Bulletin of the **S**ection of **L**ogic
Volume 52/2 (2023), pp. 145–186
https://doi.org/10.18778/0138-0680.2023.19

Emma van Dijk
David Ripley
Julian Gutierrez

CORE TYPE THEORY

Abstract

Neil Tennant's core logic is a type of bilateralist natural deduction system based on proofs and refutations. We present a proof system for propositional core logic, explain its connections to bilateralism, and explore the possibility of using it as a type theory, in the same kind of way intuitionistic logic is often used as a type theory. Our proof system is not Tennant's own, but it is very closely related. The difference matters for our purposes, and we discuss this. We then turn to the question of strong normalization, showing that although Tennant's proof system for core logic is not strongly normalizing, our modified system is.

Keywords: core logic, type theory, strong normalization.

2020 Mathematical Subject Classification: 03A05, 03B38, 03B47, 03F05.

1. Introduction

Neil Tennant's core logic is a type of bilateralist natural deduction system based on proofs and refutations. We present a proof system for propositional core logic, explain its connections to bilateralism, and explore the possibility of using it as a type theory, in the same kind of way intuitionistic logic is often used as a type theory. Our proof system is not Tennant's own, but it is very closely related. The difference matters for our purposes, and we discuss this. We then turn to the question of strong normalization, showing that although Tennant's proof system for core logic is not strongly normalizing, our modified system is.

Presented by: Sara Ayhan
Received: August 29, 2022
Published online: August 9, 2023

2. Core logic

We open by presenting a natural deduction system for core logic. This is not Tennant's own system, although it is closely related. (As the paper progresses, we'll get more and more perspective on the differences; we discuss them in Sections 2.4, 3.5 and 5.1.) The language is an ordinary propositional language with connectives $\wedge, \vee, \to, \neg$ of arities 2, 2, 2, 1, respectively. We use $p, q, r, \ldots$ for atomic formulas and $\varphi, \psi, \theta, \ldots$ for arbitrary formulas. We suppress parentheses according to the following conventions: the connectives $\wedge$ and $\vee$ bind more tightly than $\to$, and $\neg$ more tightly still; and $\to$ associates to the right. Thus $\neg p \wedge q \to r \vee s \to t$ is $((\neg p) \wedge q) \to ((r \vee s) \to t)$.

2.1. Natural deduction

We first present core logic via a natural deduction system, following presentations such as [15, 21, 22]. This proceeds in the style of [5, 12], with an important modification: not every node in a derivation needs to be a formula. There is one additional symbol ☹ that can also occupy nodes in a derivation. It is important to keep in mind, though, that ☹ is *not* a formula, and does not enter into formula construction. As a result, things like '¬☹' and '☹ ∧ p' make no sense.[1]

We will call the things that can stand at nodes of a derivation *hats* (for reasons that will emerge). That is, a hat is either a formula or else ☹. Recall that we use $\varphi, \psi, \theta, \ldots$ for arbitrary *formulas*; for arbitrary *hats*, we use $\mathfrak{C}, \mathfrak{D}$. There is an important partial order on hats: $\mathfrak{C} \leq \mathfrak{D}$ iff either $\mathfrak{C}$ is ☹ or $\mathfrak{C} = \mathfrak{D}$. That is, any two distinct formulas are $\leq$-incomparable, and ☹ is $\leq$-below all formulas. We will also use the maximum $\max(\mathfrak{C}, \mathfrak{D})$ of two hats $\mathfrak{C}, \mathfrak{D}$ according to this order; note that this is only defined when either $\mathfrak{C} = \mathfrak{D}$ or one of $\mathfrak{C}, \mathfrak{D}$ is ☹. A *sequent*, as we use the term, is a set of premise *formulas* and a conclusion *hat*; we write $\Gamma \succ \mathfrak{C}$ for the sequent with premises Γ and conclusion $\mathfrak{C}$. We draw a distinction between sequents and arguments: an *argument* is a sequent with a formula as its conclusion.

The role of ☹ in these systems is not to carry content, the way a formula might. Rather, when it occurs in a derivation, it should be seen as part of the structure of that derivation, the surrounds that the content-bearing

[1] Tennant uses the symbol $\bot$ for this purpose; we use ☹ instead because $\bot$ is in common use in other work as a formula. To reduce potential confusion, we've chosen a symbol that is not usually used as a formula.

formulas fit into. It plays, then, the same kind of role in a derivation as the horizontal bar separating nodes from each other, or the rule labels decorating such bars, or markers of which assumptions are discharged; it indicates (in concert with other such apparatus) relations between the formulas in play.

Assumptions work as usual in these natural deduction systems, and in particular only *formulas* may be assumed. Any derivation, then, has a set Γ of open assumptions, all of which are formulas, and it has a conclusion node, which is a hat $\mathfrak{C}$. We refer to $\Gamma \succ \mathfrak{C}$ as *the sequent of* the derivation, and the derivation as *a derivation of* its sequent. What we understand a derivation as telling us depends on whether the derivation's sequent is an argument or not. A derivation with sequent $\Gamma \succ \varphi$ should be understood as a proof of φ from the assumptions Γ, or, as we will also say, a proof of the argument $\Gamma \succ \varphi$. On the other hand, a derivation with sequent $\Gamma \succ$ ☹ should be understood as a *refutation* of the set Γ. It is very much not a proof of ☹—that wouldn't make sense, as ☹ does not carry content. We have here two fundamentally different roles for a derivation to play: a proof of an argument, or a refutation of a set of formulas.

This is the bilateralism in core logic: a bilateralism of proofs and refutations. In this setting, it would not be right to understand either proofs or refutations as a special kind of the other. The rules of derivation allow us to build proofs and refutations both, from components that themselves may be proofs and refutations both. In this sense, then, core logic derivations are bilateralist: based on two core notions, one positive and one negative, neither of which should be understood as a special case of the other. In this regard, the bilateralism in core logic is like the bilateralisms explored in [1, 23, 24, 25]. Tennant's discussion of these issues in [19] is useful here.

To forestall any misunderstandings, however, we note that core logic is not at all *symmetrical* in the way that many bilateralist theories are. Proofs and refutations in these systems are not at all each other's mirror image. Even before we present the rules, we can see this already, as they apply to different things. A proof is a proof of an *argument*: a pair of a set of premises and a single conclusion; while a refutation is a refutation of just a set of formulas. Both are species of derivation, to be sure, but neither is reducible to the other.

2.2. Rules for core logic

With that understood, derivations are otherwise relatively standard. What makes core logic distinctive, other than some care about the difference between formulas and hats, is its use of mostly *general* eliminations (see for example [17] or [10, Ch. 8]), and a bit of fuss around discharge policies.

Derivations begin, as usual, from *assumptions*. Any formula may be assumed; recall that ☹, which is not a formula, may not be assumed. An assumption of φ counts as a proof of $\varphi \succ \varphi$: a proof of φ from the open assumption φ.

2.2.1. Conjunction

From here, rules proceed connective by connective, with introduction and elimination rules for each connective. Each elimination rule has a major premise, which will be indicated as we proceed. Many of these rules have particular restrictions against certain kinds of vacuous discharge, which we will describe as we go.

$$\wedge\mathrm{I}\ \frac{\varphi \qquad \psi}{\varphi \wedge \psi} \qquad\qquad \wedge\mathrm{E}^n\ \frac{\varphi \wedge \psi \qquad \begin{matrix}[\varphi, \psi]^n \\ \vdots \\ \mathfrak{C}\end{matrix}}{\mathfrak{C}}$$

Discharged assumptions are marked with [square brackets]; other assumptions, including other occurrences of these discharged formulas, may also occur as assumptions.[2] We use numeral annotations (here schematized as ^{n}) to indicate which rule discharges which discharged assumption: in any derivation, we assume that each occurrence of each discharging rule wears a distinct discharge numeral, and that each discharged assumption wears the numeral corresponding to the rule occurrence that discharged it.

Discharge restriction: in $\wedge$E, the discharge $[\varphi, \psi]$ may not be completely vacuous. That is, it must discharge at least one occurence of φ *or* at least one occurrence of ψ. The major premise of $\wedge$E is $\varphi \wedge \psi$.

[2]See Section 2.4 for discussion.

2.2.2. Disjunction

$$\vee\mathrm{I}_l\ \frac{\varphi}{\varphi \vee \psi} \qquad \vee\mathrm{I}_r\ \frac{\psi}{\varphi \vee \psi} \qquad \vee\mathrm{E}^n\ \frac{\varphi \vee \psi \quad \begin{matrix}[\varphi]^n \\ \vdots \\ \mathfrak{C}\end{matrix} \quad \begin{matrix}[\psi]^n \\ \vdots \\ \mathfrak{D}\end{matrix}}{\max(\mathfrak{C}, \mathfrak{D})}$$

Discharge restriction: in $\vee$E, *neither* discharge $[\varphi]$ nor $[\psi]$ may be vacuous. Recall as well that $\max(\mathfrak{C}, \mathfrak{D})$ is only defined when either $\mathfrak{C} = \mathfrak{D}$ or at least one of $\mathfrak{C}, \mathfrak{D}$ is ☹; in other cases the rule $\vee$E is not applicable. The major premise of $\vee$E is $\varphi \vee \psi$.

2.2.3. Implication

$$\to\mathrm{I}^n\ \frac{\begin{matrix}[\varphi]^n \\ \vdots \\ \mathfrak{C}\end{matrix}}{\varphi \to \psi} \qquad \to\mathrm{E}^n\ \frac{\varphi \to \psi \quad \varphi \quad \begin{matrix}[\psi]^n \\ \vdots \\ \mathfrak{C}\end{matrix}}{\mathfrak{C}}$$

In the rule $\to$I, we must have $\mathfrak{C} \leq \psi$. In addition, *if* $\mathfrak{C}$ is ☹, then the discharge of $[\varphi]$ must not be vacuous. However, in cases where $\mathfrak{C}$ is ψ itself, the discharge $[\varphi]$ may be vacuous. In $\to$E, the discharge $[\psi]$ may not be vacuous. The major premise of $\to$E is $\varphi \to \psi$.

2.2.4. Negation

$$\neg\mathrm{I}^n\ \frac{\begin{matrix}[\varphi]^n \\ \vdots \\ ☹\end{matrix}}{\neg\varphi} \qquad \neg\mathrm{E}\ \frac{\neg\varphi \quad \varphi}{☹}$$

Discharge restriction: in $\neg$I, the discharge $[\varphi]$ may not be vacuous. The major premise of $\neg$E is $\neg\varphi$.

2.3. Core derivations and core logic

What we have in view so far is in fact a proof system for *intuitionistic* logic, not core logic. That is, an argument $\Gamma \succ \varphi$ is provable in this system iff it is intuitionistically valid, and a set Γ of formulas is refutable in this system iff it is intuitionistically inconsistent.[3]

To get to core logic, we use the notion of a *core derivation*, which we now present. A derivation is *core* iff every major premise of every elimination rule in it is an assumption, and a sequent is *core derivable* iff it is the sequent of some core derivation. We say that an argument is *core provable* iff it has a proof that is core, and that a set of formulas is *core refutable* iff it has a refutation that is core.

Not every provable argument is core provable. For example, $\neg p, p \succ q$ is provable as follows:

$$
{\to}\mathrm{E}^2\ \dfrac{{\to}\mathrm{I}^1\ \dfrac{\neg\mathrm{E}\ \dfrac{\neg p \qquad [p]^1}{\text{☹}}}{p \to q} \qquad p \qquad [q]^2}{q}
$$

This derivation is not core, as the major premise of $\to$E in it is the conclusion of a step of $\to$I rather than an assumption. And indeed there is no core proof of $\neg p, p \succ q$. To see this, note (by checking the rules) that in a core derivation, every formula that occurs must be a subformula either of some open assumption or of the conclusion. That gives very little room to work with when attempting to prove $\neg p, p \succ q$, and it's not hard to see that the task can't be done. The closest we can get is instead a core refutation of the set $\{\neg p, p\}$:

$$
\neg\mathrm{E}\ \dfrac{\neg p \qquad p}{\text{☹}}
$$

Similarly, not every refutable set of formulas is core refutable. For example, the set $\{\neg p, p, q\}$ is refutable as follows:

$$
\neg\mathrm{E}\ \dfrac{\neg p \qquad \wedge\mathrm{E}^1\ \dfrac{\wedge\mathrm{I}\ \dfrac{p \qquad q}{p \wedge q} \qquad [p]^1}{p}}{\text{☹}}
$$

[3] For discussion of this point, see [13, 20].

However, this set has no core refutation, by similar reasoning to the above. Again, the closest we can get is a core refutation of the distinct set $\{\neg p, p\}$.

One way to see core logic as a consequence relation is this: say that a sequent $\Gamma \succ \mathfrak{C}$ is in core logic iff it is core derivable. As we've just seen, then, neither $\neg p, p \succ q$ nor $\neg p, p, q \succ ☹$ is in core logic, but $\neg p, p \succ ☹$ is in core logic. In this sense, then, core logic is nonmonotonic on both sides: neither $\subseteq$ on the left nor $\leq$ on the right preserves core derivability.

Core logic is probably best known for not admitting *cut*: there are cases where both $\Gamma \succ \varphi$ and $\varphi, \Delta \succ \mathfrak{C}$ are in core logic, but where $\Gamma, \Delta \succ \mathfrak{C}$ is not. For example, $p \succ p \vee q$ and $\neg p, p \vee q \succ q$ are both core derivable, but we've just seen that $\neg p, p \succ q$ is not. What holds instead is a property Tennant calls *epistemic gain*: whenever both $\Gamma \succ \varphi$ and $\varphi, \Delta \succ \mathfrak{C}$ are in core logic, then there is some $\Sigma \succ \mathfrak{D}$ in core logic such that $\Sigma \subseteq \Gamma \cup \Delta$ and $\mathfrak{D} \leq \mathfrak{C}$. Tennant appeals to epistemic gain to defuse criticisms of core logic based on its not admitting cut, and we will depend on epistemic gain in much of our reasoning that follows. It's not our purpose here, however, to evaluate core logic, so we don't discuss such defenses further; our purposes just involve noting that this epistemic gain property holds.

2.4. The Prawitz restriction

That, then, is the natural deduction system we will work with in what follows. It differs from Tennant's own systems for core logic and its relatives in one important respect, which is the topic of this subsection and Sections 3.5 and 5.1. Tennant's systems, as we interpret them, impose a further restriction on discharges, one that we do not impose: that whenever a rule application *can* discharge an occurrence of an open assumption, it *must* discharge that occurrence.

The first thing to note about this restriction is that it has nothing special to do with core logic. Restrictions like this can be imposed, or not, in ordinary natural deduction systems for logics of all sorts. For example, Gentzen's original system NJ (in [5]) for intuitionistic logic does not impose any such restriction; but Prawitz's closely-related system I (in [12]) for intuitionistic logic adds this restriction. Accordingly, we call this restriction 'the Prawitz restriction', and call a derivation 'Prawitz' when it obeys this restriction.[4]

[4]For Tennant's imposing this restriction, see, e.g., [16, p. 674], [22, §§2.3.2, 4.6]. In some other places, however, Tennant is less explicit. For example, [21, p. 454] imposes the restriction explicitly only for those cases of $\rightarrow$I where vacuous discharge

2.4.1. Keeping track of discharge

The main reason to impose the Prawitz restriction, as we see it, is that it saves on some bookkeeping. (This is discussed in [12, § I.4].) With the restriction imposed, there is no need to mark separately in a derivation which assumptions are discharged, and no need to mark what rules do the discharging work. In a Prawitz derivation, each assumption is discharged if and only if it can be, and discharged by the earliest rule that could have done the discharging.[5]

For example, take our above-presented natural deduction system. Now consider this:

$$\dfrac{\dfrac{\dfrac{p \qquad p}{p \wedge p}\wedge\mathrm{I}}{p \to p \wedge p}{\to}\mathrm{I}}{p \to p \to p \wedge p}{\to}\mathrm{I}$$

If this is to be understood as a Prawitz derivation, both assumptions of p must in fact be discharged—despite the fact that these occurrences of $\to$I allow for vacuous discharges. This is because the Prawitz restriction requires every rule to discharge every assumption it can. Since these occurrences of $\to$I introduce formulas with antecedent p, they can discharge assumptions of p; and so they must discharge any such assumptions not already discharged. This means, in addition, that both assumptions of p must be discharged by the *upper* instance of $\to$I. The lower instance, then, does feature vacuous discharge, since by the time it is reached there are no further open assumptions.

It is the Prawitz restriction that allows us to conclude all this from the structure above. Without the Prawitz restriction in place, there are

---

would be permissible; and [20] does not state any explicit policy, but on p. 315 includes discussion that seems to require the Prawitz restriction. We (tentatively) think it's probably best to interpret these sources too as imposing the restriction.

[5] An anonymous referee suggests that another motivation for the Prawitz restriction might come from searching for derivations of a given sequent, because the restriction 'allows for faster breakdown in the complexity of sequents for which proofs are being sought'.

However, we think that imposing the Prawitz restriction simply cannot be an aid to finding derivations of a given sequent. Any derivation-search strategy that succeeds in finding a Prawitz derivation thereby succeeds in finding a derivation. So any strategy that works in the presence of the Prawitz restriction will work exactly as well in its absence.

options. Since these uses of →I both allow vacuous discharge, each assumption of p might be discharged by the upper →I, by the lower →I, or not at all; and these choices can be made independently. This means that the above display, read as containing no information about discharges, corresponds to nine distinct derivations.[6]

Working in systems without the Prawitz restriction, then, more bookkeeping is needed to indicate which assumptions are discharged and which are not, and to indicate which rules do the discharging. Our convention is a usual one: every occurrence of a discharging rule in a derivation must be annotated with a distinct numeral, and every discharged assumption in a derivation must appear surrounded by [square brackets] and annotated with the numeral of the rule that discharged it.

Using this convention, we could indicate the Prawitz derivation described above like so:

$$\dfrac{\dfrac{\dfrac{[p]^1 \qquad [p]^1}{p \wedge p}\,\wedge\mathrm{I}}{p \to (p \wedge p)}\,{\to}\mathrm{I}^1}{p \to p \to (p \wedge p)}\,{\to}\mathrm{I}^2$$

However, we can also use this convention to indicate non-Prawitz derivations, for example this one:

$$\dfrac{\dfrac{\dfrac{[p]^2 \qquad [p]^1}{p \wedge p}\,\wedge\mathrm{I}}{p \to p \wedge p}\,{\to}\mathrm{I}^1}{p \to p \to p \wedge p}\,{\to}\mathrm{I}^2$$

Indeed, one of the key reasons we do not impose the Prawitz restriction is because we want to study derivations like this latter example. Already, though, we can see one important effect of the restriction on Tennant's own natural deduction systems: the property of *being a Prawitz derivation* is not closed under substitution of arbitrary formulas for atomic formulas. To see this, return to the most recent displayed derivation, the non-Prawitz

[6] According to some conventions, this display would be read as *containing* the information that no discharges have occurred, thus picking out a particular one of these nine.

one, and note that it is a substitution instance (substituting p for q) of the following derivation, which is Prawitz:

$$\dfrac{\dfrac{\dfrac{[p]^2 \qquad [q]^1}{p \wedge q}\wedge\mathrm{I}}{q \to p \wedge q}{\to}\mathrm{I}^1}{p \to q \to p \wedge q}{\to}\mathrm{I}^2$$

By dropping the Prawitz restriction, we ensure that our derivations are closed under substitutions. We will look at some other reasons for dropping this restriction in Sections 3.5 and 5.1.

2.4.2. Prawitz derivations and Prawitz derivability

Before moving on, we pause to explore the effects of the Prawitz restriction on derivability and on core derivability.[7] It turns out that for simple derivability, imposing the Prawitz restriction or not makes no difference:

Proposition 2.1. If a sequent has a derivation, it has a Prawitz derivation.

Proof: Take a sequent with a derivation D. If D itself is Prawitz, we're done. If D is not Prawitz, suppose that all of D's proper subderivations are Prawitz. (By induction on D, it is enough to consider this situation only.)

For example, suppose D ends in an application of $\to$I:

$$\dfrac{\begin{matrix}[\varphi]^n \\ \vdots \\ \mathfrak{C}\end{matrix}}{\varphi \to \psi}{\to}\mathrm{I}^n$$

If D is not Prawitz, but all its proper subderivations are, then this final $\to$I leaves some assumptions of φ undischarged. D is then a derivation of $\varphi, \Gamma \succ \varphi \to \psi$, for some set Γ that does not contain φ. By modifying D to discharge all open assumptions of φ at this final step, we reach a Prawitz derivation D' of $\Gamma \succ \varphi \to \psi$. We can then extend D' as follows (with fresh discharge numerals m, o):

[7] Thanks to an anonymous referee for encouraging us to develop this material.

$$\dfrac{\dfrac{\begin{matrix} D' \\ \varphi \to \psi \end{matrix}}{\varphi \to \varphi \to \psi}\,{\to}\mathrm{I}^m \qquad \varphi \qquad [\psi]^o}{\varphi \to \psi}\,{\to}\mathrm{E}^o$$

Note that the discharge labeled m is vacuous, as we know that there are no open assumptions of φ in D'. This resulting derivation is Prawitz, and is a derivation of $\varphi, \Gamma \succ \varphi \to \psi$, just as D itself was.

This strategy works in general: if D is not Prawitz at its final rule occurrence, it must be because this occurrence leaves some assumption open that it could have discharged. So we first modify D to a Prawitz D' that does discharge everything it can at this final step, and then use →I and →E in tandem to restore the needed open assumptions. □

So removing the Prawitz restriction has no effect on which sequents are derivable, and thus no effect on provability or refutability. Since derivability itself is closed under substitutions, then, it follows that Prawitz derivability is also closed under substitutions, even though the property of being a Prawitz derivation is not.

The strategy adopted in the above proof, however, produces non-core derivations, even starting from a core derivation. And indeed, the situation is different when it comes to core derivability: there are sequents that have core derivations but no Prawitz core derivations. For example, consider $p \succ p \to p \wedge p$; this has the following core derivation:

$$\dfrac{\dfrac{p \qquad [p]^1}{p \wedge p}\,\wedge\mathrm{I}}{p \to p \wedge p}\,{\to}\mathrm{I}^1$$

It does not, however, have any Prawitz core derivation. To see this, note that any core derivation of $p \succ p \to p \wedge p$ must end in a step of →I; no elimination rule is possible as a last step, since the major premise of that elimination rule would have to be an open assumption, and p cannot stand as a major premise of any elimination rule. This final step of →I, however, is able to discharge any open assumptions of p in the derivation, so in a Prawitz derivation it must do so; p cannot stand as an open assumption at the end of such a derivation. Accordingly, there is no Prawitz core derivation of $p \succ p \to p \wedge p$.

So imposing the Prawitz restriction or not *does* make a difference as to which sequents are core derivable. Moreover, Prawitz core derivability is not closed under substitution: witness the following Prawitz core derivation of $p \succ q \to p \wedge q$.

$$\dfrac{\dfrac{p \qquad [q]^1}{p \wedge q}\ \wedge\mathrm{I}}{q \to p \wedge q}\ {\to}\mathrm{I}^1$$

Since Tennant's own version of core logic imposes the Prawitz restriction, then, it is not closed under substitutions. However, our liberalized version, which does not impose the Prawitz restriction, is.

3. Terms and reductions

Here, we define a language of terms, and consider reduction relations on these terms. The motivating idea is to develop, for the above natural deduction system, a term calculus that corresponds to it in the usual Curry-Howard way, the way that the calculus of [8] corresponds to a more usual intuitionistic natural deduction system. (This work is begun in [13], which explores the $\neg, \to$ fragment of core logic in this way; this section extends that work to take account of $\wedge, \vee$ as well.) The usual Curry-Howard correspondence allows us to see intuitionistic proofs as programs in a simply-typed lambda calculus, and reduction on proofs as execution of those programs. Similarly, the system presented here allows us to see derivations in the above-presented proof system as programs, and reduction of those derivations as execution.[8]

Our *types* for this system are the formulas of our language. *Hats* are as before: a hat is either a type or ☹.

3.1. Terms and eliminators

We use a mutual induction to define terms, eliminators, and the free variables in a term or eliminator. We use M, N, O, etc for terms; each term M wears a hat $\mathfrak{C}$, indicated as $M^{\mathfrak{C}}$. Every term is either *typed* or *exceptional*, according to its hat: if its hat is a type, the term is typed; and if its hat is ☹, the term is exceptional. We use $\mathcal{E}, \mathcal{F}$, etc for eliminators;

[8] For background and details, see for example [6, 14].

each eliminator $\mathcal{E}$ wears both a type φ and a separate hat $\mathfrak{C}$, indicated as ${}_{\varphi}\mathcal{E}^{\mathfrak{C}}$. We sometimes have use for metavariables that can be either terms or eliminators; for this purpose we use $\mathbb{X}, \mathbb{Y}$, etc. For every *type* φ we assume denumerably many variables x^{φ}, y^{φ}, etc; there are no variables with hat ☹. For any term or eliminator $\mathbb{X}$ there is a set $\mathtt{FV}(\mathbb{X})$ of variables that are $\mathbb{X}$'s *free variables.*

Definition 3.1. (Terms and eliminators)
Terms:

- All variables are terms; for any variable x, we have $\mathtt{FV}(x) = \{x\}$.
- For any terms M^{φ} and N^{ψ}, there is a term $\langle M, N\rangle^{\varphi\wedge\psi}$. We have $\mathtt{FV}(\langle M, N\rangle) = \mathtt{FV}(M) \cup \mathtt{FV}(N)$.
- For any term M^{φ} and type ψ, there are terms $(\mathsf{inl}(M))^{\varphi\vee\psi}$ and $(\mathsf{inr}(M))^{\psi\vee\varphi}$. We have $\mathtt{FV}(\mathsf{inl}(M)) = \mathtt{FV}(\mathsf{inr}(M)) = \mathtt{FV}(M)$.
- For any term $M^{☹}$ with $x^{\varphi} \in \mathtt{FV}(M)$, there is a term $(\lambda^{\neg}x.M)^{\neg\varphi}$, and in addition for each type ψ a term $(\lambda^{\rightarrow}x.M)^{\varphi\rightarrow\psi}$. We have $\mathtt{FV}(\lambda^{\neg}x.M) = \mathtt{FV}(\lambda^{\rightarrow}x.M) = \mathtt{FV}(M) \setminus \{x\}$.
- For any term M^{ψ} and variable x^{φ}, there is a term $(\lambda^{\rightarrow}x.M)^{\varphi\rightarrow\psi}$. Again, $\mathtt{FV}(\lambda^{\rightarrow}x.M) = \mathtt{FV}(M) \setminus \{x\}$.
- For any term M^{φ} and eliminator ${}_{\varphi}\mathcal{E}^{\mathfrak{C}}$, there is a term $(M\mathcal{E})^{\mathfrak{C}}$. We have $\mathtt{FV}(M\mathcal{E}) = \mathtt{FV}(M) \cup \mathtt{FV}(\mathcal{E})$.

Eliminators:

- For any term $N^{\mathfrak{C}}$ with $\{x^{\varphi}, y^{\psi}\} \cap \mathtt{FV}(M) \neq \emptyset$, there is an eliminator ${}_{\varphi\wedge\psi}(\!|\langle x, y\rangle.N|\!)^{\mathfrak{C}}$. We have $\mathtt{FV}((\!|\langle x, y\rangle.N|\!)) = \mathtt{FV}(N) \setminus \{x, y\}$.
- For any terms $N^{\mathfrak{C}}$ and $O^{\mathfrak{D}}$ with $x^{\varphi} \in \mathtt{FV}(N)$ and $y^{\psi} \in \mathtt{FV}(O)$, such that either $\mathfrak{C} \leq \mathfrak{D}$ or $\mathfrak{D} \leq \mathfrak{C}$, there is an eliminator ${}_{\varphi\vee\psi}(\!|x.N, y.O|\!)^{\max(\mathfrak{C},\mathfrak{D})}$. We have $\mathtt{FV}((\!|x.N, y.O|\!)) = (\mathtt{FV}(N) \setminus \{x\}) \cup (\mathtt{FV}(O) \setminus \{y\})$.
- For any terms N^{φ} and $O^{\mathfrak{C}}$ with $x^{\psi} \in \mathtt{FV}(O)$, there is an eliminator ${}_{\varphi\rightarrow\psi}(\!|N, x.O|\!)^{\mathfrak{C}}$. We have $\mathtt{FV}((\!|N, x.O|\!)) = \mathtt{FV}(N) \cup (\mathtt{FV}(O) \setminus \{x\})$.
- For any term N^{φ}, there is an eliminator ${}_{\neg\varphi}(\!|N|\!)^{☹}$. We have $\mathtt{FV}((\!|N|\!)) = \mathtt{FV}(N)$.

All terms and eliminators are identified up to change in bound variables, and we make free use of this identification without further comment. As you may have noticed in the above definition, we often omit hats, either where they can be inferred or where we are generalizing.

By comparing the above definitions to the natural deduction system, you can see the following correspondences:

Open assumption of φ	Free variable of type φ
Discharging an assumption of φ	Binding a variable of type φ
Derivation of the sequent $\Gamma \succ \mathfrak{C}$	Term $M^{\mathfrak{C}}$ with $\mathtt{FV}(M)$ having types in Γ

Let's look at two examples, to get the flavour. First, our earlier proof of $\neg p, p \succ q$:

$$\dfrac{\dfrac{\dfrac{\neg p \qquad [p]^1}{\frownie}\neg\mathrm{E}}{p \to q}{\to}\mathrm{I}^1 \qquad p \qquad [q]^2}{q}{\to}\mathrm{E}^2$$

We can annotate this derivation as follows:

$$\dfrac{\dfrac{\dfrac{w : \neg p \qquad [x : p]^1}{w\llparenthesis x \rrparenthesis : \frownie}\neg\mathrm{E}}{\lambda^{\to} x.w\llparenthesis x \rrparenthesis : p \to q}{\to}\mathrm{I}^1 \qquad y : p \qquad [z : q]^2}{(\lambda^{\to} x.w\llparenthesis x \rrparenthesis)\llparenthesis y, z.z \rrparenthesis : q}{\to}\mathrm{E}^2$$

This derivation thus corresponds to the term $(\lambda^{\to} x.w\llparenthesis x \rrparenthesis)\llparenthesis y, z.z \rrparenthesis$, which, fully spelled out with all hats visible, is

$$(\lambda^{\to} x^p.(w^{\neg p}\llparenthesis x^p \rrparenthesis)^{\frownie})^{p\to q}({}_{p\to q}\llparenthesis y^p, z^q.z^q \rrparenthesis^q)^q.$$

Second, our earlier example of a derivation that violates the Prawitz restriction:

$$\dfrac{\dfrac{\dfrac{[p]^2 \qquad [p]^1}{p \wedge p}\wedge\mathrm{I}}{p \to (p \wedge p)}{\to}\mathrm{I}^1}{p \to p \to (p \wedge p)}{\to}\mathrm{I}^2$$

We can annotate this derivation as follows:

$$\dfrac{\dfrac{\dfrac{[x:p]^2 \qquad [y:p]^1}{\langle x,y\rangle : p\wedge p}\wedge\mathrm{I}}{\lambda^{\to}y.\langle x,y\rangle : p\to(p\wedge p)}\to\mathrm{I}^1}{\lambda^{\to}x.\lambda^{\to}y.\langle x,y\rangle : p\to p\to(p\wedge p)}\to\mathrm{I}^2$$

This derivation thus corresponds to the term $(\lambda^{\to}x.\lambda^{\to}y.\langle x,y\rangle)$, which, fully spelled out, is $(\lambda^{\to}x^p.(\lambda^{\to}y^p.(\langle x^p,y^p\rangle)^{p\wedge p})^{p\to p\wedge p})^{p\to p\to p\wedge p}$. Hopefully it is by now apparent why we often suppress hats where they are not needed!

3.2. Terminology

Terms of the form $\langle M,N\rangle$, $\mathsf{inl}(M)$, $\mathsf{inr}(M)$, $\lambda^{\to}x.M$, or $\lambda^{\neg}x.M$ are *introductions*. Terms of the form $M\mathcal{E}$ are *eliminations*. So every term is a variable, an introduction, or an elimination.

Variables have no *immediate subterms*. The immediate subterms of an introduction or an eliminator are what you'd expect. (For example, the immediate subterms of $(\!|N,x.O|\!)$ are N and O.) The immediate subterms of an elimination $M\mathcal{E}$ are M and the immediate subterms of $\mathcal{E}$. The subterm relation is the reflexive transitive closure of the immediate subterm relation.

All immediate subterms of an eliminator are *minor* subterms of that eliminator. In eliminators of the form $(\!|\langle x,y\rangle.N|\!)$ or $(\!|x.N,y.O|\!)$, these minor subterms are also *commuting* subterms. In eliminators of the form $(\!|N,x.O|\!)$, only O is a commuting subterm. And in eliminators of the form $(\!|N|\!)$, there are no commuting subterms. The minor and commuting subterms of an elimination $M\mathcal{E}$ are those of the eliminator $\mathcal{E}$. The *major* subterm of an elimination $M\mathcal{E}$ is M. Note that every immediate subterm of an elimination is either major or minor.

3.3. Composition of eliminators

Given two eliminators ${}_{\varphi}\mathcal{E}^{\psi}$ and ${}_{\psi}\mathcal{F}^{\mathfrak{C}}$, the eliminator ${}_{\varphi}(\!|\mathcal{E}\mathcal{F}|\!)^{\mathfrak{C}}$ is the eliminator like $\mathcal{E}$, but with each commuting subterm P of $\mathcal{E}$ replaced with $P\mathcal{F}$.[9] For example, if $\mathcal{E}$ is ${}_{\varphi\to\psi}(\!|N^{\varphi},x.O^{\theta\wedge\rho}|\!)^{\theta\wedge\rho}$ and $\mathcal{F}$ is ${}_{\theta\wedge\rho}(\!|\langle y,z\rangle.P^{\mathfrak{C}}|\!)^{\mathfrak{C}}$, then $(\!|\mathcal{E}\mathcal{F}|\!)$ is $(\!|N,x.O\mathcal{F}|\!)$. As the commuting subterms of an eliminator always wear the same hat as the eliminator's right (output) hat, this is well-defined.

[9] Change to bound variables in $\mathcal{E}$ might be needed here to avoid capturing any variables free in $\mathcal{F}$.

3.4. Substitution

Capture-avoiding substitution of terms for variables in this calculus works as it does in similar calculi; there's nothing particularly remarkable about it. We pause to go through the details nonetheless; many aspects of core type theory do *not* work as usual, so it's worth checking the details even of those aspects that do.

Where $x_1^{\varphi_1}, \dots, x_n^{\varphi_n}$ are distinct variables and $N_1^{\varphi_1}, \dots, N_n^{\varphi_n}$ terms of corresponding types, then $[x_1 \mapsto N_1, \dots, x_n \mapsto N_n]$ is a *substitution*. (Note that all substitutions are finite.) Given a substitution σ, the substitution $\sigma^{\downarrow y}$ is just like σ except that it does not substitute anything for the variable y. That is, $[x_1 \mapsto N_1, \dots, x_n \mapsto N_n]^{\downarrow x_i}$ is $[x_1 \mapsto N_1, \dots, x_{i-1} \mapsto N_{i-1}, x_{i+1} \mapsto N_{i+1}, \dots, x_n \mapsto N_n]$; and $[x_1 \mapsto N_1, \dots, x_n \mapsto N_n]^{\downarrow y}$ is just $[x_1 \mapsto N_1, \dots, x_n \mapsto N_n]$ if y is not one of the x_is. Say that a variable y is *free* in $[x_1 \mapsto N_1, \dots, x_n \mapsto N_n]$ iff it is free in some N_i; and say that y is *acted on* by $[x_1 \mapsto N_1, \dots, x_n \mapsto N_n]$ iff it is one of the x_i.

Given a term or eliminator, capture-avoiding substitution works as usual:

- $x_i[x_1 \mapsto N_1, \dots, x_n \mapsto N_n] = N_i$;
- $y[x_1 \mapsto N_1, \dots, x_n \mapsto N_n] = y$, where y is not one of the x_is;
- $\langle M, N\rangle\sigma = \langle M\sigma, N\sigma\rangle$;
- $\mathsf{inl}(M)\sigma = \mathsf{inl}(M\sigma)$; $\mathsf{inr}(M)\sigma = \mathsf{inr}(M\sigma)$;
- $(\lambda^{\rightarrow} y.M)\sigma = \lambda^{\rightarrow} y.(M\sigma^{\downarrow y})$, assuming y is not free in σ;[10]
- $(\lambda^{\neg} y.M)\sigma = \lambda^{\neg} y.(M\sigma^{\downarrow y})$, assuming y is not free in σ;
- $(M\mathcal{E})\sigma = (M\sigma)(\mathcal{E}\sigma)$;
- ${}_{\neg\varphi}(\!|M|\!)\sigma = {}_{\neg\varphi}(\!|M\sigma|\!)$;
- ${}_{\varphi\wedge\psi}(\!|\langle x, y\rangle.M|\!)\sigma = {}_{\varphi\wedge\psi}(\!|\langle x, y\rangle.M\sigma^{\downarrow x \downarrow y}|\!)$, assuming neither x nor y is free in σ;

[10] Recall that we identify terms up to change of bound variable. So if y *is* free in σ, we first change the bound variable y in $\lambda^{\rightarrow} y.M$ to some variable that is *not* free in σ. (Since all substitutions are finite, there is always some such.) All similar assumptions in this definition should be read the same way.

- ${}_{\varphi\vee\psi}(\!|x.M, y.N|\!)\sigma = {}_{\varphi\vee\psi}(\!|x.N\sigma^{\downarrow x}, y.O\sigma^{\downarrow y}|\!)$, assuming neither x nor y is free in σ; and
- ${}_{\varphi\to\psi}(\!|M, x.N|\!)\sigma = {}_{\varphi\to\psi}(\!|M\sigma, x.N\sigma^{\downarrow x}|\!)$, assuming x is not free in σ.

Note two things: first that, since there are no variables with hat ☹, that $M[x \mapsto N^{☹}]$ is never defined; and second that substitution never affects hats: that is, the hat on $M^{\mathfrak{C}}[x \mapsto N]$ is always exactly $\mathfrak{C}$.

Substitution interacts pleasantly with composition of eliminators:

LEMMA 3.2. *Given eliminators $\mathcal{E}$ and $\mathcal{F}$ such that $(\!|\mathcal{E}\mathcal{F}|\!)$ is defined, and a substitution σ, the eliminator $(\!|(\mathcal{E}\sigma)(\mathcal{F}\sigma)|\!)$ is $(\!|\mathcal{E}\mathcal{F}|\!)\sigma$.*

PROOF: Unpacking definitions. □

3.5. The Prawitz restriction on terms

Recall that the Prawitz restriction on derivations requires that when any rule application in a derivation *can* discharge any open assumption, it *must* discharge that open assumption. The corresponding restriction on terms is this: that whenever a component of a term binds a variable of type φ, it binds *all* free variables of type φ in its scope. Equivalently, the Prawitz restriction corresponds to a term system with a *single* variable of each type, rather than the denumerably many variables of each type that we have assumed.[11]

We noted in Section 2.4 that there are many derivations in our system that do not obey the Prawitz restriction, such as the derivation repeated here:

$$\dfrac{\dfrac{\dfrac{[p]^2 \qquad [p]^1}{p \wedge p}\,\wedge\text{I}}{p \to p \wedge p}\,{\to}\text{I}^1}{p \to p \to p \wedge p}\,{\to}\text{I}^2$$

This derivation corresponds to the term $(\lambda^{\to} x^p.\lambda^{\to} y^p.(\langle x, y\rangle)^{p\wedge p})^{p\to p\to p\wedge p}$. This term requires two distinct variables of type p. This is because $\lambda^{\to} y$

[11] Term systems like this are not often explored, because they do not allow for a definition of capture-avoiding substitution; our definition in Section 3.4, like other definitions, relies crucially on being able to draw on fresh variables of a given type to avoid clashes between free and bound variables. (As we will see in Section 5.1, this interference with substitution also blocks strong normalization.)

must bind the y in $\langle x^p, y^p \rangle$ *without* binding the x, so that the outer $\lambda^{\rightarrow}x$ can bind the x instead.

This brings us to the main reason we've chosen to go without the Prawitz restriction: the terms it excludes include terms with natural and important computational behaviour. The term $\lambda^{\rightarrow}x.\lambda^{\rightarrow}y.\langle x, y \rangle$ is a very simple pairing function, a function that takes inputs x and y and returns their ordered pair.[12] Imposing the Prawitz restriction would allow us to define this function only in the case where the two inputs have distinct types, but it is also perfectly natural to want to pair up two pieces of data that have the same type.

Indeed, the Prawitz restriction prevents us from defining *any* functions that take multiple inputs of the same type: the binding required for the final input is required by the Prawitz restriction to bind all free variables of that type; any outer bindings of that same type turn out vacuous. It would be impossible, for example, to build basic arithmetic on the Church numerals (see [7, Ch. 4]) in a system obeying the Prawitz restriction, since this requires defining addition and multiplication functions, each of which takes two inputs of the same (numeric) type.

We take it, then, that most standard term systems work without the Prawitz restriction for good reason, and so we develop core type theory without any such restriction.

4. Reduction

In this section, we define two relations of *reduction* on terms of our calculus: what we call *principal reduction* and *full reduction*. The difference is that full reduction includes commuting conversions; principal reduction does not. We then prove a number of lemmas about these reduction relations, in the leadup to Section 5, where we prove that principal reduction is strongly normalizing. We conjecture that full reduction is also strongly normalizing, but leave that question for future work.

4.1. Redexes and reducts

Both reduction relations are defined by identifying a class of special terms called *redexes*, and assigning to each redex a term called its *reduct*. The

[12] This is the function written `(,)` in Haskell, for example.

difference between principal reduction and full reduction is entirely in which terms are redexes. Then, given a chosen notion of redex, for any term M that contains a redex R as a subterm, we define a specific term as the *one-step reduction of M at R*. The move from redexes to one-step reduction is very much *not* as usual; this is one of the more distinctive features of core type theory, and it is a key motivation of this work to explore this nonstandard notion. Let's dive in.

4.1.1. Principal redexes

The following table displays the forms of all *principal redexes* and their corresponding reducts.

Redex	Reduct
$\langle M, N \rangle (\!| \langle x, y \rangle . O |\!)$	$O[x \mapsto M, y \mapsto N]$
$\mathsf{inl}(M)(\!| x.N, y.O |\!)$	$N[x \mapsto M]$
$\mathsf{inr}(M)(\!| x.N, y.O |\!)$	$O[y \mapsto M]$
$(\lambda^{\rightarrow} x.(M^{\psi}))(\!| N, y.O |\!)$	$O[y \mapsto M[x \mapsto N]]$
$(\lambda^{\rightarrow} x.(M^{\frown}))(\!| N, y.O |\!)$	$M[x \mapsto N]$
$(\lambda^{\neg} x.M)(\!| N |\!)$	$M[x \mapsto N]$

In defining *principal reduction*, all and only the principal redexes count as redexes.

4.1.2. Commuting redexes

Any term of the form $(M\mathcal{E})\mathcal{F}$ is a *commuting redex*; its reduct is $M(\!|\mathcal{E}\mathcal{F}|\!)$. Note that $(\!|\mathcal{E}\mathcal{F}|\!)$ is defined, and $M(\!|\mathcal{E}\mathcal{F}|\!)$ well-formed, whenever $(M\mathcal{E})\mathcal{F}$ is well-formed. Note as well that no commuting redex is a principal redex, so given a redex (of either kind), the reduct of that redex is unambiguously determined. In defining *full reduction*, both principal redexes and commuting redexes count as redexes.

Since we focus on principal reduction rather than full reduction in Section 5, we don't linger specifically on commuting redexes. However, the definitions and lemmas in this section don't care about the difference; when we speak of 'reduction' unqualified, we are making a definition or claim that applies to both principal and full reduction.[13]

4.2. One-step reduction

Using these redexes and their reducts, we define a relation of *one-step reduction* between terms. (Since we have two different choices for what counts as a redex—principal only or principal plus commuting—we end up with two different choices for a one-step reduction relation: principal or full.) Given any term that contains an occurrence of a redex at a subterm, we define the unique result of reducing that term at that redex occurrence. That much is as usual for term systems like this.

However—and this is not usual—reduction in this system is not a *compatible* relation. That is, we do not always simply replace a redex with its reduct in place, leaving its context alone. Such a procedure could not work in core type theory. The reason is that the result of such a procedure is not always well-formed in this system.

For example, consider the redex $((\lambda^{\to} y^{\varphi}.x^{\psi})w^{\varphi})^{\psi}$ with reduct x^{ψ} as it occurs in the term $(\lambda^{\to} w.(z^{\neg\psi}(\!|(\lambda^{\to} y.x)w|\!))^{\circledcirc})^{\varphi\to\theta}$. Replacing this redex with its reduct would yield $(\lambda^{\to} w.(z^{\neg\psi}(\!|x^{\psi}|\!))^{\circledcirc})^{\varphi\to\theta}$. This latter, however, is not a term, as it violates a restriction on $\lambda^{\to}$, which may not bind w vacuously in this situation. (This restriction corresponds to the restrictions against certain cases of vacuous discharge in the rule $\to$I.)

This is an example of the following. Many of our formation rules (in the above example, using $\lambda^{\to}$ to bind into an exceptional term) require certain variables to appear free; but some redexes, because they themselves involve vacuous binding, contain free variables that are not contained in their reducts. That is, core type theory allows vacuous binding in some

[13]There are two more potential sources of redexes that might come to mind, although we use neither in this paper.

First, uses of an explosion rule like typical $\perp$E in natural deduction systems create possible violations of the subformula property, and so reduction steps are sometimes introduced to prevent these violations, as in [12, p. 40]. However, core logic contains no such explosion rules, so no such reduction steps are needed or even possible.

Second, [18] considers a type of reduction there called 'shrinking', which in effect allows a one-step reduction directly from $M^{\mathfrak{C}}$ to $N^{\mathfrak{C}}$ whenever N is a subterm of M. This makes havoc for computational interpretations of the term language, for reasons discussed in [11]; we leave it aside here.

circumstances but not all, and it is the interaction between these circumstances that creates the phenomenon of interest.[14]

For a different kind of example, consider the redex

$$((\lambda^{\to} y^{\varphi}.(z^{\neg\varphi} y)^{\circledcirc})^{\varphi\to\psi}(\!|x^{\varphi}, w^{\varphi}.w|\!))^{\psi}$$

with redex $(zx)^{\circledcirc}$ as it occurs in the term $(\langle(\lambda^{\to} y.zy)(\!|x, w.w|\!), v^{\theta}\rangle)^{\psi\wedge\theta}$. Replacing this redex with its reduct would yield $\langle (zx)^{\circledcirc}, w\rangle$. This latter, however, is not a term, as the constructor $\langle\ ,\ \rangle$ requires two *typed* subterms, and $(zx)^{\circledcirc}$ is exceptional. This corresponds to the rule $\wedge$I's requiring formulas as premises.

This is an example of a different kind of phenomenon. Many of our formation rules for terms (in the above example, using $\langle\ ,\ \rangle$) require terms to be typed; but some redexes are typed and yet have exceptional reducts. Reducing such a redex in place, then, yields a nonsensical result.

The troubles with reducing in place, then, are twofold: moving from a redex to its reduct can drop free variables, and it can move from a typed term to an exceptional one. But these reductions can happen in places where free variables or types are required. Leaving everything else in place, then, won't do in general. In what follows, we show how to handle these problems. We start by noting two important facts about redexes and their reducts: for any redex $R^{\mathfrak{C}}$ with reduct $R'^{\mathfrak{D}}$, we always have $\mathtt{FV}(R') \subseteq \mathtt{FV}(R)$ and $\mathfrak{D} \leq \mathfrak{C}$. That is, free variables and hats do not always remain constant between a redex and its reduct, but they cannot change freely; when there is a change, it is always in the same direction. We repeatedly use this constraint—which is the term-level reflection of epistemic gain—in what follows.

Basically, our strategy works like this: where we can get away with reducing in place, leaving the immediate context alone, that's what we do. Where the result would not be well-formed, we simply drop the immediate context altogether. That's the intuition, anyhow; here's the precise definition of one-step reduction.

---

[14] Contrast a usual simply-typed lambda calculus, where vacuous binding is always allowed; but also contrast the lambda calculus of [3], standardly now called the λI calculus, where vacuous binding is never allowed; also see [2, Ch. 9]. In this calculus, redexes and their corresponding reducts always have exactly the same free variables (see [2, Lemma 9.1.2]), so any nonvacuous binding into a redex remains nonvacuous into its reduct.

Definition 4.1 (One-step reduction). First, if R is a redex and S its reduct, then R reduces to S in one step; as we write, $R \leadsto_1 S$. The rest of the definition contains a number of conditions. These are expressed in the form:

$$\frac{\mathbb{X} \leadsto_1 \mathbb{Y}}{\mathbb{Z} \leadsto_1 \mathbb{W}}$$

Here is how such a condition should be read. We only apply it if $\mathbb{X}, \mathbb{Y}, \mathbb{Z}$ are each well-formed, without any assumption that $\mathbb{W}$ is well-formed. Under these conditions, if $\mathbb{X} \leadsto_1 \mathbb{Y}$ *and* $\mathbb{W}$ *is well-formed*, then $\mathbb{Z} \leadsto_1 \mathbb{W}$; on the other hand, if $\mathbb{X} \leadsto_1 \mathbb{Y}$ and $\mathbb{W}$ is *not* well-formed, then $\mathbb{Z} \leadsto_1 \mathbb{Y}$ instead.

This fallback condition—that when $\mathbb{W}$ is not well-formed we have $\mathbb{Z} \leadsto_1 \mathbb{Y}$—is what gives one-step core reduction its distinctive flavour. Note that there is no indeterminism or choice introduced here: if $\mathbb{W}$ *is* well-formed we do not have $\mathbb{Z} \leadsto_1 \mathbb{Y}$ from such a condition. Only in the case that $\mathbb{W}$ is not well-formed do we fall back to $\mathbb{Z} \leadsto_1 \mathbb{Y}$. Here, then, are the conditions:

$$\frac{M \leadsto_1 M'}{M\mathcal{E} \leadsto_1 M'\mathcal{E}} \qquad \frac{\mathcal{E} \leadsto_1 \mathcal{E}'}{M\mathcal{E} \leadsto_1 M\mathcal{E}'} \qquad \frac{\mathcal{E} \leadsto_1 N}{M\mathcal{E} \leadsto_1 N}$$

$$\frac{M \leadsto_1 M'}{\langle M, N\rangle \leadsto_1 \langle M', N\rangle} \qquad \frac{N \leadsto_1 N'}{\langle M, N\rangle \leadsto_1 \langle M, N'\rangle}$$

$$\frac{M \leadsto_1 M'}{\mathsf{inl}(M) \leadsto_1 \mathsf{inl}(M')} \qquad \frac{M \leadsto_1 M'}{\mathsf{inr}(M) \leadsto_1 \mathsf{inr}(M')}$$

$$\frac{M \leadsto_1 M'}{\lambda^{\to}x.M \leadsto_1 \lambda^{\to}x.M'} \qquad \frac{M \leadsto_1 M'}{\lambda^{\neg}x.M \leadsto_1 \lambda^{\neg}x.M'}$$

$$\frac{M \leadsto_1 M'}{(\!| M |\!) \leadsto_1 (\!| M' |\!)} \qquad \frac{M \leadsto_1 M'}{(\!| \langle x, y\rangle.M |\!) \leadsto_1 (\!| \langle x, y\rangle.M' |\!)}$$

$$\frac{M \rightsquigarrow_1 M'}{(\!|M, x.N|\!) \rightsquigarrow_1 (\!|M', x.N|\!)} \qquad \frac{N \rightsquigarrow_1 N'}{(\!|M, x.N|\!) \rightsquigarrow_1 (\!|M, x.N'|\!)}$$

$$\frac{M \rightsquigarrow_1 M'}{(\!|x.M, y.N|\!) \rightsquigarrow_1 (\!|x.M', y.N|\!)} \qquad \frac{N \rightsquigarrow_1 N'}{(\!|x.M, y.N|\!) \rightsquigarrow_1 (\!|x.M, y.N'|\!)}$$

Expressed in this way, these conditions might *look* like usual reduce-in-place conditions. But recall our distinctive way of reading these, involving fallback in case the lower-right component is not well-formed; this is the key to the definition.

Since this is an unusual way to handle one-step reduction, let's look at an example. Consider the condition for $\mathsf{inl}()$, reproduced here:

$$\frac{M \rightsquigarrow_1 M'}{\mathsf{inl}(M) \rightsquigarrow_1 \mathsf{inl}(M')}$$

Suppose first that M^ψ is $(\lambda^{\rightarrow} x^\varphi . y^\psi)(\!|z, v.v|\!)$. Then M is a redex, with reduct y. So, according to the condition for $\mathsf{inl}()$, we can conclude that $\mathsf{inl}(M)^{\psi \vee \theta}$ can be reduced in one step to $\mathsf{inl}(y)$. So far, so normal.

Suppose instead, though, that M^ψ is $(\lambda^{\rightarrow} x^\varphi . y^{\neg\varphi}(\!|x|\!))(\!|z, v.v|\!)$. Then M is again a redex, now with reduct $(y(\!|z|\!))^{\text{☹}}$. By the same condition, then, $\mathsf{inl}(M)^{\psi\vee\theta}$ can be reduced. However, note that $\mathsf{inl}(y(\!|z|\!))$ is not well-formed; $\mathsf{inl}()$ can only be applied to *typed* terms, and $y(\!|z|\!)$ is exceptional. Thus, $\mathsf{inl}(M)$ cannot reduce to $\mathsf{inl}(y(\!|z|\!))$, since the latter isn't a term at all. So, according to the condition for $\mathsf{inl}()$, we conclude that $\mathsf{inl}(M)$ reduces in one step directly to $y(\!|z|\!)$.

Three important facts about one-step reduction. First, terms always reduce to terms, while eliminators sometimes reduce to eliminators and sometimes to terms. Second, if $M^{\mathfrak{C}} \rightsquigarrow_1 N^{\mathfrak{D}}$, then $\mathfrak{D} \leq \mathfrak{C}$. Finally, if $M \rightsquigarrow_1 N$, then $\mathtt{FV}(N) \subseteq \mathtt{FV}(M)$. (All these can be shown by induction on the above definition.)

Let's look at an example that demonstrates some of these complexities. Consider the term $M^{\neg(\varphi\wedge\psi)} = (\lambda^{\neg} x^{\varphi\wedge\psi} . (w^{\neg\theta}(\!|x(\!|\langle y^\varphi, z^\psi\rangle . (\lambda^{\rightarrow} v^\varphi . u^\theta) y^\varphi|\!)|\!))^{\text{☹}})$. The free variables of this term are $w^{\neg\theta}$ and u^θ, and so this term corresponds to a derivation of the sequent $\neg\theta, \theta \succ \neg(\varphi\wedge\psi)$. It contains a redex $(\lambda^{\rightarrow} v.u)y$ with reduct u, inside the eliminator $(\!|\langle y, z\rangle . (\lambda^{\rightarrow} v.u)y|\!)$. Let's go through the one-step reduction of M at this redex.

First, we note that $(\!|\langle y,z\rangle.u|\!)$ is not well-formed, since a conjunction eliminator cannot bind fully vacuously; so we reduce $(\!|\langle y,z\rangle.(\lambda^{\to}v.u)y|\!)$ directly to u itself. Having done this, we note that $x^{\varphi\wedge\psi}u^{\theta}$ is also not well-formed; no rule allows us to juxtapose two terms at all. So we reduce $x(\!|\langle y,z\rangle.(\lambda^{\to}v.u)y|\!)$ also directly to u. The next two layers do work in place, so we reduce $w(\!|x(\!|\langle y,z\rangle.(\lambda^{\to}v.u)y|\!)|\!)$ to $w(\!|u|\!)$. The final layer, however, runs into trouble again; as x is not free in $w(\!|u|\!)$, the binder $\lambda^{\neg}x$ may not bind into $w(\!|u|\!)$. So M itself reduces to $(w(\!|u|\!))^{\circledcirc}$. Although we have here worked through this reduction layer by layer, we emphasize that this is *one-step* reduction; this is the result of reducing a single term at a single redex.

4.3. Reduction concepts

DEFINITION 4.2 (Reduction paths). Given a relation $\rightsquigarrow_1$ of one-step reduction, a *reduction path from* $\mathbb{X}$ is a sequence (finite or infinite) $\mathbb{X}_0,\ldots,\mathbb{X}_n,\ldots$ such that $\mathbb{X}_0=\mathbb{X}$, and for each n, $\mathbb{X}_n \rightsquigarrow_1 \mathbb{X}_{n+1}$. For a finite reduction path $\mathbb{X}_0,\ldots,\mathbb{X}_n$, we say it is a reduction path *from* $\mathbb{X}_0$ *to* $\mathbb{X}_n$, and its length is the number n of reduction steps in it.

DEFINITION 4.3 (Normal, strongly normalizing). A term or eliminator is *normal* iff all reduction paths from it have length 0. A term or eliminator is *strongly normalizing* iff all reduction paths from it are finite.

If a term M is strongly normalizing, then $|M|$ is the length of its longest reduction path. (If M is not strongly normalizing, $|M|$ is not defined.) We also define $|\mathcal{E}|$ for eliminators $\mathcal{E}$, but slightly differently: $|\mathcal{E}|$ is the total of all $|N|$ for $\mathcal{E}$'s immediate subterms N, and is undefined if any such $|N|$ is undefined.

DEFINITION 4.4 (Multistep reductions). We say $\mathbb{X}$ *reduces to* $\mathbb{Y}$, written $\mathbb{X}\rightsquigarrow\mathbb{Y}$, iff there is a (necessarily finite) reduction path from $\mathbb{X}$ to $\mathbb{Y}$. We say $\mathbb{X}$ *properly reduces* to $\mathbb{Y}$, written $\mathbb{X}\rightsquigarrow^{+}\mathbb{Y}$, iff there is a reduction path from $\mathbb{X}$ to $\mathbb{Y}$ with length at least 1.

Note, now by induction on reduction paths, that if $M^{\mathfrak{C}}\rightsquigarrow N^{\mathfrak{D}}$ (and so also if $M\rightsquigarrow^{+}N$), then $\mathfrak{D}\leq\mathfrak{C}$ and $\mathtt{FV}(N)\subseteq\mathtt{FV}(M)$.

Since we have two different notions of reduction in view (principal and full), we also have two different notions of normal form, strongly normalizing, etc. It's worth pausing here to think a bit about relations between these. Since full reduction is defined in terms of all the principal redexes (and then some), we have that any principal reduction path is also a full reduction path. This gives us that any term in full normal form is also in principal normal form, and that any term that is fully strongly normalizing is also principally strongly normalizing.[15]

We also note that the full normal forms are exactly the *core* terms. Corresponding to our definition of core derivations, we say that a term is *core* iff in all its subterms of the form $M\mathcal{E}$, the term M is a variable. This is also what it takes to be a full normal form: M is an introduction iff $M\mathcal{E}$ is a principal redex, and M is an elimination iff $M\mathcal{E}$ is a commuting redex.

4.4. Reduction lemmas

Here we prove a number of facts about reduction, and about interactions between reduction and substitution, that will be used in Section 5. These facts hold for both principal and full reduction.

LEMMA 4.5. *All the clauses of Definition 4.1 hold as well for $\rightsquigarrow$. That is, where*

$$\frac{\mathbb{X} \rightsquigarrow_1 \mathbb{Y}}{\mathbb{Z}(\mathbb{X}) \rightsquigarrow_1 \mathbb{Z}(\mathbb{Y})}$$

is a condition appearing in Definition 4.1, for any terms or eliminators $\mathbb{X}, \mathbb{Y}, \mathbb{Z}(\mathbb{X})$ such that $\mathbb{X} \rightsquigarrow \mathbb{Y}$: if $\mathbb{Z}(\mathbb{Y})$ is well-formed we have $\mathbb{Z}(\mathbb{X}) \rightsquigarrow \mathbb{Z}(\mathbb{Y})$, and if $\mathbb{Z}(\mathbb{Y})$ is not well-formed we have $\mathbb{Z}(\mathbb{X}) \rightsquigarrow \mathbb{Y}$.[16]

PROOF: Induction on the reduction path from $\mathbb{X}$ to $\mathbb{Y}$. At each step, we need to know that if $\mathbb{Z}(\mathbb{Y})$ is well-formed and $\mathbb{W} \rightsquigarrow_1 \mathbb{Y}$, then $\mathbb{Z}(\mathbb{W})$ is also well-formed—this way, if $\mathbb{Z}(\mathbb{Y})$ is well-formed, we can ensure that all the needed intermediate links from $\mathbb{Z}(\mathbb{X})$ to $\mathbb{Z}(\mathbb{Y})$ are also well-formed. This holds, though, because of what we know about how reduction affects hats and free variables. □

[15] We do not consider in this paper, outside this footnote, the notion of *weak* normalization, where a term M counts as weakly normalizing iff there is some normal form N with $M \rightsquigarrow N$. In general, when we have two notions of reduction $\rightsquigarrow_a \subseteq \rightsquigarrow_b$, like our principal and full reductions, nothing useful follows about a relationship between weak normalization for a and b. In this regard, weak normalization is unlike both strong normalization and normal forms.

[16] Here, $\mathbb{Z}(\mathbb{X})$ should be understood as a term or eliminator with $\mathbb{X}$ as an immediate constituent, and similarly for $\mathbb{Z}(\mathbb{Y})$.

Lemma 4.6. *If $N \rightsquigarrow_1 N'$ and N is a subterm of M, then there is some M' with $M \rightsquigarrow_1 M'$ and N' a subterm of M'.*

Proof: Induction on N's being a subterm of M.

- If $N = M$ then reducing the same way yields $M' = N'$ and we're done.
- Otherwise, let O be the immediate subterm of M that contains N. By the induction hypothesis, there is some O' with $O \rightsquigarrow_1 O'$ and N' a subterm of O'. By inspecting the one-step reduction rules, we can see that there is some M' with $M \rightsquigarrow_1 M'$ and O' as a subterm. □

Lemma 4.7. *If there is a reduction path of length n from N to N' and N is a subterm of M, then there is a reduction path of length n from M to some M' such that N' is a subterm of M'.*

Proof: Induction on the reduction path from N to N', using Lemma 4.6 at each step. □

Lemma 4.8. *If M is strongly normalizing and N is a subterm of M, then N is also strongly normalizing, and $|N| \leq |M|$.*

Proof: Immediate from Lemma 4.7. □

Lemma 4.9. *If M is strongly normalizing and $M \rightsquigarrow^+ M'$, then M' is strongly normalizing and $|M'| < |M|$.*

Proof: Immediate from definitions. □

Lemma 4.10 (Substitution lemma (see [2, 2.1.16])).
Let $\sigma = [x_1 \mapsto P_1, \ldots, x_m \mapsto P_m]$ and $\tau = [y_1 \mapsto Q_1, \ldots, y_n \mapsto Q_n]$ be substitutions such that all x_i are distinct from all y_j and no x_i occurs free in any Q_j. Let (σ^τ) be the substitution $[x_1 \mapsto P_1\tau, \ldots, x_m \mapsto P_m\tau]$. Then $\mathbb{X}\sigma\tau = \mathbb{X}\tau(\sigma^\tau)$.

Proof: Induction on $\mathbb{X}$.

- $\mathbb{X}$ is a variable. If $\mathbb{X}$ is no x_i or y_j, then both sides are M. If $\mathbb{X}$ is x_i, then both sides are $P_i\tau$. if $\mathbb{X}$ is y_j, then both sides are Q_j.
- $\mathbb{X}$ is $(\!|O|\!)$ or $\langle N, O\rangle$ or $\mathsf{inl}(N)$ or $\mathsf{inr}(N)$ or $N\mathcal{E}$. These cases follow immediately from the induction hypothesis.

- $\mathbb{X}$ is $\lambda^{\rightarrow} z.N$. Set up $\lambda^{\rightarrow} z.N$'s bound variables so that z is no x_i or y_j, and so that z is not free in any P_i or Q_j. The the induction hypothesis suffices, since $\mathbb{X}\sigma\tau = \lambda^{\rightarrow} z.(N\sigma\tau)$ and $\mathbb{X}\tau(\sigma^{\tau}) = \lambda^{\rightarrow} z.(N\tau(\sigma^{\tau}))$.

- $\mathbb{X}$ is a $\lambda^{\neg}$ term or an eliminator other than $(\!|N|\!)$. The reasoning in these cases is parallel to the $\lambda^{\rightarrow}$ case. □

LEMMA 4.11 (Substitution in redexes). *If R is a redex and R' is its reduct, then $R[x_1 \mapsto P_1, \ldots, x_n \mapsto P_n]$ is a redex and $R'[x_1 \mapsto P_1, \ldots, x_n \mapsto P_n]$ is its reduct.*

PROOF: Verifying is a matter of checking each kind of redex in turn. That substitution preserves redexhood is relatively straightforward, so we turn to the second part of the claim. Let σ be the substitution $[x_1 \mapsto P_1, \ldots, x_n \mapsto P_n]$, and change bound variables in R so that no x_i is bound in R and no variable free in any P_i is bound in R.

Principal redexes:

- If R is $(\lambda^{\rightarrow} x.(M^{\psi}))(\!|N, y.O|\!)$, then R' is $O[y \mapsto M[x \mapsto N]]$. By setting up R's bound variables (which certainly include x and y) as we have, $R\sigma = (\lambda^{\rightarrow} x.M\sigma)(\!|N\sigma, y.O\sigma|\!)$, and so its reduct is $O\sigma[y \mapsto M\sigma[x \mapsto N\sigma]]$. By Lemma 4.10 (twice) this is $O[y \mapsto M[x \mapsto N]]\sigma$, which is $R'\sigma$.

- If R is $(\lambda^{\rightarrow} x.(M^{☹}))(\!|N, y.O|\!)$, then R' is $M[x \mapsto N]$. By setting up bound variables as we have, $R\sigma = (\lambda^{\rightarrow} x.M\sigma)(\!|N\sigma, y.O\sigma|\!)$, and so its reduct is $M\sigma[x \mapsto N\sigma]$. By Lemma 4.10, this is $M[x \mapsto N]\sigma$, which is $R'\sigma$.

- If R is $\langle M, N\rangle(\!|\langle x, y\rangle.O|\!)$, then R' is $O[x \mapsto M, y \mapsto N]$. By setting up bound variables as we have, $R\sigma = \langle M\sigma, N\sigma\rangle(\!|\langle x, y\rangle.O\sigma|\!)$, and so its reduct is $O\sigma[x \mapsto M\sigma, y \mapsto N\sigma]$. By Lemma 4.10 this is $O[x \mapsto M, y \mapsto N]\sigma$, which is $R'\sigma$.

- If R is $\mathsf{inl}(M)(\!|x.N, y.O|\!)$ or $\mathsf{inr}(M)(\!|x.N, y.O|\!)$ or $(\lambda^{\neg} x.M)(\!|N|\!)$, the reasoning is parallel to the above cases.

As for commuting redexes: If R is $(M\mathcal{E})\mathcal{F}$, then R' is $M(\!|\mathcal{E}\mathcal{F}|\!)$, and $R\sigma = ((M\sigma)(\mathcal{E}\sigma))(\mathcal{F}\sigma)$. The reduct of $R\sigma$ is thus $(M\sigma)(\!|(\mathcal{E}\sigma)(\mathcal{F}\sigma)|\!)$. By Lemma 3.2 this is $M\sigma((\!|\mathcal{E}\mathcal{F}|\!)\sigma)$; and by Lemma 4.10 this is in turn $(M(\!|\mathcal{E}\mathcal{F}|\!))\sigma$, which is $R'\sigma$. □

Lemma 4.12 (Substitution and reduction).
If $\mathbb{X} \rightsquigarrow \mathbb{Y}$*, then* $\mathbb{X}[x_1 \mapsto P_1, \ldots, x_n \mapsto P_n] \rightsquigarrow \mathbb{Y}[x_1 \mapsto P_1, \ldots, x_n \mapsto P_n]$*.*

Proof: Because of the complications in our notion of one-step reduction, Lemma 4.11 does not immediately suffice for this claim; it needs to be worked through.

It suffices to show that if $\mathbb{X} \rightsquigarrow_1 \mathbb{Y}$, then for all substitutions σ we have $\mathbb{X}\sigma \rightsquigarrow_1 \mathbb{Y}\sigma$. This we show by induction on the formation of $\mathbb{X}$, explicitly stating only some representative cases. (Recall that all substitutions preserve hat exactly.)

- If $\mathbb{X}$ is a variable x, then there's nothing to show, since it's false that $x \rightsquigarrow_1 \mathbb{Y}$.

- If $\mathbb{X}$ is $N\mathcal{E}$, there are three possibilities for $\mathbb{X} \rightsquigarrow_1 \mathbb{Y}$: the redex is in N, in $\mathcal{E}$, or is $N\mathcal{E}$ itself.

  – If the redex is inside N, let N' be the result of reducing N at that redex. Applying the induction hypothesis, $N\sigma \rightsquigarrow_1 N'\sigma$; moreover, N' and $N'\sigma$ have the same hat.

    * If this hat is ☹, then $\mathbb{Y} = N'$, and so $\mathbb{X}\sigma = (N\sigma)(\mathcal{E}\sigma) \rightsquigarrow_1 N'\sigma = \mathbb{Y}\sigma$.
    * If it is some φ, then $\mathbb{Y} = N'\mathcal{E}$, and so $\mathbb{X}\sigma = (N\sigma)(\mathcal{E}\sigma) \rightsquigarrow_1 (N'\sigma)(\mathcal{E}\sigma) = \mathbb{Y}\sigma$.

  – If the redex is inside $\mathcal{E}$, the reasoning is parallel, except instead of concern for hats, we are concerned whether $\mathcal{E}$ reduces at this redex to an eliminator or a term.

  – If the redex is $N\mathcal{E}$ itself, we're covered by Lemma 4.11.

- If $\mathbb{X}$ is $\lambda^{\rightarrow}x.N$, change its bound variables so that x is not among the x_i and not free in any P_i. The redex securing $\mathbb{X} \rightsquigarrow_1 \mathbb{Y}$ must be inside N. Let N' be the result of reducing N at that redex. Applying the induction hypothesis, $N\sigma \rightsquigarrow_1 N'\sigma$. Moreover, N' and $N'\sigma$ have the same hat, and x is free in N' iff it is free in $N'\sigma$. Thus, $\lambda^{\rightarrow}x.N'$ is well-formed iff $\lambda^{\rightarrow}x.(N'\sigma)$ is.

  – If they are well-formed, then $\mathbb{Y} = \lambda^{\rightarrow}x.N'$, and so $\mathbb{X}\sigma = \lambda^{\rightarrow}x.(N\sigma) \rightsquigarrow_1 \lambda^{\rightarrow}x.(N'\sigma) = \mathbb{Y}\sigma$.

– If they are not, then $\mathbb{Y} = N'$, and so $\mathbb{X}\sigma = \lambda^{\rightarrow}x.(N\sigma) \rightsquigarrow_1 N'\sigma = \mathbb{Y}\sigma$.

- Other cases without bound variables are like the case of $N\mathcal{E}$; other cases with bound variables are like the case of $\lambda^{\rightarrow}x.N$. □

5. Strong normalization

The foregoing discussion covers both principal and full reduction. In this section, we narrow our attention to principal reduction only, and show that every term in our system is (principally) strongly normalizing. In this, we closely follow the approach of [4]. (Again, we conjecture that full reduction is also strongly normalizing, but leave that question, which requires different techniques, for future work.)

5.1. The Prawitz restriction revisited

First, however, we return briefly to the topic of Sections 2.4 and 3.5: the Prawitz restriction, which Tennant imposes and we do not. In Section 2.4 we saw that the Prawitz restriction rules out a range of derivations that we allow, and in Section 3.5 we saw that these derivations include some with important computational interpretations. That much alone, we think, motivates our dropping the Prawitz restriction. However, there is another interesting effect of the restriction, which we point out here: it blocks strong normalization, even for principal reduction (and therefore for full reduction as well). To show this, we use a (slightly modified) example of [9]. (Spelling this out in our term language would save space, but at the cost of even lower readability, so we return to derivations for the example.)

Look to the three derivations in Figure 1. Note that the first principally reduces (at the redex indicated with ⋆) to the second, and the second principally reduces (at the redex indicated with ⋆) to the third. Note also that the first and second obey the Prawitz restriction, but the third does not; the step of →I indicated with † in the third derivation can discharge open assumptions of p, and indeed there are two open assumptions of p in scope at that step in the derivation, also indicated with †.

Reduction in a system obeying the Prawitz restriction, then, could not reduce the second derivation here to the third, since the third does not belong in such a system. Rather, it would reduce the second derivation

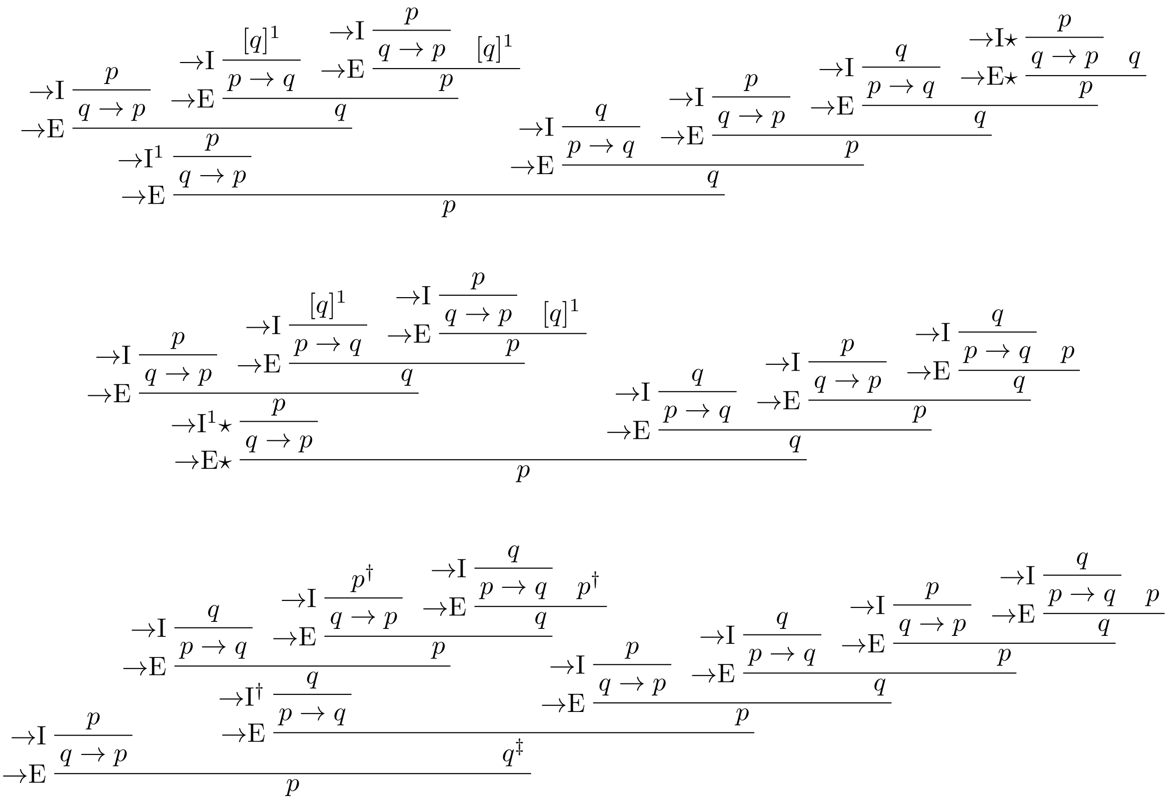

Figure 1. Strong normalization fails in Tennant's original system

here to a derivation much like the third, but which discharges the indicated open assumptions of p at the indicated step of $\to$I.

That, in turn, would defeat strong normalization: look to the q node indicated with ‡ in the third derivation, and consider the subderivation from that node upwards. With the binding in place needed to meet the Prawitz restriction, this subderivation is isomorphic to the original derivation, just with the roles of p and q switched. So we can repeat the cycle endlessly, producing an infinite reduction path.

Without the Prawitz restriction, on the other hand, the second derivation reduces to the third, with no additional binding needed. No cycle is created. And as we now show, indeed strong normalization does hold for our system.

5.2. Proving strong normalization

Definition 5.1. We define a notion of *strongly computable term* (SC term) by induction on hats:

- For an atomic type p, a term M^p is SC iff it is strongly normalizing;
- A term $M^{☹}$ is SC iff it is strongly normalizing;
- A term $M^{\varphi \wedge \psi}$ is SC iff it is strongly normalizing and whenever it reduces to a term $\langle N, O \rangle$, both N and O are SC;
- A term $M^{\varphi \vee \psi}$ is SC iff it is strongly normalizing and whenever it reduces to either $\mathsf{inl}(N)$ or $\mathsf{inr}(N)$, then N is SC; and
- A term $M^{\varphi \to \psi}$ is SC iff it is strongly normalizing and whenever it reduces to a term $\lambda^{\to} x.N$, then for all SC terms O^{φ}, the term $N[x \mapsto O]$ is SC.[17]
- A term $M^{\neg \varphi}$ is SC iff it is strongly normalizing and whenever it reduces to a term $\lambda^{\neg} x.N$, then for all SC terms O^{φ}, the term $N[x \mapsto O]$ is SC.

It is clear from this definition that every SC term is strongly normalizing. Then we show by induction on *terms* that every term is SC. This

[17] [13], which features a similar proof, has a slightly different definition here, following [7, Appendix A3], but that doesn't consider conjunction or disjunction. Here, we follow [4].

works because the inductive structures of terms and of types do not align, so we can play them off against each other.

Lemma 5.2 (Variables). *For any type φ, every variable of type φ is SC.*

Proof: All variables x^φ do not contain any redexes as subterms, thus do not have any one-step reductions, and hence all reduction paths from x^φ are of length 0, so finite. When φ is complex, the additional conditions following "whenever it reduces" are vacuously fulfilled, as variables never reduce to such forms. So all variables are SC. □

Lemma 5.3 (Closure by reduction). *If M is SC and $M \rightsquigarrow N$, then N is SC.*[18]

Proof: Note first that if M is strongly normalizing and $M \rightsquigarrow N$, then N too must be strongly normalizing; any infinite reduction path starting from N would give rise to an infinite reduction path starting from M. Since M is SC, it must be strongly normalizing, so N too must be strongly normalizing.

It remains only to check the additional requirements for N to be SC, according to N's hat. Recall that if N is N^φ, then M must be M^φ.

- If N is $N^{☹}$, then there are no additional requirements, and we're done.
- If N is N^p for an atomic type p, then there are no additional requirements, and we're done.
- If $M^{\varphi\wedge\psi} \rightsquigarrow N^{\varphi\wedge\psi}$, then if $N^{\varphi\wedge\psi}$ reduces to $\langle O, P\rangle$ so does M. Since M is SC, in this case O and P must be SC, so the additional requirement on N is met.
- If $M^{\varphi\vee\psi} \rightsquigarrow N^{\varphi\vee\psi}$, then if $N^{\varphi\vee\psi}$ reduces to $\mathsf{inl}(O)$ or $\mathsf{inr}(O)$ so does M. Since M is SC, in these cases O must be SC, so the additional requirement on N is met.
- If $M^{\varphi\to\psi} \rightsquigarrow N^{\varphi\to\psi}$, then if N reduces to $\lambda^{\to}x.O$ so does M. Since M is SC, in these cases it must be that for all SC terms P^φ, the term $O[x \mapsto P]$ is SC. So the additional requirement on N is met.

[18] Note that M and N needn't have the same hat, so this claim precisely as stated in [4] would be false.

- If $M^{\neg\varphi} \rightsquigarrow N^{\neg\varphi}$, then if N reduces to $\lambda^{\neg}x.O$ so does M. Since M is SC, in these cases it must be that for all SC terms P^{φ}, the term $O[x \mapsto P]$ is SC. So the additional requirement on N is met. □

LEMMA 5.4 (Girard's lemma). *Let M be a term that is not an introduction, such that for all N with $M \rightsquigarrow_1 N$, N is SC. Then M is SC.*

PROOF: If there does not exist such an N then M is SC because M does not have any one-step reductions, hence all reduction paths from M are of finite 0 length and additional requirements depending on hat do not apply.

Since N is SC, every reduction path is finite from N, hence M is strongly normalizing because M reduces finitely in one step to N.

- If all N have hat ☹, then M is SC because M is SN and additional requirements depending on hat don't apply because M does not reduce to any introductions.

- If there exists N with an atomic hat, then M has an atomic hat and is SC because M is SN.

Since M is not an introduction, it is not, in reduction to itself, required to satisfy the additional conditions for M to be SC for the following hats:

- If there exists N with a hat of the form $\varphi \wedge \psi$, then M has hat $\varphi \wedge \psi$. If $M \rightsquigarrow_1 N \rightsquigarrow \langle O, P \rangle$, O and P are SC because N is SC. Since M is strongly normalizing and whenever M reduces to a term $\langle O, P \rangle$, O and P are SC, M is SC.

- If there exists N with a hat of the form $\varphi \vee \psi$, then M has hat $\varphi \vee \psi$. If $M \rightsquigarrow_1 N \rightsquigarrow \mathsf{inl}(O)$ or $M \rightsquigarrow_1 N \rightsquigarrow \mathsf{inr}(O)$, O is SC because N is strongly normalizing. Since M is SN and whenever M reduces to a term $\mathsf{inl}(O)$ or $\mathsf{inr}(O)$, O is SC, M is SC.

- If there exists N with hat $\varphi \to \psi$, then M has hat $\varphi \to \psi$. If $M \rightsquigarrow_1 N \rightsquigarrow \lambda^{\to}x.O$, for all SC terms P^{φ}, the term $O[x \mapsto P]$ is SC. Since M is strongly normalizing and whenever M reduces to a term $\lambda^{\to}x.O$, for all SC terms P^{φ}, the term $O[x \mapsto P]$ is SC, M is SC.

- If there exists N with hat $\neg\varphi$, then M has hat $\neg\varphi$. If $M \rightsquigarrow_1 N \rightsquigarrow \lambda^{\neg}x.O$, for all SC terms P^{φ}, the term $O[x \mapsto P]$ is SC. Since M is strongly normalizing and whenever M reduces to a term $\lambda^{\neg}x.O$, for all SC terms P^{φ}, the term $O[x \mapsto P]$ is SC, M is SC. □

LEMMA 5.5 (Adequacy of λ (I)). *If for all SC M^{φ} we have $N^{\psi}[x \mapsto M]$ is SC, then $(\lambda^{\rightarrow}x.N)^{\varphi\rightarrow\psi}$ is SC.*

PROOF: By Lemma 5.2 , all variables are SC. Let $M := x$, $N[x \mapsto x] = N$ is SC and hence N is strongly normalizing. Thus, $\lambda^{\rightarrow}x.N$ is strongly normalizing because the only possible reductions involve reducing N within the term or reduction to an exceptional term. Thus, the reduction paths of N bind the reduction paths of $\lambda^{\rightarrow}x.N$.

If $\lambda^{\rightarrow}x.N \rightsquigarrow \lambda^{\rightarrow}x.N'$, then $N \rightsquigarrow N'$ by the reduction rules. By Lemma 4.12, $N[x \mapsto M] \rightsquigarrow N'[x \mapsto M]$ and $N'[x \mapsto M]$ is SC by Lemma 5.3.

Thus, $\lambda^{\rightarrow}x.N$ is SC because it is strongly normalizing and whenever it reduces to $\lambda^{\rightarrow}x.N'$, for any SC M^{φ}, $N'[x \mapsto M]$ is SC. □

LEMMA 5.6 (Adequacy of λ (II)). *If for all SC M^{φ} we have $N^{\circledast}[x \mapsto M]$ is SC (and so long as $x \in \texttt{FV}(N)$), then $(\lambda^{\rightarrow}x.N)^{\varphi\rightarrow\psi}$ and $(\lambda^{\neg}x.N)^{\neg\varphi}$ are both SC.*

PROOF: By Lemma 5.2 , all variables are SC. Let $M := x$, $N[x \mapsto x] = N$ is SC and hence N is strongly normalizing. Thus, both $\lambda^{\rightarrow}x.N$ and $\lambda^{\neg}x.N$ are strongly normalizing because the only possible reductions involve reducing N within the term or reduction to an exceptional term. Thus, the reduction paths of N bind the reduction paths of $\lambda^{\rightarrow}x.N$ and $\lambda^{\neg}x.N$.

If $\lambda^{\rightarrow}x.N \rightsquigarrow \lambda^{\rightarrow}x.N'$ or $\lambda^{\neg}x.N \rightsquigarrow \lambda^{\neg}x.N'$, then $N \rightsquigarrow N'$ by the reduction rules. By Lemma 4.12, $N[x \mapsto M] \rightsquigarrow N'[x \mapsto M]$ and $N'[x \mapsto M]$ is SC by Lemma 5.3.

Thus, $\lambda^{\rightarrow}x.N$ and $\lambda^{\neg}x.N$ are SC because they are strongly normalizing and whenever they respectively reduce to $\lambda^{\rightarrow}x.N'$ and $\lambda^{\neg}x.N'$, for any SC M^{φ}, $N'[x \mapsto M]$ is SC. □

LEMMA 5.7 (Adequacy of $\langle,\rangle$). *If M^{φ} and N^{ψ} are both SC, then $\langle M, N\rangle^{\varphi\wedge\psi}$ is SC.*

PROOF: $\langle M, N\rangle$ is strongly normalizing because the only possible reductions involve reducing M and N within the term or reduction to an exceptional term. Thus, since M and N are strongly normalizing, their reduction paths bind the reduction paths of $\langle M, N\rangle$.

By Lemma 5.3, if $M \rightsquigarrow M'$ and $N \rightsquigarrow N'$ then M' and N' are SC.

Whenever $\langle M, N\rangle$ reduces to an introduction $\langle M', N'\rangle$, M' and N' are SC, thus, since $\langle M, N\rangle$ is also strongly normalizing, by Definition 5.1 it is SC. □

LEMMA 5.8 (Adequacy of inl, inr). *If M^{φ} is SC, then $\mathsf{inl}(M)$ and $\mathsf{inr}(M)$ are both SC.*

PROOF: Wlog, we consider just $\mathsf{inl}(M)$.

$\mathsf{inl}(M)$ is strongly normalizing because the only possible reductions involve reducing M within the term or reduction to an exceptional term. Thus, since M is strongly normalizing, reduction paths from $\mathsf{inl}(M)$ are bound by reduction paths of M.

By Lemma 5.3 if $M \rightsquigarrow M'$, then M' is SC.

Whenever $\mathsf{inl}(M)$ reduces to an introduction $\mathsf{inl}(M')$, M' is SC, thus, since $\mathsf{inl}(M)$ is also strongly normalizing, by Definition 5.1 it is SC. □

LEMMA 5.9 (Adequacy of application (I)). *If $M^{\varphi\rightarrow\psi}$ is SC, N^{φ} is SC, and for all SC Q^{ψ}, $O[x \mapsto Q]$ is SC, then $M(\!|N, x.O|\!)$ is SC.*

PROOF: Let $Q = x$ where x is SC by Lemma 5.2, thus $O[x \mapsto x] = O$ is SC. Since M, N and O are SC, they are strongly normalising and hence $|M|$, $|N|$ and $|O|$ are defined. We proceed by induction on $|M|+|N|+|O|$. By Lemma 5.4, to prove that $M(\!|N, x.O|\!)$ is SC, we need to prove that all one-step reducts are SC. Given $M \rightsquigarrow_1 M'$ or $N \rightsquigarrow_1 N'$ or $O \rightsquigarrow_1 O'$ where M', N', and O' are SC by Lemma 5.3:

- If $M(\!|N, x.O|\!) \rightsquigarrow_1 M'(\!|N, x.O|\!)$ or $M(\!|N, x.O|\!) \rightsquigarrow_1 M(\!|N', x.O|\!)$ or $M(\!|N, x.O|\!) \rightsquigarrow_1 M(\!|N, x.O'|\!)$, then we apply the induction hypothesis and Lemma 4.9 to obtain $|M| + |N| + |O| > |M'| + |N| + |O|$, $|M|+|N|+|O| > |M|+|N'|+|O|$ or $|M|+|N|+|O| > |M'|+|N|+|O'|$.

- If $M(\!|N, x.O|\!) \rightsquigarrow_1 M'^{☹}$ or $M(\!|N, x.O|\!) \rightsquigarrow_1 N'^{☹}$ or $M(\!|N, x.O|\!) \rightsquigarrow_1 O'^{☹}$, then we already have M', N', or O' SC.

- If $M(\!|N, x.O|\!)$ is a principal redex, then M is of the form $\lambda^{\rightarrow} y.P^{\mathfrak{D}}$. If $\mathfrak{D} = ☹$, then $M(\!|N, x.O|\!) \rightsquigarrow_1 P[y \mapsto N]$ which is SC by Definition 5.1. Otherwise $M(\!|N, x.O|\!) \rightsquigarrow_1 O[x \mapsto P[y \mapsto N]]$ which is SC by the lemma statement. □

Lemma 5.10 (Adequacy of application (II)). *If $M^{\neg\varphi}$ is SC and N^{φ} is SC, then $M(\!|N|\!)$ is SC.*

Proof: Since M and N are SC, they are strongly normalising and hence $|M|$ and $|N|$ are defined. We proceed by induction on $|M| + |N|$. By Lemma 5.4, to prove that $M(\!|N|\!)$ is SC, we need to prove that all one-step reducts are SC. Given $M \rightsquigarrow_1 M'$ or $N \rightsquigarrow_1 N'$ where M' and N' are SC by Lemma 5.3:

- If $M(\!|N|\!) \rightsquigarrow_1 M'(\!|N|\!)$ or $M(\!|N|\!) \rightsquigarrow_1 M(\!|N'|\!)$ then we apply the induction hypothesis and Lemma 4.9 to obtain $|M| + |N| > |M'| + |N|$ or $|M| + |N| > |M| + |N'|$.
- If $M(\!|N|\!) \rightsquigarrow_1 M'^{☹}$ or $M(\!|N|\!) \rightsquigarrow_1 N'^{☹}$, then we already have M' or N' SC.
- If $M(\!|N|\!)$ is a principal redex, then M is of the form $\lambda^{\neg}x.O$, and $M(\!|N|\!) \rightsquigarrow_1 O[x \mapsto N]$ which is SC by Definition 5.1. □

Lemma 5.11 (Adequacy of Conjunction elimination). *If $M^{\varphi\wedge\psi}$ is SC, and for all SC P^{φ}, Q^{ψ} the term $N[x \mapsto P, y \mapsto Q]$ is SC, then $M(\!|\langle x,y\rangle.N|\!)$ is SC (if well-formed).*

Proof: Let $P = x$ and $Q = y$ where x and y are SC by Lemma 5.2, thus $N[x \mapsto x, y \mapsto y] = N$ is SC. We proceed by induction on $|M| + |N|$. By Lemma 5.4, to prove that $M(\!|\langle x,y\rangle.N|\!)$ is SC, we need to prove that all one-step reducts are SC. Given $M \rightsquigarrow_1 M'$ and $N \rightsquigarrow_1 N'$ where M' and N' are SC by Lemma 5.3:

- If $M(\!|\langle x,y\rangle.N|\!) \rightsquigarrow_1 M'(\!|\langle x,y\rangle.N|\!)$ or $M(\!|\langle x,y\rangle.N|\!) \rightsquigarrow_1 M(\!|\langle x,y\rangle.N'|\!)$ then we apply the induction hypothesis and Lemma 4.9 to obtain $|M| + |N| > |M'| + |N|$ or $|M| + |N| > |M| + |N'|$.
- If $M(\!|\langle x,y\rangle.N|\!) \rightsquigarrow_1 M'^{☹}$ or $M(\!|\langle x,y\rangle.N|\!) \rightsquigarrow_1 N'^{☹}$, then we already have M' and N' SC.
- If $M(\!|\langle x,y\rangle.N|\!)$ is a principal redex, then M is of the form $\langle R, S\rangle$ and $M(\!|\langle x,y\rangle.N|\!) \rightsquigarrow_1 N[x \mapsto R, y \mapsto S]$ which is SC by the lemma statement and Definition 5.1. □

LEMMA 5.12 (Adequacy of Disjunction elimination). *If $M^{\varphi\vee\psi}$ is SC, and for all SC P^φ the term $N[x \mapsto P]$ is SC, and for all SC Q^ψ the term $O[y \mapsto Q]$ is SC, then $M(\!|x.N, y.O|\!)$ is SC (if well-formed).*

PROOF: Let $P = x$ and $Q = y$ where x and y are SC by Lemma 5.2, thus $N[x \mapsto x] = N$ and $O[y \mapsto y] = O$ are SC. Since M, N and O are SC, they are strongly normalising and hence $|M|$, $|N|$ and $|O|$ are defined. We proceed by induction on $|M| + |N| + |O|$. By Lemma 5.4, to prove that $M(\!|x.N, y.O|\!)$ is SC, we need to prove that all one-step reducts are SC. Given $M \leadsto_1 M'$ or $N \leadsto_1 N'$ or $O \leadsto_1 O'$ where M', N', and O' are SC by Lemma 5.3:

- If $M(\!|x.N, y.O|\!) \leadsto_1 M'(\!|x.N, y.O|\!)$ or $M(\!|x.N, y.O|\!) \leadsto_1 M(\!|x.N', y.O|\!)$ or $M(\!|x.N, y.O|\!) \leadsto_1 M(\!|x.N, y.O'|\!)$, then we apply the induction hypothesis and Lemma 4.9 to obtain $|M|+|N|+|O| > |M'|+|N|+|O|$, $|M|+|N|+|O| > |M|+|N'|+|O|$ or $|M|+|N|+|O| > |M'|+|N|+|O'|$.
- If $M(\!|x.N, y.O|\!) \leadsto_1 M'$ or $M(\!|x.N, y.O|\!) \leadsto_1 N'$ or $M(\!|x.N, y.O|\!) \leadsto_1 O'$, then we already have M', N', or O' SC.
- If $M(\!|x.N, y.O|\!)$ is a principal redex, then M is of the form $\mathsf{inl}(R)$ or $\mathsf{inr}(R)$ and $M(\!|x.N, y.O|\!) \leadsto_1 N[x \mapsto R]$ or $M(\!|x.N, y.O|\!) \leadsto_1 O[y \mapsto R]$ which are both SC by the lemma statement and Definition 5.1. □

DEFINITION 5.13. A substitution $[x_1 \mapsto P_1, \ldots, x_n \mapsto P_n]$ is *SC* iff $P_1, \ldots, P_n$ are all SC. A term M is *SC under substitution* iff for all SC substitutions σ, the term $M\sigma$ is SC.

THEOREM 5.14. *All terms are SC under substitution.*

PROOF: Take any term M. To see that M is SC under substitution, proceed by induction on M's formation.

- If M is x^φ then any substitution for x will be a variable and Lemma 5.2 applies.
- If M is $\langle N, O\rangle$: take any SC substitution σ. By the induction hypothesis, N and O are SC under substitution, so $N\sigma$ and $O\sigma$ are SC. Thus, by Lemma 5.7, $\langle N\sigma, O\sigma\rangle$ is SC; but this is just $M\sigma$.
- If M is $\mathsf{inl}(N)$ or $\mathsf{inr}(N)$, the reasoning is similar to the $\langle,\rangle$ case.

- If M is $\lambda^{\rightarrow} x^{\varphi}.N$: take any SC substitution σ, and change M's bound variables so that x is neither acted on by σ nor free in σ. By the induction hypothesis, N is SC under substitution, so for any SC term P^{φ}, we have that $N\sigma[x \mapsto P]$ is SC. Thus, by Lemma 5.5 and Lemma 5.6, $\lambda^{\rightarrow} x.(N\sigma)$ is SC; but this is just $M\sigma$.

- If M is $\lambda^{\neg} x.M$, the reasoning is similar to the $\lambda^{\rightarrow}$ case.

- If M is $N(\!|O, x.P|\!)$: take any SC substitution σ, and change M's bound variables so that x is neither acted on by σ nor free in σ. By the induction hypothesis, N, O and P are SC under substitution, so $N\sigma$, $O\sigma$ and $P\sigma$ are SC. Given SC Q^{φ}, we have $P\sigma[x \mapsto Q]$ is SC. Thus, by Lemma 5.9, $N\sigma(\!|O\sigma, x.P\sigma|\!)$ is SC; but this is just $M\sigma$.

- If M is $N(\!|O|\!)$: take any SC substitution σ. By the induction hypothesis, N and O are SC under substitution, so $N\sigma$ and $O\sigma$ are SC. Thus, by Lemma 5.10, $N\sigma(\!|O\sigma|\!)$ is SC; but this is just $M\sigma$.

- If M is $N(\!|\langle x, y\rangle.O|\!)$: take any SC substitution σ, and change M's bound variables so that x and y are neither acted on by σ nor free in σ. By the induction hypothesis, N and O are SC under substitution, so $N\sigma$ and $O\sigma$ are SC. Given SC P^{φ} and Q^{ψ}, $O[x \mapsto P, y \mapsto Q]$ is SC. Thus, by Lemma 5.11, $N\sigma(\!|\langle x, y\rangle.O\sigma|\!)$ is SC; but this is just $M\sigma$.

- If M is $N(\!|x.O, y.P|\!)$: take any SC substitution σ, and change M's bound variables so that x and y are neither acted on by σ nor free in σ. By the induction hypothesis, N, O and P are SC under substitution, so $N\sigma$, $O\sigma$ and $P\sigma$ are SC. Given SC Q^{φ} and R^{ψ}, $O\sigma[x \mapsto Q]$ and $P\sigma[y \mapsto R]$ are SC. Thus, by Lemma 5.12, $N\sigma(\!|x.O\sigma, y.P\sigma|\!)$ is SC; but this is just $M\sigma$. □

Corollary 5.15. All terms are strongly normalizing.

Proof: Take any term M. By Theorem 5.14, M is SC under substitution; clearly, then, M is SC. (Consider the substitution $[x^{\varphi} \mapsto x^{\varphi}]$.) By Definition 5.1, then, M is strongly normalizing. □

6. Conclusion

In this paper, we've presented a natural deduction system for core logic, and developed a term calculus that corresponds to this natural deduction system. We've defined two reduction relations on this term calculus—principal and full reduction—and explored the ways that core logic's restrictions make reduction somewhat different from reduction in more familiar term calculi. We've discussed the Prawitz restriction and our reasons for dropping it. And finally, we've shown that principal reduction in this system is strongly normalizing (although it would not be with the Prawitz restriction in place). In future work, we hope to extend this strong normalization to full reduction as well, but as that will require different techniques, only time will tell.

Acknowledgements. This research was supported by the Monash Faculty of Information Technology, the Monash Laboratory for the Foundations of Computing, the Australian Research Council fellowship "Substructural logics for bounded resources" (FT190100147), and PLEXUS (grant agreement 101086295), a Marie-Skłodowska-Curie action funded by the EU under the Horizon Europe Research and Innovation Programme. Many thanks also to Sara Ayhan and Neil Tennant for helpful discussions, to Elaine Pimentel for pointing us to [4], and to two anonymous referees for their comments.

References

[1] S. Ayhan, H. Wansing, *On synonymy in proof-theoretic semantics: The case of 2Int*, **Bulletin of the Section of Logic**, vol. Online First (2023), DOI: https://doi.org/10.18778/0138-0680.2023.18.

[2] H. P. Barendregt, **The Lambda Calculus: Its Syntax and Semantics**, Elsevier, Amsterdam (1984).

[3] A. Church, **The Calculi of Lambda-Conversion**, Princeton University Press, Princeton, New Jersey (1941).

[4] A. Díaz-Caro, G. Dowek, *A new connective in natural deduction, and its application to quantum computing*, [in:] **International Colloquium on Theoretical Aspects of Computing**, Springer (2021), pp. 175–193, DOI: https://doi.org/10.1007/978-3-030-85315-0_11.

[5] G. Gentzen, *Investigations Into Logical Deduction (1935)*, [in:] M. E. Szabo (ed.), **The Collected Papers of Gerhard Gentzen**, North-Holland Publishing Company, Amsterdam (1969), pp. 68–131.

[6] J. Girard, P. Taylor, Y. Lafont, **Proofs and Types**, Cambridge University Press, Cambridge (1989).

[7] J. Hindley, J. P. Seldin, **Lambda-Calculus and Combinators: an Introduction**, Cambridge University Press, Cambridge (2008), DOI: https://doi.org/10.1017/CBO9780511809835.

[8] W. H. Howard, *The formulae-as-types notion of construction (1969)*, [in:] **To HB Curry: Essays on combinatory logic, lambda calculus, and formalism**, Academic Press (1980), pp. 479–490.

[9] D. Leivant, *Assumption classes in natural deduction*, **Mathematical Logic Quarterly**, vol. 25(1–2) (1979), pp. 1–4, DOI: https://doi.org/10.1002/malq.19790250102.

[10] S. Negri, J. von Plato, **Structural Proof Theory**, Cambridge University Press, Cambridge (2001), DOI: https://doi.org/10.1017/CBO9780511527340.

[11] M. Petrolo, P. Pistone, *On paradoxes in normal form*, **Topoi**, vol. 38(3) (2019), pp. 605–617, DOI: https://doi.org/10.1007/s11245-018-9543-7.

[12] D. Prawitz, **Natural Deduction: A Proof-Theoretical Study**, Almqvist and Wiksell, Stockholm (1965).

[13] D. Ripley, *Strong normalization in core type theory*, [in:] I. Sedlár, M. Blicha (eds.), **The Logica Yearbook 2019**, College Publications (2020), pp. 111–130.

[14] M. H. Sørensen, P. Urzyczyn, **Lectures on the Curry-Howard isomorphism**, Elsevier (2006).

[15] N. Tennant, **Anti-Realism and Logic**, Oxford University Press, Oxford (1987).

[16] N. Tennant, *Natural deduction and sequent calculus for intuitionistic relevant logic*, **Journal of Symbolic Logic**, vol. 52(3) (1987), pp. 665–680, DOI: https://doi.org/10.2307/2274355.

[17] N. Tennant, **Autologic: Proof theory and automated deduction**, Edinburgh University Press, Edinburgh (1992).

[18] N. Tennant, *On Paradox without Self-Reference*, **Analysis**, vol. 55(3) (1995), pp. 199–207, DOI: https://doi.org/10.2307/3328581.

[19] N. Tennant, *What is Negation?*, [in:] D. M. Gabbay, H. Wansing (eds.), **Negation, Absurdity, and Contrariety**, Kluwer Academic Publishers, Dordrecht (1999), pp. 199–222, DOI: https://doi.org/10.1007/978-94-015-9309-0_10.

[20] N. Tennant, *Ultimate normal forms for parallelized natural deductions*, **Logic Journal of the IGPL**, vol. 10(3) (2002), pp. 299–337, DOI: https://doi.org/doi.org/10.1093/jigpal/10.3.299.

[21] N. Tennant, *Cut for Core Logic*, **Review of Symbolic Logic**, vol. 5(3) (2012), pp. 450–479, DOI: https://doi.org/10.1017/S1755020311000360.

[22] N. Tennant, **Core Logic**, Oxford University Press, Oxford (2017).

[23] H. Wansing, *Proofs, disproofs, and their duals*, [in:] L. Beklemishev, V. Goranko, V. Shehtman (eds.), **Advances in Modal Logic**, vol. 8, College Publications (2010), pp. 483–505.

[24] H. Wansing, *Falsification, natural deduction and bi-intuitionistic logic*, **Journal of Logic and Computation**, vol. 26(1) (2016), pp. 425–450, DOI: https://doi.org/10.1093/logcom/ext035.

[25] H. Wansing, *A more general general proof theory*, **Journal of Applied Logic**, vol. 25(1) (2017), pp. 23–46, DOI: https://doi.org/10.1016/j.jal.2017.01.002.

Emma van Dijk
Independent scholar
Melbourne, VIC, Australia
e-mail: emmavandijk168@gmail.com

David Ripley
Monash University
Philosophy Department, SOPHIS
Building 11, Monash University
Clayton, VIC, Australia
e-mail: davewripley@gmail.com

Julian Gutierrez
Monash University
Department of Data Science & AI
Woodside Building, Monash University
Clayton, VIC, Australia
e-mail: julian.gutierrez@monash.edu