Programming Research Group

INVESTIGATIONS ON THE DUAL CALCULUS

Nikos Tzevelekos[1]

PRG-RR-04-21

Oxford University Computing Laboratory
Wolfson Building, Parks Road, Oxford OX1 3QD

[1]Nikolaos.Tzevelekos@comlab.ox.ac.uk
This work was submitted in partial fulfillment of the requirements for the Degree of MSc in Mathematics and Foundations of Computer Science.

Abstract

The Dual Calculus, proposed by Wadler in 2003, is the outcome of two distinct lines of research in theoretical computer science:
A. Efforts to extend the Curry-Howard isomorphism, established between simply-typed lambda calculus and intuitionistic logic, to classical logic.
B. Efforts to establish the tacit conjecture that call-by-value reduction in lambda calculus is dual to call-by-name reduction.
This project is aiming at introducing the Dual Calculus, examining its syntactic behaviour, and investigating possible extensions of it to polymorphic types.

Contents

1 Introduction

1.1 Two lines of research leading to the Dual Calculus

The Dual Calculus, proposed by Wadler in [Wad03a], is the outcome of two distinct lines of research in theoretical computer science:

1. Efforts to extend the Curry-Howard isomorphism, established between simply-typed lambda calculus and intuitionistic logic, to classical logic.

2. Efforts to establish the tacit conjecture that call-by-value reduction in lambda calculus is dual to call-by-name reduction.

Regarding the first line of investigation, the Curry-Howard isomorphism correlates two seemingly alien scientific fields, namely proof theory and type theory. It states a correspondence between systems of formal logic and computational calculi: logic formulas are related to types, and logic proofs are related to typed terms. More than that, proof normalization is related to term reduction. Of course, this correspondence is of great importance, since it allows to use methods and properties of the one field for the other, and it drives to a deeper understanding of foundational matters in theoretical computer science.

Traditionally, classical logic was not taken into account in the Curry-Howard isomorphism (see, for example, [GTL89, SU98]). The first attempt to add classical constructs to a computational calculus is present in the work of Griffin [Gri90], who defined a simply-typed lambda calculus in which the law of double-negation elimination was expressed in the typing rules. Griffin's rule would read:

If M is a term of type $\neg\neg A$, then $\mathcal{C}(M)$ is a term of type A

$\mathcal{C}$ is a control operator[2] which adds further expressive power to the simply-typed lambda calculus by allowing for some non-trivial jumps in computation. For example, using $\mathcal{C}$ we can define the `call/cc` operator of Scheme language.

After the work of Griffin, the view that classical constructs could be used to extend programming control features which would otherwise not be expressible in logical terms became increasingly widespread. Parigot [Par92] refined the idea of Griffin to a more concrete calculus, the $\lambda\mu$-calculus. This calculus is an extension of lambda calculus where one has the ability to name arbitrary subterms of a term by μ-variables and to abstract on these names. Thus, operations can be applied directly to subterms of a term and control features such as $\mathcal{C}$ can be easily simulated in the $\lambda\mu$-calculus. Using this "naming mechanism", Parigot was able to derive a typed $\lambda\mu$-calculus corresponding to a natural deduction system with multiple conclusions. This latter system, called Classical Natural Deduction, is a system of classical logic.

A different approach was taken by Barbanera and Berardi [BB96], who proposed a classical simply-typed lambda calculus equipped with the following set of types:

$$\text{Type } A, B \ ::= \ X \mid \neg X \mid A \vee B \mid A \wedge B$$

[2]In fact, $\mathcal{C}$ was introduced by Felleisen, see more, for example, in [FH92].

with X standing for type variables. Thus, negation is a primitive type constructor in this calculus, yet constrained only to type variables. Negation is extended to all types by the usual De Morgan laws, and thus the authors manage to identify any type A with $\neg\neg A$. Hence, having the law of double negation embedded in the syntax, this calculus (named Symmetric λ-calculus) corresponds to propositional classical logic. Further investigation on this calculus to second-order was done by Parigot [Par00].
Regarding the second line of investigation, the notion of 'duality' between call-by-value (CBV) and call-by-name (CBN) reduction was first suggested by Filinski [Fil89]. Filinski defined a symmetric lambda-calculus (SLC) in which there exist two distinct syntactic classes: *values* and *continuations*. The notion of a continuation was a well established one at the time:

> in any computation being part of a program there is some "rest of the program" ready to absorb the result of the given computation and *continue* with execution of following commands.

This "rest of the program" is called a continuation ([SW74]). Thus, there is some kind of duality (or symmetry) between values and continuations in programming languages, in that values yield data whereas continuations absorb data. This duality is part of SLC and Filinski suggests that a similar kind of duality holds between CBV and CBN reduction (or evaluation) strategies for SLC.
The suggestions of Filinski where established by Selinger in [Sel01][3], who worked in the $\lambda\mu$-calculus. Selinger showed categorical duality between CBV and CBN reduction in the $\lambda\mu$-calculus by use of control categories to model the CBN semantics and co-control categories to model the CBV semantics.
Finally, Curien and Herbelin [CH00] defined the $\bar{\lambda}\mu\tilde{\mu}$-calculus, which is an extension of the $\lambda\mu$-calculus having duals for λ- and μ-abstraction. In order to type these dual abstractions, a difference $(-)$ type constructor is included in the typed version of the calculus. This selection is due to the fact that difference is the De Morgan dual of implication – even though the operational interpretation of difference is not very intuitive. For this calculus it is shown that CBV is dual to CBN in a De Morgan fashion.

1.1.1 Summary

In this chapter we will firstly present the definition of the Dual Calculus of Wadler. Next, we are going to demonstrate the Curry-Howard isomorphism of this calculus to classical logic, and also show the duality between CBV and CBN reduction relations in the calculus. Both of these topics were, of course, discussed in [Wad03a]. Finally, we are going to present a simple embedding of simply-typed lambda calculus to the Dual Calculus, thus clarifying the point that the latter is an extension of the former.
In the second chapter we will investigate syntactic properties of the Dual Calculus under CBV reduction. Namely, we will examine the Church-Rosser and Strong Normalization properties. These are original contributions and, in rough lines, follow and extend standard techniques from the literature.

[3] first published in 1998

Finally, in the third chapter we will attempt to extend the calculus to polymorphic types, by adding second-order quantifiers over types. However, this extension is not an easy one, since problems with subject reduction can easily arise. We will give several formulations in order to overcome and to understand the difficulties that arise.

1.2 The Dual Calculus

The lines of investigation presented above lead to the Dual Calculus ([Wad03a]), which epitomizes both properties of being the Curry-Howard equivalent of classical logic and of its CBV and CBN reduction relations being De Morgan duals. More than that, the calculus has the advantage of simplicity in syntax and operational semantics.

1.2.1 Definition

The Dual Calculus (`DuCa`) consists of *types* and *objects*, in the same way that simply-typed lambda calculus consists of types and terms. The types are the same as the formulas of propositional logic, whereas the objects are divided in *terms, coterms* and *statements.* The intended computational interpretation is this of terms being objects yielding data, whereas coterms absorb data[4]. In fact, this is very similar to the notion of *values* and *continuations*, as presented in [Fil89]. The statements of `DuCa` represent *cuts* of terms upon coterms, that is constructions consisting of a term and a coterm, where the term is yielding data to be absorbed by the coterm.
Below we give the definition of `DuCa` we will be using throughout this project.

Definition 1.1 (Dual Calculus. [Wad03b])
`DuCa` consists of Types and Objects. The set of objects is the union of the sets of Terms, Coterms and Statements:

$$
\begin{array}{llll}
\text{Type} & A, B & ::= & X \mid A\&B \mid A \vee B \mid \neg A \\
\text{Object} & G, H & ::= & M \mid K \mid S \\
\text{Term} & M, N & ::= & x \mid \langle M, N\rangle \mid \langle M\rangle\mathtt{inl} \mid \langle N\rangle\mathtt{inr} \mid [K]\mathtt{not} \mid (S).\alpha \\
\text{Coterm} & K, L & ::= & \alpha \mid [K, L] \mid \mathtt{fst}[K] \mid \mathtt{snd}[L] \mid \mathtt{not}\langle M\rangle \mid x.(S) \\
\text{Statement} & S, T & ::= & M \bullet K
\end{array}
$$

The typing rules (or inference rules) involve three forms of sequents:

$$
\begin{array}{rl}
\text{Left Sequent} & K : A \,\rule{2pt}{8pt}\, \Gamma \rightarrow \Theta \\
\text{Right Sequent} & \Gamma \rightarrow \Theta \,\rule{2pt}{8pt}\, M : A \\
\text{Center Sequent} & \Gamma \,\rule{2pt}{8pt}\, M \bullet K \,\rule{2pt}{8pt}\!\rightarrow \Theta
\end{array}
$$

where Γ and Θ are *antecedent* and *succedent* sets respectively:

$$
\begin{array}{llll}
\text{Antecedent} & \Gamma, \Delta & ::= & \{x_1 : A_1, \ldots, x_n : A_n\} \\
\text{Succecedent} & \Theta, \mathrm{I} & ::= & \{\alpha_1 : B_1, \ldots, \alpha_m : B_m\}
\end{array}
$$

[4]The use of "data" is preferred here instead of the standard "value", since the latter is reserved for terms of special syntax.

We will usually omit the outer brackets in succedent and antecedent sets and use comma notation for set union (e.g. $\Gamma, \Gamma' \equiv \Gamma \cup \Gamma'$).
The typing (or inference) rules of `DuCa` are:

$$\frac{}{\alpha : A \mid \Gamma \rightarrow \Theta, \alpha : A}\ \mathtt{id}L \qquad \frac{}{x : A, \Gamma \rightarrow \Theta \mid x : A}\ \mathtt{id}R$$

$$\frac{K : A \mid \Gamma \rightarrow \Theta}{\mathtt{fst}[K] : A\&B \mid \Gamma \rightarrow \Theta} \quad \frac{L : B \mid \Gamma \rightarrow \Theta}{\mathtt{snd}[L] : A\&B \mid \Gamma \rightarrow \Theta}\ \&L \qquad \frac{\Gamma \rightarrow \Theta \mid M : A \quad \Gamma \rightarrow \Theta \mid N : B}{\Gamma \rightarrow \Theta \mid \langle M, N \rangle : A\&B}\ \&R$$

$$\frac{\Gamma \rightarrow \Theta \mid M : A}{\Gamma \rightarrow \Theta \mid \langle M \rangle \mathtt{inl} : A \vee B} \quad \frac{\Gamma \rightarrow \Theta \mid N : B}{\Gamma \rightarrow \Theta \mid \langle N \rangle \mathtt{inr} : A \vee B}\ \vee R \qquad \frac{K : A \mid \Gamma \rightarrow \Theta \quad L : B \mid \Gamma \rightarrow \Theta}{[K, L] : A \vee B \mid \Gamma \rightarrow \Theta}\ \vee L$$

$$\frac{\Gamma \rightarrow \Theta \mid M : A}{\mathtt{not}\langle M \rangle : \neg A \mid \Gamma \rightarrow \Theta}\ \neg L \qquad \frac{K : A \mid \Gamma \rightarrow \Theta}{\Gamma \rightarrow \Theta \mid [K]\mathtt{not} : \neg A}\ \neg R$$

$$\frac{x : A, \Gamma \mid S \mid\!\rightarrow \Theta}{x.(S) : A \mid \Gamma \rightarrow \Theta}\ LI \qquad \frac{\Gamma \mid S \mid\!\rightarrow \Theta, \alpha : A}{\Gamma \rightarrow \Theta \mid (S).\alpha : A}\ RI$$

$$\frac{\Gamma \rightarrow \Theta \mid M : A \quad K : A \mid \Gamma \rightarrow \Theta}{\Gamma \mid M \bullet K \mid\!\rightarrow \Theta}\ \mathrm{Cut}$$

The above rules form the sequent calculus `GW`. $\dashv$

In the definition above note that there is no need to include rules for Weakening or Contraction, since these are derived from the above (proposition 1.6).
The letters we use to denote components of `DuCa` are standard in this project. As above, for types we use capital letters opening the alphabet $(A, B, C, ...)$; for variables we use $x, y, z, ...$; for terms M, N; for covariables $\alpha, \beta, \gamma, ...$; for coterms K, L; for objects G, H; for antecedent sets Γ, Δ; for succedent sets Θ, I.
In `DuCa`, variable and covariable abstractions are performed by dots '.' . For example, the rule LI introduces the variable abstraction $x.(S)$. Consequently, we have the usual convention for free and bound occurrences of variables and covariables. For example, in $x.(S)$ all occurrences of x inside S are bound, while the occurrence of x right before the dot is transparent. Similar things hold for covariables.
The intended computational interpretation of objects in `DuCa` is as aforementioned: terms stand for computations yielding data, whereas coterms stand for computations absorbing data. Thus, sequents stand for computational scenarios where one supplies data to all variables (and coterms) of the sequent, and expects the computation to pass data to some covariable (or a term to yield data).
Under this interpretation, a term x trivially yields the data supplied to x and a term $\langle M, N \rangle$ yields a pair of data of type $A\&B$, consisting of the data yielded by terms M and N. Hence, the type conjunction $A\&B$ corresponds to product of types. Dually, a coterm α absorbs the data passed to α and a coterm $[K, L]$ absorbs a data of type $A \vee B$, which is passed on to K or L according to this data being a left or right injection. Therefore, $A \vee B$ corresponds to sum of types. For further details on the intended computational

interpretation see [Wad03a].
Note that the inference rules of GW are very similar to the inference rules of system LK of Gentzen [Gen35] restricted to propositional logic[5]. In the following subsection we will see that this similarity is in fact a Curry-Howard isomorphism.
Now, though we advertised that DuCa is an extension of simply-typed lambda calculus, one can notice that in the definition above there aren't any λ's whatsoever. This is not important though, since we can define abbreviations standing for constructions similar to λ-abstractions and applications. Such abbreviations will be presented shortly (definition 1.5), and in section 1.3 we will see in detail why DuCa is an extension of simply-typed lambda calculus.
Back to GW, since we'll be using sequent derivations, it is useful to give them a formal definition.

Definition 1.2
Let $\mathcal{S}$ be some sequent calculus. A derivation $\mathcal{D}$ in $\mathcal{S}$ is a labelled tree, whose nodes are labelled by sequents. A node σ is connected to parental nodes $\sigma_1, ..., \sigma_n$, if there is an instance of an inference rule in $\mathcal{S}$ having $\sigma_1, ..., \sigma_n$ as premises and σ as conclusion.
We say that $\mathcal{D}$ derives the sequent σ, if $\mathcal{D}$ is some derivation with root σ. σ is derivable in $\mathcal{S}$, if there is some $\mathcal{D}$ which derives σ. $\dashv$

It is also useful to introduce some notation in relation to objects of DuCa being typed by sequents of GW.

Definition 1.3
Let $G, G' \in$ DuCa and σ, σ' be sequents in GW, then

- $\sigma \in T_G(A, \Gamma, \Theta)$ if σ is derivable in GW and either

$$\begin{array}{lll} \sigma & \equiv & \Gamma \rightarrow \Theta \,\rvert\, M : A \quad \text{and } G \text{ is a term } M, \text{ or} \\ \sigma & \equiv & K : A \,\rvert\, \Gamma \rightarrow \Theta \quad \text{and } G \text{ is a coterm } K, \text{ or} \\ \sigma & \equiv & \Gamma \,\rvert\, S \,\rvert\!\rightarrow \Theta \quad \text{and } G \text{ is a statement } S. \end{array}$$

- $\sigma \in T_G(A)$ if $\sigma \in T_G(A, \Gamma, \Theta)$ for some Γ, Θ.
- $\sigma \in T_G(\Gamma, \Theta)$ if $\sigma \in T_G(A, \Gamma, \Theta)$ for some type A.
- $\sigma \in T_G$ if $\sigma \in T_G(A, \Gamma, \Theta)$ for some A, Γ, Θ. *G is typed* if $T_G \neq \emptyset$.

$\dashv$

According to the above, if $\sigma \in T_G$ and $\sigma \in T_H$, then $G \equiv H$. Moreover, for all statements S and types A, $T_S = T_S(A)$. This reflects the fact that **statements are not assigned types**, but are typed iff their components are typed with the same type. Further, an object G of the calculus can be assigned more than one type and, on the other hand, we allow for elements G with $T_G = \emptyset$. For example, $\langle x, x \rangle \bullet [a, a]$ is not typed.

[5] One may also note that symbols for logical connectives follow Gentzen's formulations.

1.2.2 Curry-Howard isomorphism

In order for the connection of `DuCa` with classical logic to become clearer, we will temporarily use an alternative sequent calculus for `DuCa`, equivalent to the initial `GW`, the system `GW1`:

Definition 1.4 (System `GW1`. [Wad03a])
`GW1` consists of left, right and center sequents. These are as in definition 1.1, with the only difference of having succedent and antecedent *sequences* instead of sets; that is,

$$\begin{array}{lll} \text{Antecedent} \quad \Gamma, \Delta & ::= & x_1 : A_1, \ldots, x_n : A_n \\ \text{Succecedent} \quad \Theta, \mathrm{I} & ::= & \alpha_1 : B_1, \ldots, \alpha_m : B_m \end{array}$$

The inference rules of the calculus are of two kinds:
Logical (or operational) rules:

$$\dfrac{}{\alpha : A \mid\ \rightarrow \alpha : A}\,\mathtt{id}L \qquad \dfrac{}{x : A \rightarrow\ \mid x : A}\,\mathtt{id}R$$

$$\dfrac{K : A \mid \Gamma \rightarrow \Theta}{\mathtt{fst}[K] : A\&B \mid \Gamma \rightarrow \Theta} \quad \dfrac{L : B \mid \Gamma \rightarrow \Theta}{\mathtt{snd}[L] : A\&B \mid \Gamma \rightarrow \Theta}\,\&L \qquad \dfrac{\Gamma \rightarrow \Theta \mid M : A \quad \Gamma \rightarrow \Theta \mid N : B}{\Gamma \rightarrow \Theta \mid \langle M, N\rangle : A\&B}\,\&R$$

$$\dfrac{\Gamma \rightarrow \Theta \mid M : A}{\Gamma \rightarrow \Theta \mid \langle M\rangle\mathtt{inl} : A \vee B} \quad \dfrac{\Gamma \rightarrow \Theta \mid N : B}{\Gamma \rightarrow \Theta \mid \langle N\rangle\mathtt{inr} : A \vee B}\,\vee R \qquad \dfrac{K : A \mid \Gamma \rightarrow \Theta \quad L : B \mid \Gamma \rightarrow \Theta}{[K, L] : A \vee B \mid \Gamma \rightarrow \Theta}\,\vee L$$

$$\dfrac{\Gamma \rightarrow \Theta \mid M : A}{\mathtt{not}\langle M\rangle : \neg A \mid \Gamma \rightarrow \Theta}\,\neg L \qquad \dfrac{K : A \mid \Gamma \rightarrow \Theta}{\Gamma \rightarrow \Theta \mid [K]\mathtt{not} : \neg A}\,\neg R$$

$$\dfrac{x : A, \Gamma \mid S \mid\rightarrow \Theta}{x.(S) : A \mid \Gamma \rightarrow \Theta}\,LI \qquad \dfrac{\Gamma \mid S \mid\rightarrow \Theta, \alpha : A}{\Gamma \rightarrow \Theta \mid (S).\alpha : A}\,RI$$

$$\dfrac{\Gamma \rightarrow \Theta \mid M : A \quad K : A \mid \Delta \rightarrow \mathrm{I}}{\Gamma, \Delta \mid M \bullet K \mid\rightarrow \Theta, \mathrm{I}}\,\text{Cut}$$

Structural rules for left, right and center sequents; for left sequents these are:

$$\dfrac{K : A \mid \Gamma \rightarrow \Theta}{K : A \mid x : A, \Gamma \rightarrow \Theta} \quad \text{Weakening} \quad \dfrac{K : A \mid \Gamma \rightarrow \Theta}{K : A \mid \Gamma \rightarrow \Theta, \alpha : A}$$

$$\dfrac{K : A \mid x : B, y : B, \Gamma \rightarrow \Theta}{K\{y/x\} : A \mid y : B, \Gamma \rightarrow \Theta} \quad \text{Contraction} \quad \dfrac{K : A \mid \Gamma \rightarrow \Theta, \alpha : B, \beta : B}{K\{\beta/\alpha\} : A \mid \Gamma \rightarrow \Theta, \beta : B}$$

$$\dfrac{K : A \mid \Delta, x : A, y : B, \Gamma \rightarrow \Theta}{K : A \mid \Delta, y : B, x : A, \Gamma \rightarrow \Theta} \quad \text{Interchange} \quad \dfrac{K : A \mid \Gamma \rightarrow \Theta, \alpha : A, \beta : B, \mathrm{I}}{K : A \mid \Gamma \rightarrow \Theta, \beta : B, \alpha : A, \mathrm{I}}$$

and similarly for center and right sequents. $\dashv$

It is time to define abbreviations for implicational types and objects in `DuCa`.

Definition 1.5
For any $A, B, x, M, N, L \in$ `DuCa` we define the following abbreviations:

$$
\begin{aligned}
A \supset B &\equiv \neg(A \& \neg B) \\
\lambda x.M &\equiv [z.(z \bullet \mathtt{fst}[x.(z \bullet \mathtt{snd}[\mathtt{not}\langle M \rangle])])]\mathtt{not} \\
N@L &\equiv \mathtt{not}\langle\langle N, [L]\mathtt{not}\rangle\rangle
\end{aligned}
$$

and thus $A \supset B$ is a type, $\lambda x.M$ a term, and $N@L$ a coterm of `DuCa`. $\dashv$

For the constructs we have just defined, we have the following familiar inference rules holding (i.e. being derivable) in `GW1`:

$$
\frac{\Gamma \rightarrow \Theta \mid M : A \quad K : B \mid \Delta \rightarrow \mathrm{I}}{M@K : A \supset B \mid \Gamma, \Delta \rightarrow \Theta, \mathrm{I}} \supset L
\qquad
\frac{x : A, \Gamma \rightarrow \Theta \mid M : B}{\Gamma \rightarrow \Theta \mid \lambda x.M : A \supset B} \supset R
$$

It is not difficult now to see that we have gathered in `GW1` all the inference rules of Gentzen's LK (presented in the Appendix), apart from rules with quantifiers. It is therefore straightforward to deduce:

> The Dual Calculus with typing rules these of `GW1` is isomorphic in Curry-Howard style to LK restricted to propositional logic. Types of `DuCa` correspond to formulas of classical propositional logic, and objects correspond to proofs.

The only purpose for introducing `GW1` was this of demonstrating the Curry-Howard isomorphism with classical logic[6]. In the sequel we will abandon `GW1` and return to our initial sequent calculus `GW` of definition 1.1. This is mainly because it is much easier to deal with antecedent and succedent sets instead of sequences. Moreover, `GW` satisfies the structural rules of `GW1`, as these are embedded inside its syntax. The latter is shown in the following proposition.

Proposition 1.6 *Weakening and Contraction are derived rules of* `GW`.

Proof: By induction on derivations. ∎

Of course, there is no need for exchange rules in `GW`, since we use antecedent and succedent sets. Note that the above proposition may be used in the sequel frequently without being mentioned.
Finally, assuming we have a standard translation from antecedent and succedent sequences to antecedent and succedent sets, and viceversa, it is not difficult to show that:

`GW` and `GW1` are equivalent.

We don't give any further details on the statement above and this is done on purpose, since there are many technical details involved and a full argument could be uncomfortably long.

[6]Strictly speaking, the isomorphism described above is a mere bijection. The true isomorphism will become evident when we add a reduction relation to `DuCa`, which will correspond to cut-elimination in propositional LK.

1.2.3 Reduction relations

In [Wad03a] two reduction relations are proposed for DuCa, representing call-by-value (CBV) and call-by-name (CBN) approaches. In fact, both these relations are restrictions of a basic reduction relation, which we present below.

Definition 1.7 (Basic Reduction R_b)
R_b is the one-step reduction relation yielded by the reduction rules listed below, when these are applied on subobjects of DuCa objects.

$$
\begin{array}{lll}
(\beta\&_1) & \langle M,N\rangle \bullet \mathtt{fst}[K] & \rightarrow M\bullet K \\
(\beta\&_2) & \langle M,N\rangle \bullet \mathtt{snd}[L] & \rightarrow N\bullet L \\
(\beta\vee_1) & \langle M\rangle\mathtt{inl} \bullet [K,L] & \rightarrow M\bullet K \\
(\beta\vee_2) & \langle N\rangle\mathtt{inr} \bullet [K,L] & \rightarrow N\bullet L \\
(\beta\neg) & [K]\mathtt{not} \bullet \mathtt{not}\langle M\rangle & \rightarrow M\bullet K \\
(\beta L) & M\bullet x.(S) & \rightarrow S\{M/x\} \\
(\beta R) & (S).\alpha\bullet K & \rightarrow S\{K/\alpha\} \\
 & & \\
(\eta L) & K & \rightarrow x.(x\bullet K) \\
(\eta R) & M & \rightarrow (M\bullet\alpha).\alpha \\
 & & \\
(\nu\&_1) & \langle M,N\rangle\bullet K & \rightarrow M\bullet x.(\langle x,N\rangle\bullet K) \\
(\nu\&_2) & \langle M,N\rangle\bullet K & \rightarrow N\bullet y.(\langle M,y\rangle\bullet K) \\
(\nu\vee_3) & \langle M\rangle\mathtt{inl}\bullet K & \rightarrow M\bullet x.(\langle x\rangle\mathtt{inl}\bullet K) \\
(\nu\vee_4) & \langle N\rangle\mathtt{inr}\bullet K & \rightarrow N\bullet y.(\langle y\rangle\mathtt{inr}\bullet K) \\
(\nu\vee_1) & M\bullet[K,L] & \rightarrow (M\bullet[\alpha,L]).\alpha\bullet K \\
(\nu\vee_2) & M\bullet[K,L] & \rightarrow (M\bullet[K,\beta]).\beta\bullet L \\
(\nu\&_3) & M\bullet\mathtt{fst}[K] & \rightarrow (M\bullet\mathtt{fst}[\alpha]).\alpha\bullet K \\
(\nu\&_4) & M\bullet\mathtt{snd}[L] & \rightarrow (M\bullet\mathtt{snd}[\beta]).\beta\bullet L
\end{array}
$$

For $G,H \in$ DuCa, $(G,H)\in R_b$ is usually written $G \longrightarrow^b H$. ⊣

In rules βL and βR we notice the introduction of *substitutions*: the statement $S\{M/x\}$ is obtained from S if we substitute M for all the free occurrences of x in S. Similarly, $S\{K/\alpha\}$ is obtained.
Note that we will often use explicit notation in reduction arrows to indicate the reduction rule responsible for a reduction[7]. For example, if $G \longrightarrow^b H$ because of some redex $\langle M,N\rangle\bullet\mathtt{fst}[K]$ inside G reducing to $M\bullet K$, we may write $G \xrightarrow{\beta\&_1}{}^b H$. Note also that in rules ηL and ηR, x and α, respectively, are fresh.
The β-reduction rules listed above correspond to familiar cut elimination techniques for LK (see for example [GTL89]), and demonstrate further the Curry-Howard isomorphism. η-expansions entangle the syntax of objects, thus their translations to LK represent non-meaningful additions of cuts. Finally, ν-rules also entangle the syntax, and their utility

[7]In this thesis "reduction" is often identified with what some authors call "reduction step".

in `DuCa` will be seen clearly later, when we introduce CBV and CBN reduction relations. At this point, it is useful to introduce some notation on reduction relation closures which is standard in this project. We introduce it as definition for easier reference.

Definition 1.8
Suppose R is some reduction relation denoted by $\longrightarrow$, then

- its reflexive closure is denoted by $\longrightarrow_=$
- its transitive closure is denoted by $\longrightarrow_+$
- its reflexive transitive closure is denoted by $\twoheadrightarrow$ $\dashv$

Due to duality being present in its reduction rules, R_b is not confluent. For example, $(x \bullet \alpha).\beta \bullet y.(z \bullet \gamma)$ reduces both to $x \bullet \alpha$ and to $z \bullet \gamma$. Confluent reduction relations can be obtained by placing restrictions on R_b. In particular, restrictions placed on R_b can lead to confluent CBV and CBN reduction relations.
In order for CBV and CBN reduction relations to be defined, one should define first which objects are to be considered as values. Due to the computational interpretation of terms and coterms, values are a subset of terms. Dually, one defines covalues as coterms of special structure. The intuition behind these definitions can be found in [Wad03a].

Definition 1.9 (Values and Covalues)

$$\begin{array}{lll} \text{Value} & V, W & ::= \; x \mid \langle V, W\rangle \mid \langle V\rangle\texttt{inl} \mid \langle W\rangle\texttt{inr} \mid [K]\texttt{not} \\ \text{Covalue} & P, Q & ::= \; \alpha \mid [P, Q] \mid \texttt{fst}[P] \mid \texttt{snd}[Q] \mid \texttt{not}\langle M\rangle \end{array}$$ $\dashv$

Thus, **variable abstractions are not values**, and similarly for covariable abstractions. The reader should not make any correlations with lambda calculus and the fact that in the latter λ-abstractions are taken as values, since variable (and covariable) abstractions of `DuCa` are not quite the same thing. This will become clearer though in the next section. The reader is also advised to keep in mind the letters used to denote values and covalues, since these are standard in this project.
It is almost evident that covalues are in a way the duals of values. In fact, covalues are to call-by-name what values are to call-by-value.

Definition 1.10 (CBV Reduction R_v and CBN Reduction R_n. [Wad03a])
The call-by-value (CBV) reduction relation R_v is the one-step reduction relation yielded by the CBV reduction rules listed below, when these are applied to subobjects of `DuCa` objects. For $G, H \in$ `DuCa`, $(G, H) \in R_v$ is usually written $G \longrightarrow^v H$.
The call-by-name (CBN) reduction relation R_n is the one-step reduction relation yielded by the CBN reduction rules listed below, when these are applied to subobjects of `DuCa`

objects. For $G, H \in \texttt{DuCa}$, $(G, H) \in R_n$ is usually written $G \longrightarrow^n H$.

	CBV rules			CBN rules		
$(\beta\&_1)$	$\langle V, W\rangle \bullet \texttt{fst}[K]$	$\to$	$V \bullet K$	$\langle M, N\rangle \bullet \texttt{fst}[P]$	$\to$	$M \bullet P$
$(\beta\&_2)$	$\langle V, W\rangle \bullet \texttt{snd}[L]$	$\to$	$W \bullet L$	$\langle M, N\rangle \bullet \texttt{snd}[Q]$	$\to$	$N \bullet Q$
$(\beta\vee_1)$	$\langle V\rangle\texttt{inl} \bullet [K, L]$	$\to$	$V \bullet K$	$\langle M\rangle\texttt{inl} \bullet [P, Q]$	$\to$	$M \bullet P$
$(\beta\vee_2)$	$\langle W\rangle\texttt{inr} \bullet [K, L]$	$\to$	$W \bullet L$	$\langle N\rangle\texttt{inr} \bullet [P, Q]$	$\to$	$N \bullet Q$
$(\beta\neg)$	$[K]\texttt{not} \bullet \texttt{not}\langle M\rangle$	$\to$	$M \bullet K$	$[K]\texttt{not} \bullet \texttt{not}\langle M\rangle$	$\to$	$M \bullet K$
(βL)	$V \bullet x.(S)$	$\to$	$S\{V/x\}$	$M \bullet x.(S)$	$\to$	$S\{M/x\}$
(βR)	$(S).\alpha \bullet K$	$\to$	$S\{K/\alpha\}$	$(S).\alpha \bullet P$	$\to$	$S\{P/\alpha\}$
(ηL)	K	$\to$	$x.(x \bullet K)$	K	$\to$	$x.(x \bullet K)$
(ηR)	M	$\to$	$(M \bullet \alpha).\alpha$	M	$\to$	$(M \bullet \alpha).\alpha$
$(\nu\&_1)$	$\langle M, N\rangle \bullet K$	$\to$	$M \bullet x.(\langle x, N\rangle \bullet K)$	$M \bullet \texttt{fst}[K]$	$\to$	$(M \bullet \texttt{fst}[\alpha]).\alpha \bullet K$
$(\nu\&_2)$	$\langle V, N\rangle \bullet K$	$\to$	$N \bullet y.(\langle V, y\rangle \bullet K)$	$M \bullet \texttt{snd}[L]$	$\to$	$(M \bullet \texttt{snd}[\beta]).\beta \bullet L$
$(\nu\vee_1)$	$\langle M\rangle\texttt{inl} \bullet K$	$\to$	$M \bullet x.(\langle x\rangle\texttt{inl} \bullet K)$	$M \bullet [K, L]$	$\to$	$(M \bullet [\alpha, L]).\alpha \bullet K$
$(\nu\vee_2)$	$\langle N\rangle\texttt{inr} \bullet K$	$\to$	$N \bullet y.(\langle y\rangle\texttt{inr} \bullet K)$	$M \bullet [P, L]$	$\to$	$(M \bullet [P, \beta]).\beta \bullet L$

$\dashv$

One clearly notes the duality in the definitions of R_v and R_n, which proposes that sums are treated as 'duals' of products.
Another property to be discussed further ahead is this of confluence. Though it is clear that the case of the critical pair $(x \bullet \alpha).\beta \bullet y.(z \bullet \gamma)$ is now resolved in CBV or CBN, it is not clear that confluence (also called the *Church-Rosser property*) holds for these reduction relations. This is studied in the next chapter, and it is shown that indeed confluence holds under CBV and CBN.

1.2.4 CBV is dual to CBN

In DuCa there is a De Morgan duality present, as advertised in several occasions above. It is time to see a translation inside DuCa demonstrating this duality. We assume that there is a bijection mapping every variable x to a covariable x^o, and every covariable α to a variable α^o, such that $x^{oo} \equiv x$ and $\alpha^{oo} \equiv \alpha$.

Definition 1.11 (De Morgan translation. [Wad03a])
For any type A and object G in DuCa, their De Morgan translations A^o and G^o are defined

recursively by:

$$
\begin{array}{lcl}
(X)^o & \equiv & X \\
(A\&B)^o & \equiv & A^o \vee B^o \\
(A \vee B)^o & \equiv & A^o \& B^o \\
(\neg A)^o & \equiv & \neg A^o
\end{array}
$$

$$
\begin{array}{lcllcl}
(x)^o & \equiv & x^o & (\alpha)^o & \equiv & \alpha^o \\
(\langle M, N \rangle)^o & \equiv & [M^o, N^o] & ([K, L])^o & \equiv & \langle K^o, L^o \rangle \\
(\langle M \rangle \texttt{inl})^o & \equiv & \texttt{fst}[M^o] & (\texttt{fst}[K])^o & \equiv & \langle K^o \rangle \texttt{inl} \\
(\langle N \rangle \texttt{inr})^o & \equiv & \texttt{snd}[N^o] & (\texttt{snd}[L])^o & \equiv & \langle L^o \rangle \texttt{inr} \\
(\texttt{not}\langle M \rangle)^o & \equiv & [M^o]\texttt{not} & ([K]\texttt{not})^o & \equiv & \texttt{not}\langle K^o \rangle \\
((S).\alpha)^o & \equiv & \alpha^o.(S^o) & (x.(S))^o & \equiv & (S^o).x^o \\
 & & (M \bullet K)^o & \equiv & K^o \bullet M^o &
\end{array}
$$

The translation is extended to antecedent and succedent sets and to sequents by:

$$
\begin{array}{lcl}
(x_1 : A_1, \ldots, x_n : A_n)^o & \equiv & x_1^o : A_1^o, \ldots, x_n^o : A_n^o \\
(\alpha_1 : B_1, \ldots, \alpha_m : B_m)^o & \equiv & \alpha_1^o : B_1^o, \ldots, \alpha_m^o : B_m^o \\
(\Gamma \rightarrow \Theta \mid M : A)^o & \equiv & M^o : A^o \mid \Theta^o \rightarrow \Gamma^o \\
(K : A \mid \Gamma \rightarrow \Theta)^o & \equiv & \Theta^o \rightarrow \Gamma^o \mid K^o : A^o \\
(\Gamma \mid S \mid\!\rightarrow \Theta)^o & \equiv & \Theta^o \mid S^o \mid\!\rightarrow \Gamma^o
\end{array}
$$

$\dashv$

Thus, the De Morgan translation of a term is its dual coterm and vice versa. Statements have statements as duals. It is interesting to see that duality defined as above is an involution, and also that derivability is preserved under duality.

Proposition 1.12 ([Wad03a])
1. *Duality is an involution: for any type A and object G and any sequent σ,*

$$A^{oo} \equiv A \quad , \quad G^{oo} \equiv G \quad , \quad \sigma^{oo} \equiv \sigma$$

2. *A sequent σ is derivable iff σ^o is.*

Proof: Straightforward. ■

We are now ready to present the main result of this subsection, namely duality between call-by-value and call-by-name reduction relations in `DuCa`.

Proposition 1.13 (CBV is dual to CBN. [Wad03a]) *Let $G, H \in$ `DuCa`, then*

$$G \longrightarrow^v H \;\; \textit{iff} \;\; G^o \longrightarrow^n H^o$$

Proof: Straightforward, because of duality in reduction rules. ■

1.3 Embedding the lambda calculus

In this section we introduce a simple embedding of the simply-typed lambda calculus in the Dual Calculus (DuCa), which is to serve us solely in clarifying the fact that DuCa is an extension of lambda calculus. Note that in [Wad03a] CPS translations are defined, taking objects of DuCa to a restriction of the lambda calculus with sums and products[8]. More than that, translations from this latter calculus to DuCa are also defined and some nice involutive properties are shown. We clarify at this point that our task here is quite different and clearly simpler, since we only opt for a translation from the simply-typed lambda calculus to DuCa, with only further ambition the preservation of β-reduction.
The three forms of reduction relations in DuCa, namely basic, call-by-value and call-by-name reduction, are also present in the simply-typed lambda calculus. Here we will use the call-by-value relation of the simply-typed lambda calculus and therefore the call-by-value relation of DuCa. Moreover, we will use, for space economy, the abbreviations of lambda abstraction and application for DuCa defined earlier, that is:

$$\begin{aligned} A \supset B &\equiv \neg(A \& \neg B) \\ \lambda x.M &\equiv [z.(z \bullet \mathtt{fst}[x.(z \bullet \mathtt{snd}[\mathtt{not}\langle M \rangle])])]\mathtt{not} \\ N@K &\equiv \mathtt{not}\langle\langle N, [K]\mathtt{not}\rangle\rangle \end{aligned}$$

Note that under these abbreviations we have the following simulation of common β-reduction:

$$\lambda x.M \bullet V@K \xrightarrow{\beta}{}^{v} V \bullet x.(M \bullet K)$$

Now, the definition of the simply-typed lambda calculus is standard ([Bar84]).

Definition 1.14
The simply-typed lambda calculus consists of Types and Terms:

$$\begin{array}{lll} \text{Type} & A, B & ::= & X \mid A \supset B \\ \text{Term} & M, N & ::= & V \mid MN \\ \text{Value} & V & ::= & x \mid \lambda x.M \end{array}$$

The set of terms is denoted by Λ. The typing rules for this calculus are:

$$\frac{}{x : A, \Gamma \vdash x : A}\,Ax \qquad \frac{\Gamma \vdash M : A \supset B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}\,App \qquad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x.M : A \supset B}\,Abs$$

where Γ is some set of assumptions $x_1 : B_1, \ldots, x_n : B_n$.
The call-by-value reduction relation R_v^λ is yielded by the reduction rule

$$(\lambda x.M)V \to M\{V/x\}$$

being applied to subterms of terms. For $M, N \in \Lambda$, we usually write $M \longrightarrow^v N$ instead of $(M, N) \in R_v^\lambda$.
$\dashv$

[8] see [Dou93] for a definition of the lambda calculus with sums and products.

The translation of the simply-typed lambda calculus is defined below. Note that all elements (terms) of the source calculus are translated to terms of `DuCa`. It is no secret that notation for translations is taken from [Wad03a].

Definition 1.15
We define the following translation from simply-typed lambda calculus to `DuCa`:

$$\begin{array}{lcl} (A)^D & \equiv & A \\ (x)^D & \equiv & x \\ (\lambda x.M)^D & \equiv & \lambda x.(M)^d \\ (V)^d & \equiv & ((V)^D \bullet \alpha).\alpha \\ (MN)^d & \equiv & ((M)^d \bullet (N)^d @ \alpha).\alpha \end{array} \qquad \dashv$$

Note that the α's in the last two lines above are fresh.
The reason for using this two-step definition for $(M)^d$ is mainly that, thus, for every value V, $(V)^D$ is a value in `DuCa`. We can prove the following propositions.

Proposition 1.16 *For any $x, M, V, \gamma, L \in$ `DuCa`, we have that:*

$$\lambda x.M \bullet (V \bullet \gamma).\gamma @ L \xrightarrow{\beta\nu}\!\!\!\!\rightarrow^{v} V \bullet x.(M \bullet L)$$

Proof: Straightforward. ■

Proposition 1.17 *For any $M, V, x \in \Lambda$:* $(M)^d\{(V)^D/x\} \equiv (M\{V/x\})^d$

Proof: By induction on $M \in \Lambda$. ■

Proposition 1.18 *For any $M \in \Lambda$ and $\alpha \in$ `DuCa`:* $((M)^d \bullet \alpha).\alpha \xrightarrow{\beta R}{}^{v} (M)^d$.

Proof: By induction on $M \in \Lambda$. ■

Thus, we can prove that the translation defined produces the desired embedding.

Proposition 1.19 (Embedding of CBV simply-typed lambda calculus)

1. *If $M \in \Lambda$ and $\Gamma \vdash M : A$ is derivable, then $\Gamma \rightarrow\ \mathbf{I}\ (M)^d : A$ is derivable.*

2. *For any $x, M, V \in \Lambda$:* $((\lambda x.M)V)^d \xrightarrow{\beta\nu}\!\!\!\!\rightarrow^{v} (M\{V/x\})^d$.

Proof: 1 is proven by induction on the derivation of the former sequent in the simply-typed lambda calculus. For 2, we have:

$$\begin{array}{rl} ((\lambda x.M)V)^d & \equiv\ ((\lambda x.M)^d \bullet (V)^d @ \alpha).\alpha \\ & \equiv\ ((\lambda x.(M)^d \bullet \beta).\beta \bullet ((V)^D \bullet \gamma).\gamma @ \alpha).\alpha \\ & \xrightarrow{\beta R}{}^{v}\ (\lambda x.(M)^d \bullet ((V)^D \bullet \gamma).\gamma @ \alpha).\alpha \\ & \underset{prop.1.16}{\xrightarrow{\beta\nu}\!\!\!\!\rightarrow^{v}}\ ((V)^D \bullet x.((M)^d \bullet \alpha)).\alpha \ \xrightarrow{\beta L}{}^{v}\ ((M)^d\{(V)^D/x\} \bullet \alpha).\alpha \\ & \underset{prop.1.17}{\equiv}\ ((M\{V/x\})^d \bullet \alpha).\alpha \ \underset{prop.1.18}{\xrightarrow{\beta R}{}^{v}}\ (M\{V/x\})^d \end{array}$$

■

It is important to note that, under the defined translation, we need to use ν-reductions in DuCa in order to preserve β-reductions of the simply-typed lambda calculus. Indeed, ν-reductions are needed in proposition 1.16 and, in particular, the $\nu\&_1$ rule is used (in the omitted proof). This fact gives us a hint on the role of ν-rules in DuCa: they are rather complementary to β-rules, facilitating some β-reductions otherwise forbidden by CBV (or CBN) restrictions, than entirely novel rules.

2 Syntactic investigations

In this chapter we investigate some syntactic properties of the Dual Calculus under call-by-value (CBV) reduction. Because of duality with call-by-name (CBN), all results proven here have analogs for the call-by-name case.
First, we examine the untyped Dual Calculus and prove confluence, or Church-Rosser property, under CBV. The proof follows, in rough lines, the proof of $\beta\eta$-reduction being Church-Rosser (CR) in the lambda calculus, as it is presented in [Bar84].
Another important syntactic property to examine is Strong Normalization (SN). For this, we investigate the typed version of the Dual Calculus under all call-by-value reduction rules except for ν-rules; the latter omission being for simplicity. Also, restrictions are applied on η-rules, so as to avoid loops in reduction sequences. In order to prove SN for this reduction relation, we first prove SN in a simpler calculus using the method of *reducibility sets* of Tait, as described in [GTL89] for the simply-typed lambda calculus. Afterwards, we introduce the call-by-value CPS translation of the Dual Calculus, as described in [Wad03a], and cite some very useful results concerning the translation and the target calculus. Finally, using the proven results, we prove the required SN theorem. The chapter ends with a consideration of a restricted version of call-by-value reduction which satisfies both the CR and SN properties.

2.1 Investigation of Church-Rosser property

In this section we are interested in the untyped version of the Dual Calculus under call-by-value reduction. The untyped dual calculus (DuCa) consists solely of Objects. The set of objects is the union of the set of Terms, Coterms and Statements:

$$
\begin{array}{llll}
\text{Object} & G, H & ::= & M \mid K \mid S \\
\text{Term} & M, N & ::= & x \mid \langle M, N \rangle \mid \langle M \rangle \mathtt{inl} \mid \langle N \rangle \mathtt{inr} \mid [K]\mathtt{not} \mid (S).\alpha \\
\text{Coterm} & K, L & ::= & \alpha \mid [K, L] \mid \mathtt{fst}[K] \mid \mathtt{snd}[L] \mid \mathtt{not}\langle M \rangle \mid x.(S) \\
\text{Statement} & S, T & ::= & M \bullet K \\
\text{Value} & V, W & ::= & x \mid \langle V, W \rangle \mid \langle V \rangle \mathtt{inl} \mid \langle W \rangle \mathtt{inr} \mid [K]\mathtt{not}
\end{array}
$$

Recall also that the call-by-value (CBV) reduction relation R_v is the one-step reduction relation yielded by the CBV reduction rules listed below, when these are applied to subobjects of DuCa objects.

$$
\begin{array}{lllll}
(\beta\&_1) & \langle V, W\rangle \bullet \mathtt{fst}[K] & \rightarrow & V \bullet K \\
(\beta\&_2) & \langle V, W\rangle \bullet \mathtt{snd}[L] & \rightarrow & W \bullet L \\
(\beta\vee_1) & \langle V\rangle \mathtt{inl} \bullet [K, L] & \rightarrow & V \bullet K \\
(\beta\vee_2) & \langle W\rangle \mathtt{inr} \bullet [K, L] & \rightarrow & W \bullet L \\
(\beta\neg) & [K]\mathtt{not} \bullet \mathtt{not}\langle M\rangle & \rightarrow & M \bullet K \\
(\beta L) & V \bullet x.(S) & \rightarrow & S\{V/x\} \\
(\beta R) & (S).\alpha \bullet K & \rightarrow & S\{K/\alpha\} \\
\\
(\eta L) & K & \rightarrow & x.(x \bullet K) \\
(\eta R) & M & \rightarrow & (M \bullet \alpha).\alpha \\
\\
(\nu\&_1) & \langle M, N\rangle \bullet K & \rightarrow & M \bullet x.(\langle x, N\rangle \bullet K) \\
(\nu\&_2) & \langle V, N\rangle \bullet K & \rightarrow & N \bullet y.(\langle V, y\rangle \bullet K) \\
(\nu\vee_1) & \langle M\rangle \mathtt{inl} \bullet K & \rightarrow & M \bullet x.(\langle x\rangle \mathtt{inl} \bullet K) \\
(\nu\vee_2) & \langle N\rangle \mathtt{inr} \bullet K & \rightarrow & N \bullet y.(\langle y\rangle \mathtt{inr} \bullet K)
\end{array}
$$

For $G, H \in \mathtt{DuCa}$, $(G, H) \in R_v$ is denoted by $G \longrightarrow^v H$.
For the rest of this section, the restriction of R_v to $\beta\nu$-rules (i.e. β-rules and ν-rules) will be denoted by $\xrightarrow{\beta\nu}$, and called simply $\beta\nu$-reduction relation. Analogously, the restriction to η-rules will be denoted by $\xrightarrow{\eta}$, and called η-reduction relation.
Finally, recall that, if $\longrightarrow$ is (denotes) some reduction relation, then $\longrightarrow_=$ is its reflexive closure, $\longrightarrow_+$ its transitive closure, and $\twoheadrightarrow$ its reflexive transitive closure.
Let us give a definition of the Church-Rosser property.

Definition 2.1
Let $R \subset U^2$ be some reduction relation denoted by $\longrightarrow$, for some universe set U. Then,

- R satisfies the Diamond Property if, for all $M, N, K \in U$,

$$\text{if } N \longleftarrow M \longrightarrow K,$$
$$\text{then there is some } L \in U \text{ such that } N \longrightarrow L \longleftarrow K.$$

- R satisfies the Church-Rosser property, or is Church-Rosser (CR), if its transitive reflexive closure ($\twoheadrightarrow$) satisfies the diamond property.

- R satisfies the Weak Church-Rosser property (WCR) if, for all $M, N, K \in U$,

$$\text{if } N \longleftarrow M \longrightarrow K,$$
$$\text{then there is some } L \in U \text{ such that } N \twoheadrightarrow L \twoheadleftarrow K.$$

$\dashv$

The purpose of this section is to show that R_v in the dual calculus is Church-Rosser. We follow the steps below.

- We show that the $\beta\nu$-reduction relation (i.e. R_v restricted to $\beta\nu$-rules) is $\mathtt{CR}$ (lemma 2.6).
- We show that the η-reduction relation is $\mathtt{CR}$ (lemma 2.7).
- We show that $\xrightarrow{\beta\nu}\!\!\!\!\rightarrow$ and $\xrightarrow{\eta}\!\!\!\!\rightarrow$ commute (lemma 2.11).

In order to show that $\beta\nu$-reduction relation is Church-Rosser, we define a respective parallel reduction relation $\longrightarrow_{\mathtt{p}}$, such that $\xrightarrow{\beta\nu}_{=} \subseteq \longrightarrow_{\mathtt{p}} \subseteq \xrightarrow{\beta\nu}\!\!\!\!\rightarrow$.

Definition 2.2
Let $G, M, N, K, L, V, W, S \in \mathtt{DuCa}$; then, $\longrightarrow_{\mathtt{p}}$ is defined by:

$$
\begin{array}{llcl}
(\mathtt{pid}) & G \longrightarrow_{\mathtt{p}} G & & \\
\\
\multicolumn{4}{l}{\text{if } M \longrightarrow_{\mathtt{p}} M',\ V \longrightarrow_{\mathtt{p}} M'',\ N \longrightarrow_{\mathtt{p}} N',\ W \longrightarrow_{\mathtt{p}} N''} \\
\multicolumn{4}{l}{\text{and } K \longrightarrow_{\mathtt{p}} K',\ L \longrightarrow_{\mathtt{p}} L',\ S \longrightarrow_{\mathtt{p}} S',\ \text{then:}} \\
(\mathtt{p}\beta\&_1) & \langle V, W\rangle \bullet \mathtt{fst}[K] & \longrightarrow_{\mathtt{p}} & M'' \bullet K' \\
(\mathtt{p}\beta\&_2) & \langle V, W\rangle \bullet \mathtt{snd}[L] & \longrightarrow_{\mathtt{p}} & N'' \bullet L' \\
(\mathtt{p}\beta\vee_1) & \langle V\rangle\mathtt{inl} \bullet [K, L] & \longrightarrow_{\mathtt{p}} & M'' \bullet K' \\
(\mathtt{p}\beta\vee_2) & \langle W\rangle\mathtt{inr} \bullet [K, L] & \longrightarrow_{\mathtt{p}} & N'' \bullet L' \\
(\mathtt{p}\beta\neg) & [K]\mathtt{not} \bullet \mathtt{not}\langle M\rangle & \longrightarrow_{\mathtt{p}} & M' \bullet K' \\
(\mathtt{p}\beta L) & V \bullet x.(S) & \longrightarrow_{\mathtt{p}} & S'\{M''/x\} \\
(\mathtt{p}\beta R) & (S).\alpha \bullet K & \longrightarrow_{\mathtt{p}} & S'\{K'/\alpha\} \\
(\mathtt{p}\nu\&_1) & \langle M, N\rangle \bullet K & \longrightarrow_{\mathtt{p}} & M' \bullet x.(\langle x, N'\rangle \bullet K') \\
(\mathtt{p}\nu\&_2) & \langle V, N\rangle \bullet K & \longrightarrow_{\mathtt{p}} & N' \bullet y.(\langle M'', y\rangle \bullet K') \\
(\mathtt{p}\nu\vee_1) & \langle M\rangle\mathtt{inl} \bullet K & \longrightarrow_{\mathtt{p}} & M' \bullet x.(\langle x\rangle\mathtt{inl} \bullet K') \\
(\mathtt{p}\nu\vee_2) & \langle N\rangle\mathtt{inr} \bullet K & \longrightarrow_{\mathtt{p}} & N' \bullet y.(\langle y\rangle\mathtt{inr} \bullet K') \\
\\
(\mathtt{p}\bullet) & M \bullet K & \longrightarrow_{\mathtt{p}} & M' \bullet K' \\
(\mathtt{p}\langle,\rangle) & \langle M, N\rangle & \longrightarrow_{\mathtt{p}} & \langle M', N'\rangle \\
(\mathtt{p}\langle\rangle\mathtt{inl}) & \langle M\rangle\mathtt{inl} & \longrightarrow_{\mathtt{p}} & \langle M'\rangle\mathtt{inl} \\
(\mathtt{p}\langle\rangle\mathtt{inr}) & \langle M\rangle\mathtt{inl} & \longrightarrow_{\mathtt{p}} & \langle N'\rangle\mathtt{inl} \\
(\mathtt{p}[]\mathtt{not}) & [K]\mathtt{not} & \longrightarrow_{\mathtt{p}} & [K']\mathtt{not} \\
(\mathtt{p}().) & (S).\alpha & \longrightarrow_{\mathtt{p}} & (S').\alpha \\
(\mathtt{p}[,]) & [K, L] & \longrightarrow_{\mathtt{p}} & [K', L'] \\
(\mathtt{pfst}[]) & \mathtt{fst}[K] & \longrightarrow_{\mathtt{p}} & \mathtt{fst}[K'] \\
(\mathtt{psnd}[]) & \mathtt{snd}[L] & \longrightarrow_{\mathtt{p}} & \mathtt{snd}[L'] \\
(\mathtt{pnot}\langle\rangle) & \mathtt{not}\langle M\rangle & \longrightarrow_{\mathtt{p}} & \mathtt{not}\langle M'\rangle \\
(\mathtt{p}.()) & x.(S) & \longrightarrow_{\mathtt{p}} & x.(S')
\end{array}
$$

$\dashv$

First, we show that parallel reduction preserves values.

Proposition 2.3 *Let V be a value and suppose $V \longrightarrow_{\mathtt{p}} M$. Then, M is also a value.*

Proof: We do induction on V. The base case is $V \equiv x$, whence $M \equiv x$.
For the inductive step, we do a case analysis on V:

- $V \equiv \langle V_1, V_2 \rangle \longrightarrow_{\mathtt{p}} M$, so by inspection of the definition, $M \equiv \langle M_1, M_2 \rangle$, with $V_1 \longrightarrow_{\mathtt{p}} M_1, V_2 \longrightarrow_{\mathtt{p}} M_2$, so by IH, M_1, M_2 are values, hence M is a value.
- the cases $V \equiv \langle V_1 \rangle \mathtt{inl}, V \equiv \langle V_2 \rangle \mathtt{inr}$ are similar to the above.
- $V \equiv [K]\mathtt{not} \longrightarrow_{\mathtt{p}} M$, so by inspection, $M \equiv [K']\mathtt{not}$, thus M is a value. ■

Moreover, parallel reduction satisfies the diamond property. To prove this, we need a substitution lemma.

Lemma 2.4 (Substitution) *Let $G, G', V, V', K, K' \in \mathtt{DuCa}$ with $G \longrightarrow_{\mathtt{p}} G'$, $V \longrightarrow_{\mathtt{p}} V'$, $K \longrightarrow_{\mathtt{p}} K'$. Then, for any variable x and covariable α,*

$$G\{V/x\} \longrightarrow_{\mathtt{p}} G'\{V'/x\}$$
$$G\{K/\alpha\} \longrightarrow_{\mathtt{p}} G'\{K'/\alpha\}$$

Proof: The proof is done by a straightforward induction on $G \in \mathtt{DuCa}$. The base cases (of $G \equiv x$, $G \equiv \alpha$ and $G \equiv x \bullet \alpha$) are trivial, since $G \equiv G'$. The induction step is done by a long case analysis on G and all cases are straightforwardly proven by using the IH. ■

Lemma 2.5 *The relation $\longrightarrow_{\mathtt{p}}$ defined above satisfies the diamond property. That is, for all $G, G_1, G_2 \in \mathtt{DuCa}$, if $G_1 \ {}_{\mathtt{p}}\!\!\longleftarrow G \longrightarrow_{\mathtt{p}} G_2$, then there exists $G_c \in \mathtt{DuCa}$ such that $G_1 \longrightarrow_{\mathtt{p}} G_c \ {}_{\mathtt{p}}\!\!\longleftarrow G_2$.*

Proof: See the Appendix. ■

Therefore, the $\beta\nu$-reduction relation is $\mathtt{CR}$.

Lemma 2.6 *The $\beta\nu$-reduction relation is Church-Rosser; that is, for all $G, G_1, G_2 \in \mathtt{DuCa}$, if $G_1 \overset{\beta\nu}{\twoheadleftarrow} G \overset{\beta\nu}{\twoheadrightarrow} G_2$, then there exists $G_c \in \mathtt{DuCa}$ such that $G_1 \overset{\beta\nu}{\twoheadrightarrow} G_c \overset{\beta\nu}{\twoheadleftarrow} G_2$.*

Proof: By definition of the parallel reduction we have that $\overset{\beta\nu}{\longrightarrow}_{=} \subseteq \longrightarrow_{\mathtt{p}} \subseteq \overset{\beta\nu}{\twoheadrightarrow}$. Taking transitive closures in this formula, we have that $\longrightarrow_{\mathtt{p}+} \equiv \overset{\beta\nu}{\twoheadrightarrow}$. But, since $\longrightarrow_{\mathtt{p}}$ satisfies the diamond property, $\longrightarrow_{\mathtt{p}+}$ is $\mathtt{CR}$, by a simple diagram chase. ■

An easier result is that the η-reduction relation is $\mathtt{CR}$.

Lemma 2.7 *The η-reduction relation is Church-Rosser; that is, for all $G, G_1, G_2 \in \mathtt{DuCa}$, if $G_1 \overset{\eta}{\twoheadleftarrow} G \overset{\eta}{\twoheadrightarrow} G_2$, then there exists $G_c \in \mathtt{DuCa}$ such that $G_1 \overset{\eta}{\twoheadrightarrow} G_c \overset{\eta}{\twoheadleftarrow} G_2$.*

Proof: It suffices to show that $\xrightarrow{\eta}$ satisfies the diamond property, since then the claim follows by a simple diagram chase.
Let C be any context, then,

$$\begin{tikzcd}
C\{K\} \arrow[r, "\eta"] \arrow[d, "\eta"] \arrow[dd, bend right=60, "\eta"'] & C\{x.(x \bullet K)\} \arrow[d, "\eta"] \arrow[dd, bend left=60, "\eta"] \\
C'\{K\} \arrow[r, "\eta"] & C'\{x.(x \bullet K)\} \\
C\{K'\} \arrow[r, "\eta"] & C\{x.(x \bullet K')\}
\end{tikzcd}$$

and similarly for $C\{M\}$. Hence, $\xrightarrow{\eta}$ satisfies the diamond property. ■

Now, regarding $\beta\nu$-reductions, we do the following distinction.

Definition 2.8
All $\beta\nu$-reductions are called simple reductions, except if they happen by application of βL or βR rules; these latter are called $\texttt{sub}_{\leq 1}$ or $\texttt{sub}_{>1}$ reductions:

> $V \bullet x.(S) \xrightarrow{\beta L} S\{V/x\}$ is a $\texttt{sub}_{\leq 1}$ reduction if x occurs at most once in S, otherwise it is a $\texttt{sub}_{>1}$ reduction. Similarly for the βR rule.

$\dashv$

The following lemma concerns η-reductions that destroy values.

Lemma 2.9 *Let V be a value, M a non-value term, and K a coterm. Then,*

$$V \bullet K \xrightarrow{\eta} M \bullet K \quad \textit{implies} \quad M \bullet K \xrightarrow{\beta\nu}\!\!\!\!\rightarrow V \bullet K$$

where $M \bullet K \xrightarrow{\beta\nu}\!\!\!\!\rightarrow V \bullet K$ involves only simple or $\texttt{sub}_{\leq 1}$ reductions.

Proof: First note that, if $\xrightarrow{\eta}$ reduces the whole of V, we trivially have:

$$V \bullet K \xrightarrow{\eta} (V \bullet x).x \bullet K \xrightarrow{\beta R} V \bullet K$$

where x is fresh, and thus the βR-reduction is $\texttt{sub}_{\leq 1}$.
So suppose that $\xrightarrow{\eta}$ reduces inside V. Since V is turned to a non-value, V cannot be of the type $[L]\texttt{not}$. Repeating this argument several times, we come to the conclusion that the η-reduction above is in fact:

$$E\{W\} \bullet K \xrightarrow{\eta} E\{(W \bullet \alpha).\alpha\} \bullet K \quad \text{where}$$
$$E ::= \{\} \mid \langle E, W' \rangle \mid \langle W', E \rangle \mid \langle E \rangle \texttt{inl} \mid \langle E \rangle \texttt{inr}$$

Therefore, we proceed by doing induction on V and a case analysis on E. The case where $E \equiv \{\}$ is dealt with above. It also includes the base case $V \equiv x$.
For the inductive step, we have the following reductions.

$$\begin{aligned}\langle E\{W\}, W'\rangle \bullet K \xrightarrow{\eta} \langle E\{(W \bullet \alpha).\alpha\}, W'\rangle \bullet K \xrightarrow{\nu} E\{(W \bullet \alpha).\alpha\} \bullet y.(\langle y, W'\rangle \bullet K) \\ \xrightarrow{(\mathbf{IH})\ \beta\nu}\!\!\!\!\rightarrow E\{W\} \bullet y.(\langle y, W'\rangle \bullet K) \xrightarrow{\beta L} \langle E\{W\}, W'\rangle \bullet K\end{aligned}$$

and similarly for the $\langle W', E\{W\}\rangle$ case. Also,

$$\begin{aligned}\langle E\{W\}\rangle\texttt{inl} \bullet K \xrightarrow{\eta} \langle E\{(W \bullet \alpha).\alpha\}\rangle\texttt{inl} \bullet K \xrightarrow{\nu} E\{(W \bullet \alpha).\alpha\} \bullet y.(\langle y\rangle\texttt{inl} \bullet K) \\ \xrightarrow{(\mathbf{IH})\ \beta\nu}\!\!\!\!\rightarrow E\{W\} \bullet y.(\langle y\rangle\texttt{inl} \bullet K) \xrightarrow{\beta L} \langle E\{W\}\rangle\texttt{inl} \bullet K\end{aligned}$$

and similarly for the $\langle E\{W\}\rangle\texttt{inr}$ case. ∎

Then, we can prove the following lemmata.

Lemma 2.10 *If $G, G_1, G_2 \in \texttt{DuCa}$, and* $G_1 \xleftarrow{\beta\nu} G \xrightarrow{\eta} G_2$, *then either*

- $G_2 \xrightarrow{\beta\nu}\!\!\!\!\rightarrow G$, *by use of simple or* $\texttt{sub}_{\leq 1}$ *reductions, or*
- *if* $G \xrightarrow{\beta\nu} G_1$ *is simple or* $\texttt{sub}_{\leq 1}$, *then there exists* G_c *such that* $G_1 \xrightarrow{\eta}_{=} G_c \xleftarrow{\beta\nu} G_2$, *and* $G_2 \xrightarrow{\beta\nu} G_c$ *is simple or* $\texttt{sub}_{\leq 1}$; *otherwise, if* $G \xrightarrow{\beta\nu} G_1$ *is* $\texttt{sub}_{>1}$, *then there exists* G_c *such that* $G_1 \xrightarrow{\eta}\!\!\!\!\rightarrow G_c \xleftarrow{\beta\nu} G_2$, *and* $G_2 \xrightarrow{\beta\nu} G_c$ *is* $\texttt{sub}_{>1}$.

Proof: We do a case analysis on the reduction $G \xrightarrow{\beta\nu} G_1$, which we label with an index: $G \xrightarrow{\beta\nu^0} G_1$.
In the following, C denotes some context and G is $C\{G'\}$, with G' being the redex of $\xrightarrow{\beta\nu^0}$. The cases of $G \xrightarrow{\eta} G_2$ being an η-reduction that can be trivially 'reverted' by a one-step β-reduction, for example
$C\{V \bullet x.(S)\} \xrightarrow{\eta} C\{V \bullet y.(y \bullet x.(S))\} \xrightarrow{\beta} C\{V \bullet x.(S)\}$, are trivial and omitted for economy. For the same reason, the cases of this η-reduction affecting solely C and not its content, for example $C\{V \bullet x.(S)\} \xrightarrow{\eta} C'\{V \bullet x.(S)\}$, are also omitted.
Hence, we have the following diagrams.

For $\xrightarrow{\beta\nu^0}$ being βL:

$$
\begin{tikzcd}
& C\{V \bullet x.(S)\} \arrow[r, "\beta\nu^0"] \arrow[d, "\eta"] \arrow[dl, shift left, two heads, "\eta"] \arrow[dd, bend right=60, "\eta"'] & C\{S\{V/x\}\} \arrow[d, "\eta"] \arrow[dd, bend left=60, two heads, "\eta^1"] \\
C\{M \bullet x.(S)\} \arrow[ur, shift left, two heads, "\beta\nu"] & C\{V \bullet x.(S')\} \arrow[r, "\beta\nu"] & C\{S'\{V/x\}\} \\
& C\{V' \bullet x.(S)\} \arrow[r, "\beta\nu^1"] & C\{S\{V'/x\}\}
\end{tikzcd}
$$

where M is a non-value, and for it lemma 2.9 is applied. Note in $\overset{\eta^1}{\twoheadrightarrow}$ that only in the occasion where $\xrightarrow{\beta\nu^0}$ is $\texttt{sub}_{>1}$ are there more than one η-steps required. In this case, $\xrightarrow{\beta\nu^1}$ is also $\texttt{sub}_{>1}$.
For $\xrightarrow{\beta\nu^0}$ being βR:

$$
\begin{tikzcd}
C\{(S).\alpha \bullet K\} \arrow[r, "\beta\nu^0"] \arrow[d, "\eta"] \arrow[dd, bend right=60, "\eta"'] & C\{S\{K/\alpha\}\} \arrow[d, "\eta"] \arrow[dd, bend left=60, two heads, "\eta^1"] \\
C\{(S').\alpha \bullet K\} \arrow[r, "\beta\nu"] & C\{S'\{K/\alpha\}\} \\
C\{(S).\alpha \bullet K'\} \arrow[r, "\beta\nu^1"] & C\{S\{K'/\alpha\}\}
\end{tikzcd}
$$

where the same comments as above apply for $\overset{\eta^1}{\twoheadrightarrow}, \xrightarrow{\beta\nu^1}$.
For $\beta\&_1$:

$$
\begin{tikzcd}
& C\{\langle V, W\rangle \bullet \texttt{fst}[K]\} \arrow[r, "\beta\nu^0"] \arrow[d, "\eta"] \arrow[dl, shift left, two heads, "\eta"] & C\{V \bullet K\} \arrow[d, "\eta", "="'] \\
C\{M \bullet \texttt{fst}[K]\} \arrow[ur, shift left, two heads, "\beta\nu"] & C\{\langle V', W'\rangle \bullet \texttt{fst}[K']\} \arrow[r, "\beta\nu"] & C\{V' \bullet K'\}
\end{tikzcd}
$$

where M is a non-value. The cases of $\beta\&_2, \beta\vee_1, \beta\vee_2$ are similar and this of $\beta\neg$ is similar but simpler.

For $\nu\&_2$:

$$\begin{tikzcd}
C\{\langle V, N\rangle \bullet K\} \arrow[rr, "\beta\nu^0"] \arrow[d, "\eta"] \arrow[dr, "\eta"] & & C\{N \bullet y.(\langle V, y\rangle \bullet K)\} \arrow[dr, "\eta"] & \\
C\{\langle M, N\rangle \bullet K\} \arrow[d, "\nu\&_1"] & C\{\langle V', N'\rangle \bullet K'\} \arrow[rr, "\beta\nu"] & & C\{N' \bullet y.(\langle V', y\rangle \bullet K')\} \\
C\{M \bullet x.(\langle x, N\rangle \bullet K)\} \arrow[d, two heads, "\beta\nu \text{ (lemma 2.9)}"] & & & \\
C\{V \bullet x.(\langle x, N\rangle \bullet K)\} \arrow[uuu, bend left=60, "\beta L"] & & &
\end{tikzcd}$$

where M is a non-value, and (since $V \xrightarrow{\eta} M$) $V \bullet x.(\langle x, N\rangle \bullet K) \xrightarrow{\eta} M \bullet x.(\langle x, N\rangle \bullet K)$, whence lemma 2.9 can be applied. The cases of $\nu\&_1, \nu\vee_1, \nu\vee_2$ are similar but simpler. ■

Lemma 2.11 (Commutativity) *$\xrightarrow{\beta\nu}\!\!\!\!\rightarrow$ and $\xrightarrow{\eta}\!\!\!\!\rightarrow$ commute; that is, for all $G, G', G'' \in$ `DuCa`, if $G' \overset{\beta\nu}{\twoheadleftarrow} G \overset{\eta}{\twoheadrightarrow} G''$, then there exists some $G_c \in$ `DuCa` such that $G' \overset{\eta}{\twoheadrightarrow} G_c \overset{\beta\nu}{\twoheadleftarrow} G''$.*

Proof: Suppose that

$$G' \equiv H_m \xleftarrow{\beta\nu} \cdots \xleftarrow{\beta\nu} H_2 \xleftarrow{\beta\nu} H_1 \xleftarrow{\beta\nu} G \xrightarrow{\eta} G_1 \xrightarrow{\eta} G_2 \xrightarrow{\eta} \cdots \xrightarrow{\eta} G_n \equiv G''$$

and assume $n > 0$ (the case $n = 0$ is trivial). We do induction on m; the base case, $m = 0$, is trivial.
So fix some $m > 0$. We claim that there exist $u_1, u_2, \ldots u_n \in$ `DuCa` such that,

$$\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, two heads, "\beta\nu^0"] & G_1 \arrow[r, "\eta"] \arrow[d, two heads, "\beta\nu^1"] & \cdots \arrow[r, "\eta"] & G_n \arrow[d, two heads, "\beta\nu^n"] \\
H_1 \arrow[r, two heads, "\eta"] & u_1 \arrow[r, two heads, "\eta"] & \cdots \arrow[r, two heads, "\eta"] & u_n
\end{tikzcd}
\qquad \text{hence} \qquad
\begin{tikzcd}
G \arrow[r, two heads, "\eta"] \arrow[d, two heads, "\beta\nu"] & G_n \arrow[d, two heads, "\beta\nu"] \\
H_1 \arrow[r, two heads, "\eta"] & u_n
\end{tikzcd}$$

Thus, applying the claim, we only need to prove commutativity for the reduction chain,

$$G' \equiv H_m \xleftarrow{\beta\nu} \cdots \xleftarrow{\beta\nu} H_2 \xleftarrow{\beta\nu} H_1 \overset{\eta}{\twoheadrightarrow} u_n$$

for which the IH on m applies.
Hence, it suffices to prove our claim. By hypothesis, $H_1 \xleftarrow{\beta\nu^0} G \xrightarrow{\eta} G_1$. Now, by lemma

2.10, if $\xrightarrow{\beta\nu^0}$ is a simple or $\texttt{sub}_{\leq 1}$ reduction, then one of the following diagrams must be the case,

$$
\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, "\beta\nu^0"] & G_1 \arrow[d, two heads, "\beta\nu^{1\prime}"] \\
H_1 \arrow[dr, "\eta(\texttt{id})", "="'] & G \arrow[d, "\beta\nu^0"] \\
& u_1 \equiv H_1
\end{tikzcd}
\qquad \text{or} \qquad
\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, "\beta\nu^0"] & G_1 \arrow[d, "\beta\nu^1"] \\
H_1 \arrow[r, "\eta", "\equiv"'] & u_1
\end{tikzcd}
$$

In both cases $\xrightarrow{\beta\nu^{1\prime}}\!\!\!\!\rightarrow$ and $\xrightarrow{\beta\nu^1}$ include only simple or $\texttt{sub}_{\leq 1}$ reductions. Thus, we can reuse this reasoning repeatedly and finally get the following diagram, which proves the claim in this case.

$$
\begin{tikzcd}
G \arrow[r, two heads, "\eta"] \arrow[d, "\beta\nu^0"] & G_{n-1} \arrow[r, "\eta"] \arrow[d, "\beta\nu"] & G_n \arrow[d, two heads, "\beta\nu"] \\
H_1 \arrow[dddd, "\eta", two heads] & w_1 \arrow[r, "\eta", "\equiv"'] \arrow[d, "\beta\nu"] & w'_1 \arrow[d, two heads, "\beta\nu"] \\
& w_2 \arrow[r, "\eta", "\equiv"'] \arrow[d, "\beta\nu"] & w'_2 \arrow[d, two heads, "\beta\nu"] \\
& \vdots \arrow[d, "\beta\nu"] & \vdots \arrow[d, two heads, "\beta\nu"] \\
& v_{n-1} \equiv w_k \arrow[r, "\eta", "\equiv"'] & w'_k \equiv u_n
\end{tikzcd}
$$

Now suppose $\xrightarrow{\beta\nu^0}$ is a $\texttt{sub}_{>1}$ reduction. Then, by lemma 2.10, one of the following diagrams must be the case.

$$
\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, "\beta\nu^0"] & G_1 \arrow[d, two heads, "\beta\nu^{1\prime}"] \\
H_1 \arrow[dr, two heads, "\eta(\texttt{id})"] & G \arrow[d, "\beta\nu^0"] \\
 & u_1 \equiv H_1
\end{tikzcd}
\quad \text{or} \quad
\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, "\beta\nu^0"] & G_1 \arrow[d, "\beta\nu^1"] \\
H_1 \arrow[r, two heads, "\eta"] & u_1
\end{tikzcd}
$$

with $\xrightarrow{\beta\nu^{1\prime}}\!\!\!\!\rightarrow$ including only simple or $\texttt{sub}_{\leq 1}$ reductions and $\xrightarrow{\beta\nu^1}$ being a $\texttt{sub}_{>1}$ reduction. Therefore, both diagrams have the form:

$$
\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, "\beta\nu^0"] & G_1 \arrow[d, two heads, "\beta\nu(\text{simple or } \texttt{sub}_{\leq 1})"] \\
H_1 \arrow[dr, two heads, "\eta"] & w_1 \arrow[d, "\beta\nu(\texttt{sub}_{>1})"] \\
 & u_1
\end{tikzcd}
$$

Reasoning thus repeatedly and handling $G_i \stackrel{\beta\nu}{\twoheadrightarrow} w_i$ in the same way as $G \stackrel{\beta\nu^0}{\longrightarrow} H_1$ previously (where $\stackrel{\beta\nu^0}{\longrightarrow}$ was simple or $\texttt{sub}_{\leq 1}$), we have the following diagram.

$$
\begin{tikzcd}
G \arrow[r, "\eta"] \arrow[d, "\beta\nu^0(\texttt{sub}_{>1})"] & G_1 \arrow[r, "\eta"] \arrow[d, two heads, "\beta\nu"] & G_2 \arrow[r, "\eta"] \arrow[d, two heads, "\beta\nu"] & \cdots \arrow[r, "\eta"] & G_{n-1} \arrow[r, "\eta"] \arrow[ddddd, two heads, "\beta\nu"] & G_n \arrow[ddddd, two heads, "\beta\nu"] \\
H_1 \arrow[dr, two heads, "\eta"] & w_1 \arrow[r, two heads, "\eta"] \arrow[d, "\beta\nu(\texttt{sub}_{>1})"] & w_1' \arrow[d, two heads, "\beta\nu"] & & & \\
& u_1 \arrow[dr, two heads, "\eta"] & w_2 \arrow[r, two heads, "\eta"] \arrow[d, "\beta\nu(\texttt{sub}_{>1})"] & \cdots & & \\
& & u_2 \arrow[dr, two heads, "\eta"] & & & \\
& & & \ddots \arrow[ddr, two heads, "\eta"] & & \\
& & & & w_{n-1} \arrow[r, two heads, "\eta"] \arrow[d, "\beta\nu(\texttt{sub}_{>1})"] & w_{n-1}' \arrow[d, two heads, "\beta\nu"] \\
& & & & u_{n-1} \arrow[dr, two heads, "\eta"] & w_n \arrow[d, "\beta\nu(\texttt{sub}_{>1})"] \\
& & & & & u_n
\end{tikzcd}
$$

which proves the claim and the lemma. ■

Combining the results above we prove confluence for the call-by-value reduction relation.

Theorem 2.12 *R_v is Church-Rosser; that is, for all $G, G_1, G_2 \in \texttt{DuCa}$*

$$\textit{if } G_1 \; {}^{v}\!\twoheadleftarrow G \twoheadrightarrow^{v} G_2 \; ,$$
$$\textit{then there exists } G_c \in \texttt{DuCa} \textit{ such that } G_1 \twoheadrightarrow^{v} G_c \; {}^{v}\!\twoheadleftarrow G_2 \; .$$

Proof: By lemmata 2.6,2.7 and 2.11, as in [Bar84]. ■

2.2 Strong Normalization

In this section we investigate the typed version of the Dual Calculus under call-by-value reduction without ν-rules and with some restrictions on η-rules. We prove that this reduction relation is strongly normalizing. The steps we follow for this proof are:

- We prove SN in a similar calculus, called Dual Calculus* (DuCa*)[9], under a similar reduction relation, using the method of *reducibility sets*.

- We introduce the call-by-value CPS translation of the Dual Calculus, as described in [Wad03a]. Under this translation reductions are preserved and in some cases one-step reductions are preserved or lengthened. Moreover, the reduction relation of the target calculus is SN.

- Using the results of the previous steps, we can straightforwardly prove SN.

Let us recall the definition of Strong Normalization.

Definition 2.13
Let $R \subset U^2$ be some reduction relation on some universe U. Then,

- if $G \in U$, then G is Strongly Normalizing (under R), or simply SN, if there is no infinite R-reduction sequence starting from G.

- R is Strongly Normalizing if all elements of U are SN.

Moreover, if $G \in U$ is SN, then $l(G)$ is the length of the longest R-reduction path starting from G.

$\dashv$

2.2.1 The reduction relation of DuCa* is SN

We introduce an auxiliary calculus similar to DuCa.

Definition 2.14 (Dual Calculus* and its reduction relation R^*)
The Dual Calculus* (DuCa*) is a typed calculus consisting of Types and Objects. The set of objects is the union of the sets of Terms, Coterms and Statements:

Type	A, B	$::=$	$X \mid A\&B \mid A \vee B \mid \neg A$
Object	G, H	$::=$	$M \mid K \mid S$
Term	M, N	$::=$	$M_{\mathfrak{n}} \mid (S).\alpha$
Neutral Term	$M_{\mathfrak{n}}$	$::=$	$M_{\mathfrak{s}} \mid (S)_{\odot}\alpha$
Simple Term	$M_{\mathfrak{s}}$	$::=$	$x \mid \langle M, N\rangle \mid \langle M\rangle\mathtt{inl} \mid \langle N\rangle\mathtt{inr} \mid [K]\mathtt{not}$
Coterm	K, L	$::=$	$K_{\mathfrak{n}} \mid x.(S)$
Neutral Coterm	$K_{\mathfrak{n}}$	$::=$	$K_{\mathfrak{s}} \mid x_{\odot}(S)$
Simple Coterm	$K_{\mathfrak{s}}$	$::=$	$\alpha \mid [K, L] \mid \mathtt{fst}[K] \mid \mathtt{snd}[L] \mid \mathtt{not}\langle M\rangle$
Statement	S, T	$::=$	$M \bullet K$

An object G of DuCa* is neutral if it is a neutral term, or a neutral coterm, or a statement. R^* is the one-step reduction relation yielded by the following rules, when these are applied

[9] DuCa* is a calculus with no value of its own; it just serves us in the proof of SN of DuCa.

to subobjects of $\mathtt{DuCa}^*$ objects.

$$\begin{array}{llcl}
(\beta L) & M_{\mathtt{n}} \bullet x.(S) & \to & S\{M_{\mathtt{n}}/x\} \\
(\beta R) & (S).\alpha \bullet K & \to & S\{K/\alpha\} \\
(\eta L) & K_{\mathtt{s}} & \to & x_{\odot}(x \bullet K_{\mathtt{s}}) \\
(\eta R) & M_{\mathtt{s}} & \to & (M_{\mathtt{s}} \bullet \alpha)_{\odot}\alpha
\end{array}$$

For $G, H \in \mathtt{DuCa}$, $(G, H) \in R^*$ is written $G \longrightarrow H$.
The η-rules are not allowed to be applied to terms [resp. coterms] that are immediately followed by [immediately follow] some cut $'\bullet'$.
The typing rules for $\mathtt{DuCa}^*$ are the same as those of $\mathtt{DuCa}$ (i.e. of system $\mathtt{GW}$), with the addition of $\mathrm{RI}_{\odot}$ and $\mathrm{LI}_{\odot}$ rules introducing $'_{\odot}'$:

$$\frac{x : A, \Gamma \,\rule{2pt}{8pt}\, S \,\rule{2pt}{8pt}\!\rightarrow\ \Theta}{x_{\odot}(S) : A \,\rule{2pt}{8pt}\, \Gamma \ \rightarrow\ \Theta}\,LI_{\odot} \qquad \frac{\Gamma \,\rule{2pt}{8pt}\, S \,\rule{2pt}{8pt}\!\rightarrow\ \Theta, \alpha : A}{\Gamma \ \rightarrow\ \Theta \,\rule{2pt}{8pt}\, (S)_{\odot}\alpha : A}\,RI_{\odot}$$

The addition of these rules yields the sequent calculus $\mathtt{GW}^*$. $\dashv$

Note that by the above definition neutral elements are preserved by reduction: if G is neutral and $G \longrightarrow G'$, then G' is neutral.
The $\mathtt{DuCa}^*$ differs from $\mathtt{DuCa}$ in the addition of $'_{\odot}'$ and the usage of the more general notion of neutral terms instead of values:

- The $'_{\odot}'$ symbol is a 'neutralizing dot' for statements, since we can't apply β-rules to it, for example:

$$M_{\mathtt{n}} \bullet x.(S) \to S\{M_{\mathtt{n}}/x\} \quad \text{but} \quad M_{\mathtt{n}} \bullet x_{\odot}(S) \not\to S\{M_{\mathtt{n}}/x\}$$

- On the other hand, since we don't have extra β-rules as in the $\mathtt{DuCa}$, we can simplify the distinctions inside terms and use neutral terms instead of values.

Regarding derivable sequents that type elements of $\mathtt{DuCa}^*$, we use the usual notation described in definition 1.2.1. We also introduce some notation regarding 'reduction' between sequents.

Definition 2.15
If $G, G' \in \mathtt{DuCa}^*$ and σ, σ' are sequents in $\mathtt{GW}^*$, then

- $\sigma \in T_G(A, \Gamma, \Theta)$ if σ is derivable in $\mathtt{GW}^*$ and either

$$\begin{array}{lcll}
\sigma & \equiv & \Gamma \ \rightarrow\ \Theta \,\rule{2pt}{8pt}\, M : A & \text{and } G \text{ is a term } M, \text{ or} \\
\sigma & \equiv & K : A \,\rule{2pt}{8pt}\, \Gamma \ \rightarrow\ \Theta & \text{and } G \text{ is a coterm } K, \text{ or} \\
\sigma & \equiv & \Gamma \,\rule{2pt}{8pt}\, S \,\rule{2pt}{8pt}\!\rightarrow\ \Theta & \text{and } G \text{ is a statement } S.
\end{array}$$

- $\sigma \in T_G(A)$ if $\sigma \in T_G(A, \Gamma, \Theta)$ for some Γ, Θ.

- $\sigma \in T_G(\Gamma, \Theta)$ if $\sigma \in T_G(A, \Gamma, \Theta)$ for some type A.
- $\sigma \in T_G$ if $\sigma \in T_G(A, \Gamma, \Theta)$ for some A, Γ, Θ. G is *typed* if $T_G \neq \emptyset$.
- If $\sigma \in T_G(A, \Gamma, \Theta)$ and $\sigma' \in T_{G'}(A, \Gamma, \Theta)$, then $\sigma \longrightarrow \sigma'$ if $G \longrightarrow G'$.

$\dashv$

As noted before, if $\sigma \in T_G$ and $\sigma \in T_{G'}$, then $G \equiv G'$. Moreover, for all statements S and types A, $T_S = T_S(A)$. Further, an element G of the calculus can be assigned more than one types and, on the other hand, we allow for elements G with $T_G = \emptyset$. For example, $G \equiv \langle x, x \rangle \bullet [a, a]$ is not typed.
Now, subject reduction holds for `DuCa`*.

Proposition 2.16 (Subject Reduction) *Let $G, G' \in$ `DuCa`* *and assume that $G \longrightarrow G'$. Then, if $\sigma \in T_G(A, \Gamma, \Theta)$, some sequent σ, then there exists a sequent $\sigma' \in T_{G'}(A, \Gamma, \Theta)$ and thus $\sigma \longrightarrow \sigma'$.*

Proof: The case of G reducing to G' by η-rules is straightforward. As far as β-rules are concerned, the claim is proven by substituting proofs (derivations) in the sequent calculus. ■

Neutral elements of `DuCa`* are like 'boxes' the inside of which cannot be accessed by outer reductions. Indeed, neutral terms are like variables with extra structure, and similarly for neutral coterms and covariables. This remark is implicitly used in the proof of the following lemma.

Lemma 2.17 *If M is a neutral term and S a statement, and both M and S are* `SN`*, then $M \bullet x.(S)$ is* `SN` *for any variable x.*
If K is a neutral coterm and S a statement, and both K and S are `SN`*, then $(S).\alpha \bullet K$ is* `SN` *for any covariable α.*

Proof: We prove only the first claim; the second is proven similarly.
We do induction on $l(M) + l(S)$. Let $S_0 \equiv M \bullet x.(S)$. If $S_0 \longrightarrow S^?$, then either $S^? \equiv M' \bullet x.(S')$, with $M \longrightarrow M'$ or $S \longrightarrow S'$, or $S^? \equiv S\{M/x\}$.
In the former case, we have $l(M') + l(S') < l(M) + l(S)$, so $M' \bullet x.(S')$ is `SN` by the IH.
Now, if we prove that in the latter case $S\{M/x\}$ is `SN`, then S_0 reduces only to `SN` elements, so S_0 is `SN`.
In order to prove this for the latter case, we show something stronger:

> For any statement $S \in$ `SN` and variable x, if we mark the occurrences of x inside S by $1, 2, \ldots, n$, then for any tuple $M_1, \ldots, M_n$ of neutral `SN` terms,
>
> $$S_1 \;\equiv\; S\{M_1/_1x, M_2/_2x, \ldots, M_n/_nx\} \in \texttt{SN}$$
>
> where $M_i/_ix$ denotes the substitution of the i-occurrence of x in S for M_i.

The proof of this claim is by induction on $l(S)$. For the base case, that is of S being in normal form, we have that the redexes inside S_1 are exactly those inside the M_is, since S doesn't contain any redexes and all M_is are neutral. But then S_1 is $\mathtt{SN}$, since all M_is are $\mathtt{SN}$.
For the inductive step, assume $l(S) > 0$ and suppose that there is some infinite reduction sequence starting from S_1. Then, since the M_is are $\mathtt{SN}$, in this sequence it must be the case that,

$$S_1 \longrightarrow\!\!\!\!\!\rightarrow S_2 \longrightarrow S_1' \text{ , with } S_2 \equiv S\{M_1'/_1x, \ldots, M_n'/_nx\} \text{ , some } M_i \longrightarrow\!\!\!\!\!\rightarrow M_i', \\ i = 1, \ldots, n, \text{ and } S_1' \equiv S'\{N_1/_1x, \ldots, N_{n+k}/_{n+k}x\}$$

where the reduction $S \longrightarrow S'$ produces $k \in \mathbb{Z}$ new occurrences of x. In S_1', all x's are substituted for some of the M_i's (denoted by $N_j,\ j = 1, \ldots, n+k$). Then, by IH, since $N_1, \ldots, N_{n+k}$ are all $\mathtt{SN}$ and neutral, and S' is $\mathtt{SN}$ with $l(S') < l(S)$, S_1' is $\mathtt{SN}$, $\lightning$ to this being an infinite reduction sequence. Hence, S_1 is $\mathtt{SN}$, and thus our initial $S\{M/x\}$ is $\mathtt{SN}$. ∎

A similar idea is applied in the following lemma.

Lemma 2.18 *If S is a statement and α a covariable not occurring immediately after a cut in S, then, if $S\{K/\alpha\}$ is $\mathtt{SN}$ for some coterm K, then S is $\mathtt{SN}$.*

Proof: We prove something stronger:

> For any statement S and covariable α not occurring in S immediately after a cut, if we mark the occurrences of α in S by $1, 2, \ldots, n$ and there exist coterms $K_1, \ldots, K_n$ such that $S^p \equiv S\{K_1/_1\alpha, \ldots, K_n/_n\alpha\} \in \mathtt{SN}$, then $S \in \mathtt{SN}$.

Above, $K_i/_i\alpha$ denotes the substitution of the i-occurrence of α in S for K_i.
The proof of this claim is by induction on $l(S^p)$. The base case is this of S^p being in normal form. Then, whenever $S \longrightarrow\!\!\!\!\!\rightarrow S^?$, we have $S^? \equiv S\{L_1/_1\alpha, \ldots, L_n/_n\alpha\}$, where, for each i, $L_i \equiv \alpha$ or $L_i \equiv x_\odot(x\bullet\alpha)$, that is the only reductions available are the η-expansions of α's; therefore, S is $\mathtt{SN}$.
For the inductive step, suppose there is some infinite reduction sequence from S. Then, in this sequence it must be the case that,

$$S \longrightarrow\!\!\!\!\!\rightarrow S_1 \longrightarrow S_1' \text{ , with } S_1 \equiv S\{L_1/_1\alpha, \ldots, L_n/_n\alpha\} \text{ , some } L_i\text{s as above,} \\ \text{and } S_1' \equiv S'\{L_1'/_1\alpha, \ldots, L_{n+k}'/_{n+k}\alpha\}$$

where the reduction $S \longrightarrow S'$ produces $k \in \mathbb{Z}$ new occurrences of α. In S_1', all α's are substituted for some of the L_is (denoted by $L_j',\ j = 1, \ldots, n+k$). But now,

$$S^p \equiv S\{K_1/_1\alpha, \ldots, K_n/_n\alpha\} \longrightarrow S'\{K_1'/_1\alpha, \ldots, K_{n+k}'/_{n+k}\alpha\} \equiv S'^p$$

where the K_i's are selected from the K_is and, since S^p is $\mathtt{SN}$, S'^p is $\mathtt{SN}$. Moreover, $l(S'^p) < l(S^p)$, thus $S' \in \mathtt{SN}$, by IH. But clearly $S' \longrightarrow\!\!\!\!\!\rightarrow S_1'$, thus there is an infinite reduction sequence starting from S', $\lightning$.

Hence, S is $\texttt{SN}$. ■

Back to the sequent calculus, we order derivable sequents by their degree.

Definition 2.19
Let $G \in \texttt{DuCa}^*$ and $\sigma \in T_G$. Then, the degree of σ, $d(\sigma)$ is:

$$d(\sigma) := (c(\sigma), \texttt{ncut}(\sigma))$$

where:

- if $\sigma \;\equiv\; K : A \mid \Gamma \;\rightarrow\; \Theta$, then $c(\sigma) = c(\Gamma) + c(\Theta)$, $\texttt{ncut}(\sigma) = 1$,
- if $\sigma \;\equiv\; \Gamma \;\rightarrow\; \Theta \mid M : A$, then $c(\sigma) = c(\Gamma) + c(\Theta)$, $\texttt{ncut}(\sigma) = 1$,
- if $\sigma \;\equiv\; \Gamma \mid S \mid\!\rightarrow\; \Theta$, then $c(\sigma) = c(\Gamma) + c(\Theta)$, $\texttt{ncut}(\sigma) = 0$,
- if $\Gamma \;\equiv\; x_1 : A_1, x_2 : A_2, \ldots, x_n : A_n$, then $c(\Gamma) = c(A_1) + c(A_2) + \cdots + c(A_n)$,
- if $\Theta \;\equiv\; \alpha_1 : B_1, \alpha_2 : B_2, \ldots, \alpha_n : B_m$, then $c(\Theta) = c(B_1) + c(B_2) + \cdots + c(B_m)$,
- if A is some type, then $c(A)$ is its complexity, that is the number of connectives contained in A.

We order degrees lexicographically. ⊣

We define the set of reducible sequents, which is a subset of derivable sequents.

Definition 2.20
The set of reducible sequents $\texttt{Red}$ is defined by induction on the degree of (derivable) sequents:

- if $d(\sigma) = (0, n)$, $n \in \{0, 1\}$ and $\sigma \in T_G$, then $\sigma \in \texttt{Red} \iff G \in \texttt{SN}$
- if $d(\sigma) = (c, n)$, $n \in \{0, 1\}$, $c > 0$ and $\sigma \in T_G$, then $\sigma \in \texttt{Red} \iff cl(\sigma) \subset \texttt{Red}$
- if $d(\sigma) = (c, 1)$, $c > 0$, $\sigma \;\equiv\; \Gamma \;\rightarrow\; \Theta \mid M : A$, $\sigma \in T_M$, then $cl(\sigma)$ is the set:

  $$\sigma_2 \;\equiv\; \Gamma, \Gamma_0 \mid M \bullet K_0 \mid\!\rightarrow\; \Theta, \Theta_0$$

  if $\sigma_1 \equiv K_0 : A \mid \Gamma_0 \;\rightarrow\; \Theta_0 \in (\texttt{Red} \cap T_{K_0})$, $d(\sigma_1) < d(\sigma)$, $d(\sigma_2) < d(\sigma)$, and if M is neutral, then K_0 is neutral.[10]

[10]This is in fact an abbreviation for:

$$cl(\sigma) := \{\sigma_2 \mid \exists A, K_0, \Gamma_0, \Theta_0, \sigma_1 .[\ (\sigma_2 \;\equiv\; \Gamma, \Gamma_0 \mid M \bullet K_0 \mid\!\rightarrow\; \Theta, \Theta_0) \;\wedge\; (\sigma_1 \;\equiv\; K_0 : A \mid \Gamma_0 \;\rightarrow\; \Theta_0) \\ \wedge (\sigma_1 \in (\texttt{Red} \cap T_{K_0})) \wedge (d(\sigma_1) < d(\sigma)) \wedge (d(\sigma_2) < d(\sigma)) \wedge (M \text{ neutral} \implies K_0 \text{ neutral})]\}$$

Note that σ, σ_1 being derivable implies that σ_2 is derivable, thus $d(\sigma_2)$ is defined.

- if $d(\sigma) = (c, 1),\ c > 0,\ \sigma\ \equiv\ K : A \,\|\, \Gamma\ \rightarrow\ \Theta,\ \sigma \in T_K$, then $cl(\sigma)$ is the set:

$$\sigma_2\ \equiv\ \Gamma, \Gamma_0 \,\|\, M_0 \bullet K \,\|\!\rightarrow\ \Theta, \Theta_0$$

  if $\sigma_1 \equiv \Gamma_0\ \rightarrow\ \Theta_0 \,\|\, M_0 : A \in (\mathtt{Red} \cap T_{M_0})$, $d(\sigma_1) < d(\sigma)$, $d(\sigma_2) < d(\sigma)$, and M_0 is neutral.

- if $d(\sigma) = (c, 0),\ c > 0,\ \sigma\ \equiv\ \Gamma \,\|\, S \,\|\!\rightarrow\ \Theta,\ \sigma \in T_S$, then $cl(\sigma)$ is the union of the sets:

$$\begin{array}{ll} \sigma_1 \equiv\ \Gamma\ \rightarrow\ \Theta - \{\beta : B\} \,\|\, (S)_{\odot}\beta : B & , \text{if } \beta : B \in \Theta,\ c(B) > 0 \\ \sigma_2 \equiv\ y_{\odot}(S) : B \,\|\, \Gamma - \{y : B\}\ \rightarrow\ \Theta & , \text{if } y : B \in \Gamma,\ c(B) > 0 \end{array}$$

$\dashv$

Note that the above definition is valid. In all cases, the question of $\sigma \in \mathtt{Red}$ reduces to questions of $\sigma' \in \mathtt{Red}$, with $d(\sigma') < d(\sigma)$. For example, in the last case we have that, for all such σ_1, $d(\sigma_1) < d(\sigma)$, since we subtract a non-base type B from Θ, and σ being derivable implies σ_1 is derivable.
Note also that , if $\sigma \longrightarrow \sigma'$, then $d(\sigma) = d(\sigma')$.
The following proposition assures that for any type A we can find terms and coterms typed with A using only variables and covariables of base type.

Proposition 2.21 *For any type A there exist derivable sequents:*

$$\begin{array}{l} \sigma_1 \equiv\ \Gamma_1\ \rightarrow\ \Theta_1 \,\|\, M : A \\ \sigma_2 \equiv\ K : A \,\|\, \Gamma_2\ \rightarrow\ \Theta_2 \end{array}$$

such that $d(\sigma_1) = d(\sigma_2) = (0, 1)$ and M, K are neutral and $\mathtt{SN}$.

Proof: We derive σ_1 by:

$$\dfrac{\dfrac{x : X\ \rightarrow\ \alpha : X, \beta : A \,\|\, x : X \qquad \alpha : X \,\|\, x : X\ \rightarrow\ \alpha : X, \beta : A}{x : X \,\|\, x \bullet \alpha \,\|\!\rightarrow\ \alpha : X, \beta : A}}{x : X\ \rightarrow\ \alpha : X \,\|\, (x \bullet \alpha)_{\odot}\beta : A}$$

and similarly σ_2. ■

The following lemma shows the relation between the reducibility set $\mathtt{Red}$ of sequents and the set $\mathtt{SN}$ of strongly normalizing elements of $\mathtt{DuCa}^*$.

Lemma 2.22 *Let σ be some derivable sequent, then,*

CR1: *If $\sigma \in (T_G \cap \mathtt{Red})$, some G, then G is* $\mathtt{SN}$.

CR3: *If $\sigma \in T_G$, some neutral G, and $\sigma \longrightarrow \sigma'$ implies that $\sigma' \in \mathtt{Red}$, then $\sigma \in \mathtt{Red}$.*
This implies:
CR3': *If $\sigma \in T_G$, G neutral and* $\mathtt{SN}$, *then $\sigma \in \mathtt{Red}$.*

CR2: *If $\sigma \in (\texttt{Red} \cap T_G)$, some G, and $\sigma \longrightarrow \sigma'$, then $\sigma' \in \texttt{Red}$.*

Proof: See the Appendix. ■

A straightforward corollary of the lemma is the following.

Corollary 2.23 *Let G be some element of* DuCa*, then:*

- *If G is neutral and* SN*, then $T_G \subset \texttt{Red}$.*
- *If $(T_G \cap \texttt{Red}) \neq \emptyset$ and G is neutral, then $T_G \subset \texttt{Red}$.*
- *If K is some coterm and $(T_K \cap \texttt{Red}) \neq \emptyset$, then $T_K \subset \texttt{Red}$.*

Proof: The first claim is clear from CR3'.
For the second, if $\sigma \in (T_G \cap \texttt{Red})$, then, by CR1, G is SN, so $T_G \subset \texttt{Red}$ by first claim.
For the last claim, if K is neutral, then we use the previous claim. Otherwise, assume $K \equiv x.(S)$ and take some $\sigma \equiv x.(S) : A \blacktriangleright \Gamma \twoheadrightarrow \Theta \in T_K$. Then, $\sigma \in \texttt{Red}$ iff for all neutral M_0 and $\sigma_1 \equiv \Gamma_0 \twoheadrightarrow \Theta_0 \blacktriangleright M_0 : A \in (\texttt{Red} \cap T_{M_0})$ with $d(\sigma_1) < d(\sigma)$:
if $\sigma_2 \equiv \Gamma, \Gamma_0 \blacktriangleright M_0 \bullet x.(S) \blacktriangleright\twoheadrightarrow \Theta, \Theta_0$ and $d(\sigma_2) < d(\sigma)$, then $\sigma_2 \in \texttt{Red}$.
For this, it suffices to show that $M_0 \bullet x.(S)$ is SN, by first claim. But $x.(S)$ is SN, by hypothesis and CR1, and thus S is SN. Moreover, by CR1, M_0 is also SN, thus, by lemma 2.17, $M_0 \bullet x.(S)$ is SN. ■

We can prove the following lemma for sequents typing non-neutral terms.

Lemma 2.24 *Let $\sigma \equiv \Gamma \twoheadrightarrow \Theta \blacktriangleright (S).\alpha : A$ be some derivable sequent. If, for all coterms L with $T_L(A) \cap \texttt{Red} \neq \emptyset$, we have $T_{S\{L/\alpha\}} \subset \texttt{Red}$, then $\sigma \in \texttt{Red}$.*

Proof: Assume the hypothesis. $\sigma \in \texttt{Red}$ iff for all coterms K_0 and sequents $\sigma_1 \in (T_{K_0} \cap \texttt{Red})$ with $d(\sigma_1) < d(\sigma)$ and $\sigma_1 \equiv K_0 : A \blacktriangleright \Gamma_0 \twoheadrightarrow \Theta_0$:

$$\text{if } \sigma_2 \equiv \Gamma, \Gamma_0 \blacktriangleright (S).\alpha \bullet K_0 \blacktriangleright\twoheadrightarrow \Theta, \Theta_0 \text{ and } d(\sigma_2) < d(\sigma)\text{ , then } \sigma_2 \in \texttt{Red}.$$

Now take any such σ_1, K_0, σ_2. By corollary 2.23, $T_\alpha \subset \texttt{Red}$, $\therefore$ $T_S \subset \texttt{Red}$, by hypothesis. Since $(S).\alpha$ is typed, S is also typed, thus, by CR1, $S \in$ SN. Since K_0 is also SN, by CR1, we show by induction on $l(S) + l(K_0)$ that, if $\sigma_1 \in (T_{K_0}(A) \cap \texttt{Red})$ and for all coterms L with $T_L(A) \cap \texttt{Red} \neq \emptyset$ we have $T_{S\{L/\alpha\}} \subset \texttt{Red}$, then $\sigma_2 \in \texttt{Red}$.
So suppose that $\sigma_2 \longrightarrow \sigma_2'$. By CR3, it suffices to show that $\sigma_2' \in \texttt{Red}$. Now, σ_2' may be:

- $\sigma_2' \equiv \Gamma, \Gamma_0 \blacktriangleright S\{K_0/\alpha\} \blacktriangleright\twoheadrightarrow \Theta, \Theta_0$, where, since $\sigma_1 \in (T_{K_0}(A) \cap \texttt{Red})$, we have $T_{K_0} \cap \texttt{Red} \neq \emptyset$ and, by hypothesis, $T_{S\{K_0/\alpha\}} \subset \texttt{Red}$, $\therefore$ $\sigma_2' \in \texttt{Red}$.
- $\sigma_2' \equiv \Gamma, \Gamma_0 \blacktriangleright (S').\alpha \bullet K_0' \blacktriangleright\twoheadrightarrow \Theta, \Theta_0$, where $S \longrightarrow S'$ or $K_0 \longrightarrow K_0'$.
By CR2, $\sigma_1' \equiv K_0' : A \blacktriangleright \Gamma_0 \twoheadrightarrow \Theta_0 \in \texttt{Red}$. Thus, if we show that $T_{S'\{L/\alpha\}} \subset \texttt{Red}$, for all relevant L, then we can use the IH on $l(S) + l(K_0)$ and get $\sigma_2' \in \texttt{Red}$.
Now take some relevant L. By corollary 2.23, it suffices to show that $S'\{L/\alpha\}$ is SN.

Since $(S).\alpha$ and L are typed with A, $(S).\alpha \bullet L$ is also typed, $\therefore$ $S\{L/\alpha\}$ is typed, $\therefore$ $S\{L/\alpha\} \in \mathtt{SN}$, by hypothesis and CR1. Therefore, if $S\{L/\alpha\} \longrightarrow S'\{L/\alpha\}$, we're done.

The only case this latter reduction cannot happen is when $S \longrightarrow S'$ is a reduction affecting some occurrence of α and which cannot happen with L in its place. By inspection of the reduction rules, the only possible case is this of α η-expanding inside S and of L being non-simple, so that L cannot η-expand. That is, there is some context C such that,

$$\begin{array}{rcl} S \equiv & C\{\alpha\} \longrightarrow C\{x_{\odot}(x \bullet \alpha)\} & \equiv S' \\ S\{L/\alpha\} \equiv & (C\{\alpha\})\{L/\alpha\} \not\longrightarrow (C\{x_{\odot}(x \bullet \alpha)\})\{L/\alpha\} & \equiv S'\{L/\alpha\} \quad , L \text{ non-simple} \end{array}$$

Take then $S_1 \equiv C\{\alpha_1\}$, with α_1 fresh. Since $S\{L/\alpha\} \in \mathtt{SN}$, $S_1\{L/\alpha\} \in \mathtt{SN}$ as well, by lemma 2.18 and the fact that α_1 doesn't follow a cut (otherwise it couldn't η-expand). Further, L is also $\mathtt{SN}$, $\therefore$ $x_{\odot}(x \bullet L)$ is $\mathtt{SN}$. Then, $(S_1\{L/\alpha\}).\alpha_1 \bullet x_{\odot}(x \bullet L)$ is $\mathtt{SN}$ by lemma 2.17, therefore $S'\{L/\alpha\} \equiv (S_1\{L/\alpha\})\{x_{\odot}(x \bullet L)/\alpha_1\}$ is $\mathtt{SN}$. ■

Now, strong normalization of $\mathtt{DuCa}^*$ follows from the next theorem.

Theorem 2.25 *Let G be some element of $\mathtt{DuCa}^*$ with free variables amongst $x_1, x_2, \ldots, x_n$ and covariables amongst $\alpha_1, \alpha_2, \ldots, \alpha_m$.*
Then, for all coterms L_i with $T_{L_i} \cap \mathtt{Red} \neq \emptyset$, $i = 1, \ldots, m$, and neutral terms N_j with $T_{N_j} \cap \mathtt{Red} \neq \emptyset$, $j = 1, \ldots, n$,

$$T_{G\{f\}} \subset \mathtt{Red} \quad , \textit{where} \quad f := N_1/x_1, \ldots, N_n/x_n, L_1/\alpha_1, \ldots, L_m/\alpha_m$$

Proof: Note first that if $G\{f\}$ is not typed, then the claim trivially holds. Assume then that $T_{G\{f\}} \neq \emptyset$. We do induction on G. The base cases, that is of $G \equiv x$ or $G \equiv \alpha$, are clear by corollary 2.23.
For the inductive step, we do a case analysis on G, showing only the most difficult cases. Let $\sigma \in T_{G\{f\}}(A)$, some type A:

– $G \equiv \langle M, N\rangle$. Then $G\{f\} \equiv \langle M\{f\}, N\{f\}\rangle$, thus both $M\{f\}, N\{f\}$ are typed. By IH, $T_{M\{f\}} \subset \mathtt{Red}$ and $T_{N\{f\}} \subset \mathtt{Red}$, so, by CR1, both $N\{f\}$ and $M\{f\}$ are $\mathtt{SN}$.
But then $\langle M\{f\}, N\{f\}\rangle \equiv \langle M, N\rangle\{f\}$ is $\mathtt{SN}$, so, by corollary 2.23, $\sigma \in \mathtt{Red}$.

– $G \equiv (S)_{\odot}\alpha$. Then $G\{f\} \equiv (S\{f, \alpha/\alpha\})_{\odot}\alpha$, thus $S\{f, \alpha/\alpha\}$ is typed. By IH, $T_{S\{f,\alpha/\alpha\}} \subset \mathtt{Red}$, therefore, by CR1, $S\{f, \alpha/\alpha\}$ is $\mathtt{SN}$.
But then $(S\{f, \alpha/\alpha\})_{\odot}\alpha$ is $\mathtt{SN}$, so, by corollary 2.23, $\sigma \in \mathtt{Red}$.

– $G \equiv (S).\alpha$. By IH, $T_{S\{f,L/\alpha\}} \subset \mathtt{Red}$, for any coterm L with $T_L \cap \mathtt{Red} \neq \emptyset$. Then, $T_{S\{f,L/\alpha\}} \subset \mathtt{Red}$ for any coterm L with $T_L(A) \cap \mathtt{Red} \neq \emptyset$. Now $S\{f, L/\alpha\} \equiv (S\{f\})\{L/\alpha\}$, since f is a substitution in $(S).\alpha$ and it doesn't introduce new α's. Moreover, $\sigma \equiv \Gamma \twoheadrightarrow \Theta \mid (S\{f\}).\alpha : A$, some Γ, Θ, so, by lemma 2.24, $\sigma \in \mathtt{Red}$.

– $G \equiv x.(S)$. By definition and corollary 2.23, it suffices to show that, for every relevant neutral M_0 and $\sigma_1 \in (T_{M_0}(A) \cap \texttt{Red})$, $M_0 \bullet (x.(S)\{f\})$ is $\texttt{SN}$. Now $G\{f\} \equiv x.(S\{f, x/x\})$, thus $S\{f, x/x\}$ is typed and, by IH and CR1, $S\{f, x/x\} \in \texttt{SN}$. Moreover, $M_0 \in \texttt{SN}$ by CR1, therefore $M_0 \bullet (x.(S)\{f\})$ is $\texttt{SN}$ by lemma 2.17.

– $G \equiv M \bullet K$. Then $G\{f\} \equiv M\{f\} \bullet K\{f\}$, thus both $M\{f\}, K\{f\}$ are typed: say with type B. There are two subcases:

- $G \equiv (S).\alpha \bullet K$. Let
$$\sigma \;\equiv\; \Gamma \blacktriangleright ((S).\alpha \bullet K)\{f\} \blacktriangleright\!\!\rightarrow\; \Theta$$
Since σ is derivable, σ_1 and σ_2 are also derivable:
$$\sigma_1 \equiv\; \Gamma \;\rightarrow\; \Theta \blacktriangleright (S).\alpha\{f\} : B$$
$$\sigma_2 \equiv\; K\{f\} : B \blacktriangleright \Gamma \;\rightarrow\; \Theta$$
By IH, $T_{(S).\alpha\{f\}} \subset \texttt{Red}$, so $\sigma_1' \in (\texttt{Red} \cap T_{(S).\alpha\{f\}})$, where
$$\sigma_1' \equiv\; \Gamma \;\rightarrow\; \Theta, \beta : A^b \blacktriangleright (S).\alpha\{f\} : B$$
with A^b some big type and β fresh.
Now note that, by IH, $\sigma_2 \in \texttt{Red}$ and, because of A^b, $d(\sigma_2) < d(\sigma_1')$. Then, by definition, $\sigma_1' \in \texttt{Red}$ implies $\sigma' \in (T_{((S).\alpha\bullet K)\{f\}} \cap \texttt{Red})$, where
$$\sigma' \;\equiv\; \Gamma \blacktriangleright ((S).\alpha \bullet K)\{f\} \blacktriangleright\!\!\rightarrow\; \Theta, \beta : A^b$$
with $d(\sigma') = (c, 0) < (c, 1) = d(\sigma_1')$, some c.
But then, by corollary 2.23, $\sigma \in \texttt{Red}$.
- $G \equiv M_0 \bullet K$, M_0 neutral: treated dually as the above case. ■

Corollary 2.26 *If $G \in \texttt{DuCa}^*$ is typed, then G is* $\texttt{SN}$.

Proof: Straightforward from the previous theorem and CR1. ■

2.2.2 The call-by-value CPS translation

CPS translations in general are very useful when examining extensions of lambda calculi, since they supply us with a way of projecting given properties of the source calculus to a well-behaved lambda calculus. In our case, using a CPS translation of the Dual Calculus enables us to prove that the reduction relation yielded by some β-reduction rules is strongly normalizing, since these reduction rules are projected in the target calculus.
The call-by-value CPS translation of the Dual Calculus we'll be using was defined in [Wad03a]. We quote that definition.

Definition 2.27 (Call-by-value CPS translation. [Wad03a])
Let M be a term, V a value, K a coterm and S a statement. Then, their call-by-value CPS translations are $(M)^v, (V)^V, (K)^v$ and $(S)^v$ respectively, defined by:

$$
\begin{array}{lcl}
(x)^V & \equiv & x \\
(\langle V, W\rangle)^V & \equiv & \langle (V)^V, (W)^V \rangle \\
(\langle V\rangle \texttt{inl})^V & \equiv & \texttt{inl}(V)^V \\
(\langle W\rangle \texttt{inr})^V & \equiv & \texttt{inr}(W)^V \\
([K]\texttt{not})^V & \equiv & K^v \\
\\
(x)^v & \equiv & \boldsymbol{\lambda}\gamma.\gamma x \\
(\langle M, N\rangle)^v & \equiv & \boldsymbol{\lambda}\gamma.(M)^v(\boldsymbol{\lambda}x.(N)^v(\boldsymbol{\lambda}y.\gamma\langle x, y\rangle)) \\
(\langle M\rangle \texttt{inl})^v & \equiv & \boldsymbol{\lambda}\gamma.(M)^v(\boldsymbol{\lambda}x.\gamma(\texttt{inl}x)) \\
(\langle N\rangle \texttt{inr})^v & \equiv & \boldsymbol{\lambda}\gamma.(N)^v(\boldsymbol{\lambda}y.\gamma(\texttt{inr}y)) \\
([K]\texttt{not})^v & \equiv & \boldsymbol{\lambda}\gamma.\gamma(\lambda z.(K)^v z) \\
((S).\alpha)^v & \equiv & \boldsymbol{\lambda}\alpha.(S)^v \\
\\
(\alpha)^v & \equiv & \boldsymbol{\lambda}z.\alpha z \\
([K, L])^v & \equiv & \boldsymbol{\lambda}z.\texttt{case}\ z\ \texttt{of}\ \texttt{inl}x \Rightarrow (K)^v x,\ \texttt{inr}y \Rightarrow (L)^v y \\
(\texttt{fst}[K])^v & \equiv & \boldsymbol{\lambda}z.\texttt{case}\ z\ \texttt{of}\ \langle x, -\rangle \Rightarrow (K)^v x \\
(\texttt{snd}[L])^v & \equiv & \boldsymbol{\lambda}z.\texttt{case}\ z\ \texttt{of}\ \langle -, y\rangle \Rightarrow (L)^v y \\
(\texttt{not}\langle M\rangle)^v & \equiv & \boldsymbol{\lambda}z.(\lambda\gamma.(M)^v\gamma)z \\
(x.(S))^v & \equiv & \boldsymbol{\lambda}x.(S)^v \\
\\
(M \bullet K)^v & \equiv & (M)^v(K)^v
\end{array}
$$

We have the following translation of types.

$$
\begin{array}{lcl}
(X)^V & \equiv & X \\
(A\&B)^V & \equiv & (A)^V \times (B)^V \\
(A \vee B)^V & \equiv & (A)^V + (B)^V \\
(\neg A)^V & \equiv & (A)^V \to R
\end{array}
$$

$\dashv$

In the above definition, boldface lambda-abstractions are administrative, that is they are reduced automatically on translation.
As follows from the following definition, the target calculus is a restriction of the simply-typed lambda calculus with products and sums.

Definition 2.28 (The target calculus. [Wad03a])
The CPS target calculus is a typed calculus containing values, terms, coterms and state-

ments:

$$\begin{array}{llcl}
\text{Type} & A, B & ::= & X \mid A \times B \mid A + B \mid A \to R \\
\\
\text{Value} & V, W & ::= & x \mid \langle V, W \rangle \mid \mathtt{inl} V \mid \mathtt{inr} W \mid K \\
\text{Term} & M, N & ::= & \lambda\alpha.S \\
\text{Coterm} & K, L & ::= & \lambda x.S \\
\text{Statement} & S, T & ::= & \alpha V \mid \mathtt{case}\ V\ \mathtt{of}\ \langle x, - \rangle \Rightarrow S \mid \mathtt{case}\ V\ \mathtt{of}\ \langle -, y \rangle \Rightarrow T \mid \\
& & & \mathtt{case}\ V\ \mathtt{of}\ \mathtt{inl} x \Rightarrow S,\ \mathtt{inr} y \Rightarrow T \mid M\,V
\end{array}$$

The reduction relation is defined by the following reduction rules.

$$\begin{array}{lllcl}
(\beta\times_1) & \mathtt{case}\ \langle V, W \rangle\ \mathtt{of}\ \langle x, - \rangle \Rightarrow S & & \to & S\{V/x\} \\
(\beta\times_2) & \mathtt{case}\ \langle V, W \rangle\ \mathtt{of}\ \langle -, y \rangle \Rightarrow T & & \to & T\{W/y\} \\
(\beta+_1) & \mathtt{case}\ \mathtt{inl} V\ \mathtt{of}\ \mathtt{inl} x \Rightarrow S,\ \mathtt{inr} y \Rightarrow T & & \to & S\{V/x\} \\
(\beta+_2) & \mathtt{case}\ \mathtt{inr} W\ \mathtt{of}\ \mathtt{inl} x \Rightarrow S,\ \mathtt{inr} y \Rightarrow T & & \to & T\{W/y\} \\
(\beta\to) & (\lambda\alpha.S)(\lambda x.T) & & \to & S\{T\{-/x\}/\alpha\,-\}
\end{array}$$

⊣

Note that in the above definition $S\{T\{-/x\}/\alpha\,-\}$ stands for S with all occurrences of the form $\alpha\,V$ replaced by $T\{V/x\}$.
Calculi of this type have been investigated in depth and many nice properties are known to hold. One such property is strong normalization.

Proposition 2.29 *The target calculus of the call-by-value CPS translation is* SN *under the given reduction relation.*

Proof: It suffices to show that the target calculus is a restriction of the lambda calculus with sums and products of Dougherty [Dou93], since the latter was shown to be SN.
But this clearly holds. For example, $\mathtt{case}\ V\ \mathtt{of}\ \langle x, - \rangle \Rightarrow S$ is an abbreviation for $(\lambda x.S)\pi_1 V$ and $\mathtt{case}\ V\ \mathtt{of}\ \mathtt{inl} x \Rightarrow S,\ \mathtt{inr} y \Rightarrow T$ is an abbreviation for $[\lambda x.S, \lambda y.T]V$.
Moreover, the reductions of the target calculus are valid in the calculus of Dougherty. ■

A very handy property of the CPS translation is that it preserves reductions.

Proposition 2.30 ([Wad03a]) *Let M, N, K, L, S, T be in the Dual Calculus. Then,*

$$\begin{array}{lcl}
M \longrightarrow^v N & \Longrightarrow & (M)^v \twoheadrightarrow (N)^v \\
K \longrightarrow^v L & \Longrightarrow & (K)^v \twoheadrightarrow (L)^v \\
S \longrightarrow^v T & \Longrightarrow & (S)^v \twoheadrightarrow (T)^v
\end{array}$$

In particular, if the left-hand side reduction is of type $\beta L, \beta R, \eta L, \eta R$ or ν, then, in the right-hand side, $\twoheadrightarrow$ can be replaced by $\equiv$. Otherwise, it can be replaced by $\longrightarrow_+$.

Proof: The proposition is proven in [Wad03a]. The last part of it is not completely stated in that paper, yet it is straightforward from the definition of the CPS translation and the results that proceed this proposition in [Wad03a]. ■

Wadler goes further by defining an inverse CPS translation, which translates elements M of the target calculus to objects $(M)_v$ of the Dual Calculus. Finally, he proves that the CPS translation is a *reflection*, as the following proposition states.

Proposition 2.31 ([Wad03a]) *Let M, K, S be in the Dual Calculus and N, L, T be in the target calculus. Then,*

$$\begin{array}{lcll} M \twoheadrightarrow^v (N)_v & \Longleftrightarrow & (M)^v \twoheadrightarrow N & , \textit{and} \quad ((N)_v)^v \equiv N \\ K \twoheadrightarrow^v (L)_v & \Longleftrightarrow & (K)^v \twoheadrightarrow L & , \textit{and} \quad ((L)_v)^v \equiv L \\ S \twoheadrightarrow^v (T)_v & \Longleftrightarrow & (S)^v \twoheadrightarrow T & , \textit{and} \quad ((T)_v)^v \equiv T \end{array}$$

Proof: As in [Wad03a]. ■

2.2.3 Strong normalization of call-by-value reduction in DuCa

Using the results of the previous sections, we will show that the call-by-value reduction relation is strongly normalizing in `DuCa` when ν-rules are omitted and some restrictions on η-rules are placed in order to avoid loops. We will call this reduction relation $R_v^{\beta\eta}$. The reason for not utilizing ν-rules as well is that, if we did so, we would have to prove `SN` for a more complicated analog of `DuCa`*, and that proof is already complicated.
Regarding the restrictions on η-rules, loops may arise, for example, in the following cases.

$$K \longrightarrow x.(x \bullet K) \longrightarrow x.(x \bullet y.(y \bullet K)) \longrightarrow \dots$$
$$K \longrightarrow x.(x \bullet K) \longrightarrow y.(y \bullet x.(x \bullet K)) \longrightarrow \dots$$

Therefore, $R_v^{\beta\eta}$ is defined as follows.

Definition 2.32
$R_v^{\beta\eta}$ is the one-step reduction relation yielded by the following rules, when these are applied to subobjects of `DuCa` objects.

$$\begin{array}{lllll}
(\beta\&_1) & \langle V, W\rangle \bullet \mathtt{fst}[K] & \rightarrow & V \bullet K \\
(\beta\&_2) & \langle V, W\rangle \bullet \mathtt{snd}[L] & \rightarrow & W \bullet L \\
(\beta\vee_1) & \langle V\rangle\mathtt{inl} \bullet [K, L] & \rightarrow & V \bullet K \\
(\beta\vee_2) & \langle W\rangle\mathtt{inr} \bullet [K, L] & \rightarrow & W \bullet L \\
(\beta\neg) & [K]\mathtt{not} \bullet \mathtt{not}\langle M\rangle & \rightarrow & M \bullet K \\
(\beta L) & V \bullet x.(S) & \rightarrow & S\{V/x\} \\
(\beta R) & (S).\alpha \bullet K & \rightarrow & S\{K/\alpha\} \\
\\
(\eta L) & K & \rightarrow & x.(x \bullet K) \\
(\eta R) & M & \rightarrow & (M \bullet \alpha).\alpha
\end{array}$$

The η-rules are not allowed to be applied to terms [resp. coterms] that are immediately followed by [immediately follow] some cut '$\bullet$'. Moreover, in ηL, K is not of the form $y.(S)$

and, in ηR, M is not of the form $(S).\beta$.
For $G, H \in \mathtt{DuCa}$, $(G, H) \in R_v^{\beta\eta}$ is written simply $G \longrightarrow H$. $\dashv$

The SN result is the following.

Theorem 2.33 *Let $G \in \mathtt{DuCa}$. Then there is no infinite $R_v^{\beta\eta}$-reduction sequence starting from G.*

Proof: Let $G \in \mathtt{DuCa}$ and suppose that there is some infinite $R_v^{\beta\eta}$-reduction sequence starting from G: say $G \longrightarrow G_1 \longrightarrow G_2 \longrightarrow \ldots$. Then, by proposition 2.30, there is a sequence:

$$(G)^v \twoheadrightarrow (G_1)^v \twoheadrightarrow (G_2)^v \twoheadrightarrow \ldots$$

in the target calculus of the CPS translation. By proposition 2.29, the target calculus is SN, so there is some last element in the sequence, say M_t. Moreover, there is some i_t such that, for all $i \geq i_t$, $(G_i)^v \equiv M_t$. But then, by proposition 2.30, in the sequence

$$G_{i_t} \longrightarrow G_{i_t+1} \longrightarrow G_{i_t+2} \longrightarrow \ldots$$

all reductions are instances of $\beta L, \beta R, \eta L$ or ηR. We claim then that this latter infinite reduction sequence produces an infinite reduction sequence in $\mathtt{DuCa}^*$.
Indeed, G_{i_t} is also an element of $\mathtt{DuCa}^*$. Moreover, the reduction relation for $\mathtt{DuCa}^*$ consists of generalizations of the call-by-value $\beta L, \beta R$ and restricted $\eta L, \eta R$ rules, with the only difference that η-rules introduce the neutralizing dot '$\odot$', instead of simple dot '.'. But note that, under the restrictions we placed, a term $(M \bullet \alpha).\alpha$ generated by ηR is not immediately followed by some cut, therefore it doesn't make any difference if the dot is neutralizing or not. Similarly for coterms $x.(x \bullet K)$ generated by ηL. Therefore, an infinite reduction sequence in the $\mathtt{DuCa}^*$ can be produced from G_{i_t}, $\lightning$ to corollary 2.26. ■

2.3 A call-by-value reduction to satisfy both SN and CR

In this section we investigate a restricted version of the call-by-value reduction relation in DuCa which satisfies both CR and SN. As done previously, ν-reductions are excluded for simplicity. In fact, the version of the call-by-value reduction relation we use is $R_v^{\beta\eta}$ of the previous section with some more restrictions on the η-rules.
Though SN is proven for $R_v^{\beta\eta}$, there are several ways in which different reductions of the same element cannot reduce to a common element. These 'separations' occur, for example, in the following cases.

$$\begin{array}{rcccl} V \bullet K & \longleftarrow & \langle V, W\rangle \bullet \mathtt{fst}[K] & \longrightarrow & \langle (V \bullet \alpha).\alpha, W\rangle \bullet \mathtt{fst}[K] \\ V \bullet K & \longleftarrow & \langle V\rangle\mathtt{inl} \bullet [K, L] & \longrightarrow & \langle (V \bullet \alpha).\alpha\rangle\mathtt{inl} \bullet [K, L] \end{array}$$

The reduction relation with the new restrictions is called $R_v^{\beta\eta\prime}$.

Definition 2.34
The reduction rules of $R_v^{\beta\eta\prime}$ are those of $R_v^{\beta\eta}$ (definition 2.32) with the addition of the following restriction.

η-rules are not allowed to be applied to values that are immediate subjects to $\langle \_, V\rangle$, $\langle V, \_\rangle$, $\langle _\rangle$inl or $\langle _\rangle$inr; and to coterms that are immediate subjects to $[_]$not.

$\dashv$

Note that, by these restrictions, values followed by a cut cannot be reduced to non-values. The separations we saw above could be solved by allowing ν-rules of reduction. However, the above restrictions on η-rules would be inevitable in order to have SN. Of course, here there is no matter for ν-rules whatsoever, since we haven't proven SN with ν-rules.
It is clear that, since $R_v^{\beta\eta\prime}$ is a restriction of $R_v^{\beta\eta}$ and the latter is SN, $R_v^{\beta\eta\prime}$ is SN. Therefore, we need only to show satisfaction of the CR property, which is much easier with SN at hand. Indeed, it suffices to prove WCR, by the following proposition[11].

Proposition 2.35 (Newman)

$$\texttt{SN} \wedge \texttt{WCR} \Rightarrow \texttt{CR}$$

Proof: As in [Bar84]. ■

It is convenient to follow a route similar to the one we followed in order to prove CR of the untyped calculus, though some results may be more strict than we need. Indeed, proving CR for the β-reduction relation and the diamond property for the η-reduction relation may not be the most clear steps in order to prove WCR; however, they are steps already proven.
Below, β-reduction relation is $R_v^{\beta\eta\prime}$ restricted to β-rules, and η-reduction relation is $R_v^{\beta\eta\prime}$ restricted to η-rules.

Lemma 2.36 *The β-reduction relation is Church-Rosser.*

Proof: The proof is identical to the one of $\beta\nu$ being CR in the untyped calculus; this latter residing in a previous section. We copy definition 2.2 (except for pν-rules) and thus define a parallel reduction relation, which we show to satisfy the diamond property by proving the analogous versions of lemmata 2.4 and 2.5. Note that the proofs for these analogous lemmata are identical to those of the original ones, with the only difference being the omission of anything that concerns ν-rules. ■

Lemma 2.37 *The η-reduction relation satisfies the diamond property.*

Proof: Nearly identical to proof of lemma 2.7. ■

We now prove WCR.

Lemma 2.38 *$R_v^{\beta\eta\prime}$ satisfies the Weak Church-Rosser property (*WCR*); that is, for all $G, G_1, G_2 \in$ DuCa, if $G_1 \longleftarrow G \longrightarrow G_2$, then there exists some $G_c \in$ DuCa such that $G_1 \twoheadrightarrow G_c \twoheadleftarrow G_2$.*

[11] See definition 2.1 for explanation of WCR (Weak Church-Rosser property).

Proof: The proof is by a case analysis on $G \longrightarrow G_1$ and the possible combinations for $G \longrightarrow G_2$. By lemmata 2.36 and 2.37, we can omit the cases of both reductions being β or both being η. Therefore, by symmetry, we may assume that $G \longrightarrow G_1$ is a β-reduction and $G \longrightarrow G_2$ an η-reduction. In the following diagrams we do the case analysis on $G \longrightarrow G_1$, which is always the topmost horizontal reduction. C is some context and note that we have omitted the trivial cases of $G \longrightarrow G_2$ affecting solely C and not its content. For $G \longrightarrow G_1$ being $\beta\&_1$:

$$
\begin{tikzcd}
C\{\langle V, W\rangle \bullet \mathtt{fst}[K]\} \arrow[r, "\beta"] \arrow[d, "\eta"] \arrow[dd, bend right=60, "\eta"'] & C\{V \bullet K\} \arrow[d, "\eta", "="'] \\
C\{\langle V', W'\rangle \bullet \mathtt{fst}[K']\} \arrow[r, "\beta"] & C\{V' \bullet K'\} \\
C\{\langle V, W\rangle \bullet \mathtt{fst}[x.(x \bullet K)]\} \arrow[r, "\beta"] & C\{V \bullet x.(x \bullet K)\} \arrow[uu, bend right=60, "\beta"']
\end{tikzcd}
$$

The cases of $\beta\&_2, \beta\vee_1, \beta\vee_2$ and $\beta\neg$ are similar.
For $G \longrightarrow G_1$ being βL:

$$
\begin{tikzcd}
 & C\{V \bullet x.(S)\} \arrow[r, "\beta"] \arrow[dl, "\eta"] \arrow[dr, "\eta"] & C\{S\{V/x\}\} \arrow[dl, "\eta"] \arrow[dr, "\eta"] & \\
C\{V' \bullet x.(S)\} \arrow[r, "\beta"] & C\{S\{V'/x\}\} & C\{V \bullet x.(S')\} \arrow[r, "\beta"] & C\{S'\{V/x\}\}
\end{tikzcd}
$$

For $G \longrightarrow G_1$ being βR, we have the dual diagram as above:

$$
\begin{tikzcd}
 & C\{(S).\alpha \bullet K\} \arrow[r, "\beta"] \arrow[dl, "\eta"] \arrow[dr, "\eta"] & C\{S\{K/\alpha\}\} \arrow[dl, "\eta"] \arrow[dr, "\eta"] & \\
C\{(S).\alpha \bullet K'\} \arrow[r, "\beta"] & C\{S\{K'/\alpha\}\} & C\{(S').\alpha \bullet K\} \arrow[r, "\beta"] & C\{S'\{K/\alpha\}\}
\end{tikzcd}
$$

but we must also consider the particular case when $S\{K/\alpha\}$ cannot reduce to $S'\{K/\alpha\}$. It is not difficult to see that this case occurs when

$$
\begin{tikzcd}
C\{\ (C_0\{\alpha\}).\alpha \bullet y.(S)\ \} \arrow[r, "\beta"] \arrow[d, "\eta"] & C\{\ (C_0\{\alpha\})\{y.(S)/\alpha\}\ \} \arrow[d, shift right, "\times" marking] \\
C\{\ (C_0\{x.(x \bullet \alpha)\}).\alpha \bullet y.(S)\ \} \arrow[r, "\beta"] & C\{\ (C_0\{x.(x \bullet \alpha)\})\{y.(S)/\alpha\}\ \} \arrow[u, shift right, "\beta"']
\end{tikzcd}
$$

where $C_0\{\alpha\}$ is some statement S_0, and we use the fact that, by alpha-conversion, we have

$$x.(x \bullet y.(S)) \longrightarrow x.(S\{x/y\}) \equiv y.(S) \;\blacksquare$$

We conclude with the main result of this section.

Theorem 2.39 *$R_v^{\beta\eta\prime}$ is both* SN *and* CR.

Proof: Since $R_v^{\beta\eta\prime}$ is a restriction of $R_v^{\beta\eta}$, every reduction sequence of the former is also a reduction sequence of the latter. By theorem 2.33, $R_v^{\beta\eta}$ is SN, therefore $R_v^{\beta\eta\prime}$ is SN. Furthermore, by proposition 2.35 and lemma 2.38, $R_v^{\beta\eta\prime}$ is CR. $\blacksquare$

3 The second-order case

Girard proposed an extension of the simply-typed lambda calculus, called polymorphic lambda calculus (system **F** [GTL89, SU98], or $\lambda 2$ [Bar92]), which is isomorphic to second-order propositional intuitionistic logic in Curry-Howard style. Second-order propositional intuitionistic logic is an extension of propositional intuitionistic logic by quantifiers ranging over propositions. The extension from the simply-typed lambda calculus to **F** is a very strong one, with regard to the functions that we can represent in each calculus.
In [FLO83] it is shown that the functions which are representable in the simply-typed lambda calculus form a proper subset of the elementary functions. The class of elementary functions is the smallest class of functions which contains the projection functions, successor, $+$, $\dot{-}$ and $\times$, and is closed under composition and bounded sums and products[12]. This is indeed a very 'small' class of functions. On the other hand, in [GTL89] it is shown that the functions representable in **F** are exactly those which are provably total[13] in second-order Peano Arithmetic. This is a substantially 'larger' class of functions.
Consequently, it is interesting and natural to study second-order extensions for the Dual Calculus.

3.1 The natural extension

In this section we generalize the Dual Calculus by adding typing rules which introduce second-order quantifiers to types. The resulting type syntax is this of second-order propositional classical logic. This rather straightforward generalization yields the Second-Order Dual Calculus, or DuCa2.

Definition 3.1
The DuCa2 consists of Types and Objects. The set of objects is the union of the sets of

[12] $x \dot{-} y$ is $x - y$ for $x \geq y$, otherwise 0.
[13] A function f is provably total in a theory T, if there is an algorithm A computing f for which T proves that A terminates on all inputs.

Terms, Coterms and Statements:

Type	A, B	$::=$	$X \mid A\&B \mid A \vee B \mid \neg A \mid \forall X.A \mid \exists X.A$
Object	G, H	$::=$	$M \mid K \mid S$
Term	M, N	$::=$	$x \mid \langle M, N\rangle \mid \langle M\rangle\texttt{inl} \mid \langle N\rangle\texttt{inr} \mid [K]\texttt{not} \mid (S).\alpha$
Coterm	K, L	$::=$	$\alpha \mid [K, L] \mid \texttt{fst}[K] \mid \texttt{snd}[L] \mid \texttt{not}\langle M\rangle \mid x.(S)$
Statement	S, T	$::=$	$M \bullet K$

The typing rules are the same as those of `DuCa` (i.e. of system `GW`):

$$\frac{}{\alpha : A \blacktriangleright \Gamma \rightarrow \Theta, \alpha : A}\,\texttt{id}L \qquad \frac{}{x : A, \Gamma \rightarrow \Theta \blacktriangleleft x : A}\,\texttt{id}R$$

$$\frac{K : A \blacktriangleright \Gamma \rightarrow \Theta}{\texttt{fst}[K] : A\&B \blacktriangleright \Gamma \rightarrow \Theta} \quad \frac{L : B \blacktriangleright \Gamma \rightarrow \Theta}{\texttt{snd}[L] : A\&B \blacktriangleright \Gamma \rightarrow \Theta}\,\&L \qquad \frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A \quad \Gamma \rightarrow \Theta \blacktriangleleft N : B}{\Gamma \rightarrow \Theta \blacktriangleleft \langle M, N\rangle : A\&B}\,\&R$$

$$\frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A}{\Gamma \rightarrow \Theta \blacktriangleleft \langle M\rangle\texttt{inl} : A \vee B} \quad \frac{\Gamma \rightarrow \Theta \blacktriangleleft N : B}{\Gamma \rightarrow \Theta \blacktriangleleft \langle N\rangle\texttt{inr} : A \vee B}\,\vee R \qquad \frac{K : A \blacktriangleright \Gamma \rightarrow \Theta \quad L : B \blacktriangleright \Gamma \rightarrow \Theta}{[K, L] : A \vee B \blacktriangleright \Gamma \rightarrow \Theta}\,\vee L$$

$$\frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A}{\texttt{not}\langle M\rangle : \neg A \blacktriangleright \Gamma \rightarrow \Theta}\,\neg L \qquad \frac{K : A \blacktriangleright \Gamma \rightarrow \Theta}{\Gamma \rightarrow \Theta \blacktriangleleft [K]\texttt{not} : \neg A}\,\neg R$$

$$\frac{x : A, \Gamma \blacktriangleleft S \blacktriangleright \rightarrow \Theta}{x.(S) : A \blacktriangleright \Gamma \rightarrow \Theta}\,LI \qquad \frac{\Gamma \blacktriangleleft S \blacktriangleright \rightarrow \Theta, \alpha : A}{\Gamma \rightarrow \Theta \blacktriangleleft (S).\alpha : A}\,RI$$

$$\frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A \quad K : A \blacktriangleright \Gamma \rightarrow \Theta}{\Gamma \blacktriangleleft M \bullet K \blacktriangleright \rightarrow \Theta}\,\text{Cut}$$

plus the second-order rules:

$$\frac{K : A\{B/X\} \blacktriangleright \Gamma \rightarrow \Theta}{K : \forall X.A \blacktriangleright \Gamma \rightarrow \Theta}\,\forall L \qquad \frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A}{\Gamma \rightarrow \Theta \blacktriangleleft M : \forall X.A}\,\forall R$$

$$\frac{K : A \blacktriangleright \Gamma \rightarrow \Theta}{K : \exists X.A \blacktriangleright \Gamma \rightarrow \Theta}\,\exists L \qquad \frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A\{B/X\}}{\Gamma \rightarrow \Theta \blacktriangleleft M : \exists X.A}\,\exists R$$

where in $\forall R$ and $\exists L$ there are no free occurrences of X in Γ, Θ. The resulting sequent calculus is called `GW2`. $\dashv$

It is useful to make a note on notation concerning variables, since now we will be using variables both in the level of term-coterm syntax and in type syntax. The set of term variables is Var and the set of coterm covariables is $cVar$. If G is some element of `DuCa2`, then its free variables form the set $FV(G)$, while its free covariables the set $FcV(G)$. On the other hand, the set of type variables is denoted by $TVar$, and, if A is some type, then its free type variables form the set $FTV(A)$.
As expected, the extension to `DuCa2` is non-trivial, in the sense that there are typed elements of `DuCa2` which are not typed in `DuCa`. Now, a common example in the lambda

calculus bibliography of a lambda-term not typed in simply-typed lambda calculus but typed in $\lambda 2$ is xx, for any variable x. By translating this term to the dual calculus (as in section 1.3) we get a term typed in `DuCa2` but not typed in `DuCa`.

Example 3.2 *The term* $((x \bullet \alpha).\alpha \bullet \mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle).\gamma$ *is typed in* `DuCa2`*, yet it is not typed in* `DuCa`*.*

Proof: In `DuCa2`, for any A, we have the following derivation.

$$\dfrac{\dfrac{\dfrac{}{\mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle : \neg(\forall X.X \& \neg A) \mid x : \forall X.X \;\to\; \gamma : A}}{\mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle : \forall X.X \mid x : \forall X.X \;\to\; \gamma : A}\,\forall L \qquad \dfrac{}{x : \forall X.X \;\to\; \gamma : A \mid (x \bullet \alpha).\alpha : \forall X.X}}{\dfrac{x : \forall X.X \mid (x \bullet \alpha).\alpha \bullet \mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle \mid\!\to \gamma : A}{x : \forall X.X \;\to\; \mid ((x \bullet \alpha).\alpha \bullet \mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle).\gamma : A}}$$

Where double lines stand for some obvious derivation steps. From a proof-theoretic point of view, the above derivation is not an unexpected one, since $\forall X.X$ represents falsity and proves anything.
Now, a derivation of a sequent which types the same term in `DuCa` must have the form:

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \to \Theta, \beta : B, \gamma : A \mid x : B}{\Gamma \to \Theta, \gamma : A \mid (x \bullet \beta).\beta : B} \qquad \dfrac{\gamma : A \mid \Gamma \to \Theta, \gamma : A}{\Gamma \to \Theta, \gamma : A \mid [\gamma]\mathtt{not} : \neg A}}{\Gamma \to \Theta, \gamma : A \mid \langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle : B \& \neg A}}{\mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle : \neg(B \& \neg A) \mid \Gamma \to \Theta, \gamma : A} \qquad \dfrac{\Gamma \to \Theta, \gamma : A, \alpha : \neg(B \& \neg A) \mid x : \neg(B \& \neg A)}{\Gamma \to \Theta, \gamma : A \mid (x \bullet \alpha).\alpha : \neg(B \& \neg A)}}{\dfrac{\Gamma \mid (x \bullet \alpha).\alpha \bullet \mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle \mid\!\to \Theta, \gamma : A}{\Gamma \to \Theta \mid ((x \bullet \alpha).\alpha \bullet \mathtt{not}\langle\langle (x \bullet \beta).\beta, [\gamma]\mathtt{not}\rangle\rangle).\gamma : A}}$$

From the extreme left and right leaves it follows that $B \equiv \neg(B \& \neg A)$, $\lightning$. ■

Though this generalization with second-order quantifiers seems to be quite natural, once we add a reduction relation to `DuCa2`, namely the basic reduction relation R_b extended to `DuCa2`, we find out that the calculus is not well-behaved. In fact, the Subject Reduction property, which was straightforward in the propositional case, does not hold for reductions involving the βL and βR reduction rules. This behavior is shown in the following example. Recall that under R_b we have that $M \bullet x.(S) \longrightarrow S\{M/x\}$.

Example 3.3 $x : \forall X.X \mid x \bullet y.(\langle y, y\rangle \bullet \alpha) \mid\!\to \alpha : X \& X$ *is derivable, and* $x : \forall X.X \mid \langle x, x\rangle \bullet \alpha \mid\!\to \alpha : X \& X$ *is not derivable. However,* $x \bullet y.(\langle y, y\rangle \bullet \alpha) \longrightarrow \langle x, x\rangle \bullet \alpha$*.*

Proof: Deriving the former sequent is routine:

$$
\dfrac{\dfrac{\dfrac{\dfrac{y : X, x : \forall X.X \rightarrow \alpha : X\&X \mid y : X}{y : X, x : \forall X.X \rightarrow \alpha : X\&X \mid \langle y, y\rangle : X\&X} \qquad \alpha : X\&X \mid y : X, x : \forall X.X \rightarrow \alpha : X\&X}{y : X, x : \forall X.X \mid \langle y, y\rangle \bullet \alpha \mid \rightarrow \alpha : X\&X}}{y.(\langle y, y\rangle \bullet \alpha) : X \mid x : \forall X.X \rightarrow \alpha : X\&X}\,\forall L}{y.(\langle y, y\rangle \bullet \alpha) : \forall X.X \mid x : \forall X.X \rightarrow \alpha : X\&X} \qquad x : \forall X.X \rightarrow \alpha : X\&X \mid x : \forall X.X}{x : \forall X.X \mid x \bullet y.(\langle y, y\rangle \bullet \alpha) \mid \rightarrow \alpha : X\&X}
$$

Now, regarding the other sequent not being derivable, in order for $x : \forall X.X \mid \langle x, x\rangle \bullet \alpha \mid \rightarrow \alpha : X\&X$ to be derived, we have to derive first $\alpha : B \mid x : \forall X.X \rightarrow \alpha : X\&X$ and $x : \forall X.X \rightarrow \alpha : X\&X \mid \langle x, x\rangle : B$, for some type B. Now, the derivation of the former must have the form:

$$
\dfrac{\alpha : X\&X \mid x : \forall X.X \rightarrow \alpha : X\&X}{\alpha : B \mid x : \forall X.X \rightarrow \alpha : X\&X}\ (1)
$$

where a sequence of $\forall L$ and $\exists L$ rules is applied in the double lines. On the other hand, the derivation of the other sequent must be:

$$
\dfrac{\dfrac{\dfrac{x : \forall X.X \rightarrow \alpha : X\&X \mid x : \forall X.X}{x : \forall X.X \rightarrow \alpha : X\&X \mid x : B_1} \qquad \dfrac{x : \forall X.X \rightarrow \alpha : X\&X \mid x : \forall X.X}{x : \forall X.X \rightarrow \alpha : X\&X \mid x : B_2}}{x : \forall X.X \rightarrow \alpha : X\&X \mid \langle x, x\rangle : B_1\&B_2}}{x : \forall X.X \rightarrow \alpha : X\&X \mid \langle x, x\rangle : B}\ (2)
$$

Where in the double lines we apply sequences of $\forall R$ and $\exists R$ rules. We assume that in (1) are included n derivation steps and that in (2) we have m steps. Now suppose that there exist such derivations and, more than that, take those derivations with least $m+n$. Further, assume that the derivation steps in (1) type α with $X\&X \equiv C_0, C_1, \ldots, C_n \equiv B$, and those in (2) type $\langle x, x\rangle$ with $B_1\&B_2 \equiv D_0, D_1, \ldots, D_m \equiv B$.
It is not difficult to see that it cannot happen $m = 0$ or $n = 0$, since $B_1\&B_2$ necessarily contains quantifiers inside B_1 and B_2. So, suppose that $mn > 0$ and that the last rule used in (1) is $\exists L$, which implies that the last rule in (2) is $\exists R$. Then $C_n \equiv \exists Z.C_{n-1}$, some Z, and thus $D_m \equiv \exists Z.C_{n-1}$, whence $D_{m-1} \equiv C_{n-1}\{C'/Z\}$.
Now note that the only free variable we may have in the C_i's is X, and $Z \not\equiv X$, since Z must be free in $\alpha : X\&X$ for $\exists L$ to be applied. Therefore, Z is not free in C_{n-1} and thus $D_{m-1} \equiv C_{n-1}$, $\lightning$ since we took the derivations with least $m+n$.
The case of the last rule in (1) being $\forall L$ and in (2) $\forall R$ is dealt with similarly, using the fact that the D_j's contain no free variables. ■

Note that the example above applies also in the cases of R_v and R_n.
Now, βL and βR rules are crucial in the dual calculus, since they are the only rules involving term (and coterm) substitution. One first thought in order to solve the problem with

subject reduction could be adding axioms of the form $x : \forall X.A, \Gamma \rightarrow \Theta \mid x : A\{B/X\}$ to our sequent calculus. Then, though the example above would be solved, we would still be able to yield a similar problematic example, that is of $x : \forall W.\forall X.X \mid x \bullet y.(\langle y, y\rangle \bullet \alpha) \mid\!\rightarrow \alpha : X\&X$ being derivable and $x : \forall W.\forall X.X \mid \langle x, x\rangle \bullet \alpha \mid\!\rightarrow \alpha : X\&X$ being not.
A second approach is that of adding in our calculus typing rules of the form:

$$\frac{x : A\{B/X\}, \Gamma \rightarrow \Theta \mid M : C}{x : \forall X.A, \Gamma \rightarrow \Theta \mid M : C}\star$$

Such an approach will be followed subsequently.

Note 3.4. In the sequel of this section we will focus on βR and βL reduction rules to study the subject reduction property. Note that subject reduction does hold for the rest of the rules of R_b. However, we avoid to give a proof of this latter result, since it is not difficult but rather technical.

3.1.1 Adding Strong Contraction rules

In this section we solve some problematic cases of subject reduction by adding typing rules of $\star$ form. In fact, we add rules that are a combination of contraction and $\star$-rules.

Definition 3.5
The calculus `DuCa2`$^+$ is an extension of `DuCa2` by the typing rules:

$$\frac{K : A \mid \Gamma \rightarrow \Theta, \alpha : B\{C/X\}, \beta : \exists X.B}{K\{\beta/\alpha\} : A \mid \Gamma \rightarrow \Theta, \beta : \exists X.B}\,SCL \qquad \frac{\Gamma \mid S \mid\!\rightarrow \Theta, \alpha : B\{C/X\}, \beta : \exists X.B}{\Gamma \mid S\{\beta/\alpha\} \mid\!\rightarrow \Theta, \beta : \exists X.B}\,SCC_{\exists}$$

$$\frac{x : B\{C/X\}, y : \forall X.B, \Gamma \rightarrow \Theta \mid M : A}{y : \forall X.B, \Gamma \rightarrow \Theta \mid M\{y/x\} : A}\,SCR \qquad \frac{x : B\{C/X\}, y : \forall X.B, \Gamma \mid S \mid\!\rightarrow \Theta}{y : \forall X.B, \Gamma \mid S\{y/x\} \mid\!\rightarrow \Theta}\,SCC_{\forall}$$

The resulting sequent calculus is called `GW2`$^+$. $\dashv$

Note that these rules are not radical to the calculus, since they can be seen as special contraction rules; in fact, we call them Strong Contraction rules. These rules are proof-theoretically valid and, for example, an equivalent of SCR in `DuCa2` (actually, in `GW2`) would be some derivation such as:

$$\begin{array}{c}
x : B\{C/X\}, y : \forall X.B, \Gamma \rightarrow \Theta, \alpha : A \mid M : A \\
\hline
x : B\{C/X\}, y : \forall X.B, \Gamma \mid M \bullet \alpha \mid\!\rightarrow \Theta, \alpha : A \\
\hline
x.(M \bullet \alpha) : B\{C/X\} \mid y : \forall X.B, \Gamma \rightarrow \Theta, \alpha : A \\
\hline
x.(M \bullet \alpha) : \forall X.B \mid y : \forall X.B, \Gamma \rightarrow \Theta, \alpha : A \\
\hline
y : \forall X.B, \Gamma \rightarrow \Theta \mid (y \bullet x.(M \bullet \alpha)).\alpha : A
\end{array}$$

The above proof yields a sequent which is logically equivalent to the one we were opting for, yet clearly not syntactically equivalent. Not surprisingly, the yielded term $(y \bullet x.(M \bullet \alpha)).\alpha$ encodes the variable substitution which occurs in SCR, since it reduces under βL to $(M\{y/x\} \bullet \alpha).\alpha$, an η-expansion of $M\{y/x\}$.
The following proposition states a basic property of the new calculus.[14]

Proposition 3.6 (Type Substitution) *Take some $G \in \texttt{DuCa2}^+$. If there exists some sequent $\sigma \in T_G(A, \Gamma, \Theta)$, derived say by $\mathcal{D}$ (in $\texttt{GW2}^+$), then, for any type B and type variable X, there exists some sequent $\sigma' \in T_G(A\{B/X\}, \Gamma\{B/X\}, \Theta\{B/X\})$ derived by $\mathcal{D}'$, such that $\mathcal{D}$ and $\mathcal{D}'$ have the same tree structure.*

Proof: The proof is done by induction on derivations and is very similar to this of proposition 3.14. ■

The results of adding SC-rules in our calculus are in essence summarized in the following proposition.

Proposition 3.7 *Let $G \in \texttt{DuCa2}^+$ and B be any type:*

1. $T_G(C, \Gamma, \Theta \cup \{\alpha : \forall X.A\}) \neq \emptyset$ *implies* $T_G(C, \Gamma, \Theta \cup \{\alpha : A\{B/X\}\}) \neq \emptyset$.

2. $T_G(C, \Gamma \cup \{x : \exists X.A\}, \Theta) \neq \emptyset$ *implies* $T_G(C, \Gamma \cup \{\alpha : A\{B/X\}\}, \Theta) \neq \emptyset$.

3. *If* $\Gamma \rightarrow \Theta \,\rule{0.3em}{0.6em}\, M : \forall X.A$ *is derivable, then so is* $\Gamma \rightarrow \Theta \,\rule{0.3em}{0.6em}\, M : A\{B/X\}$.

4. *If* $K : \exists X.A \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta$ *is derivable, then so is* $K : A\{B/X\} \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta$.

Proof: We show 1 and 3; 2 and 4 are shown similarly.
For 1 assume that $\sigma \in T_G(C, \Gamma, \Theta \cup \{\alpha : \forall X.A\})$; we do induction on its derivation. For the non-trivial base case we replace the axiom $\alpha : \forall X.A \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta, \alpha : \forall X.A$ with

$$\frac{\alpha : A\{B/X\} \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta, \alpha : A\{B/X\}}{\alpha : \forall X.A \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta, \alpha : A\{B/X\}}\,\forall L$$

For the inductive step we do a case analysis on the last rule in the derivation. The only non-straightforward cases are those where we want a certain type variable to be free in the premise of the rule, that is when the last rule is $\forall R$ or $\exists L$. We deal only with the latter case, the other being one resolved similarly. Assume that the last rule in the derivation is:

$$\frac{K : C_0 \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta, \alpha : \forall X.A}{K : \exists Y.C_0 \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta, \alpha : \forall X.A}\,\exists L$$

with $Y \notin FTV(\Gamma, \Theta, \forall X.A)$. In case $Y \notin FTV(B)$ or $X \notin FTV(A)$, we prove the claim directly by using the IH, since then $Y \notin FTV(A\{B/X\})$. Otherwise, by proposition 3.6, we can derive $K : C_0\{Z/Y\} \,\rule{0.3em}{0.6em}\, \Gamma \rightarrow \Theta, \alpha : \forall X.A$, for some fresh Z, and, since we

[14] Recall the notation we use for sequents typing elements of the calculus (definition 1.3), and also the definition of derivations (definition 1.2) in chapter 1.

don't have an increase on the size of the derivation, we can use the IH and thus derive $K : C_0\{Z/Y\} \mid \Gamma \rightarrow \Theta, \alpha : A\{B/X\}$. From the latter, the desired sequent is derived by $\exists L$ (note that $\exists Z.C_0\{Z/Y\} \equiv \exists Y.C_0$).
For 3 assume that $\Gamma \rightarrow \Theta \mid M : \forall X.A$ is derivable; we do induction on its derivation. For the base case we utilize the SCR rule: we replace the axiom $x : \forall X.A, \Gamma' \rightarrow \Theta \mid x : \forall X.A$ with

$$\frac{x : \forall X.A, y : A\{B/X\}, \Gamma' \rightarrow \Theta \mid y : A\{B/X\}}{x : \forall X.A, \Gamma' \rightarrow \Theta \mid x : A\{B/X\}}\, SCR$$

For the induction step we do a case analysis on the last rule in the derivation. Since M is typed with $\forall X.A$, there are only three cases:

$$\frac{\Gamma \rightarrow \Theta \mid M : A}{\Gamma \rightarrow \Theta \mid M : \forall X.A}\,\forall R \quad \text{or} \quad \frac{\Gamma \mid S \mid\rightarrow \Theta, \alpha : \forall X.A}{\Gamma \rightarrow \Theta \mid (S).\alpha : \forall X.A}\, RI$$

$$\text{or} \quad \frac{x : C\{D/Y\}, y : \forall Y.C, \Gamma' \rightarrow \Theta \mid M' : \forall X.A}{y : \forall Y.C, \Gamma' \rightarrow \Theta \mid M'\{y/x\} : \forall X.A}\, SCR$$

The first case is clear by proposition 3.6, the second by previous claim (1), and the third by IH. ■

In the next proposition we use vector notation $\vec{x} : \vec{B}$ for $x_1 : B_1, \ldots, x_n : B_n$, and $\vec{M}/\vec{x}$ for $M_1/x_1, \ldots, M_n/x_n$. Similar conventions apply for $\vec{\alpha}$ and $\vec{K}$.

Proposition 3.8 *Suppose that $G, M_1, \ldots, M_n, K_1, \ldots, K_m \in \mathtt{DuCa2}^+$.*

1. *If $T_G(A, \Gamma \cup \{\vec{x} : \vec{B}\}, \Theta) \neq \emptyset$, and $\Gamma \rightarrow \Theta \mid M_i : B_i$ is derivable for any $i = 1, ..., n$, then $T_{G\{\vec{M}/\vec{x}\}}(A, \Gamma, \Theta) \neq \emptyset$.*

2. *If $T_G(A, \Gamma, \Theta \cup \{\vec{\alpha} : \vec{B}\}) \neq \emptyset$, and $K_j : B_j \mid \Gamma \rightarrow \Theta$ is derivable for any $j = 1, ..., m$, then $T_{G\{\vec{K}/\vec{\alpha}\}}(A, \Gamma, \Theta) \neq \emptyset$.*

Proof: We prove only 1; 2 is proven similarly.
Assume that $\sigma \in T_G(A, \Gamma \cup (\vec{x} : \vec{B}), \Theta)$; we do induction on its derivation. The base cases are trivial and for the inductive step we do a case analysis on the last rule in the derivation. In fact, all cases are straightforward, except for the last rule being some strong contraction rule. But for these cases we utilize proposition 3.7 and resolve them. ■

The above propositions allow us to prove subject reduction under βL and βR rules in case we don't use $\exists L$ and $\forall R$ rules in certain crucial steps of a derivation.

Lemma 3.9 *Suppose that $\Gamma \rightarrow \Theta \mid M : A$ and $K : A \mid \Gamma \rightarrow \Theta$ are derivable; then,*

1. *If $x.(S) : A \mid \Gamma \rightarrow \Theta$ is derived via $x : A_0, \Gamma \mid S \mid\rightarrow \Theta$ without using $\exists L$ in between, then $\Gamma \mid S\{M/x\} \mid\rightarrow \Theta$ is derivable.*

2. *If* $\Gamma \rightarrow \Theta \mid (S).\alpha : A$ *is derived via* $\Gamma \mid S \mid\rightarrow \Theta, \alpha : A_0$ *without using* $\forall R$ *in between, then* $\Gamma \mid S\{K/\alpha\} \mid\rightarrow \Theta$ *is derivable.*

Proof: We show 1; 2 is shown dually.
By hypothesis, the derivation of $x.(S) : A \mid \Gamma \rightarrow \Theta$ ends with:

$$\cfrac{\cfrac{x : A_0, \Gamma \mid S \mid\rightarrow \Theta}{x.(S) : A_0 \mid \Gamma \rightarrow \Theta}}{x.(S) : A \mid \Gamma \rightarrow \Theta}\ (1)$$

where in (1) we have condensed $n \geq 0$ applications of the $\forall L$ rule. We do induction on n. The case of $n = 0$ implies $A \equiv A_0$, so proposition 3.8 proves the claim.
In case $n > 0$, we have that $A \equiv \forall X.A'$, where $x.(S) : A'\{B/X\} \mid \Gamma \rightarrow \Theta$ is the (n-1)-th sequent in (1). Now, $\Gamma \rightarrow \Theta \mid M : A$ is derivable, so $\Gamma \rightarrow \Theta \mid M : A'\{B/X\}$ is also derivable by proposition 3.7. Hence, the claim follows from the IH. ■

Corollary 3.10 *Suppose that* $G, H \in$ `DuCa2`$^+$, $\sigma \in T_G(A_0, \Gamma_0, \Theta_0)$ *for some* $\sigma, A_0, \Gamma_0, \Theta_0$, *and that* σ *is derived by* $\mathcal{D}$. *Then,*

1. *If* $G \longrightarrow H$ *by rule* βR *being applied to some subobject* $M \bullet x.(S)$ *of* G, *and* $\mathcal{D}$ *respects the restrictions posed for* $x.(S)$ *in the previous lemma, then* $T_H(A_0, \Gamma_0, \Theta_0) \neq \emptyset$.

2. *If* $G \longrightarrow H$ *by rule* βL *being applied to some subobject* $(S).\alpha \bullet K$ *of* G, *and* $\mathcal{D}$ *respects the restrictions posed for* $(S).\alpha$ *in the previous lemma, then* $T_H(A_0, \Gamma_0, \Theta_0) \neq \emptyset$.

Proof: Straightforward from previous lemma. ■

3.1.2 Some cases are still problematic

In lemma 3.9 we have placed an important restriction in deriving the objects $x.(S)$ and $(S).\alpha$. The restriction was that, once we have constructed them, we are not allowed to change their types by applying $\exists L$ or $\forall R$ rules respectively. Thus, we are still left with problematic cases of the forms:

$$\cfrac{\Gamma \rightarrow \Theta \mid M : \exists X.A \qquad \cfrac{\cfrac{x : A, \Gamma \mid S \mid\rightarrow \Theta}{x.(S) : A \mid \Gamma \rightarrow \Theta}}{x.(S) : \exists X.A \mid \Gamma \rightarrow \Theta}}{\Gamma \mid M \bullet x.(S) \mid\rightarrow \Theta}$$

$$\cfrac{\cfrac{\cfrac{\Gamma \mid S \mid\rightarrow \Theta, \alpha : A}{\Gamma \rightarrow \Theta \mid (S).\alpha : A}}{\Gamma \rightarrow \Theta \mid (S).\alpha : \forall X.A} \qquad K : \forall X.A \mid \Gamma \rightarrow \Theta}{\Gamma \mid (S).\alpha \bullet K \mid\rightarrow \Theta}$$

where X is not free in Γ, Θ. Let's focus, for example, in the latter case, and name that derivation $\mathcal{D}$. The complication which arises if we try to derive $K : A \mid \Gamma \rightarrow \Theta$, so that we can substitute its derivation for all axioms introducing α in the derivation of $\Gamma \mid S \mid\rightarrow \Theta, \alpha : A$, is that not only can we not derive it given $K : \forall X.A \mid \Gamma \rightarrow \Theta$, but it is not even logically valid to do so! Thus, our only chance lies on deriving $\Gamma \mid S \mid\rightarrow \Theta, \alpha : \forall X.A$ instead. Note that adding some explicit rule such as:

$$\frac{\Gamma \mid S \mid\rightarrow \Theta, \alpha : A}{\Gamma \mid S \mid\rightarrow \Theta, \alpha : \forall X.A}\,(X \notin FTV(\Gamma, \Theta))$$

would not save us now, since we wouldn't then be able to prove proposition 3.8. In fact, if we used such a rule, we would still have to substitute the derivation typing K for the axioms introducing α in the derivation of $\Gamma \mid S \mid\rightarrow \Theta, \alpha : A$. Therefore, we have to prove that the above rule *is a derived one* in $\mathtt{DuCa2}^{+}$.

The difficulties that arise when trying to show the latter can be viewed in the following example. We use a derivation for the law of the excluded middle ($\forall X.X \vee \neg X$). For any covariable α, let $\phi(\alpha)$ stand for $(\langle[x.(\langle x\rangle\mathtt{inl} \bullet \alpha)]\mathtt{not}\rangle\mathtt{inr} \bullet \alpha)$. For economy in space we use weakening rules; we have the following derivation being an occurrence of $\mathcal{D}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x : X \rightarrow\ \mid x : X}{x : X \rightarrow\ \mid \langle x\rangle\mathtt{inl} : X \vee \neg X}}{x.(\langle x\rangle\mathtt{inl} \bullet \alpha) : X \mid\ \rightarrow \alpha : X \vee \neg X}}{\rightarrow \alpha : X \vee \neg X \mid [x.(\langle x\rangle\mathtt{inl} \bullet \alpha)]\mathtt{not} : \neg X}}{\rightarrow \alpha : X \vee \neg X \mid \langle[x.(\langle x\rangle\mathtt{inl} \bullet \alpha)]\mathtt{not}\rangle\mathtt{inr} : X \vee \neg X}}{\mid \phi(\alpha) \mid\rightarrow \alpha : X \vee \neg X}}{\rightarrow\ \mid (\phi(\alpha)).\alpha : X \vee \neg X}}{\rightarrow\ \mid (\phi(\alpha)).\alpha : \forall X.X \vee \neg X} \quad x.(x \bullet \beta) : \forall X.X \vee \neg X \mid\ \rightarrow \beta : \forall X.X \vee \neg X}{\mid (\phi(\alpha)).\alpha \bullet x.(x \bullet \beta) \mid\rightarrow \beta : \forall X.X \vee \neg X}$$

It is clear that, in order to derive $\mid \phi(\alpha) \mid\rightarrow \alpha : \forall X.X \vee \neg X$, we must introduce α with type $\forall X.X \vee \neg X$ from the beginning (i.e. from the axiom introducing α). But then, we must be able to derive cuts of the form:

$$\frac{x : X \rightarrow\ \mid \langle x\rangle\mathtt{inl} : X \vee \neg X \quad \alpha : \forall X.X \vee \neg X \mid\ \rightarrow \alpha : \forall X.X \vee \neg X}{x : X \mid \langle x\rangle\mathtt{inl} \bullet \alpha \mid\rightarrow \alpha : \forall X.X \vee \neg X}$$

However, it is not clear what extra typing rules would allow us to derive such cuts, without allowing for the formulation of logically invalid derivations[15].

Summing up, though it is embarrassing to admit, we are not able to show subject reduction for some 'natural' extension of $\mathtt{DuCa}$ with second-order quantifiers. Nevertheless, in the next section we propose a different approach of a 'constructive' extension, which is

[15]note that when we talk about logical validity we have, of course, in mind the Curry-Howard isomorphism.

based on the connection of `DuCa` with classical logic and indeed has the subject reduction property.

3.2 A constructive approach

In second-order propositional classical logic the quantification over propositional variables is in fact a quantification over true and false propositions. That is, if $\perp$ is some contradiction (for example $\perp \equiv X_0 \;\&\; \neg X_0$), then, for any formula A, $\forall X.A$ is logically equivalent to $A\{\perp/X\} \;\&\; A\{\neg\perp/X\}$. A similar property holds for existential quantification. Note that in the intuitionistic second-order propositional logic this is not the case, since there is no converse of falsity; so, quantification is in a sense stronger in that logic.
Below we are going to define a quantification construction over types by using the above remark explicitly; that is universal quantification will be the abbreviation of a conjunction and existential quantification the abbreviation of a disjunction. Moreover, universal types will be assigned to product constructs in the calculus, while existential types to sum constructs. For this purpose, some new construction rules for terms and coterms will be defined, so as to capture quantification in the cases where the existing construction rules are not enough. All this, of course, has as a final objective to obtain a calculus where subject reduction holds under R_b. Note that in this section we will deal with the usual R_b, including all reduction rules. The resulting calculus differs a lot from the intended `DuCa2` and, in fact, the typed terms of the new calculus restricted to the old construction rules are exactly the typed terms of `DuCa`. Therefore, it is clear that in this section the algorithmic interpretation of the calculus is not our primary interest. Soon the reasons for calling this approach a constructive one will become clear.
We begin with some definitions.

Definition 3.11
Define the set of Types:

$$\text{Type} \quad A, B \;::=\; \digamma \mid X \mid A\&B \mid A \vee B \mid \neg A$$

$\digamma$ is a special type variable. Therefore, the set $TVar$ of common propositional variables is the set of all type variables except for $\digamma$. For any type A, the set $FTV(A)$ is formed from all members of $TVar$ which are free in A.
We define the following abbreviations. For any type A and $X \in TVar$:

$$\forall X.A \equiv A\{\top/X\} \;\&\; A\{\perp/X\} \quad , \quad \exists X.A \equiv A\{\top/X\} \vee A\{\perp/X\}$$
$$\text{and} \quad \top \equiv \digamma \vee \neg\digamma \quad , \quad \perp \equiv \digamma\&\neg\digamma \qquad \dashv$$

The distinction we have made between common type variables and $\digamma$ is because of the facts that we don't want to allow quantification over $\digamma$ and that we want to keep the usual free-variables-over-quantifiers definition. In the sequel, X, Y, Z will denote members of $TVar$, unless otherwise specified. Note that under the above abbreviations alpha-equivalence is valid, that is, for all types A and $X, Y \in TVar$ with $Y \notin FTV(A)$, $\forall X.A \equiv \forall Y.A\{Y/X\}$ and $\exists X.A \equiv \exists Y.A\{Y/X\}$.
Now we can define the calculus which will be using these types, namely `DuCa2`$^{\mathtt{C}}$.

Definition 3.12
The $\mathtt{DuCa2^C}$ consists of Types as in definition 3.11 and Objects. The set of objects is the union of the sets of Terms, Coterms and Statements:

Object	G, H	$::=$	$M \mid K \mid S$
Term	M, N	$::=$	$x \mid \langle M, N\rangle \mid \langle M\rangle\mathtt{in} \mid \langle M\rangle\mathtt{inl} \mid \langle N\rangle\mathtt{inr} \mid [K]\mathtt{not} \mid (S).\alpha$
Coterm	K, L	$::=$	$\alpha \mid [K, L] \mid \mathtt{one}[K] \mid \mathtt{fst}[K] \mid \mathtt{snd}[L] \mid \mathtt{not}\langle M\rangle \mid x.(S)$
Statement	S, T	$::=$	$M \bullet K$

The typing rules are the same as those of $\mathtt{DuCa}$ (i.e. of system $\mathtt{GW}$) with the addition of the rules:

$$\frac{K : A\{B/X\} \blacktriangleright \Gamma \rightarrow \Theta}{\mathtt{one}[K] : \forall X.A \blacktriangleright \Gamma \rightarrow \Theta}\,\forall L \qquad \frac{\Gamma \rightarrow \Theta \blacktriangleleft M : A\{B/X\}}{\Gamma \rightarrow \Theta \blacktriangleleft \langle M\rangle\mathtt{in} : \exists X.A}\,\exists R$$

The resulting sequent calculus is called $\mathtt{GW2^C}$. $\dashv$

The intuition behind these new construction rules is that $\langle M\rangle\mathtt{in}$ produces a sum value built up from M being **either its first or its second element**, whereas $\mathtt{one}[K]$ absorbs a product value and offers **one of its elements** to K. Therefore, these new rules introduce terms with **uncertainty** regarding their inner structure. This is exactly the proof-theoretic uncertainty contained in the $\forall L$ and $\exists R$ rules, since when we move, for example, from $A\{B/X\}$ in $\exists R$ to $\exists X.A$, we don't know if B represents a true or false proposition (and we don't care).
It may be observed that up to now the constructivism involved in our definitions is rather vague, in the sense that the constructions we introduce contain uncertainty. However, for the $\forall R$ and $\exists L$ rules we follow a purely constructive approach, which justifies the title of this section. Suppose we can derive $\Gamma \rightarrow \Theta \blacktriangleleft M : A$ and $X \notin FTV(\Gamma, \Theta)$. Then it is not difficult to see that we can also derive $\Gamma \rightarrow \Theta \blacktriangleleft M : A\{\top/X\}$ and $\Gamma \rightarrow \Theta \blacktriangleleft M : A\{\bot/X\}$ and thus derive $\Gamma \rightarrow \Theta \blacktriangleleft \langle M, M\rangle : A\{\top/X\}\&A\{\bot/X\}$. Therefore, a proof of $\forall X.A$ is a construction merging together a proof of $A\{\top/X\}$ and one of $A\{\bot/X\}$. In fact, this is the analog of the Brouwer-Heyting-Kolmogorov interpretation of the proof of $\forall X.A$ in intuitionistic logic. Similar remarks can be made for the $\exists L$ rule.
Let us now formulate the above remark formally. First, we need the following results.

Proposition 3.13 (Weakening) *Suppose that $T_G(A, \Gamma, \Theta) \neq \emptyset$, some $G \in \mathtt{DuCa2^C}$; then,*
$T_G(A, \Gamma \cup (x : B), \Theta) \neq \emptyset$, any fresh variable x, and
$T_G(A, \Gamma, \Theta \cup (\alpha : B)) \neq \emptyset$, any fresh covariable α.

Proof: By induction on the derivation of $\sigma \in T_G(A, \Gamma, \Theta)$. ∎

Proposition 3.14 (Type Substitution) *Take some $G \in \mathtt{DuCa2^C}$. If there exists some sequent $\sigma \in T_G(A, \Gamma, \Theta)$, derived say by $\mathcal{D}$, then for any type B there exists some sequent $\sigma' \in T_G(A\{B/X\}, \Gamma\{B/X\}, \Theta\{B/X\})$ derived by $\mathcal{D}'$, such that $\mathcal{D}$ and $\mathcal{D}'$ have the same tree structure.*

Proof: Let $\sigma \in T_G(A, \Gamma, \Theta)$; we do induction on the derivation of σ. By a case analysis on the last rule of the derivation, the only non-straightforward cases are those of rules with quantifiers.

Assume the last rule is $\dfrac{K : A\{C/Y\} \blacktriangleright \Gamma \;\rightarrow\; \Theta}{\texttt{one}[K] : \forall Y.A \blacktriangleright \Gamma \;\rightarrow\; \Theta}\,\forall L$

By IH, we can derive

$$K : (A\{C/Y\})\{B/X\} \blacktriangleright \Gamma\{B/X\} \;\rightarrow\; \Theta\{B/X\}$$

Now, $(A\{C/Y\})\{B/X\} \equiv ((A\{Z/Y\})\{B/X\})\{C\{B/X\}/Z\}$, some fresh Z, and $\forall Z.(A\{Z/Y\})\{B/X\} \equiv (\forall Z.A\{Z/Y\})\{B/X\} \equiv (\forall Y.A)\{B/X\}$; thus, by $\forall L$ we derive

$$\texttt{one}[K] : (\forall Y.A)\{B/X\} \blacktriangleright \Gamma\{B/X\} \;\rightarrow\; \Theta\{B/X\}$$

as required. The case of $\exists R$ is dealt with similarly. ■

Now we can prove that there exist $\exists L$ and $\forall R$ derived rules.

Proposition 3.15 *The following are derived rules of* $\texttt{DuCa2}^{\texttt{C}}$.

$$\frac{K : A \blacktriangleright \Gamma \;\rightarrow\; \Theta}{[K, K] : \exists X.A \blacktriangleright \Gamma \;\rightarrow\; \Theta}\,\exists L \qquad \frac{\Gamma \;\rightarrow\; \Theta \blacktriangleright M : A}{\Gamma \;\rightarrow\; \Theta \blacktriangleright \langle M, M\rangle : \forall X.A}\,\forall R$$

where $X \in (TVar \setminus FTV(\Gamma, \Theta))$.

Proof: Straightforward by proposition 3.14 and the fact that $X \notin FTV(\Gamma, \Theta)$. ■

Our objective is to prove that the subject reduction property holds for this calculus under R_b. The set of reduction rules, which is the same as in DuCa, is given below and defines the $R_b^{2\texttt{C}}$ reduction relation, that is the basic reduction relation for $\texttt{DuCa2}^{\texttt{C}}$.

Definition 3.16
$R_b^{2\texttt{C}}$ is the compatible one-step reduction relation which is yielded by the reduction rules listed below, when these are applied to subobjects of $\texttt{DuCa2}^{\texttt{C}}$ objects.

$$\begin{array}{llcl}
(\beta\&_1) & \langle M, N\rangle \bullet \texttt{fst}[K] & \rightarrow & M \bullet K \\
(\beta\&_2) & \langle M, N\rangle \bullet \texttt{snd}[L] & \rightarrow & N \bullet L \\
(\beta\vee_1) & \langle M\rangle\texttt{inl} \bullet [K, L] & \rightarrow & M \bullet K \\
(\beta\vee_2) & \langle N\rangle\texttt{inr} \bullet [K, L] & \rightarrow & N \bullet L \\
(\beta\neg) & [K]\texttt{not} \bullet \texttt{not}\langle M\rangle & \rightarrow & M \bullet K \\
(\beta L) & M \bullet x.(S) & \rightarrow & S\{M/x\} \\
(\beta R) & (S).\alpha \bullet K & \rightarrow & S\{K/\alpha\} \\
 & & & \\
(\eta L) & K & \rightarrow & x.(x \bullet K) \\
(\eta R) & M & \rightarrow & (M \bullet \alpha).\alpha
\end{array}$$

$$\begin{array}{llcl}
(\nu\&_1) & \langle M,N\rangle \bullet K & \rightarrow & M \bullet x.(\langle x,N\rangle \bullet K) \\
(\nu\&_2) & \langle M,N\rangle \bullet K & \rightarrow & N \bullet y.(\langle M,y\rangle \bullet K) \\
(\nu\vee_3) & \langle M\rangle\mathtt{inl} \bullet K & \rightarrow & M \bullet x.(\langle x\rangle\mathtt{inl} \bullet K) \\
(\nu\vee_4) & \langle N\rangle\mathtt{inr} \bullet K & \rightarrow & N \bullet y.(\langle y\rangle\mathtt{inr} \bullet K) \\
(\nu\vee_1) & M \bullet [K,L] & \rightarrow & (M \bullet [\alpha,L]).\alpha \bullet K \\
(\nu\vee_2) & M \bullet [K,L] & \rightarrow & (M \bullet [K,\beta]).\beta \bullet L \\
(\nu\&_3) & M \bullet \mathtt{fst}[K] & \rightarrow & (M \bullet \mathtt{fst}[\alpha]).\alpha \bullet K \\
(\nu\&_4) & M \bullet \mathtt{snd}[L] & \rightarrow & (M \bullet \mathtt{snd}[\beta]).\beta \bullet L
\end{array}$$

For $G,H \in \mathtt{DuCa2^C}$, $(G,H) \in R_b^{2\mathtt{C}}$ will be denoted $G \longrightarrow H$, when this isn't confusing. $\dashv$

The following lemma will allow us to prove subject reduction.

Lemma 3.17 *Suppose that $G, M, K \in \mathtt{DuCa2^C}$.*

1. *If $T_G(A, \Gamma \cup (x:B), \Theta) \neq \emptyset$ and $\Gamma \rightarrow \Theta \mid M : B$ is derivable, then $T_{G\{M/x\}}(A,\Gamma,\Theta) \neq \emptyset$.*

2. *If $T_G(A, \Gamma, \Theta \cup (\alpha:B)) \neq \emptyset$ and $K : B \mid \Gamma \rightarrow \Theta$ is derivable, then $T_{G\{K/\alpha\}}(A,\Gamma,\Theta) \neq \emptyset$.*

Proof: In 1 we take some $\sigma \in T_G(A,\Gamma \cup (x:B),\Theta)$ and do induction on the derivation of σ. In 2 we proceed similarly. ■

Subject reduction is now straightforward, as it was in the case of $\mathtt{DuCa}$.

Theorem 3.18 (Subject Reduction) *Suppose that $G,H \in \mathtt{DuCa2^C}$ and that $G \longrightarrow H$. If $T_G(A_0,\Gamma_0,\Theta_0) \neq \emptyset$, some A_0,Γ_0,Θ_0, then $T_H(A_0,\Gamma_0,\Theta_0) \neq \emptyset$.*

Proof: The proof is by a case analysis on the rule used for the reduction and it is rather straightforward. We are going to show some characteristic cases.
Suppose that $G \longrightarrow H$ by use of the $\beta\&_1$ rule. Then, it suffices to show that $T_{\langle M,N\rangle \bullet \mathtt{fst}[K]}(A,\Gamma,\Theta) \neq \emptyset$ implies $T_{M\bullet K}(A,\Gamma,\Theta) \neq \emptyset$. Suppose that $\Gamma \mid \langle M,N\rangle \bullet \mathtt{fst}[K] \mid\rightarrow \Theta$ is derivable. By inspection of the rules, its derivation must end with:

$$\dfrac{\dfrac{K : B \mid \Gamma \rightarrow \Theta}{\mathtt{fst}[K] : B\&C \mid \Gamma \rightarrow \Theta} \qquad \dfrac{\Gamma \rightarrow \Theta \mid M : B \quad \Gamma \rightarrow \Theta \mid N : C}{\Gamma \rightarrow \Theta \mid \langle M,N\rangle : B\&C}}{\Gamma \mid \langle M,N\rangle \bullet \mathtt{fst}[K] \mid\rightarrow \Theta}$$

and the claim straightforwardly follows.
Suppose that $G \longrightarrow H$ by use of the $\nu\&_1$ rule. Then it suffices to show that $T_{\langle M,N\rangle \bullet K}(A,\Gamma,\Theta) \neq \emptyset$ implies $T_{M\bullet x.(\langle x,N\rangle\bullet K)}(A,\Gamma,\Theta) \neq \emptyset$. Let $\Gamma \mid \langle M,N\rangle \bullet K \mid\rightarrow \Theta$ be derivable. By inspection of the typing rules, its derivation must end with

$$\dfrac{K : B_1\&B_2 \mid \Gamma \rightarrow \Theta \qquad \dfrac{\Gamma \rightarrow \Theta \mid M : B_1 \quad \Gamma \rightarrow \Theta \mid N : B_2}{\Gamma \rightarrow \Theta \mid \langle M,N\rangle : B_1\&B_2}}{\Gamma \mid \langle M,N\rangle \bullet K \mid\rightarrow \Theta}$$

and our claim follows by a simple derivation which uses also the axiom $x : B_1, \Gamma \rightarrow \Theta \mid x : B_1$ and Weakening.
Suppose that $G \longrightarrow Q$ by use of the βR rule. Then it suffices to show that $T_{(S).\alpha \bullet K}(A, \Gamma, \Theta) \neq \emptyset$ implies $T_{S\{K/\alpha\}}(A, \Gamma, \Theta) \neq \emptyset$. Let $\Gamma \mid (S).\alpha \bullet K \mid\rightarrow \Theta$ be derivable. By inspection of the typing rules, its derivation must end with

$$\dfrac{K : B \mid \Gamma \rightarrow \Theta \qquad \dfrac{\Gamma \mid S \mid\rightarrow \Theta, \alpha : B}{\Gamma \rightarrow \Theta \mid (S).\alpha : B}}{\Gamma \mid (S).\alpha \bullet K \mid\rightarrow \Theta}$$

and our claim follows from lemma 3.17. ■

Thus, we have defined an extension of the dual calculus that corresponds to second-order propositional classical logic in Curry-Howard isomorphism, and is well-behaved in that it satisfies subject reduction (under R_b, and therefore under R_v and R_n too). The expressive power of this calculus is limited, since in essence there are no polymorphic types. Indeed, to capture polymorphism we have to allow transition from $M : A$ to $M : \forall X.A$, and not to some new construct containing M. In fact, the former behavior is the cornerstone of system **F**. However, we think that the calculus studied in this section has a value of its own, since it extends even further the Curry-Howard isomorphism in a well-behaved way.
Note further that the syntactic properties studied in the previous chapter for `DuCa` can be readily proven for $\mathtt{DuCa2}^{\mathtt{C}}$. This is because there is no essential difference in syntax between the two calculi and, more than that, the proof of `SN` doesn't seem to take into consideration the details in structure of types, but merely the derivations in which these types are involved.

3.2.1 Two additional reduction rules

Since we have added new construction rules for our calculus, it is very tempting to add some reduction rules that refer especially to the new constructs:

$$\begin{array}{lllll} (\beta\&_c) & \langle M, M\rangle \bullet \mathtt{one}[K] & \rightarrow & M \bullet K \\ (\beta\vee_c) & \langle M\rangle\mathtt{in} \bullet [K, K] & \rightarrow & M \bullet K \end{array}$$

These new rules have the task to remove uncertainty when it is not important, and they seem quite natural. Nevertheless, the problem of subject reduction not holding arises again, as it is shown in the following example.

Example 3.19 $x : \bot \mid \langle\langle x\rangle\mathtt{in}, \langle x\rangle\mathtt{in}\rangle \bullet \mathtt{one}[\alpha] \mid\rightarrow \alpha : B \vee \bot$ *is derivable in* $\mathtt{DuCa2}^{\mathtt{C}}$ *for any type* B*, yet* $x : \bot \mid \langle x\rangle\mathtt{in} \bullet \alpha \mid\rightarrow \alpha : B \vee \bot$ *is derivable only if* $B \equiv \top$ *or* $B \equiv \bot$*.*

Proof: Note first that $\forall X.X \vee \bot \equiv (\top \vee \bot)\&(\bot \vee \bot) \equiv (\exists X.X)\&(\exists X.\bot)$. If we set $A \equiv B \vee \bot$, then a derivation of the former sequent is:

$$\dfrac{\dfrac{\dfrac{x : \bot \rightarrow \alpha : A \mid x : \bot}{x : \bot \rightarrow \alpha : A \mid \langle x \rangle\mathtt{in} : \exists X.X} \quad \dfrac{x : \bot \rightarrow \alpha : A \mid x : \bot}{x : \bot \rightarrow \alpha : A \mid \langle x \rangle\mathtt{in} : \exists X.\bot}}{x : \bot \rightarrow \alpha : A \mid \langle\langle x \rangle\mathtt{in}, \langle x \rangle\mathtt{in}\rangle : \forall X.X \vee \bot} \quad \dfrac{\alpha : B \vee \bot \mid x : \bot \rightarrow \alpha : A}{\mathtt{one}[\alpha] : \forall X.X \vee \bot \mid x : \bot \rightarrow \alpha : A}}{x : \bot \mid \langle\langle x \rangle\mathtt{in}, \langle x \rangle\mathtt{in}\rangle \bullet \mathtt{one}[\alpha] \mid \rightarrow \alpha : B \vee \bot}$$

Now, in order to derive $x : \bot \mid \langle x \rangle\mathtt{in} \bullet \alpha \mid \rightarrow \alpha : B \vee \bot$, we have to derive first $\alpha : C \mid x : \bot \rightarrow \alpha : B \vee \bot$ and $x : \bot \rightarrow \alpha : B \vee \bot \mid \langle x \rangle\mathtt{in} : C$, for some type C; that is we need to derive

$$x : \bot \rightarrow \alpha : B \vee \bot \mid \langle x \rangle\mathtt{in} : B \vee \bot$$

which is possible only in case $B \equiv \top$ or $B \equiv \bot$. ■

In case we wish to fix this problem, we have to add new typing rules in the calculus. In fact, the previous example is revealing on what type of rules we have to add: we need such rules that for any type B the following derivation be possible.

$$\dfrac{x : \bot \rightarrow \; \mid \langle x \rangle\mathtt{in} : \top \vee \bot \quad x : \bot \rightarrow \; \mid \langle x \rangle\mathtt{in} : \bot \vee \bot}{x : \bot \rightarrow \; \mid \langle x \rangle\mathtt{in} : B \vee \bot}$$

Allowing such rules to our calculus ruins the notion of constructivism we had and, in essence, allows of quantification of the form:

$$\dfrac{\Gamma \rightarrow \Theta \mid M : A\{\top/X\} \quad \Gamma \rightarrow \Theta \mid M : A\{\bot/X\}}{\Gamma \rightarrow \Theta \mid M : \forall X.A}$$

which is exactly what we were avoiding throughout this section.

4 Conclusion

In this project we examined an extension of the lambda calculus, the Dual Calculus of Wadler. We saw that under Curry-Howard isomorphism this calculus corresponds to classical logic. Moreover, we saw that two reduction relations defined for this calculus, one corresponding to call-by-value and the other to call-by-name, are De Morgan duals. We studied two basic syntactic properties relative to CBV reduction in the dual calculus: Church-Rosser property for the untyped calculus, and Strong Normalization for the typed calculus. Finally, we tried in several ways to extend the calculus to polymorphic types, and analyzed the difficulties that arose in these attempts. In one of these attempts we proposed an extension of the calculus to non-polymorphic types with second-order quantifiers which corresponds to second-order propositional classical logic.

Of course, some of the matters we studied could be studied more thoroughly, and many important matters weren't examined at all. Thus, in our study of the strong normalization property some reduction rules of R_v (in particular, the ν-rules) were exempted for reasons of simplicity. One could study strong normalization with these rules included; in fact, the `SN` result for `DuCa`* can be modified for this task. Moreover, polymorphism for the dual calculus is still an open question, since all our attempts to polymorphism were flawed in that they couldn't guarantee the Subject Reduction property.

Finally, a very interesting aspect we didn't examine at all, due to lack of time, is that of categorical semantics for the dual calculus. It would be very useful to study such semantics, so that we could obtain an idea on what is the kind of structures that are needed in order to model objects of the calculus. Such a study would help us build more intuition on this new calculus.

5 Acknowledgements

I would like to thank Prof Samson Abramsky for his encouragement and guidance as my supervisor. I would also like to thank my family for their faith and support. Finally, none of this would be possible without Note.

A Gentzen's system LK

The system LK of Gentzen ([Gen35]) is a sequent calculus consisting of:

I. Structural rules:

$$
\begin{array}{lcc}
\text{Weakening} & \dfrac{\Gamma \rightarrow \Theta}{A, \Gamma \rightarrow \Theta} & \dfrac{\Gamma \rightarrow \Theta}{\Gamma \rightarrow \Theta, A} \\[2ex]
\text{Contraction} & \dfrac{A, A, \Gamma \rightarrow \Theta}{A, \Gamma \rightarrow \Theta} & \dfrac{\Gamma \rightarrow \Theta, A, A}{\Gamma \rightarrow \Theta, A} \\[2ex]
\text{Interchange} & \dfrac{\Delta, A, B, \Gamma \rightarrow \Theta}{\Delta, B, A, \Gamma \rightarrow \Theta} & \dfrac{\Gamma \rightarrow \Theta, A, B, \mathrm{I}}{\Gamma \rightarrow \Theta, B, A, \mathrm{I}} \\[2ex]
\text{Cut} & \multicolumn{2}{c}{\dfrac{\Gamma \rightarrow \Theta, A \quad A, \Delta \rightarrow \mathrm{I}}{\Gamma, \Delta \rightarrow \Theta, \mathrm{I}}}
\end{array}
$$

II. Logical rules:

$$\frac{}{A \rightarrow A}\ \texttt{id}$$

$$\frac{A, \Gamma \rightarrow \Theta}{A \& B, \Gamma \rightarrow \Theta} \quad \frac{B, \Gamma \rightarrow \Theta}{A \& B, \Gamma \rightarrow \Theta}\ \& L \qquad \frac{\Gamma \rightarrow \Theta, A \quad \Gamma \rightarrow \Theta, B}{\Gamma \rightarrow \Theta, A \& B}\ \& R$$

$$\frac{\Gamma \rightarrow \Theta, A}{\Gamma \rightarrow \Theta, A \vee B} \quad \frac{\Gamma \rightarrow \Theta, B}{\Gamma \rightarrow \Theta, A \vee B}\ \vee R \qquad \frac{A, \Gamma \rightarrow \Theta \quad B, \Gamma \rightarrow \Theta}{A \vee B, \Gamma \rightarrow \Theta}\ \vee L$$

$$\frac{\Gamma \rightarrow \Theta, A \quad B, \Delta \rightarrow \mathrm{I}}{A \supset B, \Gamma, \Delta \rightarrow \Theta, \mathrm{I}}\ \supset L \qquad \frac{A, \Gamma \rightarrow \Theta, B}{\Gamma \rightarrow \Theta, A \supset B}\ \supset R$$

$$\frac{\Gamma \rightarrow \Theta, A}{\neg A, \Gamma \rightarrow \Theta}\ \neg L \qquad \frac{A, \Gamma \rightarrow \Theta}{\Gamma \rightarrow \Theta, \neg A}\ \neg R$$

$$\frac{A, \Gamma \rightarrow \Theta}{\forall x.A, \Gamma \rightarrow \Theta}\ \forall L \qquad \frac{\Gamma \rightarrow \Theta, A}{\Gamma \rightarrow \Theta, \forall x.A}\ \forall R$$

$$\frac{A, \Gamma \rightarrow \Theta}{\exists x.A, \Gamma \rightarrow \Theta}\ \exists L \qquad \frac{\Gamma \rightarrow \Theta, A}{\Gamma \rightarrow \Theta, \exists x.A}\ \exists R$$

where in $\forall R$ and $\exists L$ there are no free occurrences of variable x inside Γ or Θ.

B Some proofs from section 2

Proof of lemma 2.5:
The proof is by induction on $G \in \texttt{DuCa}$ and case analysis on the two reductions. The `pid` cases are trivial. We examine the other cases. Note that we may use the previous lemma on substitution without mentioning it.
The cases for $G \longrightarrow_{\texttt{p}} G_1$ are:
$\rightarrow$ $\texttt{p}\beta\&_1$: then $G \equiv \langle V, W\rangle \bullet \texttt{fst}[K]$ and $G_1 \equiv V' \bullet K'$; for some $V \longrightarrow_{\texttt{p}} V', W \longrightarrow_{\texttt{p}} W'$,$K \longrightarrow_{\texttt{p}} K'$.
By inspection of the other rules we have the following choices for the rule in $G \longrightarrow_{\texttt{p}} G_2$,

- $\texttt{p}\beta\&_1$: in this case $G_2 \equiv V'' \bullet K''$; for some $V \longrightarrow_{\texttt{p}} V'', W \longrightarrow_{\texttt{p}} W'', K \longrightarrow_{\texttt{p}} K''$. By IH, there exist V_c, K_c such that $V', V'' \longrightarrow_{\texttt{p}} V_c$ and $K', K'' \longrightarrow_{\texttt{p}} K_c$. Thus, $G_1, G_2 \longrightarrow_{\texttt{p}} V_c \bullet K_c$, by use of $\texttt{p}\bullet$.

- $\texttt{p}\nu\&_1$: then $G_2 \equiv V'' \bullet x.(\langle x, W''\rangle \bullet K^?)$; for some $V \longrightarrow_{\texttt{p}} V'', W \longrightarrow_{\texttt{p}} W''$ and $\texttt{fst}[K] \longrightarrow_{\texttt{p}} K^?$.
By inspection, $K^? \equiv \texttt{fst}[K'']$ for some $K \longrightarrow_{\texttt{p}} K''$, and so, by IH, there exist V_c, K_c

with $V', V'' \longrightarrow_{\mathtt{p}} V_c$ and $K', K'' \longrightarrow_{\mathtt{p}} K_c$; thus,

$$\begin{aligned}
&\langle x, W''\rangle \bullet \mathtt{fst}[K''] \xrightarrow{\mathtt{p}\beta\&_1}_{\mathtt{p}} x \bullet K_c \\
&\therefore\ G_2 \equiv V'' \bullet x.(\langle x, W''\rangle \bullet \mathtt{fst}[K'']) \xrightarrow{\mathtt{p}\beta L}_{\mathtt{p}} V_c \bullet K_c \\
&\text{and } G_1 \equiv V' \bullet K' \xrightarrow{\mathtt{p}\bullet}_{\mathtt{p}} V_c \bullet K_c
\end{aligned}$$

- $\mathtt{p}\nu\&_2$: then $G_2 \equiv W'' \bullet y.(\langle V'', y\rangle \bullet \mathtt{fst}[K''])$ and we follow the same steps as above.
- $\mathtt{p}\bullet$: then $G_2 \equiv M^? \bullet K^?$, with $\langle V, W\rangle \longrightarrow_{\mathtt{p}} M^?, \mathtt{fst}[K] \longrightarrow_{\mathtt{p}} K^?$. By inspection, $M^? \equiv \langle V'', W''\rangle$ and $K^? \equiv \mathtt{fst}[K'']$, some $V \longrightarrow_{\mathtt{p}} V'', W \longrightarrow_{\mathtt{p}} W'', K \longrightarrow_{\mathtt{p}} K''$. Hence, using the IH,
  $G_2 \equiv \langle V'', W''\rangle \bullet \mathtt{fst}[K''] \longrightarrow_{\mathtt{p}} V_c \bullet K_c \ {}_{\mathtt{p}}\!\longleftarrow V' \bullet K' \equiv G_1$.

→ $\mathtt{p}\beta\&_2, \mathtt{p}\beta\vee_1, \mathtt{p}\beta\vee_2, \mathtt{p}\beta\neg$: proven similarly.
→ $\mathtt{p}\beta L$: then $G \equiv V \bullet x.(S)$ and $G_1 \equiv S'\{V'/x\}$; for some $V \longrightarrow_{\mathtt{p}} V', S \longrightarrow_{\mathtt{p}} S'$.
By inspection, we have the following choices for $G \longrightarrow_{\mathtt{p}} G_2$.

- $\mathtt{p}\beta L$: then $G_2 \equiv S''\{V''/x\}$ and the result is straightforward from lemma 2.4.
- $\mathtt{p}\nu\&_1$: then $G \equiv \langle V, W\rangle \bullet x.(S)$, $G_1 \equiv S'\{\langle V', W'\rangle/x\}$ and $G_2 \equiv V'' \bullet y.(\langle y, W''\rangle \bullet x.(S''))$; for some $V \longrightarrow_{\mathtt{p}} V', V'', W \longrightarrow_{\mathtt{p}} W', W'', S \longrightarrow_{\mathtt{p}} S', S''$. Then, by IH, there exist $V', V'' \longrightarrow_{\mathtt{p}} V_c$, $W', W'' \longrightarrow_{\mathtt{p}} W_c$, $S', S'' \longrightarrow_{\mathtt{p}} S_c$, and thus,

$$\begin{aligned}
&\langle y, W''\rangle \bullet x.(S'') \xrightarrow{\mathtt{p}\beta L} S_c\{\langle y, W_c\rangle/x\} \\
&\therefore V'' \bullet y.(\langle y, W''\rangle \bullet x.(S'')) \xrightarrow{\mathtt{p}\beta L} (S_c\{\langle y, W_c\rangle/x\})\{V_c/y\} \overset{y \text{ fresh}}{\equiv} S_c\{\langle V_c, W_c\rangle/x\} \\
&\text{and } S'\{\langle V', W'\rangle/x\} \longrightarrow_{\mathtt{p}} S_c\{\langle V_c, W_c\rangle/x\}\ \text{, by lemma 2.4}
\end{aligned}$$

- $\mathtt{p}\nu\&_2, \mathtt{p}\nu\neg_1, \mathtt{p}\nu\neg_2$: proven similarly.
- $\mathtt{p}\bullet$: this case is straightforward by applying lemma 2.4.

→ $\mathtt{p}\beta R$: this case is proven in a similar, yet simpler, way as $\mathtt{p}\beta R$.
→ $\mathtt{p}\nu\&_1$: then $G \equiv \langle M, N\rangle \bullet K$ and $G_1 \equiv M' \bullet x.(\langle x, N'\rangle \bullet K')$; for some $M \longrightarrow_{\mathtt{p}} M'$, $N \longrightarrow_{\mathtt{p}} N'$ and $K \longrightarrow_{\mathtt{p}} K'$. Then, the possible choices for $G \longrightarrow_{\mathtt{p}} G_2$ which have not been considered above in symmetry are:

- $\mathtt{p}\nu\&_2$: then M is a value and $G_2 \equiv N'' \bullet y.(\langle M'', y\rangle \bullet K'')$; for some $M \longrightarrow_{\mathtt{p}} M''$, $N \longrightarrow_{\mathtt{p}} N''$ and $K \longrightarrow_{\mathtt{p}} K''$. Using the IH, we have that,

$$\begin{aligned}
&\langle x, N'\rangle \bullet K' \xrightarrow{\mathtt{p}\nu\&_2} N_c \bullet y.(\langle x, y\rangle \bullet K_c) \\
&\therefore\ G_1 \xrightarrow{\mathtt{p}\beta L} N_c \bullet y.(\langle M_c, y\rangle \bullet K_c) \\
&\text{and clearly } G_2 \longrightarrow_{\mathtt{p}} N_c \bullet y.(\langle M_c, y\rangle \bullet K_c)
\end{aligned}$$

- $\texttt{p}\bullet$: this case is straightforward.

$\rightharpoonup$ $\texttt{p}\nu\&_2, \texttt{p}\nu\vee_1, \texttt{p}\nu\vee_2$: proven similarly.
$\rightharpoonup$ $\texttt{p}\bullet$: this case is simple, since all reductions to which it may be combined are already checked in the previous cases, by symmetry.
$\rightharpoonup$ $\texttt{p}\langle,\rangle, \texttt{p}\langle\rangle\texttt{inl}, \texttt{p}\langle\rangle\texttt{inr}, \texttt{p}[]\texttt{not}, \texttt{pnot}\langle\rangle, \texttt{p}().,\texttt{p}.(), \texttt{pfst}[], \texttt{psnd}[]$: these cases are trivial, since no other reduction can be combined to any one of them. ■

Proof of lemma 2.22:
First note that CR3' is implied by CR3 by induction on $l(G)$, where G is neutral and $\texttt{SN}$, using the fact that all its reducts are also neutral and $\texttt{SN}$.
For CR$[1,2,3]$, we do induction on $d(\sigma)$. The base case, of $d(\sigma) = (0, n)$, is straightforward:

- CR1 is a tautology.
- For CR3, if $\sigma' \in \texttt{Red}$, then G' is $\texttt{SN}$. Then G reduces only to strongly normalizing elements, thus G is $\texttt{SN}$, $\therefore\ \sigma \in \texttt{Red}$.
- For CR2, if $\sigma \in (T_G \cap \texttt{Red})$, then G is $\texttt{SN}$ and so is any G' to which it may reduce.

Now assume $d(\sigma) = (c, n),\ c > 0$ and $\sigma \in T_G$:

CR1: [If $\sigma \in (T_G \cap \texttt{Red})$, some G, then G is $\texttt{SN}$]
– If $G \equiv S$, some $S \equiv M \bullet K$, and $\sigma \equiv \Gamma \,❙\, S \,❙\!\!\rightarrow\, \Theta$, then, since $c > 0$, there exists some $\beta : B \in \Theta$ or $y : B \in \Gamma$ with $c(B) > 0$, and thus, for example in the former case:

$$\sigma' \ \equiv\ \Gamma \ \rightarrow\ \Theta - \{\beta : B\} \,❙\, (S)_{\odot}\beta : B \ \in\ \texttt{Red}$$

Then, by IH, $(S)_{\odot}\beta \in \texttt{SN}$, $\therefore\ S \in \texttt{SN}$.

– If $G \equiv M$ and $\sigma \equiv \Gamma \ \rightarrow\ \Theta \,❙\, M : A$, then, by proposition 2.21, there exists some derivable sequent $\sigma_1 \equiv K_0 : A \,❙\, \Gamma_0 \ \rightarrow\ \Theta_0$, for some K_0 neutral and $\texttt{SN}$, with $d(\sigma_1) = (0, 1) < d(\sigma)$. By definition, $\sigma_1 \in \texttt{Red}$. Then,

$$\sigma_2 \ \equiv\ \Gamma, \Gamma_0 \,❙\, M \bullet K_0 \,❙\!\!\rightarrow\ \Theta, \Theta_0$$

has $d(\sigma_2) = (c, 0) < (c, 1) = d(\sigma)$. Since $\sigma \in \texttt{Red}$, by definition, $\sigma_2 \in \texttt{Red}$,
$\therefore$ by IH, $M \bullet K_0 \in \texttt{SN}$, $\therefore\ M \in \texttt{SN}$.
For the latter implication note that all reductions from M can be translated to reductions from $M \bullet K_0$, except for the case of $M \xrightarrow{\star} (M \bullet \alpha)_{\odot}\alpha$, when M must be simple (and $\star$ is a label). But in any infinite sequence of reductions starting from M, $\star$ can happen at most once, since afterwards M's reduct is immediately followed by a cut. Thus, an infinite reduction sequence from $M \bullet K_0$ can be produced by simply ignoring the $\star$-reduction.

– If $G \equiv K$, we work dually as above.

CR3: [If $\sigma \in T_G$, G neutral, and $\sigma \longrightarrow \sigma'$ implies $\sigma' \in \mathtt{Red}$, then $\sigma \in \mathtt{Red}$]
– If $G \equiv M$ and $\sigma \equiv \Gamma \rightarrow \Theta \shortmid M : A$, then we need to show that for all relevant K_0 and $\sigma_1 \in (T_{K_0} \cap \mathtt{Red})$, we have $\sigma_2 \in \mathtt{Red}$, where:

$$\begin{aligned} \sigma_2 \ &\equiv \ \Gamma, \Gamma_0 \shortmid M \bullet K_0 \shortmid\rightarrow \ \Theta, \Theta_0 \ \ \text{, with } d(\sigma_2) < d(\sigma) \\ \sigma_1 \ &\equiv \ K_0 : A \shortmid \Gamma_0 \ \rightarrow \ \Theta_0 \ \ , \ \ d(\sigma_1) < d(\sigma) \end{aligned}$$

Since $d(\sigma_1) < d(\sigma)$, by IH on CR1, K_0 is $\mathtt{SN}$. Thus, prove that $\sigma_2 \in \mathtt{Red}$ by induction on $l(K_0)$. Since M, K_0 are neutral, $\sigma_2 \longrightarrow \sigma_2'$ implies:

$$\sigma_2' \ \equiv \ \Gamma, \Gamma_0 \shortmid M' \bullet K_0' \shortmid\rightarrow \ \Theta, \Theta_0$$

where either $M \longrightarrow M'$, or $K_0 \longrightarrow K_0'$.
In the former case, $\sigma_2' \in \mathtt{Red}$ by hypothesis.
In the latter case, by IH on CR2, $\sigma_1' \ \equiv \ K_0' : A \shortmid \Gamma_0 \ \rightarrow \ \Theta_0 \in \mathtt{Red}$ and $l(K_0') < l(K_0)$, thus the IH on $l(K_0)$ applies and $\sigma_2' \in \mathtt{Red}$.
In any case, $\sigma_2 \longrightarrow \sigma_2'$ implies $\sigma_2' \in \mathtt{Red}$, $\therefore$ $\sigma_2 \in \mathtt{Red}$, by IH on CR3.

– If $G \equiv K$, we work dually as above.

– If $G \equiv S \equiv M \bullet K$, and, say $\sigma \ \equiv \ \Gamma \shortmid S \shortmid\rightarrow \ \Theta \in T_S$, then $\sigma \in \mathtt{Red}$ iff for all $x : B \in \Gamma$, $\alpha : C \in \Theta$, with $c(B) > 0$, $c(C) > 0$, we have $\sigma_1, \sigma_2 \in \mathtt{Red}$, where,

$$\begin{aligned} \sigma_1 &\equiv \ \Gamma \ \rightarrow \ \Theta - \{\alpha : C\} \shortmid (S)_\odot \alpha : C \\ \sigma_2 &\equiv \ x_\odot(S) : B \shortmid \Gamma - \{x : B\} \ \rightarrow \ \Theta \end{aligned}$$

But $\sigma_1 \longrightarrow \sigma_1'$ implies $\sigma_1' \ \equiv \ \Gamma \ \rightarrow \ \Theta - \{\alpha : C\} \shortmid (S')_\odot \alpha : C$, some $S \longrightarrow S'$, and, by hypothesis, $\sigma' \ \equiv \ \Gamma \shortmid S' \shortmid\rightarrow \ \Theta \ \in \ \mathtt{Red}$, thus $\sigma_1' \in \mathtt{Red}$, $\therefore$ by IH, $\sigma_1 \in \mathtt{Red}$.
Similarly, $\sigma_2 \in \mathtt{Red}$; hence, $\sigma \in \mathtt{Red}$.

CR2: [If $\sigma \in (\mathtt{Red} \cap T_G)$, some G, and $\sigma \longrightarrow \sigma'$, then $\sigma' \in \mathtt{Red}$]
– If $G \equiv M$ and $\sigma \ \equiv \ \Gamma \ \rightarrow \ \Theta \shortmid M : A$, then $\sigma \longrightarrow \sigma'$ implies that $\sigma' \ \equiv \ \Gamma \ \rightarrow \ \Theta \shortmid M' : A$, with $M \longrightarrow M'$.
Now $\sigma \in \mathtt{Red}$; so, for all relevant K_0, $\sigma_1 \in (T_{K_0} \cap \mathtt{Red})$, and

$$\sigma_2 \ \equiv \ \Gamma, \Gamma_0 \shortmid M \bullet K_0 \shortmid\rightarrow \ \Theta, \Theta_0 \quad \text{with } d(\sigma_2) < d(\sigma),$$

we have that $\sigma_2 \in \mathtt{Red}$. Now, if $M' \not\equiv \ (M \bullet \alpha)_\odot \alpha$, any α, then

$$\sigma_2 \longrightarrow \sigma_2' \ \equiv \ \Gamma, \Gamma_0 \shortmid M' \bullet K_0 \shortmid\rightarrow \ \Theta, \Theta_0$$

so, by IH, $\sigma_2' \in \mathtt{Red}$ for all such σ_2', and thus $\sigma' \in \mathtt{Red}$.
In case $M' \equiv (M \bullet \alpha)_\odot \alpha$, we use CR$[1,3]$ for σ, σ'. Indeed, this is the reason for having typed CR[1,3,2] when stating this lemma. Now, by CR1, M is $\mathtt{SN}$, therefore M' is $\mathtt{SN}$, since M is simple. But M' is neutral, so, by CR3', $\sigma' \in \mathtt{Red}$.

– If $G \equiv K : A$, we work dually as above.

– If $G \equiv M \bullet K \equiv S$, $\sigma \equiv \Gamma \blacktriangleright S \blacktriangleright\!\!\rightarrow \Theta$, then $\sigma \longrightarrow \sigma'$ implies $\sigma' \equiv \Gamma \blacktriangleright S' \blacktriangleright\!\!\rightarrow \Theta$, some $S \longrightarrow S'$. Since $\sigma \in \texttt{Red}$, for all $y : B \in \Gamma, \beta : C \in \Theta$ with $c(B) > 0, c(C) > 0$, we have $\sigma_1, \sigma_2 \in \texttt{Red}$, where,

$$\begin{aligned} \sigma_1 &\equiv \Gamma \rightarrow \Theta - \{\beta : C\} \blacktriangleright (S)_{\odot}\beta : C \\ \sigma_2 &\equiv y_{\odot}(S) : B \blacktriangleright \Gamma - \{y : B\} \rightarrow \Theta \end{aligned}$$

By IH, since $\sigma_1 \longrightarrow \sigma_1' \equiv \Gamma \rightarrow \Theta - \{\beta : C\} \blacktriangleright (S')_{\odot}\beta : C$, we have $\sigma_1' \in \texttt{Red}$, and similarly $\sigma_2' \in \texttt{Red}$, for all relevant σ_1', σ_2', $\therefore$ $\sigma' \in \texttt{Red}$. ■

References

[Bar84] Hendrik P. Barendregt. *The Lambda Calculus. Its Syntax and Semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics.* North-Holland, 1984.

[Bar92] Hendrik P. Barendregt. Lambda calculi with types. In Abramsky, Gabbay, and Maibaum, editors, *Handbook of logic in computer science (vol. 2). Background: computational structures*, pages 117–309. Oxford University Press, Inc., 1992.

[BB96] Franco Barbanera and Stefano Berardi. A symmetric lambda calculus for classical program extraction. *Inf. Comput.*, 125(2):103–117, 1996.

[CH00] Pierre-Louis Curien and Hugo Herbelin. The duality of computation. In *Proceedings of the fifth ACM SIGPLAN international conference on Functional programming*, pages 233–243. ACM Press, 2000.

[Dou93] Daniel J. Dougherty. Some lambda calculi with categorical sums and products. In Claude Kirchner, editor, *Proc. 5th International Conference on Rewriting Techniques and Applications (RTA)*, volume 690 of *Lecture Notes in Computer Science*, pages 137–151, Berlin, 1993. Springer-Verlag.

[FH92] Matthias Felleisen and Robert Hieb. The revised report on the syntactic theories of sequential control and state. *Theor. Comput. Sci.*, 103(2):235–271, 1992.

[Fil89] Andrzej Filinski. Declarative continuations: an investigation of duality in programming language semantics (lecture notes in computer science 389). In D. H. Pitt et al., editors, *Category Theory and Computer Science*, pages 224–249. Springer-Verlag, 1989.

[FLO83] Steven Fortune, Daniel Leivant, and Michael O'Donnell. The expressiveness of simple and second-order type structures. *J. ACM*, 30(1):151–185, 1983.

[Gen35] Gerhard Gentzen. Untersuchungen über das logische schließen. *Mathematische Zeitschrift*, 39:176–210, 405–431, 1935. English translation in [Gen69].

[Gen69] Gerhard Gentzen. Investigations into logical deductions, 1935. In M. E. Szabo, editor, *The Collected Works of Gerhard Gentzen*, pages 68–131. North-Holland, Amsterdam, 1969.

[Gri90] Timothy G. Griffin. A formulae-as-types notion of control. In *Conf. Record 17th Annual ACM Symp. on Principles of Programming Languages, POPL'90, San Francisco, CA, USA, 17–19 Jan 1990*, pages 47–57. ACM Press, New York, 1990.

[GTL89] Jean-Yves Girard, Paul Taylor, and Yves Lafont. *Proofs and Types.* 1989.

[Par92] Michel Parigot. $\lambda\mu$-calculus: An algorithmic interpretation of classical natural deduction. In A. Voronkov, editor, *Proceedings of the International Conference on Logic Programming and Automated Reasoning*, pages 190–201, St. Petersburg, Russia, July 1992. Springer-Verlag LNCS 624.

[Par00] Michel Parigot. Strong normalization of second order symmetric lambda-calculus. In *Proceedings of the 20th Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 442–453. Springer-Verlag, 2000.

[Sel01] Peter Selinger. Control categories and duality: On the categorical semantics of the lambda-mu calculus. *Mathematical Structures in Computer Science*, 11(2):207–260, 2001.

[SU98] Morten Heine Sørensen and Pawel Urzyczyn. Lectures on the curry-howard isomorphism. Available as DIKU Rapport 98/14, 1998.

[SW74] Christopher Strachey and Christopher P. Wadsworth. Continuations: A mathematical semantics for handling full jumps. Technical Monograph PRG-11, Oxford University Computing Laboratory Programming Research Group, 1974. Republished in [SW00].

[SW00] Christopher Strachey and Christopher P. Wadsworth. Continuations: A mathematical semantics for handling full jumps. *Higher-Order and Symbolic Computation*, 13(1–2):135–152, April 2000.

[Wad03a] Philip Wadler. Call-by-value is dual to call-by-name. *ICFP 2003, Uppsala, Sweden*, 25-29 August 2003.

[Wad03b] Philip Wadler. Call-by-value is dual to call-by-name reloaded. Unpublished, July 2003.