treated analogously by passing from E to $E'[E]$. $\square$

Lemma 33 *Let $\mathcal{M}, \mathcal{N}, \mathcal{M}_0, \mathcal{M}_1$ be P-saturated. Then*

$$\frac{t \in \mathsf{S}_x(\mathcal{M}, \mathcal{N})}{\lambda x.t \in \mathcal{M} \to \mathcal{N}} \qquad \frac{t \in \mathcal{M}_i}{\mathsf{inj}_i\, t \in \mathcal{M}_0 + \mathcal{M}_1}$$

PROOF.

- Let $t \in \mathsf{S}_x(\mathcal{M}, \mathcal{N}) \subseteq \mathsf{SN}$. Then $\lambda x.t \in \mathsf{SN}$. Let $s \in \mathcal{M} \subseteq \mathsf{SN}$. Show $(\lambda x.t)s \in \mathcal{N}$. Since $\mathcal{N} \in \mathsf{SAT}$ by the previous lemma, it suffices to show $t[x := s], s \in \mathsf{SN}$.
- Let $t \in \mathcal{M}_i \subseteq \mathsf{SN}$. Then $\mathsf{inj}_i\, t \in \mathsf{SN}$. Let $\mathcal{N} \in \mathsf{PSAT}$ and x_0, t_0, x_1, t_1 as in the definition of $\mathcal{M}_0 + \mathcal{M}_1$. Show $\mathsf{case}\,(\mathsf{inj}_i\, t,\ x_0.\, t_0,\ x_1.\, t_1) \in \mathcal{N}$. Since $\mathcal{N} \in \mathsf{SAT}$ by the previous lemma, it suffices to show $t_i[x_i := t], t_{1-i} \in \mathcal{N}$ and $t \in \mathsf{SN}$. $\square$

We are now in the position to redo the whole development from definition 23 (strong computability) to corollary 26 (strong normalization) with PSAT instead of SAT and F^+ instead of $\mathsf{F}^{+\neg}$. One only has to omit the references to RAA.

Remark 34 *Parigot [5, p. 1469] uses a strictly-positive (but infinitely branching) inductive definition of his set of reducibility candidates. In our notation, we would define a set MSAT inductively by*

$$\frac{}{\mathsf{SN} \in \mathsf{MSAT}} \qquad \frac{U \subseteq \mathsf{MSAT}}{\bigcap U \in \mathsf{MSAT}} \qquad \frac{\mathcal{M}, \mathcal{N} \in \mathsf{MSAT}}{\mathcal{M} \to \mathcal{N} \in \mathsf{MSAT}}$$

From lemma 31, we get $\mathsf{MSAT} \subseteq \mathsf{PSAT}$ and well-definedness of $\mathcal{M} \to \mathcal{N}$. Unfortunately, this idea cannot be extended to sums as defined in that lemma: The definition of $\mathcal{M}_0 + \mathcal{M}_1$ refers to arbitrary $\mathcal{N} \in \mathsf{PSAT}$, hence PSAT cannot be replaced by MSAT in that definition while maintaining an inductive definition of MSAT. For first-order $\lambda\mu$-calculus, Nour and Saber [24] give a definition of $\mathcal{M}_0 + \mathcal{M}_1$ that allows to treat $+$ like $\to$ in the definition of MSAT above: In our notation, it reads

$$\mathcal{M}_0 + \mathcal{M}_1 := \{r \mid \forall x_0 \forall t_0 \in \mathsf{S}_{x_0}(\mathcal{M}_0, \mathsf{sn}) \forall x_0 \forall t_1 \in \mathsf{S}_{x_1}(\mathcal{M}_1, \mathsf{sn}).\ \mathsf{case}\,(r,\ x_0.\, t_0,\ x_1.\, t_1) \in \mathsf{sn}\}.$$

They even show that every $\mathcal{M} \in \mathsf{MSAT}$ (defined with sn instead of SN) can be written as $\mathcal{E} \to \mathsf{sn}$ with $\star \in \mathcal{E}$ and every $E \in \mathcal{E}$ being of the form F_+, see remark 7.

To conclude, a higher degree of modularization of Parigot's proof method is achieved for F^+. Our construction for sums only works with (an adaptation of) Parigot's generalized reducibility candidates, not with his inductive definition of candidates.

6.2 The Classical Case

In this section, it is shown that the definition of PSAT for F^+ even works for $\mathsf{F}^{+\neg}$ in the sense that $\mathsf{PSAT} \subseteq \mathsf{SAT}$. Consequently, we arrive at an alternative proof of strong normalization for $\mathsf{F}^{+\neg}$, using Parigot's approach in combination with the one presented in section 4.

Let the term rewrite system be that of $\mathsf{F}^{+\neg}$ with its typing and SN and SAT be the original notions, defined in section 4. Interpret the definition of P-saturated sets over the richer term system and the richer notion of SN. Then, the closure properties of PSAT hold without any change to the proof of lemma 31. The real task are the closure properties of individual P-saturated sets: They are saturated.

Lemma 35 (for $\mathsf{F}^{+\neg}$) *$\mathsf{PSAT} \subseteq \mathsf{SAT}$.*

PROOF. Let $\mathcal{M} \in \mathsf{PSAT}$. Show that it satisfies the closure rules of SAT. The proof of lemma 32 does not need any change for the rules treated there. Even the rule for $E[e[\mu x.r]] \in \mathcal{M}$ can be treated analogously to the one for $E[(\lambda x.t)s] \in \mathcal{M}$. Finally, we have to show

$$\frac{r \in \mathsf{SN} \quad \forall z \forall s.\, s \in \mathsf{S}_z(\mathcal{M}, \mathsf{SN}) \Rightarrow r[x := \lambda z.s] \in \mathsf{SN}}{\mu x.r \in \mathcal{M}}$$

Assume the premises, let $\mathcal{M} = \mathcal{E} \to \mathsf{SN}$ and $E \in \mathcal{E}$. Show $E[\mu x.r] \in \mathsf{SN}$.

This is trivial for $E = \star$ since $r \in \mathsf{SN}$. Otherwise, $E = E'[e]$, and it suffices to show $E'\Big[\mu y.r[x := \lambda z.y(e[z])]\Big], E[x'] \in \mathsf{SN}$ for some variable x'. The second term is in SN by the observation immediately after the definition of PSAT. For the first term, we will have some difficulties. If $E' = \star$, it suffices to show $r[x := \lambda z.y(e[z])] \in \mathsf{SN}$. We use the second premise with $s := y(e[z])$. Let $t \in \mathcal{M}$. We have to show $s[z := t] = y(e[t]) \in \mathsf{SN}$. Since $E' = \star$, $e[t] = E[t] \in \mathsf{SN}$ by our choice of $\mathcal{E}$. Hence, $y(e[t]) \in \mathsf{SN}$. To conclude, this case can be dealt with comfortably. This is not true of $E' = E''[e']$: It would suffice to show

$$E''\Big[\mu y_1.r[x := \lambda z.y(e[z])][y := \lambda z_1.y_1(e'[z_1])]\Big], E'[x'] \in \mathsf{SN}.$$

The first term is $s := E''\Big[\mu y_1.r\Big[x := \lambda z.\Big(\lambda z_1.y_1(e'[z_1])\Big)(e[z])\Big]\Big]$ which reduces in finitely many (because it happens in the substituted term) "administrative" β-reduction steps to $s' := E''\Big[\mu y_1.r[x := \lambda z.y_1(e'[e[z]])]\Big]$, but we have no means of inferring $s \in \mathsf{SN}$ from $s' \in \mathsf{SN}$. If we had it, we could go on and finally reduce our goal to prove $\mu y_n.r[x := \lambda z.y_n(E[z])] \in \mathsf{SN}$. And this would be provable just as in the case of $E' = \star$. Moreover, we forgot the second term $E'[x']$ above. It satisfies $(E'[x'])[x' := e[x']] = E[x'] \in \mathsf{SN}$, but we cannot directly see that $E'[x'] \in \mathsf{SN}$, since certainly nothing allows us to assume that $\mathcal{E}$ were closed under the operation that transforms $E'[e]$ into E'. However, one can show the following derived rule of SN by induction on SN:

$$\frac{r[x := s] \in \mathsf{SN}}{r \in \mathsf{SN}}$$

(This rule would be evident with SN replaced by sn since $r \longrightarrow r'$ implies $r[x := s] \longrightarrow r'[x := s]$.) Hence, the problem with the second term is solved.[8] Unfortunately, a second derived rule of SN is needed in order to get the right invariants in the development below:

$$\frac{e[t] \in \mathsf{SN}}{t \in \mathsf{SN}}$$

It can be proven by course-of-value induction on SN, i. e., by induction on SN with use of the inductive hypothesis for all terms "found in SN before" the respective term under consideration (this is crucial for the treatment of the fourth rule of SN with $E := \star$). The proof needs to refer to the derived rule above in the clauses of SN which introduce a β-redex $(\lambda x.t)s$ or a μ-redex.

We will now rigorously prove that $E[\mu x.r] \in \mathsf{SN}$ for $E \neq \star$. Then there is an $\ell \geq 1$ and there are eliminations e_n, $1 \leq n \leq \ell$ such that $E = E_\ell$ with

$$\begin{aligned} E_0 &:= \star \\ E_{n+1} &:= e_{n+1}[E_n] \quad \text{for } 0 \leq n < \ell \end{aligned}$$

Define the "tails" of E by

$$\begin{aligned} E_{>\ell} &:= \star \\ E_{>n} &:= E_{>(n+1)}[e_{n+1}] \quad \text{for } 0 \leq n < \ell \end{aligned}$$

[8] The solution does not fit with the promise made immediately preceding definition 8 of SN. The normalization proof in section 4 is in better harmony with the SN method.

Then $E_{>n}[E_n] = E$ for all $0 \leq n \leq \ell$, to be proven by induction from ℓ downwards to 0. Hence, $E_{>0} = E$. Let $y_0, \dots, y_\ell$ be different variables not occurring free in $\mu x.r$ or in E. Define the iterated substitutions stemming from the repeated unfolding of the rule of SN pertaining to μ-reduction:

$$\begin{aligned} s_0 &:= y_0 \\ s_{n+1} &:= s_n\Big[y_n := \lambda z_{n+1}.y_{n+1}(e_{n+1}[z_{n+1}])\Big] \quad \text{for } 0 \leq n < \ell \end{aligned}$$

Define $t_n := E_{>n}[\mu y_n.r[x := s_n]]$ for $0 \leq n \leq \ell$. Then $t_0 = E[\mu x.r]$ by the variable convention. This is the end of the preparations for the first stage in the two-stage process of the proof, namely the unwinding of the conditions ensuring $E[\mu x.r] \in \mathsf{SN}$ along the whole multiple elimination E.

Let x' be a variable. Then $E[x'] \in \mathsf{SN}$, hence $E_{>n}[x'] \in \mathsf{SN}$ for $0 \leq n \leq \ell$ by the first derived rule of SN, mentioned above. For every $0 \leq n < \ell$, we have

$$\frac{t_{n+1} \in \mathsf{SN}}{t_n \in \mathsf{SN}}$$

This is proven as follows: Let $t_{n+1} \in \mathsf{SN}$. $t_n = E_{>(n+1)}[e_{n+1}[\mu y_n.r[x := s_n]]]$. Hence, for $t_n \in \mathsf{SN}$ it suffices to show

$$E_{>(n+1)}\Big[\mu y_{n+1}.\underbrace{r[x := s_n][y_n := \lambda z_{n+1}.y_{n+1}(e_{n+1}[z_{n+1}])]}_{r[x:=s_n[y_n:=\lambda z_{n+1}.y_{n+1}(e_{n+1}[z_{n+1}])]]=r[x:=s_{n+1}]}\Big], E_{>n}[x'] \in \mathsf{SN}.$$

The first term is $t_{n+1} \in \mathsf{SN}$, the second one is in SN, as shown above.

By applying the rule ℓ times, we can infer $t_0 = E[\mu x.r] \in \mathsf{SN}$ from $t_\ell \in \mathsf{SN}$. Hence, it remains to show $t_\ell = \mu y_\ell.r[x := s_\ell] \in \mathsf{SN}$. Before the second stage of the proof can be entered, the term s_ℓ has to be represented differently.

Let $z_1, \dots, z_\ell$ be "fresh" variables. Define the "building blocks" of s_ℓ by

$$\begin{aligned} u_\ell &:= y_\ell(e_\ell[z_\ell]) \\ u_n &:= (\lambda z_{n+1}.u_{n+1})(e_n[z_n]) \quad \text{for } 1 \leq n < \ell \end{aligned}$$

Then $z_1, \dots, z_\ell$ are not free in $\lambda z_n.u_n$ for $1 \leq n \leq \ell$. One shows that $s_\ell = \lambda z_1.u_1$ by showing $s_\ell = s_n[y_n := \lambda z_{n+1}.u_{n+1}]$ for $0 \leq n < \ell$ which can be done by induction from $\ell - 1$ downwards to 0.

It remains to show $t_\ell = \mu y_\ell.r[x := s_\ell] = \mu y_\ell.r[x := \lambda z_1.u_1] \in \mathsf{SN}$.

We use the second premise (of the rule deriving $\mu x.r \in \mathcal{M}$) to show that $r[x := \lambda z_1.u_1] \in \mathsf{SN}$. It hence suffices to show $u_1 \in \mathsf{S}_{z_1}(\mathcal{M}, \mathsf{SN})$. Let $t \in \mathcal{M}$. Show $u_1[z_1 := t] \in \mathsf{SN}$. The second stage of the proof consists in unwinding the conditions for this to hold.

We have $E[t] \in \mathsf{SN}$, hence by the second derived rule for SN above (applied repeatedly), we get $E_n[t] \in \mathsf{SN}$ for every $0 \leq n \leq \ell$. Set $v_n := u_n[z_n := E_{n-1}[t]]$ for $1 \leq n \leq \ell$. We have to show $v_1 \in \mathsf{SN}$. For every $1 \leq n < \ell$, we have

$$\frac{v_{n+1} \in \mathsf{SN}}{v_n \in \mathsf{SN}}$$

For a proof, assume $v_{n+1} \in \mathsf{SN}$. $v_n = (\lambda z_{n+1}.u_{n+1})(e_n[E_{n-1}[t]])$ since z_n does not occur free in u_{n+1}. Therefore, $v_n \in \mathsf{SN}$ if $e_n[E_{n-1}[t]] = E_n[t] \in \mathsf{SN}$ and $u_{n+1}[z_{n+1} := E_n[t]] \in \mathsf{SN}$. The first term is dealt with above, the second one is $v_{n+1} \in \mathsf{SN}$.

We use this rule $\ell - 1$ times and therefore get $v_1 \in \mathsf{SN}$ from $v_\ell \in \mathsf{SN}$. Hence, it only remains to show $v_\ell = u_\ell[z_\ell := E_{\ell-1}[t]] = y_\ell(e_\ell[E_{\ell-1}[t]]) = y_\ell(E[t]) \in \mathsf{SN}$. This is true since $t \in \mathcal{M}$ implies $E[t] \in \mathsf{SN}$. $\Box$

Corollary 36 $\mathcal{M} \in \mathsf{PSAT}$ *and* $r \in \mathsf{S}_x(\mathcal{M} \to \mathsf{SN}, \mathsf{SN})$ *imply* $\mu x.r \in \mathcal{M}$.

PROOF. Loosely speaking, we only need to appeal to lemma 22. Strictly speaking, one has to reprove that lemma since $\mathcal{M} \to \mathsf{SN}$ has been differently defined in lemma 31. However, the only needed change is reference to lemma 33 instead of that to lemma 21 which both say that $s \in \mathsf{S}_z(\mathcal{M}, \mathsf{SN})$ implies $\lambda z.s \in \mathcal{M} \to \mathsf{SN}$. $\Box$

As for the intuitionistic case in the previous section, we can again redo the development from definition 23 (strong computability) to corollary 26 (strong normalization) with PSAT instead of SAT. We again conclude strong normalization of $\mathsf{F}^{+\neg}$.

7 Final Comments

In this article, strong normalization of the second-order system $\mathsf{F}^{+\neg}$ with disjunction, permutative conversions, reductio ad absurdum (RAA) and μ-reductions for RAA is proven with two quite different constructions of sets of strongly computable terms, i. e., candidates of reducibility. The tradeoff is as follows:

- In the first proof (in section 4) the notion of saturated set uses a complicated clause that amounts to a non-strictly positive operator, the second proof (the previous section) just refers to a set $\mathcal{E}$ of multiple eliminations.
- The first proof defines $\mathcal{M} \to \mathcal{N}$ and $\mathcal{M}_0 + \mathcal{M}_1$ in a way that immediately yields the soundness of these constructions for the introduction rules of the typing system (the first and third rule in lemma 21). The hard work concerns the elimination rules (the second and fourth rule in that lemma). Therefore, it is reasonable to call those definitions *introduction-based*, in short, I-based. (Definitions of validity such as the one used by Prawitz in the proof of strong normalization also for disjunction [11] are I-based.) For the second proof, the situation is reversed: The soundness of the elimination rules is directly built into the definitions of $\mathcal{M} \to \mathcal{N}$ and $\mathcal{M}_0 + \mathcal{M}_1$, the introduction rules treated in lemma 33 need lemma 32. We would like to call those definitions elimination-based, in short, E-based. Interestingly, Prawitz expressed in [11, p. 290] that disjunctions "seem impossible to handle in this way". Certainly, the second proof only works due to the quantification over all P-saturated sets in the definition of $\mathcal{M}_0 + \mathcal{M}_1$.
- $\mu x.r$ falls outside the introduction/elimination-dichotomy. In the first proof, it needs an argument in the soundness proof of every elimination rule in lemma 21. In the second proof, this is just done once in the proof of lemma 35.
- The first proof works very smoothly with our inductive definition SN of the "interesting strongly normalizing terms", the second proof needs derived closure rules of SN which would be obvious for the set sn of *all* strongly normalizing terms.
- Concededly, the proof of lemma 35 which is central to the second proof, is not very elegant, although indispensable with our formulation of $\mathsf{F}^{+\neg}$. A restriction of the admissible terms would simplify the argument considerably: The idea is to require occurrences of a variable x that are bound by some μ to be in a subterm of the form $x\,s$. Then the right-hand side of the μ-reduction rule would be changed from $\mu y.r[x := \lambda z.y(e[z])]$ to $\mu y.r[x\star := y\,e]$, meaning that every subterm in r of the form $x\,s$ is replaced by $y(e[s])$, with the same operation recursively applied to s, i. e., in terms of $\mathsf{F}^{+\neg}$, a reduction strategy is imposed that requires to carry out the "administrative" β-reductions immediately after the μ-reduction. This approach is taken by Andou [19], and the admissible terms are called $\lambda^{\perp}$-regular. A change to the syntax itself leads to Parigot's $\lambda\mu$-calculus [12]: Variables that may be bound by μ are taken from another set of variables, the *μ-variables*, and free occurrences of those μ-variables a only come from an application $a\,s$ to a term s (written as $[a]s$ and called a *named* term by Parigot). Parigot goes even further and avoids $\perp$ altogether in his formulation of $\lambda\mu$-calculus.

In the more liberal formulation by de Groote [6] [9], the μ-reduction rule would become $e[\mu a.r] \longrightarrow_\mu \mu b.r[a\star := b\,e]$ with b a "new" μ-variable. SN and SAT could be adapted to that formulation, and the problem with the "administrative" β-reductions mentioned at the beginning of the proof of lemma 35 would disappear: There would be no serious difference between μ-reduction with an elimination e or a multiple elimination E in place of e, as in the original proof by Parigot [5]. [10]
- In the first proof, everything is defined in order to get an interpretation of RAA, in the second proof this just happens to be the case.

As regards extensions, we remark the following:

- The addition of products $A \times B$, i. e., conjunction, would not cause the least difficulty: They would give rise to a new kind of special eliminations f, namely the projections, applied to $\star$. These would also go into the notions e, F and E and hence extend the notion of $\longrightarrow_\pi$ and $\longrightarrow_\mu$. The definitions of SN and of SAT would be extended by rules, analogous to those for lambda abstraction, a construction $\mathcal{M}_0 \times \mathcal{M}_1$ on saturated sets would be needed and its soundness w. r. t. the typing rules would have to be established. Also with the approach à la Parigot, no new ideas would be needed.
- System $\mathsf{PF}^{+\neg}$ which adds non-strictly positive and interleaving fixed-points to $\mathsf{F}^{+\neg}$ has been treated in section 5 by the I-based method. It is an exercise to dualize the construction of $\mathsf{fix}\,\Phi$ to the E-based approach of the previous section.
- Also, we do not expect any problem with the extension by second-order existential quantifiers that come with their own permutative conversions.
- Finally, we could easily prove strong normalization for the extension of $\mathsf{F}^{+\neg}$ by primitive recursion (and iteration) on monotone inductive types with new μ-reduction rules that "reduce" stability of those fixed-points to that of the arbitrary target types of the recursively defined functions. This is not done by extending the normalization proof for $\mathsf{PF}^{+\neg}$, but by a reduction-preserving embedding of that extension into $\mathsf{PF}^{+\neg}$, see [29] for a presentation with fixed types and $\lambda\mu$-calculus. μ-reduction and permutation rules are preserved since that embedding translates the recursor (and the iterator)—which gives rise to an elimination—into a multiple elimination of $\mathsf{PF}^{+\neg}$.

[9] De Groote claims that his treatment of first-order $\lambda\mu$-calculus with disjunction and permutative conversions can be raised to second order [6, p. 185]. It seems that one would have to argue about typed terms instead of using Curry-style, though.
[10] Only the obligation to prove $r \in \mathsf{SN}$ from $r[x := s] \in \mathsf{SN}$ would remain a small spot in the elegant picture.

References

[1] H. Schwichtenberg, Type theory, Lecture course notes, Mathematical Institute of the University of Munich (1992).

[2] J. van de Pol, H. Schwichtenberg, Strict functionals for termination proofs, in: M. Dezani-Ciancaglini, G. Plotkin (Eds.), Typed Lambda Calculi and Applications, Vol. 902 of Lecture Notes in Computer Science, Springer Verlag, 1995, pp. 350–364.

[3] H. Schwichtenberg, Proof theory, Lecture course notes, Mathematical Institute of the University of Munich (1994).

[4] H. Schwichtenberg, Proofs, lambda terms and control operators, in: Logic of Computation, Vol. 157 of NATO Advanced Science Institute Series F: Computer and System Sciences, Springer Verlag, 1997, pp. 309–347.

[5] M. Parigot, Proofs of strong normalisation for second order classical natural deduction, The Journal of Symbolic Logic 62 (4) (1997) 1461–1479.

[6] P. de Groote, Strong normalization of classical natural deduction with disjunction, in: S. Abramsky (Ed.), Proceedings of TLCA 2001, Vol. 2044 of Lecture Notes in Computer Science, Springer Verlag, 2001, pp. 182–196.

[7] W. W. Tait, A realizability interpretation of the theory of species, in: R. Parikh (Ed.), Logic Colloquium Boston 1971/72, Vol. 453 of Lecture Notes in Mathematics, Springer Verlag, 1975, pp. 240–251.

[8] F. van Raamsdonk, P. Severi, On normalisation, Tech. Rep. CS-R9545, Centrum voor Wiskunde en Informatica (CWI), Amsterdam (June 1995).

[9] F. van Raamsdonk, P. Severi, M. H. Sørensen, H. Xi, Perpetual reductions in lambda calculus, Information and Computation 149 (2) (1999) 173–225.

[10] J.-L. Krivine, Lambda-calculus, types and models, Masson, Paris and Ellis Horwood, Hemel Hempstead, 1993, english translation of Lambda-calcul, types et modèles, Masson, 1990.

[11] D. Prawitz, Ideas and results in proof theory, in: J. E. Fenstad (Ed.), Proceedings of the Second Scandianvian Logic Symposium, North–Holland, Amsterdam, 1971, pp. 235–307.

[12] M. Parigot, $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction, in: A. Voronkov (Ed.), Logic Programming and Automated Reasoning, International Conference LPAR'92, St. Petersburg, Russia, Vol. 624 of Lecture Notes in Computer Science, Springer Verlag, 1992, pp. 190–201.

[13] J. Rehof, M. H. Sørensen, The λ_Δ-calculus, in: M. Hagiya, J. C. Mitchell (Eds.), Theoretical Aspects of Computer Software, International Conference TACS '94, Sendai, Japan, Proceedings, Vol. 789 of Lecture Notes in Computer Science, Springer Verlag, 1994, pp. 516–542.

[14] D. Prawitz, Natural Deduction. A Proof-Theoretical Study., Almquist and Wiksell, 1965.

[15] T. Joly, An embedding of 2nd order classical logic into functional arithmetic FA2, C. R. Acad. Sci. Paris, Série I 325 (1997) 1–4.

[16] G. Stålmarck, Normalization theorems for full first order classical natural deduction, The Journal of Symbolic Logic 56 (1) (1991) 129–149.

[17] E. Ritter, D. Pym, L. Wallen, On the intuitionistic force of classical search, Theoretical Computer Science 232 (1–2) (2000) 299–333.

[18] M. Takahashi, Parallel reduction in λ-calculus, Information and Computation 118 (1) (1995) 120–127.

[19] Y. Andou, Church-Rosser property of a simple reduction for full first-order classical natural deduction, Annals of Pure and Applied Logic 119 (2003) 225–237.

[20] F. Joachimski, R. Matthes, Short proofs of normalization for the simply-typed lambda-calculus, permutative conversions and Gödel's T, Archive for Mathematical Logic 42 (1) (2003) 59–87.

[21] R. David, K. Nour, A short proof of the strong normalization of classical natural deduction with disjunction, The Journal of Symbolic Logic, to appear.

[22] F. Joachimski, Reduction Properties of ΠIE-Systems, Ph.D. thesis, LMU München (2001).

[23] J.-Y. Girard, Interprétation fonctionnelle et élimination des coupures dans l'arithmétique d'ordre supérieur, Thèse de Doctorat d'État, Université de Paris VII (1972).

[24] K. Nour, K. Saber, A semantical proof of the strong normalization theorem of full propositional classical natural deduction, submitted for publication, 2003.

[25] H. Geuvers, Inductive and coinductive types with iteration and recursion, in: B. Nordström, K. Pettersson, G. Plotkin (Eds.), Proceedings of the Workshop on Types for Proofs and Programs, Båstad, Sweden, 1992, pp. 193–217, `ftp://ftp.cs.chalmers.se/pub/cs-reports/baastad.92/proc.dvi.Z`, only electronically available.

[26] N. P. Mendler, Recursive types and type constraints in second-order lambda calculus, in: Proceedings of the Second Annual IEEE Symposium on Logic in Computer Science, Ithaca, N.Y., IEEE Computer Society Press, 1987, pp. 30–36.

[27] R. Matthes, Monotone fixed-point types and strong normalization, in: G. Gottlob, E. Grandjean, K. Seyr (Eds.), Computer Science Logic, 12th International Workshop, Brno, Czech Republic, August 24–28, 1998, Proceedings, Vol. 1584 of Lecture Notes in Computer Science, Springer Verlag, 1999, pp. 298–312.

[28] Z. Spławski, P. Urzyczyn, Type Fixpoints: Iteration vs. Recursion, SIGPLAN Notices 34 (9) (1999) 102–113, proceedings of the 1999 International Conference on Functional Programming (ICFP), Paris, France.

[29] R. Matthes, Parigot's second order $\lambda\mu$-calculus and inductive types, in: S. Abramsky (Ed.), Proceedings of TLCA 2001, Vol. 2044 of Lecture Notes in Computer Science, Springer Verlag, 2001, pp. 329–343.