The vectorial λ-calculus

Pablo Arrighi, Alejandro Díaz-Caro, Benoît Valiron

The Vectorial λ-Calculus☆

Pablo Arrighi[a], Alejandro Díaz-Caro[b], Benoît Valiron[c]

[a] *Aix-Marseille Université, CNRS, LIF, Marseille and IXXI, Lyon, France*
[b] *Universidad Nacional de Quilmes & CONICET, B1876BXD Bernal, Buenos Aires, Argentina*
[c] *LRI, CentraleSuplec, Univ. Paris Sud, CNRS, Université Paris-Saclay, 91405 Orsay Cedex, France*

Abstract

We describe a type system for the linear-algebraic λ-calculus. The type system accounts for the linear-algebraic aspects of this extension of λ-calculus: it is able to statically describe the linear combinations of terms that will be obtained when reducing the programs. This gives rise to an original type theory where types, in the same way as terms, can be superposed into linear combinations. We prove that the resulting typed λ-calculus is strongly normalising and features weak subject reduction. Finally, we show how to naturally encode matrices and vectors in this typed calculus.

1. Introduction

1.1. (Linear-)algebraic λ-calculi

A number of recent works seek to endow the λ-calculus with a vector space structure. This agenda has emerged simultaneously in two different contexts.

- The field of *Linear Logic* considers a logic of resources where the propositions themselves stand for those resources – and hence cannot be discarded nor copied. When seeking to find models of this logic, one obtains a particular family of vector spaces and differentiable functions over these. It is by trying to capture these mathematical structures back into a programming language that Ehrhard and Regnier have defined the *differential λ-calculus* [28], which has an intriguing differential operator as a built-in primitive and an algebraic module of the λ-calculus terms over natural numbers. Vaux [43] has focused his attention on a 'differential λ-calculus without differential operator', extending the algebraic module to positive real numbers. He obtained a confluence result in this case, which stands even in the untyped setting. More recent works on this *algebraic λ-calculus* tend to consider arbitrary scalars [1, 27, 40].

- The field of *Quantum Computation* postulates that, as computers are physical systems, they may behave according to quantum theory. It proves that, if this

☆Partially supported by the STIC-AmSud project FoQCoSS and PICT-PRH 2015-1208.

Email addresses: `pablo.arrighi@univ-amu.fr` (Pablo Arrighi), `alejandro.diaz-caro@unq.edu.ar` (Alejandro Díaz-Caro), `benoit.valiron@monoidal.net` (Benoît Valiron)

is the case, novel, more efficient algorithms are possible [30, 39] – which have no classical counterpart. Whilst partly unexplained, it is nevertheless clear that the algorithmic speed-up arises by tapping into the parallelism granted to us 'for free' by the *superposition principle*, which states that if $\mathbf{t}$ and $\mathbf{u}$ are possible states of a system, then so is the formal linear combination of them $\alpha \cdot \mathbf{t} + \beta \cdot \mathbf{u}$ (with α and β some arbitrary complex numbers, up to a normalizing factor). The idea of a module of λ-terms over an arbitrary scalar field arises quite naturally in this context. This was the motivation behind the *linear-algebraic λ-calculus*, or *Lineal* for short, by Dowek and one of the authors [7], who obtained a confluence result which holds for arbitrary scalars and again covers the untyped setting.

These two languages are rather similar: they both merge higher-order computation, be they terminating or not, in its simplest and most general form (namely the untyped λ-calculus) together with linear algebra also in its simplest and most general form (the axioms of vector spaces). In fact they can simulate each other [8]. Our starting point is the second one, *Lineal*, because its confluence proof allows arbitrary scalars and because one has to make a choice. Whether the models developed for the first language, and the type systems developed for the second language, carry through to one another via their reciprocal simulations, is a topic of future investigation.

1.2. Other motivations to study (linear-)algebraic λ-calculi

The two languages are also reminiscent of other works in the literature:

- *Algebraic and symbolic computation.* The functional style of programming is based on the λ-calculus together with a number of extensions, so as to make everyday programming more accessible. Hence since the birth of functional programming there have been several theoretical studies on extensions of the λ-calculus in order to account for basic algebra (see for instance Dougherty's algebraic extension [26] for normalising terms of the λ-calculus) and other basic programming constructs such as pattern-matching [3, 16], together with sometimes non-trivial associated type theories [37]. Whilst this was not the original motivation behind (linear-)algebraic λ-calculi, they could still be viewed as an extension of the λ-calculus in order to handle operations over vector spaces and make programming more accessible with them. The main difference in approach is that the λ-calculus is not seen here as a control structure which sits on top of the vector space data structure, controlling which operations to apply and when. Rather, the λ-calculus terms themselves can be summed and weighted, hence they actually are vectors, upon which they can also act.

- *Parallel and probabilistic computation.* The above intertwinings of concepts are essential if seeking to represent parallel or probabilistic computation as it is the computation itself which must be endowed with a vector space structure. The ability to superpose λ-calculus terms in that sense takes us back to Boudol's parallel λ-calculus [12] or de Liguoro and Piperno's work on non-deterministic extensions of λ-calculus [18], as well as more recent works such as [14, 24, 36]. It may also be viewed as being part of a series of works on probabilistic extensions of calculi, e.g. [13, 31] and [19, 21, 35] for λ-calculus more specifically.

Hence (linear-)algebraic λ-calculi can be seen as a platform for various applications, ranging from algebraic computation, probabilistic computation, quantum computation and resource-aware computation.

1.3. The language

The language we consider in this paper will be called the *vectorial λ-calculus*, denoted by λ^{vec}. It is derived from *Lineal* [7]. This language admits the regular constructs of λ-calculus: variables $x, y, \ldots$, λ-abstractions $\lambda x.\mathbf{s}$ and application $(\mathbf{s})\,\mathbf{t}$. But it also admits linear combinations of terms: $\mathbf{0}$, $\mathbf{s}+\mathbf{t}$ and $\alpha\cdot\mathbf{s}$ are terms, where the scalar α ranges over a ring. As in [7], it behaves in a call-by-value oriented manner, in the sense that $(\lambda x.\mathbf{r})\,(\mathbf{s}+\mathbf{t})$ first reduces to $(\lambda x.\mathbf{r})\,\mathbf{s}+(\lambda x.\mathbf{r})\,\mathbf{t}$ until *basis terms* (i.e. values) are reached, at which point beta-reduction applies.

The set of the normal forms of the terms can then be interpreted as a module and the term $(\lambda x.\mathbf{r})\,\mathbf{s}$ can be seen as the application of the linear operator $(\lambda x.\mathbf{r})$ to the vector $\mathbf{s}$. The goal of this paper is to give a formal account of linear operators and vectors at the level of the type system.

1.4. Our contributions: The types

Our goal is to characterize the vectoriality of the system of terms, as summarized by the slogan:

If $\mathbf{s} : T$ and $\mathbf{t} : R$ then $\alpha\cdot\mathbf{s}+\beta\cdot\mathbf{t} : \alpha\cdot T+\beta\cdot R$.

In the end we achieve a type system such that:

- The typed language features a slightly weakened subject reduction property (Theorem 4.1).
- The typed language features strong normalization (cf. Theorem 5.7).
- In general, if $\mathbf{t}$ has type $\sum_i \alpha_i\cdot U_i$, then it must reduce to a $\mathbf{t}'$ of the form $\sum_{ij}\beta_{ij}\cdot\mathbf{b}_{ij}$, where: the $\mathbf{b}_{ij}$'s are basis terms of unit type U_i, and $\sum_{ij}\beta_{ij}=\alpha_i$. (cf. Theorem 6.1).
- In particular finite vectors and matrices and tensorial products can be encoded within λ^{vec}. In this case, the type of the encoded expressions coincides with the result of the expression (cf. Theorem 6.2).

Beyond these formal results, this work constitutes a first attempt to describe a natural type system with type constructs $\alpha\cdot$ and $+$ and to study their behaviour.

1.5. Directly related works

This paper is part of a research path [2, 4, 7, 15, 25, 41, 42] to design a typed language where terms can be linear combinations of terms (they can be interpreted as probability distributions or quantum superpositions of data and programs) and where the types capture some of this additional structure (they provide the propositions for a probabilistic or quantum logic via Curry-Howard).

Along this path, a first step was accomplished in [4] with scalars in the type system. If α is a scalar and $\Gamma\vdash\mathbf{t}:T$ is a valid sequent, then $\Gamma\vdash\alpha\cdot\mathbf{t}:\alpha\cdot T$ is a valid sequent.

When the scalars are taken to be positive real numbers, the developed language actually provides a static analysis tool for *probabilistic* computation. However, it fails to address the following issue: without sums but with negative numbers, the term representing "$\mathbf{true} - \mathbf{false}$", namely $\lambda x.\lambda y.x - \lambda x.\lambda y.y$, is typed with $0 \cdot (X \to (X \to X))$, a type which fails to exhibit the fact that we have a superposition of terms.

A second step was accomplished in [25] with sums in the type system. In this case, if $\Gamma \vdash \mathbf{s} : S$ and $\Gamma \vdash \mathbf{t} : T$ are two valid sequents, then $\Gamma \vdash \mathbf{s} + \mathbf{t} : S + T$ is a valid sequent. However, the language considered is only the *additive* fragment of *Lineal*, it leaves scalars out of the picture. For instance, $\lambda x.\lambda y.x - \lambda x.\lambda y.y$, does not have a type, due to its minus sign. Each of these two contributions required renewed, careful and lengthy proofs about their type systems, introducing new techniques.

The type system we propose in this paper builds upon these two approaches: it includes both scalars and sums of types, thereby reflecting the vectorial structure of the terms at the level of types. Interestingly, combining the two separate features of [4, 25] raises subtle novel issues, which we identify and discuss (cf. Section 3). Equipped with those two vectorial type constructs, the type system is indeed able to capture some fine-grained information about the vectorial structure of the terms. Intuitively, this means keeping track of both the 'direction' and the 'amplitude' of the terms.

A preliminary version of this paper has appeared in [5].

1.6. Plan of the paper

In Section 2, we present the language. We discuss the differences with the original language *Lineal* [7]. In Section 3, we explain the problems arising from the possibility of having linear combinations of types, and elaborate a type system that addresses those problems. Section 4 is devoted to subject reduction. We first say why the standard formulation of subject reduction does not hold. Second we state a slightly weakened notion of the subject reduction theorem, and we prove this result. In Section 5, we prove strong normalisation. Finally we close the paper in Section 6 with theorems about the information brought by the type judgements, both in the general and the finitary cases (matrices and vectors).

2. The terms

We consider the untyped language λ^{vec} described in Figure 1. It is based on *Lineal* [7]: terms come in two flavours, basis terms which are the only ones that will substitute a variable in a β-reduction step, and general terms. We use Krivine's notation [34] for function application: The term $(\mathbf{s})\,\mathbf{t}$ passes the argument $\mathbf{t}$ to the function $\mathbf{s}$.

In addition to β-reduction, there are fifteen rules stemming from the oriented axioms of vector spaces [7], specifying the behaviour of sums and products. We divide the rules in groups: Elementary (E), Factorisation (F), Application (A) and the Beta reduction (B). Essentially, the rules E and F, presented in [6], consist in capturing the equations of vector spaces in an oriented rewrite system. For example, $0 \cdot \mathbf{s}$ reduces to $\mathbf{0}$, as $0 \cdot \mathbf{s} = \mathbf{0}$ is valid in vector spaces. It should also be noted that this set of algebraic rule is confluent, and does not introduce loops. In particular, the two rules stating $\alpha \cdot (\mathbf{t} + \mathbf{r}) \to \alpha \cdot \mathbf{t} + \alpha \cdot \mathbf{r}$ and $\alpha \cdot \mathbf{t} + \beta \cdot \mathbf{t} \to (\alpha + \beta) \cdot \mathbf{t}$ are not inverse one of the other when $\mathbf{r} = \mathbf{t}$. Indeed,

$$\alpha \cdot (\mathbf{t} + \mathbf{t}) \to \alpha \cdot \mathbf{t} + \alpha \cdot \mathbf{t} \to (\alpha + \alpha) \cdot \mathbf{t}$$

Terms:	$\mathbf{r},\mathbf{s},\mathbf{t},\mathbf{u} ::=$	$\mathbf{b} \mid (\mathbf{t})\ \mathbf{r} \mid \mathbf{0} \mid \alpha\cdot\mathbf{t} \mid \mathbf{t}+\mathbf{r}$
Basis terms:	$\mathbf{b} ::=$	$x \mid \lambda x.\mathbf{t}$

Group E:	*Group F:*	*Group A:*
$0\cdot\mathbf{t}\to\mathbf{0}$	$\alpha\cdot\mathbf{t}+\beta\cdot\mathbf{t}\to(\alpha+\beta)\cdot\mathbf{t}$	$(\mathbf{t}+\mathbf{r})\ \mathbf{u}\to(\mathbf{t})\ \mathbf{u}+(\mathbf{r})\ \mathbf{u}$
$1\cdot\mathbf{t}\to\mathbf{t}$	$\alpha\cdot\mathbf{t}+\mathbf{t}\to(\alpha+1)\cdot\mathbf{t}$	$(\mathbf{t})\ (\mathbf{r}+\mathbf{u})\to(\mathbf{t})\ \mathbf{r}+(\mathbf{t})\ \mathbf{u}$
$\alpha\cdot\mathbf{0}\to\mathbf{0}$	$\mathbf{t}+\mathbf{t}\to(1+1)\cdot\mathbf{t}$	$(\alpha\cdot\mathbf{t})\ \mathbf{r}\to\alpha\cdot(\mathbf{t})\ \mathbf{r}$
$\alpha\cdot(\beta\cdot\mathbf{t})\to(\alpha\times\beta)\cdot\mathbf{t}$	$\mathbf{t}+\mathbf{0}\to\mathbf{t}$	$(\mathbf{t})\ (\alpha\cdot\mathbf{r})\to\alpha\cdot(\mathbf{t})\ \mathbf{r}$
$\alpha\cdot(\mathbf{t}+\mathbf{r})\to\alpha\cdot\mathbf{t}+\alpha\cdot\mathbf{r}$	*Group B:*	$(\mathbf{0})\ \mathbf{t}\to\mathbf{0}$
	$(\lambda x.\mathbf{t})\ \mathbf{b}\to\mathbf{t}[\mathbf{b}/x]$	$(\mathbf{t})\ \mathbf{0}\to\mathbf{0}$

$$\frac{\mathbf{t}\to\mathbf{r}}{\alpha\cdot\mathbf{t}\to\alpha\cdot\mathbf{r}}\qquad\frac{\mathbf{t}\to\mathbf{r}}{\mathbf{u}+\mathbf{t}\to\mathbf{u}+\mathbf{r}}\qquad\frac{\mathbf{t}\to\mathbf{r}}{(\mathbf{u})\ \mathbf{t}\to(\mathbf{u})\ \mathbf{r}}\qquad\frac{\mathbf{t}\to\mathbf{r}}{(\mathbf{t})\ \mathbf{u}\to(\mathbf{r})\ \mathbf{u}}\qquad\frac{\mathbf{t}\to\mathbf{r}}{\lambda x.\mathbf{t}\to\lambda x.\mathbf{r}}$$

Figure 1: Syntax, reduction rules and context rules of λ^{vec}.

but not the other what around.

The group of A rules formalize the fact that a general term $\mathbf{t}$ is thought of as a linear combination of terms $\alpha\cdot\mathbf{r}+\beta\cdot\mathbf{r}'$ and the face that the application is distributive on the left *and* on the right. When we apply $\mathbf{s}$ to such a superposition, $(\mathbf{s})\ \mathbf{t}$ reduces to $\alpha\cdot(\mathbf{s})\ \mathbf{r}+\beta\cdot(\mathbf{s})\ \mathbf{r}'$. The term $\mathbf{0}$ is the empty linear combination of terms, explaining the last two rules of Group A.

Terms are considered modulo associativity and commutativity of the operator $+$, making the reduction into an *AC-rewrite system* [32]. Scalars (notation $\alpha,\beta,\gamma,\dots$) form a ring $(\mathsf{S},+,\times)$, where the scalar 0 is the unit of the addition and 1 the unit of the multiplication. We use the shortcut notation $\mathbf{s}-\mathbf{t}$ in place of $\mathbf{s}+(-1)\cdot\mathbf{t}$. Note that although the typical ring we consider in the examples is the ring of complex numbers, the development works for any ring: the ring of integer $\mathbb{Z}$, the finite ring $\mathbb{Z}/2\mathbb{Z}$...

The set of free variables of a term is defined as usual: the only operator binding variables is the λ-abstraction. The operation of substitution on terms (notation $\mathbf{t}[\mathbf{b}/x]$) is defined in the usual way for the regular λ-term constructs, by taking care of variable renaming to avoid capture. For a linear combination, the substitution is defined as follows: $(\alpha\cdot\mathbf{t}+\beta\cdot\mathbf{r})[\mathbf{b}/x]=\alpha\cdot\mathbf{t}[\mathbf{b}/x]+\beta\cdot\mathbf{r}[\mathbf{b}/x]$.

Note that we need to choose a reduction strategy. For example, the term $(\lambda x.(x)\ x)\ (y+z)$ cannot reduce to both $(\lambda x.(x)\,x)\,y+(\lambda x.(x)\,x)\,z$ and $(y+z)\,(y+z)$. Indeed, the former normalizes to $(y)\,y+(z)\,z$ whereas the latter normalizes to $(y)\,z+(y)\,y+(z)\,y+(z)\,z$; which would break confluence. As in [4, 7, 25], we consider a call-by-value reduction strategy: The argument of the application is required to be a base term, cf. Group B.

2.1. Relation to Lineal

Although strongly inspired from *Lineal*, the language λ^{vec} is closer to [4, 8, 25]. Indeed, *Lineal* considers some restrictions on the reduction rules, for example $\alpha\cdot\mathbf{t}+\beta\cdot\mathbf{t}\to(\alpha+\beta)\cdot\mathbf{t}$ is only allowed when $\mathbf{t}$ is a closed normal term. These restrictions are enforced to ensure confluence in the untyped setting. Consider the following example. Let $\mathbf{Y_b}=(\lambda x.(\mathbf{b}+(x)\ x))\ \lambda x.(\mathbf{b}+(x)\ x)$. Then $\mathbf{Y_b}$ reduces to $\mathbf{b}+\mathbf{Y_b}$. So the term $\mathbf{Y_b}-\mathbf{Y_b}$ reduces to $\mathbf{0}$, but also reduces to $\mathbf{b}+\mathbf{Y_b}-\mathbf{Y_b}$ and hence to $\mathbf{b}$, breaking confluence. The above restriction forbids the first reduction, bringing back confluence. In our setting we do not need it because $\mathbf{Y_b}$ is not well-typed. If one considers a typed language enforcing strong normalisation, one can waive many of the restrictions and consider a more canonical set of rewrite rules [4, 8, 25]. Working with a type system enforcing strong normalisation (as shown in Section 5), we follow this approach.

2.2. Booleans in the vectorial λ-calculus

We claimed in the introduction that the design of *Lineal* was motivated by quantum computing; in this section we develop this analogy.

Both in λ^{vec} and in quantum computation one can interpret the notion of booleans. In the former we can consider the usual booleans $\lambda x.\lambda y.x$ and $\lambda x.\lambda y.y$ whereas in the latter we consider the regular quantum bits $\mathbf{true} = |0\rangle$ and $\mathbf{false} = |1\rangle$.

In λ^{vec}, a representation of *if* $\mathbf{r}$ *then* $\mathbf{s}$ *else* $\mathbf{t}$ needs to take into account the special relation between sums and applications. We cannot directly encode this test as the usual $((\mathbf{r})\,\mathbf{s})\,\mathbf{t}$. Indeed, if $\mathbf{r}$, $\mathbf{s}$ and $\mathbf{t}$ were respectively the terms $\mathbf{true}$, $\mathbf{s}_1+\mathbf{s}_2$ and $\mathbf{t}_1+\mathbf{t}_2$, the term $((\mathbf{r})\,\mathbf{s})\,\mathbf{t}$ would reduce to $((\mathbf{true})\,\mathbf{s}_1)\,\mathbf{t}_1 + ((\mathbf{true})\,\mathbf{s}_1)\,\mathbf{t}_2 + ((\mathbf{true})\,\mathbf{s}_2)\,\mathbf{t}_1 + ((\mathbf{true})\,\mathbf{s}_2)\,\mathbf{t}_2$, then to $2\cdot\mathbf{s}_1 + 2\cdot\mathbf{s}_2$ instead of $\mathbf{s}_1+\mathbf{s}_2$. We need to "freeze" the computations in each branch of the test so that the sum does not distribute over the application. For that purpose we use the well-known notion of *thunks* [7]: we encode the test as $\{((\mathbf{r})\,[\mathbf{s}])\,[\mathbf{t}]\}$, where $[-]$ is the term $\lambda f.-$ with f a fresh, unused term variable and where $\{-\}$ is the term $(-)\lambda x.x$. The former "freezes" the linearity while the latter "releases" it. Then the term *if* $\mathbf{true}$ *then* $(\mathbf{s}_1+\mathbf{s}_2)$ *else* $(\mathbf{t}_1+\mathbf{t}_2)$ reduces to the term $\mathbf{s}_1+\mathbf{s}_2$ as one could expect. Note that this test is linear, in the sense that the term *if* $(\alpha\cdot\mathbf{true}+\beta\cdot\mathbf{false})$ *then* $\mathbf{s}$ *else* $\mathbf{t}$ reduces to $\alpha\cdot\mathbf{s}+\beta\cdot\mathbf{t}$.

This is similar to the *quantum test* that can be found e.g. in [2, 42]. Quantum computation deals with complex, linear combinations of terms, and a typical computation is run by applying linear unitary operations on the terms, called *gates*. For example, the Hadamard gate $\mathbf{H}$ acts on the space of booleans spanned by $\mathbf{true}$ and $\mathbf{false}$. It sends $\mathbf{true}$ to $\frac{1}{\sqrt{2}}(\mathbf{true}+\mathbf{false})$ and $\mathbf{false}$ to $\frac{1}{\sqrt{2}}(\mathbf{true}-\mathbf{false})$. If x is a quantum bit, the value $(\mathbf{H})\,x$ can be represented as the quantum test

$$(\mathbf{H})\,x \quad := \quad \textit{if } x \textit{ then } \frac{1}{\sqrt{2}}(\mathbf{true}+\mathbf{false}) \textit{ else } \frac{1}{\sqrt{2}}(\mathbf{true}-\mathbf{false}).$$

As developed in [7], one can simulate this operation in λ^{vec} using the test construction we just described:

$$\{(\mathbf{H})\,x\} \quad := \quad \left\{\left((x)\,\left[\frac{1}{\sqrt{2}}\cdot\mathbf{true}+\frac{1}{\sqrt{2}}\cdot\mathbf{false}\right]\right)\,\left[\frac{1}{\sqrt{2}}\cdot\mathbf{true}-\frac{1}{\sqrt{2}}\cdot\mathbf{false}\right]\right\}.$$

Note that the thunks are necessary: without thunks the term

$$\left((x)\,\left(\frac{1}{\sqrt{2}}\cdot\mathbf{true}+\frac{1}{\sqrt{2}}\cdot\mathbf{false}\right)\right)\,\left(\frac{1}{\sqrt{2}}\cdot\mathbf{true}-\frac{1}{\sqrt{2}}\cdot\mathbf{false}\right)$$

would reduce to the term

$$\frac{1}{2}(((x)\,\mathbf{true})\,\mathbf{true} + ((x)\,\mathbf{true})\,\mathbf{false} + ((x)\,\mathbf{false})\,\mathbf{true} + ((x)\,\mathbf{false})\,\mathbf{false}),$$

which is fundamentally different from the term $\mathbf{H}$ we are trying to emulate.

With this procedure we can "encode" any matrix. If the space is of some general dimension n, instead of the basis elements **true** and **false** we can choose for $i = 1$ to n the terms $\lambda x_1.\cdots.\lambda x_n.x_i$'s to encode the basis of the space. We can also take tensor products of qubits. We come back to these encodings in Section 6.

3. The type system

This section presents the core definition of the paper: the vectorial type system.

3.1. Intuitions

Before diving into the technicalities of the definition, we discuss the rationale behind the construction of the type-system.

3.1.1. Superposition of types

We want to incorporate the notion of scalars in the type system. If A is a valid type, the construction $\alpha \cdot A$ is also a valid type and if the terms $\mathbf{s}$ and $\mathbf{t}$ are of type A, the term $\alpha \cdot \mathbf{s} + \beta \cdot \mathbf{t}$ is of type $(\alpha + \beta) \cdot A$. This was achieved in [4] and it allows us to distinguish between the functions $\lambda x.(1 \cdot x)$ and $\lambda x.(2 \cdot x)$: the former is of type $A \to A$ whereas the latter is of type $A \to (2 \cdot A)$.

The terms **true** and **false** can be typed in the usual way with $\mathcal{B} = X \to (X \to X)$, for a fixed type X. So let us consider the term $\frac{1}{\sqrt{2}} \cdot (\mathbf{true} - \mathbf{false})$. Using the above addition to the type system, this term should be of type $0 \cdot \mathcal{B}$, a type which fails to exhibit the fact that we have a superposition of terms. For instance, applying the Hadamard gate to this term produces the term **false** of type $\mathcal{B}$: the norm would then jump from 0 to 1. This time, the problem comes from the fact that the type system does not keep track of the "direction" of a term.

To address this problem we must allow sums of types. For instance, provided that $\mathcal{T} = X \to (Y \to X)$ and $\mathcal{F} = X \to (Y \to Y)$, we can type the term $\frac{1}{\sqrt{2}} \cdot (\mathbf{true} - \mathbf{false})$ with $\frac{\sqrt{2}}{2} \cdot (\mathcal{T} - \mathcal{F})$, which has L_2-norm 1, just like the type of **false** has norm one.

At this stage the type system is able to type the term $\mathbf{H} = \lambda x.\{((x)\,[\frac{1}{\sqrt{2}} \cdot \mathbf{true} + \frac{1}{\sqrt{2}} \cdot \mathbf{false}])\,[\frac{1}{\sqrt{2}} \cdot \mathbf{true} - \frac{1}{\sqrt{2}} \cdot \mathbf{false}]\}$. Indeed, remember that the thunk construction $[-]$ is simply $\lambda f.(-)$ where f is a fresh variable and that $\{-\}$ is $(-)\lambda x.x$. So whenever $\mathbf{t}$ has type A, $[\mathbf{t}]$ has type $\mathbf{I} \to A$ with $\mathbf{I}$ an identity type of the form $Z \to Z$, and $\{\mathbf{t}\}$ has type A whenever $\mathbf{t}$ has type $\mathbf{I} \to A$. The term $\mathbf{H}$ can then be typed with $((\mathbf{I} \to \frac{1}{\sqrt{2}}.(\mathcal{T} + \mathcal{F})) \to (\mathbf{I} \to \frac{1}{\sqrt{2}}.(\mathcal{T} - \mathcal{F})) \to \mathbf{I} \to T) \to T$, where T any fixed type.

Let us now try to type the term $(\mathbf{H})\,\mathbf{true}$. This is possible by taking T to be $\frac{1}{\sqrt{2}} \cdot (\mathcal{T} + \mathcal{F})$. But then, if we want to type the term $(\mathbf{H})\,\mathbf{false}$, T needs to be equal to $\frac{1}{\sqrt{2}} \cdot (\mathcal{T} - \mathcal{F})$. It follows that we cannot type the term $(\mathbf{H})\,(\frac{1}{\sqrt{2}} \cdot \mathbf{true} + \frac{1}{\sqrt{2}} \cdot \mathbf{false})$ since there is no possibility to conciliate the two constraints on T.

To address this problem, we need a forall construction in the type system, making it *à la System F*. The term $\mathbf{H}$ can now be typed with $\forall T.((\mathbf{I} \to \frac{1}{\sqrt{2}} \cdot (\mathcal{T} + \mathcal{F})) \to (\mathbf{I} \to \frac{1}{\sqrt{2}} \cdot (\mathcal{T} - \mathcal{F})) \to \mathbf{I} \to T) \to T$ and the types $\mathcal{T}$ and $\mathcal{F}$ are updated to be respectively $\forall XY.X \to (Y \to X)$ and $\forall XY.X \to (Y \to Y)$. The terms $(\mathbf{H})\,\mathbf{true}$ and $(\mathbf{H})\,\mathbf{false}$ can both be well-typed with respective types $\frac{1}{\sqrt{2}} \cdot (\mathcal{T} + \mathcal{F})$ and $\frac{1}{\sqrt{2}} \cdot (\mathcal{T} - \mathcal{F})$, as expected.

3.1.2. Type variables, units and general types

Because of the call-by-value strategy, variables must range over types that are not linear combination of other types, i.e. *unit types.* To illustrate this necessity, consider the following example. Suppose we allow variables to have scaled types, such as $\alpha \cdot U$. Then the term $\lambda x.x + y$ could have type $(\alpha \cdot U) \to \alpha \cdot U + V$ (with y of type V). Let $\mathbf{b}$ be of type U, then $(\lambda x.x + y)\ (\alpha \cdot \mathbf{b})$ has type $\alpha \cdot U + V$, but then

$$(\lambda x.x + y)\ (\alpha \cdot \mathbf{b}) \to \alpha \cdot (\lambda x.x + y)\ \mathbf{b} \to \alpha \cdot (\mathbf{b} + y) \to \alpha \cdot \mathbf{b} + \alpha \cdot y\,,$$

which is problematic since the type $\alpha \cdot U + V$ does not reflect such a superposition. Hence, the left side of an arrow will be required to be a unit type. This is achieved by the grammar defined in Figure 2.

Type variables, however, do not always have to be unit type. Indeed, a forall of a general type was needed in the previous section in order to type the term $\mathbf{H}$. But we need to distinguish a general type variable from a unit type variable, in order to make sure that only unit types appear at the left of arrows. Therefore, we define two sorts of type variables: the variables $\mathcal{X}$ to be replaced with unit types, and $\mathbb{X}$ to be replaced with any type (we use just X when we mean either one). The type $\mathcal{X}$ is a unit type whereas the type $\mathbb{X}$ is not.

In particular, the type $\mathcal{T}$ is now $\forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X}$, the type $\mathcal{F}$ is $\forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{Y}$ and the type of $\mathbf{H}$ is

$$\forall \mathbb{X}. \left(\left(\mathbf{I} \to \frac{1}{\sqrt{2}} \cdot (\mathcal{T} + \mathcal{F}) \right) \to \left(\mathbf{I} \to \frac{1}{\sqrt{2}} \cdot (\mathcal{T} - \mathcal{F}) \right) \to \mathbf{I} \to \mathbb{X} \right) \to \mathbb{X}.$$

Notice how the left sides of all arrows remain unit types.

3.1.3. The term $\mathbf{0}$

The term $\mathbf{0}$ will naturally have the type $0 \cdot T$, for any inhabited type T (enforcing the intuition that the term $\mathbf{0}$ is essentialy a normal form of programs of the form $\mathbf{t} - \mathbf{t}$).

We could also consider to add the equivalence $R + 0 \cdot T \equiv R$ as in [4]. However, consider the following example. Let $\lambda x.x$ be of type $U \to U$ and let $\mathbf{t}$ be of type T. The term $\lambda x.x + \mathbf{t} - \mathbf{t}$ is of type $(U \to U) + 0 \cdot T$, that is, $(U \to U)$. Now choose $\mathbf{b}$ of type U: we are allowed to say that $(\lambda x.x + \mathbf{t} - \mathbf{t})\,\mathbf{b}$ is of type U. This term reduces to $\mathbf{b} + (\mathbf{t})\,\mathbf{b} - (\mathbf{t})\,\mathbf{b}$. But if the type system is reasonable enough, we should at least be able to type $(\mathbf{t})\,\mathbf{b}$. However, since there is no constraints on the type T, this is difficult to enforce.

The problem comes from the fact that along the typing of $\mathbf{t} - \mathbf{t}$, the type of $\mathbf{t}$ is lost in the equivalence $(U \to U) + 0 \cdot T \equiv U \to U$. The only solution is to not discard $0 \cdot T$, that is, to not equate $R + 0 \cdot T$ and R.

3.2. Formalisation

We now give a formal account of the type system: we first describe the language of types, then present the typing rules.

$$
\begin{array}{lrcl}
\textit{Types:} & T, R, S & ::= & U \mid \alpha \cdot T \mid T + R \mid \mathbb{X} \\
\textit{Unit types:} & U, V, W & ::= & \mathcal{X} \mid U \to T \mid \forall \mathcal{X}.U \mid \forall \mathbb{X}.U
\end{array}
$$

$$
\begin{array}{rclcrcl}
1 \cdot T & \equiv & T & \qquad & \alpha \cdot T + \beta \cdot T & \equiv & (\alpha + \beta) \cdot T \\
\alpha \cdot (\beta \cdot T) & \equiv & (\alpha \times \beta) \cdot T & & T + R & \equiv & R + T \\
\alpha \cdot T + \alpha \cdot R & \equiv & \alpha \cdot (T + R) & & T + (R + S) & \equiv & (T + R) + S
\end{array}
$$

$$
\frac{}{\Gamma, x : U \vdash x : U}\ ax
\qquad
\frac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \mathbf{0} : 0 \cdot T}\ 0_I
\qquad
\frac{\Gamma, x : U \vdash \mathbf{t} : T}{\Gamma \vdash \lambda x.\mathbf{t} : U \to T}\ \to_I
$$

$$
\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i) \qquad \Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j / \vec{X}]}{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j / \vec{X}]}\ \to_E
$$

$$
\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i \quad X \notin FV(\Gamma)}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i}\ \forall_I
\qquad
\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i[A/X]}\ \forall_E
$$

$$
\frac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \alpha \cdot \mathbf{t} : \alpha \cdot T}\ \alpha_I
\qquad
\frac{\Gamma \vdash \mathbf{t} : T \qquad \Gamma \vdash \mathbf{r} : R}{\Gamma \vdash \mathbf{t} + \mathbf{r} : T + R}\ +_I
\qquad
\frac{\Gamma \vdash \mathbf{t} : T \qquad T \equiv R}{\Gamma \vdash \mathbf{t} : R}\ \equiv
$$

Figure 2: Types and typing rules of λ^{vec}. We use X when we do not want to specify if it is $\mathcal{X}$ or $\mathbb{X}$, that is, unit variables or general variables respectively. In $T[A/X]$, if $X = \mathcal{X}$, then A is a unit type, and if $X = \mathbb{X}$, then A can be any type. We also may write $\forall_I$ and $\forall_{\mathbb{I}}$ (resp. $\forall_{\mathcal{E}}$ and $\forall_{\mathbb{E}}$) when we need to specify which kind of variable is being used.

3.2.1. Definition of types

Types are defined in Figure 2 (top). They come in two flavours: *unit types* and general types, that is, linear combinations of types. Unit types include all types of *System F* [29, Ch. 11] and intuitively they are used to type basis terms. The arrow type admits only a unit type in its domain. This is due to the fact that the argument of a λ-abstraction can only be substituted by a basis term, as discussed in Section 3.1.2. As discussed before, the type system features two sorts of variables: unit variables $\mathcal{X}$ and general variables $\mathbb{X}$. The former can only be substituted by a unit type whereas the latter can be substituted by any type. We use the notation X when the type variable is unrestricted. The substitution of $\mathcal{X}$ by U (resp. $\mathbb{X}$ by S) in T is defined as usual and is written $T[U/\mathcal{X}]$ (resp. $T[S/\mathbb{X}]$). We use the notation $T[A/X]$ to say: "if X is a unit variable, then A is a unit type and otherwise A is a general type". In particular, for a linear combination, the substitution is defined as follows: $(\alpha \cdot T + \beta \cdot R)[A/X] = \alpha \cdot T[A/X] + \beta \cdot R[A/X]$. We also use the vectorial notation $T[\vec{A}/\vec{X}]$ for $T[A_1/X_1] \cdots [A_n/X_n]$ if $\vec{X} = X_1, \dots, X_n$ and $\vec{A} = A_1, \dots, A_n$, and also $\forall\vec{X}$ for $\forall X_1 \dots X_n = \forall X_1 . \dots . \forall X_n$.

The equivalence relation $\equiv$ on types is defined as a congruence. Notice that this equivalence makes the types into a weak module over the scalars: they almost form a module save from the fact that there is no neutral element for the addition. The type $0 \cdot T$ is not the neutral element of the addition.

We may use the summation ($\sum$) notation without ambiguity, due to the associativity and commutativity equivalences of $+$.

3.2.2. Typing rules

The typing rules are given also in Figure 2 (bottom). Contexts are denoted by Γ, Δ, etc. and are defined as sets $\{x : U, \dots\}$, where x is a term variable appearing only once in the set, and U is a unit type. The axiom (ax) and the arrow introduction rule $(\to_I)$ are the usual ones. The rule (0_I) to type the term $\mathbf{0}$ takes into account the discussion in Section 3.1.3. This rule also ensures that the type of $\mathbf{0}$ is inhabited, discarding problematic types like $0 \cdot \forall X.X$. Any sum of typed terms can be typed using Rule $(+_I)$. Similarly, any scaled typed term can be typed with (α_I). Rule $(\equiv)$ ensures that equivalent types can be used to type the same terms. Finally, the particular form of the arrow-elimination rule $(\to_E)$ is due to the rewrite rules in group A that distribute sums and scalars over application. The need and use of this complicated arrow elimination can be illustrated by the following three examples.

Example 3.1. Rule $(\to_E)$ is easier to read for trivial linear combinations. It states that provided that $\Gamma \vdash \mathbf{s} : \forall X.U \to S$ and $\Gamma \vdash \mathbf{t} : V$, if there exists some type W such that $V = U[W/X]$, then since the sequent $\Gamma \vdash \mathbf{s} : V \to S[W/X]$ is valid, we also have $\Gamma \vdash (\mathbf{s})\,\mathbf{t} : S[W/X]$. Hence, the arrow elimination here performs an arrow and a forall elimination at the same time.

Example 3.2. Consider the terms $\mathbf{b}_1$ and $\mathbf{b}_2$, of respective types U_1 and U_2. The term $\mathbf{b}_1 + \mathbf{b}_2$ is of type $U_1 + U_2$. We would reasonably expect the term $(\lambda x.x)\,(\mathbf{b}_1 + \mathbf{b}_2)$ to also be of type $U_1 + U_2$. This is the case thanks to Rule $(\to_E)$. Indeed, type the term $\lambda x.x$ with the type $\forall X.X \to X$ and we can now apply the rule. Notice that we could not type such a term unless we eliminate the forall together with the arrow.

Example 3.3. A slightly more involved example is the projection of a pair of elements. It is possible to encode in *System F* the notion of pairs and projections: $\langle \mathbf{b}, \mathbf{c} \rangle = \lambda x.((x)\ \mathbf{b})\ \mathbf{c}$, $\langle \mathbf{b}', \mathbf{c}' \rangle = \lambda x.((x)\ \mathbf{b}')\ \mathbf{c}'$, $\pi_1 = \lambda x.(x)\ (\lambda y.\lambda z.y)$ and $\pi_2 = \lambda x.(x)\ (\lambda y.\lambda z.z)$. Provided that $\mathbf{b}$, $\mathbf{b}'$, $\mathbf{c}$ and $\mathbf{c}'$ have respective types U, U', V and V', the type of $\langle \mathbf{b}, \mathbf{c} \rangle$ is $\forall X.(U \to V \to X) \to X$ and the type of $\langle \mathbf{b}', \mathbf{c}' \rangle$ is $\forall X.(U' \to V' \to X) \to X$. The term π_1 and π_2 can be typed respectively with $\forall XYZ.((X \to Y \to X) \to Z) \to Z$ and $\forall XYZ.((X \to Y \to Y) \to Z) \to Z$. The term $(\pi_1 + \pi_2)\,(\langle \mathbf{b}, \mathbf{c} \rangle + \langle \mathbf{b}', \mathbf{c}' \rangle)$ is then typable of type $U + U' + V + V'$, thanks to Rule $(\to_E)$. Note that this is consistent with the rewrite system, since it reduces to $\mathbf{b} + \mathbf{c} + \mathbf{b}' + \mathbf{c}'$.

3.3. Example: Typing Hadamard

In this Section, we formally show how to retrieve the type that was discussed in Section 3.1.2, for the term $\mathbf{H}$ encoding the Hadamard gate.

Let $\mathbf{true} = \lambda x.\lambda y.x$ and $\mathbf{false} = \lambda x.\lambda y.y$. It is easy to check that

$$\vdash \mathbf{true} : \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X},$$
$$\vdash \mathbf{false} : \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{Y}.$$

We also define the following superpositions:

$$|+\rangle = \frac{1}{\sqrt{2}} \cdot (\mathbf{true} + \mathbf{false}) \qquad \text{and} \qquad |-\rangle = \frac{1}{\sqrt{2}} \cdot (\mathbf{true} - \mathbf{false}).$$

In the same way, we define

$$\boxplus = \frac{1}{\sqrt{2}} \cdot ((\forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X}) + (\forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{Y})),$$
$$\boxminus = \frac{1}{\sqrt{2}} \cdot ((\forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X}) - (\forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{Y})).$$

Finally, we recall $[\mathbf{t}] = \lambda x.\mathbf{t}$, where $x \notin FV(\mathbf{t})$ and $\{\mathbf{t}\} = (\mathbf{t})\ I$. So $\{[\mathbf{t}]\} \to \mathbf{t}$. Then it is easy to check that $\vdash [|+\rangle] : I \to \boxplus$ and $\vdash [|-\rangle] : I \to \boxminus$. In order to simplify the notation, let $F = (I \to \boxplus) \to (I \to \boxminus) \to (I \to \mathbb{X})$. Then

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x : F \vdash x : F}\,ax \qquad x : F \vdash [|+\rangle] : I \to \boxplus}{x : F \vdash (x)\ [|+\rangle] : (I \to \boxminus) \to (I \to \mathbb{X})}\,\to_E \qquad x : F \vdash [|-\rangle] : I \to \boxminus}{x : F \vdash (x)\ [|+\rangle][|-\rangle] : I \to \mathbb{X}}\,\to_E}{x : F \vdash \{(x)\ [|+\rangle][|-\rangle]\} : \mathbb{X}}\,\to_E}{\vdash \lambda x.\{(x)\ [|+\rangle][|-\rangle]\} : F \to \mathbb{X}}\,\to_I}{\vdash \lambda x.\{(x)\ [|+\rangle][|-\rangle]\} : \forall \mathbb{X}.((I \to \boxplus) \to (I \to \boxminus) \to (I \to \mathbb{X})) \to \mathbb{X}}\,\forall_{\mathbb{I}}$$

Now we can apply Hadamard to a qubit and get the right type. Let H be the term $\lambda x.\{(x)\ [|+\rangle][|-\rangle]\}$

$$\dfrac{\dfrac{\vdash H : \forall\mathbb{X}.((I \to \boxplus) \to (I \to \boxminus) \to (I \to \mathbb{X})) \to \mathbb{X}}{\vdash H : ((I \to \boxplus) \to (I \to \boxminus) \to (I \to \boxplus)) \to \boxplus}\ \forall_{\mathbb{E}} \quad \dfrac{\dfrac{\vdash \mathbf{true} : \forall \mathcal{X}.\forall\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X}}{\vdash \mathbf{true} : \forall\mathcal{Y}.(I \to \boxplus) \to \mathcal{Y} \to (I \to \boxplus)}\ \forall_{\mathcal{E}}}{\vdash \mathbf{true} : (I \to \boxplus) \to (I \to \boxminus) \to (I \to \boxplus)}\ \forall_{\mathcal{E}}}{\vdash (H)\ \mathbf{true} : \boxplus}\ \to_E$$

Yet a more interesting example is the following. Let

$$\boxplus_I = \frac{1}{\sqrt{2}} \cdot (((I \to \boxplus) \to (I \to \boxminus) \to (I \to \boxplus)) + ((I \to \boxplus) \to (I \to \boxminus) \to (I \to \boxminus)))$$

That is, $\boxplus$ where the forall have been instantiated. It is easy to check that $\vdash |+\rangle : \boxplus_I$. Hence,

$$\dfrac{\vdash H : \forall\mathbb{X}.((I \to \boxplus) \to (I \to \boxminus) \to (I \to \mathbb{X})) \to \mathbb{X} \qquad \vdash |+\rangle : \boxplus_I}{\vdash (H)\ |+\rangle : \frac{1}{\sqrt{2}} \cdot \boxplus + \frac{1}{\sqrt{2}} \cdot \boxminus}\ \to_E$$

And since $\frac{1}{\sqrt{2}} \cdot \boxplus + \frac{1}{\sqrt{2}} \cdot \boxminus \equiv \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X}$, we conclude that

$$\vdash (H)\ |+\rangle : \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to \mathcal{Y} \to \mathcal{X}.$$

Notice that $(H)\ |+\rangle \to^* \mathbf{true}$.

4. Subject reduction

As we will now explain, the usual formulation of subject reduction is not directly satisfied. We discuss the alternatives and opt for a weakened version of subject reduction.

4.1. Principal types and subtyping alternatives

Since the terms of λ^{vec} are not explicitly typed, we are bound to have sequents such as $\Gamma \vdash \mathbf{t} : T_1$ and $\Gamma \vdash \mathbf{t} : T_2$ with distinct types T_1 and T_2 for the same term $\mathbf{t}$. Using Rules $(+_I)$ and (α_I) we get the valid typing judgement $\Gamma \vdash \alpha \cdot \mathbf{t} + \beta \cdot \mathbf{t} : \alpha \cdot T_1 + \beta \cdot T_2$. Given that $\alpha \cdot \mathbf{t} + \beta \cdot \mathbf{t}$ reduces to $(\alpha + \beta) \cdot \mathbf{t}$, a regular subject reduction would ask for the valid sequent $\Gamma \vdash (\alpha + \beta) \cdot \mathbf{t} : \alpha \cdot T_1 + \beta \cdot T_2$. But since in general we do not have $\alpha \cdot T_1 + \beta \cdot T_2 \equiv (\alpha + \beta) \cdot T_1 \equiv (\alpha + \beta) \cdot T_2$, we need to find a way around this.

A first approach would be to use the notion of principal types. However, since our type system includes *System F*, the usual examples for the absence of principal types apply to our settings: we cannot rely upon this method.

A second approach would be to ask for the sequent $\Gamma \vdash (\alpha + \beta) \cdot \mathbf{t} : \alpha \cdot T_1 + \beta \cdot T_2$ to be valid. If we force this typing rule into the system, it seems to solve the issue but then the type of a term becomes pretty much arbitrary: with typing context Γ, the term $(\alpha+\beta)\cdot\mathbf{t}$ would then be typed with any combination $\gamma \cdot T_1 + \delta \cdot T_2$, where $\alpha+\beta = \gamma+\delta$.

The approach we favour in this paper is via a notion of order on types. The order, denoted with $\sqsupseteq$, will be chosen so that the factorisation rules make the types of terms smaller. We will ask in particular that $(\alpha + \beta) \cdot T_1 \sqsupseteq \alpha \cdot T_1 + \beta \cdot T_2$ and $(\alpha + \beta) \cdot T_2 \sqsupseteq \alpha \cdot T_1 + \beta \cdot T_2$ whenever T_1 and T_2 are types for the same term. This approach can also be extended to solve a second pitfall coming from the rule $\mathbf{t} + \mathbf{0} \to \mathbf{t}$. Indeed, although $x : \mathcal{X} \vdash x + \mathbf{0} : \mathcal{X} + 0 \cdot T$ is well-typed for any inhabited T, the sequent $x : \mathcal{X} \vdash x : \mathcal{X} + 0 \cdot T$ is not valid in general. We therefore extend the ordering to also have $\mathcal{X} \sqsupseteq \mathcal{X} + 0 \cdot T$.

Notice that we are not introducing a subtyping relation with this ordering. For example, although $\vdash (\alpha + \beta) \cdot \lambda x.\lambda y.x : (\alpha + \beta) \cdot \forall \mathcal{X}.\mathcal{X} \to (\mathcal{X} \to \mathcal{X})$ is valid and $(\alpha + \beta) \cdot \forall \mathcal{X}.\mathcal{X} \to (\mathcal{X} \to \mathcal{X}) \sqsupseteq \alpha \cdot \forall \mathcal{X}.\mathcal{X} \to (\mathcal{X} \to \mathcal{X}) + \beta \cdot \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to (\mathcal{Y} \to \mathcal{Y})$, the sequent $\vdash (\alpha + \beta) \cdot \lambda x.\lambda y.x : \alpha \cdot \forall \mathcal{X}.\mathcal{X} \to (\mathcal{X} \to \mathcal{X}) + \beta \cdot \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to (\mathcal{Y} \to \mathcal{Y})$ is not valid.

4.2. Weak subject reduction

We define the (antisymmetric) ordering relation $\sqsupseteq$ on types discussed above as the smallest reflexive transitive and congruent relation satisfying the rules:

1. $(\alpha+\beta)\cdot T \sqsupseteq \alpha\cdot T+\beta\cdot T'$ if there are $\Gamma, \mathbf{t}$ such that $\Gamma \vdash \alpha\cdot\mathbf{t} : \alpha\cdot T$ and $\Gamma \vdash \beta\cdot\mathbf{t} : \beta\cdot T'$.
2. $T \sqsupseteq T + 0.R$ for any type R.
3. If $T \sqsupseteq R$ and $U \sqsupseteq V$, then $T + S \sqsupseteq R + S$, $\alpha \cdot T \sqsupseteq \alpha \cdot R$, $U \to T \sqsupseteq U \to R$ and $\forall X.U \sqsupseteq \forall X.V$.

Note the fact that $\Gamma \vdash \mathbf{t} : T$ and $\Gamma \vdash \mathbf{t} : T'$ does not imply that $\beta \cdot T \sqsupseteq \beta \cdot T'$. For instance, although $\beta \cdot T \sqsupseteq 0 \cdot T + \beta \cdot T'$, we do not have $0 \cdot T + \beta \cdot T' \equiv \beta \cdot T'$.

Let R be any reduction rule from Figure 1, and $\to_R$ a one-step reduction by rule R. A weak version of the subject reduction theorem can be stated as follows.

Theorem 4.1 (Weak subject reduction)**.** *For any terms $\mathbf{t}$, $\mathbf{t}'$, any context Γ and any type T, if $\mathbf{t} \to_R \mathbf{t}'$ and $\Gamma \vdash \mathbf{t} : T$, then:*

1. *if $R \notin$ Group F, then $\Gamma \vdash \mathbf{t}' : T$;*
2. *if $R \in$ Group F, then $\exists S \sqsupseteq T$ such that $\Gamma \vdash \mathbf{t}' : S$ and $\Gamma \vdash \mathbf{t} : S$.*

4.3. Prerequisites to the proof

The proof of Theorem 4.1 requires some machinery that we develop in this section. Omitted proofs can be found in Appendix A.1.

The following lemma gives a characterisation of types as linear combinations of unit types and general variables.

Lemma 4.2 (Characterisation of types)**.** *For any type T in $\mathcal{G}$, there exist $n, m \in \mathbb{N}$, $\alpha_1, \ldots, \alpha_n$, $\beta_1, \ldots, \beta_m \in \mathsf{S}$, distinct unit types $U_1, \ldots, U_n$ and distinct general variables $\mathbb{X}_1, \ldots, \mathbb{X}_m$ such that*

$$T \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i + \sum_{j=1}^{m} \beta_j \cdot \mathbb{X}_j \ . \qquad \square$$

Our system admits weakening, as stated by the following lemma.

Lemma 4.3 (Weakening)**.** *Let $\mathbf{t}$ be such that $x \notin FV(\mathbf{t})$. Then $\Gamma \vdash \mathbf{t} : T$ is derivable if and only if $\Gamma, x : U \vdash \mathbf{t} : T$ is derivable.*

Proof. By a straightforward induction on the type derivation. $\square$

Lemma 4.4 (Equivalence between sums of distinct elements (up to $\equiv$))**.** *Let $U_1, \dots, U_n$ be a set of distinct (not equivalent) unit types, and let $V_1, \dots, V_m$ be also a set distinct unit types. If $\sum_{i=1}^{n} \alpha_i \cdot U_i \equiv \sum_{j=1}^{m} \beta_j \cdot V_j$, then $m = n$ and there exists a permutation p of m such that $\forall i$, $\alpha_i = \beta_{p(i)}$ and $U_i \equiv V_{p(i)}$.* $\square$

Lemma 4.5 (Equivalences $\forall_I$)**.**

1. $\sum_{i=1}^{n} \alpha_i \cdot U_i \equiv \sum_{j=1}^{m} \beta_j \cdot V_j \Leftrightarrow \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^{m} \beta_j \cdot \forall X.V_j$.
2. $\sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^{m} \beta_j \cdot V_j \Rightarrow \forall V_j, \exists W_j \ / \ V_j \equiv \forall X.W_j$.
3. $T \equiv R \Rightarrow T[A/X] \equiv R[A/X]$. $\square$

For the proof of subject reduction, we use the standard strategy developed by Barendregth [10][1]. It consists in defining a relation betwen types of the form $\forall X.T$ and T. For our vectorial type system, we take into account linear combinations of types

Definition 4.6. *For any types T, R, any context Γ and any term $\mathbf{t}$ such that*

$$\cfrac{\Gamma \vdash \mathbf{t} : T}{\cfrac{\vdots}{\Gamma \vdash \mathbf{t} : R}}$$

1. *if $X \notin FV(\Gamma)$, write $T \succ_{X,\Gamma}^{\mathbf{t}} R$ if either*
   - *$T \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i$ and $R \equiv \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i$, or*
   - *$T \equiv \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i$ and $R \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i[A/X]$.*
2. *if $\mathcal{V}$ is a set of type variables such that $\mathcal{V} \cap FV(\Gamma) = \emptyset$, we define $\succeq_{\mathcal{V},\Gamma}^{\mathbf{t}}$ inductively by*
   - *If $X \in \mathcal{V}$ and $T \succ_{X,\Gamma}^{\mathbf{t}} R$, then $T \succeq_{\{X\},\Gamma}^{\mathbf{t}} R$.*
   - *If $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathcal{V}$, $T \succeq_{\mathcal{V}_1,\Gamma}^{\mathbf{t}} R$ and $R \succeq_{\mathcal{V}_2,\Gamma}^{\mathbf{t}} S$, then $T \succeq_{\mathcal{V}_1 \cup \mathcal{V}_2,\Gamma}^{\mathbf{t}} S$.*
   - *If $T \equiv R$, then $T \succeq_{\mathcal{V},\Gamma}^{\mathbf{t}} R$.*

Example 4.7. *Let the following be a valid derivation.*

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash \mathbf{t} : T \qquad T \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i \qquad \mathcal{X} \notin FV(\Gamma)}\equiv}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \mathcal{X}.U_i}\forall_I}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i[V/\mathcal{X}] \qquad \mathbb{Y} \notin FV(\Gamma)}\forall_{\mathcal{E}}}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \mathbb{Y}.U_i[V/\mathcal{X}] \qquad \sum_{i=1}^{n} \alpha_i \cdot \forall \mathbb{Y}.U_i[V/\mathcal{X}] \equiv R}\forall_I}{\Gamma \vdash \mathbf{t} : R}\equiv$$

[1] Note that Barendregth's original proof contains a mistake [20]. We use the corrected proof proposed in [4]

Then $T \succeq^{\mathbf{t}}_{\{x,\mathbb{Y}\},\Gamma} R$.

Note that this relation is stable under reduction in the following way:

Lemma 4.8 ($\succeq$-stability)**.** *If* $T \succeq^{\mathbf{t}}_{\mathcal{V},\Gamma} R$*,* $\mathbf{t} \to \mathbf{r}$ *and* $\Gamma \vdash \mathbf{r} : T$*, then* $T \succeq^{\mathbf{r}}_{\mathcal{V},\Gamma} R$*.* □

The following lemma states that if two arrow types are ordered, then they are equivalent up to some substitutions.

Lemma 4.9 (Arrows comparison)**.** *If* $V \to R \succeq^{\mathbf{t}}_{\mathcal{V},\Gamma} \forall \vec{X}.(U \to T)$*, then* $U \to T \equiv (V \to R)[\vec{A}/\vec{Y}]$*, with* $\vec{Y} \notin FV(\Gamma)$*.*

Before proving Theorem 4.1, we need to prove some basic properties of the system.

Lemma 4.10 (Scalars)**.** *For any context* Γ*, term* $\mathbf{t}$*, type* T *and scalar* α*, if* $\Gamma \vdash \alpha \cdot \mathbf{t} : T$*, then there exists a type* R *such that* $T \equiv \alpha \cdot R$ *and* $\Gamma \vdash \mathbf{t} : R$*. Moreover, if the minimum size of the derivation of* $\Gamma \vdash \alpha \cdot \mathbf{t} : T$ *is* s*, then if* $T = \alpha \cdot R$*, the minimum size of the derivation of* $\Gamma \vdash \mathbf{t} : R$ *is at most* $s - 1$*, in other case, its minimum size is at most* $s - 2$*.* □

The following lemma shows that the type for $\mathbf{0}$ is always $0 \cdot T$.

Lemma 4.11 (Type for zero)**.** *Let* $\mathbf{t} = \mathbf{0}$ *or* $\mathbf{t} = \alpha \cdot \mathbf{0}$*, then* $\Gamma \vdash \mathbf{t} : T$ *implies* $T \equiv 0 \cdot R$*.* □

Lemma 4.12 (Sums)**.** *If* $\Gamma \vdash \mathbf{t} + \mathbf{r} : S$*, then* $S \equiv T + R$ *with* $\Gamma \vdash \mathbf{t} : T$ *and* $\Gamma \vdash \mathbf{r} : R$*. Moreover, if the size of the derivation of* $\Gamma \vdash \mathbf{t} + \mathbf{r} : S$ *is* s*, then if* $S = T + R$*, the minimum sizes of the derivations of* $\Gamma \vdash \mathbf{t} : T$ *and* $\Gamma \vdash \mathbf{r} : R$ *are at most* $s - 1$*, and if* $S \neq T + R$*, the minimum sizes of these derivations are at most* $s - 2$*.* □

Lemma 4.13 (Applications)**.** *If* $\Gamma \vdash (\mathbf{t})\ \mathbf{r} : T$*, then* $\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i)$ *and* $\Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$ *where* $\sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{(\mathbf{t})\mathbf{r}}_{\mathcal{V},\Gamma} T$ *for some* $\mathcal{V}$*.* □

Lemma 4.14 (Abstractions)**.** *If* $\Gamma \vdash \lambda x.\mathbf{t} : T$*, then* $\Gamma, x : U \vdash \mathbf{t} : R$ *where* $U \to R \succeq^{\lambda x.\mathbf{t}}_{\mathcal{V},\Gamma} T$ *for some* $\mathcal{V}$*.* □

A basis term can always be given a unit type.

Lemma 4.15 (Basis terms)**.** *For any context* Γ*, type* T *and basis term* $\mathbf{b}$*, if* $\Gamma \vdash \mathbf{b} : T$ *then there exists a unit type* U *such that* $T \equiv U$*.* □

The final stone for the proof of Theorem 4.1 is a lemma relating well-typed terms and substitution.

Lemma 4.16 (Substitution lemma)**.** *For any term* $\mathbf{t}$*, basis term* $\mathbf{b}$*, term variable* x*, context* Γ*, types* T*,* U*, type variable* X *and type* A*, where* A *is a unit type if* X *is a unit variables, otherwise* A *is a general type, we have,*

1. *if* $\Gamma \vdash \mathbf{t} : T$*, then* $\Gamma[A/X] \vdash \mathbf{t} : T[A/X]$*;*
2. *if* $\Gamma, x : U \vdash \mathbf{t} : T$*,* $\Gamma \vdash \mathbf{b} : U$ *then* $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : T$*.* □

The proof of subject reduction (Theorem 4.1), follows by induction using the previous defined lemmas. It can be foun in full details in Appendix A.2.

5. Strong normalisation

For proving strong normalisation of well-typed terms, we use reducibility candidates, a well-known method described for example in [29, Ch. 14]. The technique is adapted to linear combinations of terms. Omitted proofs in this section can be found in Appendix B.

A *neutral term* is a term that is not a λ-abstraction and that does reduce to something. The set of *closed neutral terms* is denoted with $\mathcal{N}$. We write Λ_0 for the set of closed terms and SN_0 for the set of closed, strongly normalising terms. If $\mathbf{t}$ is any term, $\mathrm{Red}(\mathbf{t})$ is the set of all terms $\mathbf{t}'$ such that $\mathbf{t} \to \mathbf{t}'$. It is naturally extended to sets of terms. We say that a set S of closed terms is a reducibility candidate, denoted with $S \in \mathsf{RC}$ if the following conditions are verified:

$\mathbf{RC}_1$ Strong normalisation: $S \subseteq SN_0$.

$\mathbf{RC}_2$ Stability under reduction: $\mathbf{t} \in S$ implies $\mathrm{Red}(\mathbf{t}) \subseteq S$.

$\mathbf{RC}_3$ Stability under neutral expansion: If $\mathbf{t} \in \mathcal{N}$ and $\mathrm{Red}(\mathbf{t}) \subseteq S$ then $\mathbf{t} \in S$.

$\mathbf{RC}_4$ The common inhabitant: $\mathbf{0} \in S$.

We define the notion of *algebraic context* over a list of terms $\vec{\mathbf{t}}$, with the following grammar:

$$F(\vec{\mathbf{t}}), G(\vec{\mathbf{t}}) \quad ::= \quad \mathbf{t}_i \mid F(\vec{\mathbf{t}}) + G(\vec{\mathbf{t}}) \mid \alpha \cdot F(\vec{\mathbf{t}}) \mid \mathbf{0},$$

where $\mathbf{t}_i$ is the i-th element of the list $\mathbf{t}$. Given a set of terms $S = \{\mathbf{s}_i\}_i$, we write $\mathcal{F}(S)$ for the set of terms of the form $F(\vec{\mathbf{s}})$ when F spans over algebraic contexts.

We introduce two conditions on contexts, which will be handy to define some of the operations on candidates:

$\mathbf{CC}_1$ If for some F, $F(\vec{\mathbf{s}}) \in S$ then $\forall i, \mathbf{s}_i \in S$.

$\mathbf{CC}_2$ If for all i, $\mathbf{s}_i \in S$ and F is an algebraic context, then $F(\vec{\mathbf{s}}) \in S$.

We then define the following operations on reducibility candidates.

1. Let A and B be in RC. $\mathsf{A} \to \mathsf{B}$ is the closure under $\mathbf{RC}_3$ and $\mathbf{RC}_4$ of the set of $\mathbf{t} \in \Lambda_0$ such that $(\mathbf{t})\,\mathbf{0} \in \mathsf{B}$ and such that for all base terms $\mathbf{b} \in \mathsf{A}$, $(\mathbf{t})\,\mathbf{b} \in \mathsf{B}$.
2. If $\{\mathsf{A}_i\}_i$ is a family of reducibility candidates, $\sum_i \mathsf{A_i}$ is the closure under $\mathbf{CC}_1$, $\mathbf{CC}_2$, $\mathbf{RC}_2$ and $\mathbf{RC}_3$ of the set $\cup_i \mathsf{A}_i$.

Remark 5.1. Notice that $\sum_{i=1}^{1} \mathsf{A} \neq \mathsf{A}$. Indeed, $\sum_{i=1}^{1} \mathsf{A}$ is in particular the closure over $\mathbf{CC}_2$, meaning that all linear combinations of terms of A belongs to $\sum_{i=1}^{1} \mathsf{A}$, whereas they might not be in A.

Remark 5.2. In the definition of algebraic contexts, a term t_i might appear at several positions in the context. However, for any given algebraic context $F(\vec{\mathbf{t}})$ there is always a *linear* algebraic context $F_l(\vec{\mathbf{t}}')$ with a suitably modified list of terms $\vec{\mathbf{t}}'$ such that $F(\vec{\mathbf{t}})$ and $F_l(\vec{\mathbf{t}}')$ are the same terms. For example, choose the following (arguably very verbose) construction: if F contains m placeholders, and if $\vec{\mathbf{t}}$ is of size n, let $\vec{\mathbf{t}}'$ be the list $\mathbf{t}_1 \ldots \mathbf{t}_1, \mathbf{t}_2 \ldots \mathbf{t}_2, \ldots, \mathbf{t}_n \ldots \mathbf{t}_n$ with each time m repetitions of each $\mathbf{t}_i$. Then construct $F_l(\vec{\mathbf{t}}')$ exactly as F except that for each ith placeholder we pick the ith copy of the corresponding term in F. By construction, each element in the list $\vec{\mathbf{t}}'$ is used at most once, and the term $F_l(\vec{\mathbf{t}}')$ is the same as the term $F(\vec{\mathbf{t}})$.

Lemma 5.3. *If* A, B *and all the* A_i*'s are in* RC*, then so are* $\mathsf{A} \to \mathsf{B}$, $\sum_i \mathsf{A}_i$ *and* $\cap_i \mathsf{A}_i$. $\square$

A *single type valuation* is a partial function from type variables to reducibility candidates, that we define as a sequence of comma-separated mappings, with $\emptyset$ denoting the empty valuation: $\rho := \emptyset \mid \rho, X \mapsto \mathsf{A}$. Type variables are interpreted using pairs of single type valuations, that we simply call *valuations*, with common domain: $\rho = (\rho_+, \rho_-)$ with $|\rho_+| = |\rho_-|$. Given a valuation $\rho = (\rho_+, \rho_-)$, the *complementary valuation* $\bar{\rho}$ is the pair (ρ_-, ρ_+). We write $(X_+, X_-) \mapsto (A_+, A_-)$ for the valuation $(X_+ \mapsto A_+, X_- \mapsto A_-)$. A valuation is called *valid* if for all X, $\rho_-(X) \subseteq \rho_+(X)$.

From now on, we will consider the following grammar

$$\mathbb{U}, \mathbb{V}, \mathbb{W} ::= U \mid \mathbb{X}.$$

That is, we will use $\mathbb{U}, \mathbb{V}, \mathbb{W}$ for unit and $\mathbb{X}$-kind of variables.

To define the interpretation of a type T, we use the following result.

Lemma 5.4. *Any type* T*, has a unique (up to* $\equiv$*) canonical decomposition* $T \equiv \sum_{i=1}^n \alpha_i \cdot \mathbb{U}_i$ *such that for all* l, k, $\mathbb{U}_l \not\equiv \mathbb{U}_k$. $\square$

The interpretation $[\![T]\!]_\rho$ of a type T in a valuation $\rho = (\rho_+, \rho_-)$ defined for each free type variable of T is given by:

$$\begin{array}{rcl} [\![X]\!]_\rho &=& \rho_+(X), \\ [\![U \to T]\!]_\rho &=& [\![U]\!]_{\bar{\rho}} \to [\![T]\!]_\rho, \\ [\![\forall X.U]\!]_\rho &=& \cap_{\mathsf{A} \in \mathsf{RC}} [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}, \\ \multicolumn{3}{l}{\text{If } T \equiv \sum_i \alpha_i \cdot \mathbb{U}_i \text{ is the canonical decomposition of } T \text{ and } T \not\equiv \mathbb{U}} \\ [\![T]\!]_\rho &=& \sum_i [\![\mathbb{U}_i]\!]_\rho \end{array}$$

From Lemma 5.3, the interpretation of any type is a reducibility candidate.

Reducibility candidates deal with closed terms, whereas proving the adequacy lemma by induction requires the use of open terms with some assumptions on their free variables, that will be guaranteed by a context. Therefore we use *substitutions* σ to close terms:

$$\sigma := \emptyset \mid (x \mapsto \mathbf{b}; \sigma) \ ,$$

then $\mathbf{t}_\emptyset = \mathbf{t}$ and $\mathbf{t}_{x \mapsto \mathbf{b};\sigma} = \mathbf{t}[\mathbf{b}/x]_\sigma$. All the substitutions ends by $\emptyset$, hence we omit it when not necessary.

Given a context Γ, we say that a substitution σ *satisfies* Γ for the valuation ρ (notation: $\sigma \in [\![\Gamma]\!]_\rho$) when $(x : U) \in \Gamma$ implies $x_\sigma \in [\![U]\!]_{\bar{\rho}}$ (Note the change in polarity). A typing judgement $\Gamma \vdash \mathbf{t} : T$, is said to be *valid* (notation $\Gamma \models \mathbf{t} : T$) if

- in case $T \equiv \mathbb{U}$, then for every valuation ρ, and for every substitution $\sigma \in [\![\Gamma]\!]_\rho$, we have $\mathbf{t}_\sigma \in [\![\mathbb{U}]\!]_\rho$.
- in other case, that is, $T \equiv \sum_{i=1}^n \alpha_i \cdot \mathbb{U}_i$ with $n > 1$, such that for all i, j, $\mathbb{U}_i \not\equiv \mathbb{U}_j$ (notice that by Lemma 5.4 such a decomposition always exists), then for every valuation ρ, and set of valuations $\{\rho_i\}_n$, where ρ_i acts on $FV(U_i) \setminus FV(\Gamma)$, and for every substitution $\sigma \in [\![\Gamma]\!]_\rho$, we have $\mathbf{t}_\sigma \in \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho,\rho_i}$.

Lemma 5.5. *For any types* T *and* A*, variable* X *and valuation* ρ*, we have* $[\![T[A/X]]\!]_\rho = [\![T]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)}$ *and* $[\![T[A/X]]\!]_{\bar{\rho}} = [\![T]\!]_{\bar{\rho},(X_-,X_+)\mapsto([\![A]\!]_\rho,[\![A]\!]_{\bar{\rho}})}$. $\square$

The proof of the Adequacy Lemma as well as the machinery of needed auxiliary lemmas can be found in Appendix B.2.

Lemma 5.6 (Adequacy Lemma)**.** *Every derivable typing judgement is valid: For every valid sequent* $\Gamma \vdash \mathbf{t} : T$*, we have* $\Gamma \models \mathbf{t} : T$*.* □

Theorem 5.7 (Strong normalisation)**.** *If* $\Gamma \vdash \mathbf{t} : T$ *is a valid sequent, then* $\mathbf{t}$ *is strongly normalising.*

Proof. If Γ is the list $(x_i : U_i)_i$, the sequent $\vdash \lambda x_1 \dots x_n.\mathbf{t} : U_1 \to (\dots \to (U_n \to T) \cdots)$ is derivable. Using Lemma 5.6, we deduce that for any valuation ρ and any substitution $\sigma \in [\![\emptyset]\!]_\rho$, we have $\lambda x_1 \dots x_n.\mathbf{t}_\sigma \in [\![T]\!]_\rho$. By construction, σ does nothing on $\mathbf{t}$: $\mathbf{t}_\sigma = \mathbf{t}$. Since $[\![T]\!]_\rho$ is a reducibility candidate, $\lambda x_1 \dots x_n.\mathbf{t}$ is strongly normalising and hence $\mathbf{t}$ is strongly normalising. □

6. Interpretation of typing judgements

6.1. The general case

In the general case the calculus can represent infinite-dimensional linear operators such as $\lambda x.x$, $\lambda x.\lambda y.y$, $\lambda x.\lambda f.(f)\,x$,... and their applications. Even for such general terms $\mathbf{t}$, the vectorial type system provides much information about the superposition of basis terms $\sum_i \alpha_i \cdot \mathbf{b}_i$ to which $\mathbf{t}$ reduces, as explained in Theorem 6.1. How much information is brought by the type system in the finitary case is the topic of Section 6.2.

Theorem 6.1 (Characterisation of terms)**.** *Let* T *be a generic type with canonical decomposition* $\sum_{i=1}^n \alpha_i.\mathbb{U}_i$*, in the sense of Lemma 5.4. If* $\vdash \mathbf{t} : T$*, then* $\mathbf{t} \to^* \sum_{i=1}^n \sum_{j=1}^{m_i} \beta_{ij} \cdot \mathbf{b}_{ij}$*, where for all* i*,* $\vdash \mathbf{b}_{ij} : \mathbb{U}_i$ *and* $\sum_{j=1}^{m_i} \beta_{ij} = \alpha_i$*, and with the convention that* $\sum_{j=1}^{0} \beta_{ij} = 0$ *and* $\sum_{j=1}^{0} \beta_{ij} \cdot \mathbf{b}_{ij} = \mathbf{0}$*.* □

The detailed proof of the previous theorem can be found in Appendix C

6.2. The finitary case: Expressing matrices and vectors

In what we call the "finitary case", we show how to encode finite-dimensional linear operators, i.e. matrices, together with their applications to vectors, as well as matrix and tensor products. Theorem 6.2 shows that we can encode matrices, vectors and operations upon them, and the type system will provide the result of such operations.

6.2.1. In 2 dimensions

In this section we come back to the motivating example introducing the type system and we show how λ^{vec} handles the Hadamard gate, and how to encode matrices and vectors.

With an empty typing context, the booleans $\mathbf{true} = \lambda x.\lambda y.x$ and $\mathbf{false} = \lambda x.\lambda y.y$ can be respectively typed with the types $\mathcal{T} = \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to (\mathcal{Y} \to \mathcal{X})$ and $\mathcal{F} = \forall \mathcal{X}\mathcal{Y}.\mathcal{X} \to (\mathcal{Y} \to \mathcal{Y})$. The superposition has the following type $\vdash \alpha \cdot \mathbf{true} + \beta \cdot \mathbf{false} : \alpha \cdot \mathcal{T} + \beta \cdot \mathcal{F}$. (Note that it can also be typed with $(\alpha + \beta) \cdot \forall \mathcal{X}.\mathcal{X} \to \mathcal{X} \to \mathcal{X}$).

The linear map $\mathbf{U}$ sending $\mathbf{true}$ to $a \cdot \mathbf{true} + b \cdot \mathbf{false}$ and $\mathbf{false}$ to $c \cdot \mathbf{true} + d \cdot \mathbf{false}$, that is

$$\mathbf{true} \mapsto a \cdot \mathbf{true} + b \cdot \mathbf{false},$$

$$\mathbf{false} \mapsto c \cdot \mathbf{true} + d \cdot \mathbf{false}$$

is written as

$$\mathbf{U} = \lambda x.\{((x)[a \cdot \mathbf{true} + b \cdot \mathbf{false}])[c \cdot \mathbf{true} + d \cdot \mathbf{false}]\}.$$

The following sequent is valid:

$$\vdash \mathbf{U} : \forall \mathbb{X}.((I \to (a \cdot \mathcal{T} + b \cdot \mathcal{F})) \to (I \to (c \cdot \mathcal{T} + d \cdot \mathcal{F})) \to I \to \mathbb{X}) \to \mathbb{X}.$$

This is consistent with the discussion in the introduction: the Hadamard gate is the case $a = b = c = \frac{1}{\sqrt{2}}$ and $d = -\frac{1}{\sqrt{2}}$. One can check that with an empty typing context, $(\mathbf{U})$ $\mathbf{true}$ is well typed of type $a \cdot \mathcal{T} + b \cdot \mathcal{F}$, as expected since it reduces to $a \cdot \mathbf{true} + b \cdot \mathbf{false}$.

The term $(\mathbf{H})$ $\frac{1}{\sqrt{2}} \cdot (\mathbf{true} + \mathbf{false})$ is well-typed of type $\mathcal{T} + 0 \cdot \mathcal{F}$. Since the term reduces to $\mathbf{true}$, this is consistent with the subject reduction: we indeed have $\mathcal{T} \sqsupseteq \mathcal{T} + 0 \cdot \mathcal{F}$.

But we can do more than typing 2-dimensional vectors 2×2-matrices: using the same technique we can encode vectors and matrices of any size.

6.2.2. Vectors in n dimensions

The 2-dimensional space is represented by the span of $\lambda x_1 x_2.x_1$ and $\lambda x_1 x_2.x_2$: the n-dimensional space is simply represented by the span of all the $\lambda x_1 \cdots x_n.x_i$, for $i = 1 \cdots n$. As for the two dimensional case where

$$\vdash \alpha_1 \cdot \lambda x_1 x_2.x_1 + \alpha_2 \cdot \lambda x_1 x_2.x_2 : \alpha_1 \cdot \forall X_1 X_2.X_1 + \alpha_2 \cdot \forall X_1 X_2.X_2,$$

an n-dimensional vector is typed with

$$\vdash \sum_{i=1}^{n} \alpha_i \cdot \lambda x_1 \cdots x_n.x_i : \sum_{i=1}^{n} \alpha_i \cdot \forall X_1 \cdots X_n.X_i.$$

We use the notations

$$\mathbf{e}_i^n = \lambda x_1 \cdots x_n.x_i, \qquad \mathbf{E}_i^n = \forall X_1 \cdots X_n.X_i$$

and we write

$$\left[\!\!\left[\begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_n \end{array} \right]\!\!\right]_n^{\text{term}} = \left(\begin{array}{c} \alpha_1 \cdot \mathbf{e}_1^n \\ + \\ \cdots \\ + \\ \alpha_n \cdot \mathbf{e}_n^n \end{array} \right) = \sum_{i=1}^{n} \alpha_i \cdot \mathbf{e}_i^n,$$

$$\left[\!\!\left[\begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_n \end{array} \right]\!\!\right]_n^{\text{type}} = \left(\begin{array}{c} \alpha_1 \cdot \mathbf{E}_1^n \\ + \\ \cdots \\ + \\ \alpha_n \cdot \mathbf{E}_n^n \end{array} \right) = \sum_{i=1}^{n} \alpha_i \cdot \mathbf{E}_i^n.$$

6.2.3. $n \times m$ matrices

Once the representation of vectors is chosen, it is easy to generalize the representation of 2×2 matrices to the $n \times m$ case. Suppose that the matrix U is of the form

$$U = \begin{pmatrix} \alpha_{11} & \cdots & \alpha_{1m} \\ \vdots & & \vdots \\ \alpha_{n1} & \cdots & \alpha_{nm} \end{pmatrix},$$

then its representation is

$$[\![U]\!]^{\text{term}}_{n\times m} = \quad \lambda x. \left\{ \left(\cdots \left((x) \begin{bmatrix} \alpha_{11} \cdot \mathbf{e}_1^n \\ + \\ \cdots \\ + \\ \alpha_{n1} \cdot \mathbf{e}_n^n \end{bmatrix} \right) \cdots \begin{bmatrix} \alpha_{1m} \cdot \mathbf{e}_1^n \\ + \\ \cdots \\ + \\ \alpha_{nm} \cdot \mathbf{e}_n^n \end{bmatrix} \right) \right\}$$

and its type is

$$[\![U]\!]^{\text{type}}_{n\times m} = \quad \forall \mathbb{X}. \left(\begin{bmatrix} \alpha_{11} \cdot \mathbf{E}_1^n \\ + \\ \cdots \\ + \\ \alpha_{n1} \cdot \mathbf{E}_n^n \end{bmatrix} \to \cdots \to \begin{bmatrix} \alpha_{1m} \cdot \mathbf{E}_1^n \\ + \\ \cdots \\ + \\ \alpha_{nm} \cdot \mathbf{E}_n^n \end{bmatrix} \to [\ \mathbb{X}\ ] \right) \to \mathbb{X},$$

that is, an almost direct encoding of the matrix U.

We also use the shortcut notation

$$\mathbf{mat}(\mathbf{t}_1, \ldots, \mathbf{t}_n) = \lambda x.(\ldots((x)\,[\mathbf{t}_1])\ldots)\,[\mathbf{t}_n]$$

6.2.4. Useful constructions

In this section, we describe a few terms representing constructions that will be used later on.

Projections. The first useful family of terms are the projections, sending a vector to its i^{th} coordinate:

$$\begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_i \\ \vdots \\ \alpha_n \end{pmatrix} \longmapsto \begin{pmatrix} 0 \\ \vdots \\ \alpha_i \\ \vdots \\ 0 \end{pmatrix}.$$

Using the matrix representation, the term projecting the i^{th} coordinate of a vector of size n is

$$\mathbf{p}_i^n = \mathbf{mat}(\mathbf{0}, \cdots, \mathbf{0},\ \overset{i^{\text{th}}\text{ position}}{\mathbf{e}_i^n},\ \mathbf{0}, \cdots, \mathbf{0}).$$

We can easily verify that

$$\vdash \mathbf{p}_i^n : \left[\!\!\left[\begin{matrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & & & \vdots \\ 0 & & 1 & & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{matrix} \right]\!\!\right]_{n\times n}^{\text{type}}$$

and that

$$(\mathbf{p}_{i_0}^n)\left(\sum_{i=1}^{n} \alpha_i \cdot \mathbf{e}_i^n\right) \longrightarrow^* \alpha_{i_0} \cdot \mathbf{e}_{i_0}^n.$$

Vectors and diagonal matrices. Using the projections defined in the previous section, it is possible to encode the map sending a vector of size n to the corresponding $n \times n$ matrix:

$$\begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} \longmapsto \begin{pmatrix} \alpha_1 & & 0 \\ & \ddots & \\ 0 & & \alpha_n \end{pmatrix}$$

with the term

$$\mathbf{diag}^n = \lambda b.\mathbf{mat}((\mathbf{p}_1^n)\,\{b\}, \ldots, (\mathbf{p}_n^n)\,\{b\})$$

of type

$$\vdash \mathbf{diag}^n : \left[\left[\!\!\left[\begin{matrix} \alpha_1 \\ \vdots \\ \alpha_n \end{matrix} \right]\!\!\right]_n^{\text{type}} \right] \to \left[\!\!\left[\begin{matrix} \alpha_1 & & 0 \\ & \ddots & \\ 0 & & \alpha_n \end{matrix} \right]\!\!\right]_{n\times n}^{\text{type}}.$$

It is easy to check that

$$(\mathbf{diag}^n)\left[\sum_{i=1}^{n} \alpha_i \cdot \mathbf{e}_i^n\right] \longmapsto^* \mathbf{mat}(\alpha_1 \cdot \mathbf{e}_1^n, \ldots, \alpha_n \cdot \mathbf{e}_n^n)$$

Extracting a column vector out of a matrix. Another construction that is worth exhibiting is the operation

$$\begin{pmatrix} \alpha_{11} & \cdots & \alpha_{1n} \\ \vdots & & \vdots \\ \alpha_{m1} & \cdots & \alpha_{mn} \end{pmatrix} \longmapsto \begin{pmatrix} \alpha_{1i} \\ \vdots \\ \alpha_{mi} \end{pmatrix}.$$

It is simply defined by multiplying the input matrix with the i^{th} base column vector:

$$\mathbf{col}_i^n = \lambda x.(x)\,\mathbf{e}_i^n$$

and one can easily check that this term has type

$$\vdash \mathbf{col}_i^n : \left[\!\!\left[\begin{matrix} \alpha_{11} & \cdots & \alpha_{1n} \\ \vdots & & \vdots \\ \alpha_{m1} & \cdots & \alpha_{mn} \end{matrix} \right]\!\!\right]_{m\times n}^{\text{type}} \to \left[\!\!\left[\begin{matrix} \alpha_{1i} \\ \vdots \\ \alpha_{mi} \end{matrix} \right]\!\!\right]_m^{\text{type}}.$$

Note that the same term $\mathbf{col}_i^n$ can be typed with several values of m.

6.2.5. A language of matrices and vectors

In this section we formalize what was informally presented in the previous sections: the fact that one can encode simple matrix and vector operations in λ^{vec}, and the fact that the type system serves as a witness for the result of the encoded operation.

We define the language *Mat* of matrices and vectors with the grammar

$$\begin{array}{ll} M, N & ::= \zeta \mid M \otimes N \mid (M)\,N \\ u, v & ::= \nu \mid u \otimes v \mid (M)\,u, \end{array}$$

where ζ ranges over the set matrices and ν over the set of (column) vectors. Terms are implicitly typed: types of matrices are (m, n) where m and n ranges over positive integers, while types of vectors are simply integers. Typing rules are the following.

$$\frac{\zeta \in \mathbb{C}^{m\times n}}{\zeta : (m,n)} \qquad \frac{M : (m,n) \quad N : (m',n')}{M \otimes N : (mm', nn')} \qquad \frac{M : (m,n') \quad N : (n',n)}{(M)\,N : (m,n)}$$

$$\frac{\nu \in \mathbb{C}^{m}}{\nu : m} \qquad \frac{u : m \quad v : n}{u \otimes v : mn} \qquad \frac{M : (m,n) \quad u : m}{(M)\,u : n}$$

The operational semantics of this language is the natural interpretation of the terms as matrices and vectors. If M computes the matrix ζ, we write $M \downarrow \zeta$. Similarly, if u computes the vector ν, we write $u \downarrow \nu$.

Following what we already said, matrices and vectors can be interpreted as types and terms in λ^{vec}. The map $[\![-]\!]^{\mathrm{term}}$ sends terms of *Mat* to terms of λ^{vec} and the map $[\![-]\!]^{\mathrm{type}}$ sends matrices and vectors to types of λ^{vec}.

- Vectors and matrices are defined as in Sections 6.2.2 and 6.2.3.
- As we already discussed, the matrix-vector multiplication is simply the application of terms in λ^{vec}:
$$[\![(M)\,u]\!]^{\mathrm{term}} = ([\![M]\!]^{\mathrm{term}})\,[\![u]\!]^{\mathrm{term}}$$
- The matrix multiplication is performed by first extracting the column vectors, then performing the matrix-vector multiplication: this gives a column of the final matrix. We conclude by recomposing the final matrix column-wise.

  That is done with the term
$$\mathbf{app} = \lambda xy.\mathbf{mat}((x)\,((\mathbf{col}_1^m)\,y), \ldots, (x)\,((\mathbf{col}_n^m)\,y))$$
  and its type is
$$\left[\!\!\left[\begin{array}{ccc} \alpha_{11} & \cdots & \alpha_{1n} \\ \vdots & & \vdots \\ \alpha_{m1} & \cdots & \alpha_{mn} \end{array}\right]\!\!\right]^{\mathrm{type}}_{m\times n} \to \left[\!\!\left[\begin{array}{ccc} \beta_{11} & \cdots & \beta_{1k} \\ \vdots & & \vdots \\ \beta_{n1} & \cdots & \beta_{nk} \end{array}\right]\!\!\right]^{\mathrm{type}}_{n\times k} \to \left[\!\!\left[(\textstyle\sum_{i=1}^{n} \alpha_{ji}\beta_{il})_{\substack{j=1\ldots m\\ l=1\ldots k}}\right]\!\!\right]^{\mathrm{type}}_{m\times k}$$
  Hence,
$$[\![(M)\ N]\!]^{\mathrm{term}} = ((\mathbf{app})\ [\![M]\!]^{\mathrm{term}})\ [\![N]\!]^{\mathrm{term}}$$

- For defining the the tensor of vectors, we need to multiply the coefficients of the vectors:

$$\begin{pmatrix}\alpha_1\\ \vdots\\ \alpha_n\end{pmatrix}\otimes\begin{pmatrix}\beta_1\\ \vdots\\ \beta_m\end{pmatrix}=\begin{pmatrix}\alpha_1\cdot\begin{pmatrix}\beta_1\\ \vdots\\ \beta_m\end{pmatrix}\\ \vdots\\ \alpha_n\cdot\begin{pmatrix}\beta_1\\ \vdots\\ \beta_m\end{pmatrix}\end{pmatrix}=\begin{pmatrix}\alpha_1\beta_1\\ \vdots\\ \alpha_1\beta_m\\ \vdots\\ \alpha_n\beta_1\\ \vdots\\ \alpha_n\beta_m\end{pmatrix}.$$

We perform this operation in several steps: First, we map the two vectors $(\alpha_i)_i$ and $(\beta_j)_j$ into matrices of size $mn \times mn$:

$$\begin{pmatrix}\alpha_1\\ \vdots\\ \alpha_n\end{pmatrix}\mapsto\begin{pmatrix}\alpha_1 &&&&&\\ &\ddots&&&&\\ &&\alpha_1&&&\\ &&&\ddots&&\\ &&&&\alpha_n&\\ &&&&&\ddots\\ &&&&&&\alpha_n\end{pmatrix}\begin{matrix}\left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\times m\\ \\ \left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\times m\end{matrix}\quad\text{and}\quad\begin{pmatrix}\beta_1\\ \vdots\\ \beta_m\end{pmatrix}\mapsto\begin{pmatrix}\beta_1 &&&&&\\ &\ddots&&&&\\ &&\beta_m&&&\\ &&&\ddots&&\\ &&&&\beta_1&\\ &&&&&\ddots\\ &&&&&&\beta_m\end{pmatrix}\left.\begin{matrix}\left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\\ \\ \left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\end{matrix}\right\}\times n.$$

These two operations can be represented as terms of λ^{vec} respectively as follows:

$$\mathbf{m}_1^{n,m}=\lambda b.\,\mathbf{mat}\begin{pmatrix}(\mathbf{p}_1^n)\{b\},\\ \vdots\\ (\mathbf{p}_1^n)\{b\},\\ \vdots\\ (\mathbf{p}_n^n)\{b\},\\ \vdots\\ (\mathbf{p}_n^n)\{b\}\end{pmatrix}\begin{matrix}\left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\times m\\ \\ \left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\times m\end{matrix}\quad\text{and}\quad\mathbf{m}_2^{m,n}=\lambda b.\,\mathbf{mat}\begin{pmatrix}(\mathbf{p}_1^m)\{b\},\\ \vdots\\ (\mathbf{p}_m^m)\{b\},\\ \vdots\\ (\mathbf{p}_1^m)\{b\},\\ \vdots\\ (\mathbf{p}_m^m)\{b\}\end{pmatrix}\left.\begin{matrix}\left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\\ \\ \left.\begin{matrix}\\ \\ \\ \end{matrix}\right\}\end{matrix}\right\}\times n.$$

It is now enough to multiply these two matrices together to retrieve the diagonal:

$$\begin{pmatrix}\alpha_1 &&&&&\\ &\ddots&&&&\\ &&\alpha_1&&&\\ &&&\ddots&&\\ &&&&\alpha_n&\\ &&&&&\ddots\\ &&&&&&\alpha_n\end{pmatrix}\begin{pmatrix}\beta_1 &&&&&\\ &\ddots&&&&\\ &&\beta_m&&&\\ &&&\ddots&&\\ &&&&\beta_1&\\ &&&&&\ddots\\ &&&&&&\beta_m\end{pmatrix}\begin{pmatrix}1\\ \vdots\\ 1\\ \vdots\\ 1\\ \vdots\\ 1\end{pmatrix}=\begin{pmatrix}\alpha_1\beta_1\\ \vdots\\ \alpha_1\beta_m\\ \vdots\\ \alpha_n\beta_1\\ \vdots\\ \alpha_n\beta_m\end{pmatrix}$$

and this can be implemented through matrix-vector multiplication:

$$\mathbf{tens}^{n,m}=\lambda bc.((\mathbf{m}_1^{n,m})\,b)\left(((\mathbf{m}_2^{m,n})\,c)\left(\sum_{i=1}^{mn}\mathbf{e}_i^n\right)\right).$$

Hence, if $u : n$ and $v : m$, we have

$$[\![u \otimes v]\!]^{\text{term}} = ((\mathbf{tens}^{n,m})\ [\![u]\!]^{\text{term}})\ [\![v]\!]^{\text{term}}$$

- The tensor of matrices is done column by column:

$$\begin{pmatrix} \alpha_{11} & \dots & \alpha_{1n} \\ \vdots & & \vdots \\ \alpha_{n'1} & \dots & \alpha_{n'n} \end{pmatrix} \otimes \begin{pmatrix} \beta_{11} & \dots & \beta_{1m} \\ \vdots & & \vdots \\ \beta_{m'1} & \dots & \beta_{m'm} \end{pmatrix} =$$
$$\left(\begin{pmatrix} \alpha_{11} \\ \vdots \\ \alpha_{n'1} \end{pmatrix} \otimes \begin{pmatrix} \beta_{11} \\ \vdots \\ \beta_{m'1} \end{pmatrix} \quad \dots \quad \begin{pmatrix} \alpha_{1n} \\ \vdots \\ \alpha_{n'n} \end{pmatrix} \otimes \begin{pmatrix} \beta_{1m} \\ \vdots \\ \beta_{m'm} \end{pmatrix} \right)$$

If M be a matrix of size $m \times m'$ and N a matrix of size $n \times n'$. Then $M \otimes N$ has size $m \times n$, and it can be implemented as

$$\begin{aligned} &\mathbf{Tens}^{m,n} = \\ &\quad \lambda bc.\mathbf{mat}(((\mathbf{tens}^{m,n})\ (\mathbf{col}_1^m)\ b)\ (\mathbf{col}_1^n)\ c, \cdots ((\mathbf{tens}^{m,n})\ (\mathbf{col}_n^m)\ b)\ (\mathbf{col}_m^n)\ c) \end{aligned}$$

Hence, if $M : (m, m')$ and $N : (n, n')$, we have

$$[\![M \otimes N]\!]^{\text{term}} = ((\mathbf{Tens}^{m,n})\ [\![M]\!]^{\text{term}})\ [\![N]\!]^{\text{term}}$$

Theorem 6.2. *The denotation of Mat as terms and types of λ^{vec} are sound in the following sense.*

$$\begin{array}{lcl} M \downarrow \zeta & \textit{implies} & \vdash [\![M]\!]^{\text{term}} : [\![\zeta]\!]^{\text{type}}, \\ u \downarrow \nu & \textit{implies} & \vdash [\![u]\!]^{\text{term}} : [\![\nu]\!]^{\text{type}}. \end{array}$$

Proof. The proof is a straightfoward structural induction on M and u. □

6.3. λ^{vec} and quantum computation

In quantum computation, data is encoded on normalised vectors in Hilbert spaces. For our purpose, their interesting property is to be modules over the ring of complex numbers. The smallest non-trivial such space is the space of *qubits*. The space of qubits is the two-dimensional vector space $\mathbb{C}^2$, together with a chosen orthonormal basis $\{|0\rangle, |1\rangle\}$. A quantum bit (or qubit) is a normalised vector $\alpha|0\rangle + \beta|1\rangle$, where $|\alpha|^2 + |\beta|^2 = 1$. In quantum computation, the operations on qubits that are usually considered are the *quantum gates*, i.e. a chosen set of unitary operations. For our purpose, their interesting property is to be *linear*.

The fact that one can encode quantum circuits in λ^{vec} is a corollary of Theorem 6.2. Indeed, a quantum circuit can be regarded as a sequence of multiplications and tensors of matrices. The language of term can faithfully represent those, where as the type system can serve as an abstract interpretation of the actual unitary map computed by the circuit.

We believe that this tool is a first step towards lifting the "quantumness" of algebraic λ-calculi to the level of a type based analysis. It could also be a step towards a "quantum theoretical logic" coming readily with a Curry-Howard isomorphism. The logic we are sketching merges intuitionistic logic and vectorial structure, which makes it intriguing.

The next step in the study of the quantumness of the linear algebraic λ-calculus is the exploration of the notion of orthogonality between terms, and the validation of this notion by means of a compilation into quantum circuits. The work of [41] shows that it is worthwhile pursuing in this direction.

6.4. λ^{vec} and other calculi

No direct connection seems to exists between λ^{vec} and intersection and union types [9, 38]. However, there is an ongoing project of a new type system based on intersections, which may take some of the ideas from the Vectorial lambda calculus. Indeed, the sum resembles as a non-idempotent intersection, with some extra quantitative information. In [17] a type system with non-idempotent intersection has been used to compute a bound on the normalisation time, and in [11, 33] to provide new characterisations on strongly normalising terms. In any case, the Scalar type system [4] seems more close to these results: only scalars are considered and so $\mathbf{t} + \mathbf{t}$ has type $2 \cdot T$ if $\mathbf{t}$ has type T, hence the difference with the non-idempotent intersection type is that the scalars are not just natural numbers, but members of a given ring. The Vectorial lambda calculus goes beyond, as it not only have the quantitative information, thought the scalars, but also allows to *intersect* different types. On the other hand, the interpretation of linear combinations we give in this paper is more in line with a union than an intersection, which may be a interesting starting question to follow.

A different path to follow has been started at [22], where a non-idempotent intersection (simply) type has been considered, with no scalars, and where all the isomorphisms on such a system are made explicit with an equivalence relation. One derivative of such a work has been a characterisation of superpositions and projective measurement [23]. A natural objective to pursue is to add scalars to it, by somehow merging it with λ^{vec}.

Acknowledgements. We would like to thank Gilles Dowek and Barbara Petit for enlightening discussions.

7. Bibliography

References

[1] Alberti, M., 2013. Normal forms for the algebraic lambda-calculus. In: Pous, D., Tasson, C. (Eds.), Journées francophones des langages applicatifs (JFLA 2013). Aussois, France, accessible online at http://hal.inria.fr/hal-00779911.

[2] Altenkirch, T., Grattage, J. J., 2005. A functional quantum programming language. In: 20th Annual IEEE Symposium on Logic in Computer Science (LICS 2005). IEEE Computer Society, pp. 249–258.

[3] Arbiser, A., Miquel, A., Ríos, A., 2009. The λ-calculus with constructors: Syntax, confluence and separation. Journal of Functional Programming 19 (5), 581–631.

[4] Arrighi, P., Díaz-Caro, A., 2012. A System F accounting for scalars. Logical Methods in Computer Science 8 (1:11).

[5] Arrighi, P., Díaz-Caro, A., Valiron, B., 2012. A type system for the vectorial aspects of the linear-algebraic lambda-calculus. In: Kashefi, E., Krivine, J., van Raamsdonk, F. (Eds.), 7th International Workshop on Developments of Computational Methods (DCM 2011). Vol. 88 of Electronic Proceedings in Theoretical Computer Science. Open Publishing Association, pp. 1–15.

[6] Arrighi, P., Dowek, G., 2005. A computational definition of the notion of vectorial space. In: Martí-Oliet, N. (Ed.), Proceedings of the Fifth International Workshop on Rewriting Logic and Its Applications (WRLA 2004). Vol. 117 of Electronic Notes in Computer Science. pp. 249–261.

[7] Arrighi, P., Dowek, G., 2017. Lineal: A linear-algebraic lambda-calculus. Logical Methods in Computer Science 13 (1:8).

[8] Assaf, A., Díaz-Caro, A., Perdrix, S., Tasson, C., Valiron, B., 2014. Call-by-value, call-by-name and the vectorial behaviour of the algebraic λ-calculus. Logical Methods in Computer Science 10 (4:8).

[9] Barbanera, F., Dezani-Ciancaglini, M., de' Liguoro, U., 1995. Intersection and Union Types: Syntax and Semantics. Information and Computation 119, 202–230.

[10] Barendregt, H. P., 1992. Lambda-calculi with types. Vol. II of Handbook of Logic in Computer Science. Oxford University Press.

[11] Bernadet, A., Lengrand, S. J., 2013. Non-idempotent intersection types and strong normalisation. Logical Methods in Computer Science 9 (4:3).

[12] Boudol, G., 1994. Lambda-calculi for (strict) parallel functions. Information and Computation 108 (1), 51–127.

[13] Bournez, O., Hoyrup, M., 2003. Rewriting logic and probabilities. In: Nieuwenhuis, R. (Ed.), Proceedings of RTA-2003. Vol. 2706 of Lecture Notes in Computer Science. Springer, pp. 61–75.

[14] Bucciarelli, A., Ehrhard, T., Manzonetto, G., 2012. A relational semantics for parallelism and non-determinism in a functional setting. Annals of Pure and Applied Logic 163 (7), 918–934.

[15] Buiras, P., Díaz-Caro, A., Jaskelioff, M., 2012. Confluence via strong normalisation in an algebraic λ-calculus with rewriting. In: Ronchi della Rocca, S., Pimentel, E. (Eds.), 6th Workshop on Logical and Semantic Frameworks with Applications (LSFA 2011). Vol. 81 of Electronic Proceedings in Theoretical Computer Science. Open Publishing Association, pp. 16–29.

[16] Cirstea, H., Kirchner, C., Liquori, L., 2001. The Rho Cube. In: Honsell, F., Miculan, M. (Eds.), Proceedings of FOSSACS-2001. Vol. 2030 of Lecture Notes in Computer Science. Springer, pp. 168–183.

[17] De Benedetti, E., Ronchi Della Rocca, S., 2013. Bounding normalization time through intersection types. In: Paolini, L. (Ed.), Proceedings of Sixth Workshop on Intersection Types and Related Systems (ITRS 2012). EPTCS. Cornell University Library, pp. 48–57.

[18] de'Liguoro, U., Piperno, A., 1995. Non deterministic extensions of untyped λ-calculus. Information and Computation 122 (2), 149–177.

[19] Di Pierro, A., Hankin, C., Wiklicky, H., 2005. Probabilistic λ-calculus and quantitative program analysis. Journal of Logic and Computation 15 (2), 159–179.

[20] Díaz-Caro, A., 2011. Barendregt's proof of subject reduction for $\lambda 2$. Accessible online on `http://cstheory.stackexchange.com/questions/8891`.

[21] Díaz-Caro, A., Dowek, G., 2014. The probability of non-confluent systems. In: Ayala-Rincn, M., Bonelli, E., Mackie, I. (Eds.), Proceedings of the 9th International Workshop on Developments in Computational Models. Vol. 144 of Electronic Proceedings in Theoretical Computer Science. Open Publishing Association, pp. 1–15.

[22] Díaz-Caro, A., Dowek, G., 2017. Simply typed lambda-calculus modulo type isomorphisms. `hal-01109104`.

[23] Díaz-Caro, A., Dowek, G., 2017. Typing quantum superpositions and measurement. `arXiv:1601.04294`.

[24] Díaz-Caro, A., Manzonetto, G., Pagani, M., 2013. Call-by-value non-determinism in a linear logic type discipline. In: Artemov, S., Nerode, A. (Eds.), International Symposium on Logical Foundations of Computer Science (LFCS 2013). Vol. 7734 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 164–178.

[25] Díaz-Caro, A., Petit, B., 2012. Linearity in the non-deterministic call-by-value setting. In: Ong, L., de Queiroz, R. (Eds.), 19th International Workshop on Logic, Language, Information and Computation (WoLLIC 2012). Vol. 7456 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 216–231.

[26] Dougherty, D. J., 1992. Adding algebraic rewriting to the untyped lambda calculus. Information and Computation 101 (2), 251–267.

[27] Ehrhard, T., 2010. A finiteness structure on resource terms. In: Proceedings of LICS-2010. IEEE Computer Society, pp. 402–410.

[28] Ehrhard, T., Regnier, L., 2003. The differential lambda-calculus. Theoretical Computer Science 309 (1), 1–41.

[29] Girard, J.-Y., Lafont, Y., Taylor, P., 1989. Proofs and Types. Vol. 7 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press.

[30] Grover, L. K., 1996. A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of computing. STOC-96. ACM, pp. 212–219.

[31] Herescu, O. M., Palamidessi, C., 2000. Probabilistic asynchronous π-calculus. In: Tiuryn, J. (Ed.), Proceedings of FOSSACS-2000. Vol. 1784 of Lecture Notes in Computer Science. Springer, pp. 146–160.

[32] Jouannaud, J.-P., Kirchner, H., 1986. Completion of a set of rules modulo a set of equations. SIAM Journal on Computing 15 (4), 1155–1194.

[33] Kesner, D., Ventura, D., 2014. Quantitative types for the linear substitution calculus. In: Diaz, J., Lanese, I., Sangiorgi, D. (Eds.), Theoretical Computer Science. Vol. 8705 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 296–310.

[34] Krivine, J.-L., 1990. Lambda-calcul: Types et Modèles. Études et Recherches en Informatique. Masson.

[35] Lago, U. D., Zorzi, M., 2012. Probabilistic operational semantics for the lambda calculus. RAIRO - Theoretical Informatics and Applications 46 (3), 413–450.

[36] Pagani, M., Ronchi Della Rocca, S., 2010. Linearity, non-determinism and solvability. Fundamental Informaticae 103 (1–4), 173–202.

[37] Petit, B., 2009. A polymorphic type system for the lambda-calculus with constructors. In: Curien, P.-L. (Ed.), Proceedings of TLCA-2009. Vol. 5608 of Lecture Notes in Computer Science. Springer, pp. 234–248.

[38] Pimentel, E., Ronchi Della Rocca, S., Roversi, L., 2012. Intersection Types from a Proof-theoretic Perspective. Fundamenta Informaticae 121 (1-4), 253—274.

[39] Shor, P. W., 1997. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing 26 (5), 1484–1509.

[40] Tasson, C., 2009. Algebraic totality, towards completeness. In: Curien, P.-L. (Ed.), 9th International Conference on Typed Lambda Calculi and Applications (TLCA 2009). Vol. 5608 of Lecture Notes in Computer Science. Springer, pp. 325–340.

[41] Valiron, B., May 29–30, 2010. Orthogonality and algebraic lambda-calculus. In: Proceedings of the 7th International Workshop on Quantum Physics and Logic (QPL 2010). Oxford, UK, pp. 169–175, `http://www.cs.ox.ac.uk/people/bob.coecke/QPL_proceedings.html`.

[42] van Tonder, A., 2004. A lambda-calculus for quantum computation. SIAM Journal of Computing 33, 1109–1135.

[43] Vaux, L., 2009. The algebraic lambda-calculus. Mathematical Structures in Computer Science 19 (5), 1029–1059.

Appendix A. Detailed proofs of lemmas and theorems in Section 4

Appendix A.1. Lemmas from Section 4.3

Lemma 4.2 (Characterisation of types)**.** For any type T in $\mathcal{G}$, there exist $n, m \in \mathbb{N}$, $\alpha_1, \dots, \alpha_n$, $\beta_1, \dots, \beta_m \in \mathsf{S}$, distinct unit types $U_1, \dots, U_n$ and distinct general variables $\mathbb{X}_1, \dots, \mathbb{X}_m$ such that

$$T \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i + \sum_{j=1}^{m} \beta_j \cdot \mathbb{X}_j \ .$$

Proof. Structural induction on T.

- Let T be a unit type, then take $\alpha = \beta = 1$, $n = 1$ and $m = 0$, and so $T \equiv \sum_{i=1}^{1} 1 \cdot U = 1 \cdot U$.
- Let $T = \alpha \cdot T'$, then by the induction hypothesis $T' \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i + \sum_{j=1}^{m} \beta_j \cdot \mathbb{X}_j$, so $T = \alpha \cdot T' \equiv \alpha \cdot (\sum_{i=1}^{n} \alpha_i \cdot U_i + \sum_{j=1}^{m} \beta_j \cdot \mathbb{X}_j) \equiv \sum_{i=1}^{n} (\alpha \times \alpha_i) \cdot U_i + \sum_{j=1}^{m} (\alpha \times \beta_j) \cdot \mathbb{X}_j$.
- Let $T = R + S$, then by the induction hypothesis $R \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i + \sum_{j=1}^{m} \beta_j \cdot \mathbb{X}_j$ and $S \equiv \sum_{i=1}^{n'} \alpha'_i \cdot U'_i + \sum_{j=1}^{m'} \beta'_j \cdot \mathbb{X}'_j$, so $T = R + S \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i + \sum_{i=1}^{n'} \alpha'_i \cdot U'_i +$

$\sum_{j=1}^{m} \beta_j \cdot \mathbb{X}_j + \sum_{j=1}^{m'} \beta'_j \cdot \mathbb{X}'_j$. If the U_i and the U'_i are all different each other, we have finished, in other case, if $U_k = U'_h$, notice that $\alpha_k \cdot U_k + \alpha'_h \cdot U'_h = (\alpha_k + \alpha'_h) \cdot U_k$.

- Let $T = \mathbb{X}$, then take $\alpha = \beta = 1$, $m = 1$ and $n = 0$, and so $T \equiv \sum_{j=1}^{1} 1 \cdot \mathbb{X} = 1 \cdot \mathbb{X}$. □

Definition Appendix A.1. Let F be an algebraic context with n holes. Let $\vec{U} = U_1, \ldots, U_n$ be a list of n unit types. If U is a unit type, we write $\bar{U}$ for the set of unit types equivalent to U:

$$\bar{U} := \{V \mid V \text{ is unit and } V \equiv U\}.$$

The *context vector* $v_F(\vec{U})$ associated with the context F and the unit types $\vec{U}$ is partial map from the set $\mathcal{S} = \{\bar{\mathcal{U}}\}$ to scalars. It is inductively defined as follows: $v_{\alpha \cdot F}(\vec{U}) := \alpha v_F(\vec{U})$, $v_{F+G}(\vec{U}) := v_F(\vec{U}) + v_G(\vec{U})$, and finally $v_{[-_i]}(\vec{U}) := \{\bar{U}_i \mapsto 1\}$. The sum is defined on these partial map as follows:

$$(f+g)(\vec{U}) = \begin{cases} f(\vec{U}) + g(\vec{U}) & \text{if both are defined} \\ f(\vec{U}) & \text{if } f(\vec{U}) \text{ is defined but not } g(\vec{U}) \\ g(\vec{U}) & \text{if } g(\vec{U}) \text{ is defined but not } f(\vec{U}) \\ \text{is not defined} & \text{if neither } f(\vec{U}) \text{ nor } g(\vec{U}) \text{ is defined.} \end{cases}$$

Scalar multiplication is defined as follows:

$$(\alpha f)(\vec{U}) = \begin{cases} \alpha(f(\vec{U})) & \text{if } f(\vec{U}) \text{ is defined} \\ \text{is not defined} & \text{if } f(\vec{U}) \text{ is not defined.} \end{cases}$$

Lemma Appendix A.2. *Let F and G be two algebraic contexts with respectively n and m holes. Let $\vec{U}$ be a list of n unit types, and $\vec{V}$ be a list of m unit types. Then $F(\vec{U}) \equiv G(\vec{V})$ implies $v_F(\vec{U}) = v_G(\vec{V})$.*

Proof. The derivation of $F(\vec{U}) \equiv F(\vec{V})$ essentially consists in a sequence of the elementary rules (or congruence thereof) in Figure 2 composed with transitivity:

$$F(\vec{U}) = W_1 \equiv W_2 \equiv \cdots \equiv W_k = G(\vec{V}).$$

We prove the result by induction on k.

- Case $k = 1$. Then $F(\vec{U})$ is syntactically equal to $G(\vec{V})$: we are done.
- Suppose that the result is true for sequences of size k, and let

$$F(\vec{U}) = W_1 \equiv W_2 \equiv \cdots \equiv W_k \equiv W_{k+1} = G(\vec{V}).$$

Let us concentrate on the first step $F(\vec{U}) \equiv W_2$: it is an elementary step from Figure 2. By structural induction on the proof of $F(\vec{U}) \equiv W_2$ (which only uses congruence and elementary steps, and not transitivity), we can show that W_2 is of the form $F'(\vec{U}')$ where $v_F(\vec{U}) = v_{F'}(\vec{U}')$. We are now in power of applying the induction hypothesis, because the sequence of elementary rewrites from $F'(\vec{U}')$ to $G(\vec{V})$ is of size k. Therefore $v_{F'}(\vec{U}') = v_G(\vec{V})$. We can then conclude that $v_F(\vec{U}) = v_G(\vec{V})$.

This conclude the proof of the lemma. $\square$

Lemma 4.4 (Equivalence between sums of distinct elements (up to $\equiv$))**.** Let $U_1, \dots, U_n$ be a set of distinct (not equivalent) unit types, and let $V_1, \dots, V_m$ be also a set distinct unit types. If $\sum_{i=1}^n \alpha_i \cdot U_i \equiv \sum_{j=1}^m \beta_j \cdot V_j$, then $m = n$ and there exists a permutation p of n such that $\forall i$, $\alpha_i = \beta_{p(i)}$ and $U_i \equiv V_{p(i)}$.

Proof. Let $S = \sum_{i=1}^n \alpha_i \cdot U_i$ and $T = \sum_{j=1}^m \beta_j \cdot V_j$. Both S and T can be respectively written as $F(\vec{U})$ and $G(\vec{V})$. Using Lemma Appendix A.2, we conclude that $v_F(\vec{U}) = v_G(\vec{V})$. Since all U_i's are pairwise non-equivalent, the $\bar{U}_i$'s are pairwise distinct.

$$v_F(\vec{U}) = \{\bar{U}_i \mapsto \alpha_i \mid i = 1 \dots n\}.$$

Similarly, the $\bar{V}_j$'s are pairwise disjoint, and

$$v_G(\vec{G}) = \{\bar{V}_j \mapsto \beta_j \mid i = 1 \dots m\}.$$

We obtain the desired result because these two partial maps are supposed to be equal. Indeed, this immplies:

- $m = n$ because the domains are equal (so they should have the same size)
- Again using the fact that the domains are equal, the sets $\{\bar{U}_i\}$ and $\{\bar{V}_j\}$ are equal: this means there exists a permutation p of n such that $\forall i$, $\bar{U}_i = \bar{V}_{p(i)}$, meaning $U_i \equiv V_{p(i)}$.
- Because the partial maps are equal, the images of a given element $\bar{U}_i = \bar{V}_{p(i)}$ under v_F and v_G are in fact the same: we therefore have $\alpha_i = \beta_{p(i)}$.

And this closes the proof of the lemma. $\square$

Lemma 4.5 (Equivalences $\forall_I$)**.** Let $U_1, \dots, U_n$ be a set of distinct (not equivalent) unit types and let $V_1, \dots, V_n$ be also a set of distinct unit types.

1. $\sum_{i=1}^n \alpha_i \cdot U_i \equiv \sum_{j=1}^m \beta_j \cdot V_j \Leftrightarrow \sum_{i=1}^n \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^m \beta_j \cdot \forall X.V_j$.
2. $\sum_{i=1}^n \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^m \beta_j \cdot V_j \Rightarrow \forall V_j, \exists W_j \;/\; V_j \equiv \forall X.W_j$.
3. $T \equiv R \Rightarrow T[A/X] \equiv R[A/X]$.

Proof. Item (1) From Lemma 4.4, $m = n$, and without loss of generality, for all i, $\alpha_i = \beta_i$ and $U_i = V_i$ in the left-to-right direction, $\forall X.U_i = \forall X.V_i$ in the right-to-left direction. In both cases we easily conclude.

Item (2) is similar.

Item (3) is a straightforward induction on the equivalence $T \equiv R$. $\square$

Lemma 4.8 ($\succeq$-stability)**.** If $T \succeq^{\mathbf{t}}_{\mathcal{V},\Gamma} R$, $\mathbf{t} \to \mathbf{r}$ and $\Gamma \vdash \mathbf{r} : T$, then $T \succeq^{\mathbf{r}}_{\mathcal{V},\Gamma} R$.

Proof. It suffices to show this for $\succ^{\mathbf{t}}_{X,\Gamma}$, with $X \in \mathcal{V}$. Observe that since $T \succ^{\mathbf{t}}_{X,\Gamma} R$, then $X \notin FV(\Gamma)$. We only have to prove that $\Gamma \vdash \mathbf{r} : R$ is derivable from $\Gamma \vdash \mathbf{r} : T$. We proceed now by cases:

- $T \equiv \sum_{i=1}^n \alpha_i \cdot U_i$ and $R \equiv \sum_{i=1}^n \alpha_i \cdot \forall X.U_i$, then using rules $\forall_I$ and $\equiv$, we can deduce $\Gamma \vdash \mathbf{r} : R$.
- $T \equiv \sum_{i=1}^n \alpha_i \cdot \forall X.U$ and $R \equiv \sum_{i=1}^n \alpha_i \cdot U_i[A/X]$, then using rules $\forall_E$ and $\equiv$, we can deduce $\Gamma \vdash \mathbf{r} : R$. $\square$

Lemma 4.9 (Arrows comparison)**.** If $V \to R \succeq^{\mathbf{t}}_{\mathcal{V},\Gamma} \forall \vec{X}.(U \to T)$, then $U \to T \equiv (V \to R)[\vec{A}/\vec{Y}]$, with $\vec{Y} \notin FV(\Gamma)$.

Proof. Let $(\ \cdot\ )^\circ$ be a map from types to types defined as follows,

$$\begin{array}{ccc} X^\circ = X & (U \to T)^\circ = U \to T & (\forall X.T)^\circ = T^\circ \\ (\alpha \cdot T)^\circ = \alpha \cdot T^\circ & & (T+R)^\circ = T^\circ + R^\circ \end{array}$$

We need three intermediate results:

1. If $T \equiv R$, then $T^\circ \equiv R^\circ$.
2. For any types U, A, there exists B such that $(U[A/X])^\circ = U^\circ[B/X]$.
3. For any types V, U, there exists $\vec{A}$ such that if $V \succeq^{\mathbf{t}}_{\mathcal{V},\Gamma} \forall \vec{X}.U$, then $U^\circ \equiv V^\circ[\vec{A}/\vec{X}]$.

Proofs.

1. Induction on the equivalence rules. We only give the basic cases since the inductive step, given by the context where the equivalence is applied, is trivial.
   - $(1 \cdot T)^\circ = 1 \cdot T^\circ \equiv T^\circ$.
   - $(\alpha \cdot (\beta \cdot T))^\circ = \alpha \cdot (\beta \cdot T^\circ) \equiv (\alpha \times \beta) \cdot T^\circ = ((\alpha \times \beta) \cdot T)^\circ$.
   - $(\alpha \cdot T + \alpha \cdot R)^\circ = \alpha \cdot T^\circ + \alpha \cdot R^\circ \equiv \alpha \cdot (T^\circ + R^\circ) = (\alpha \cdot (T + R))^\circ$.
   - $(\alpha \cdot T + \beta \cdot T)^\circ = \alpha \cdot T^\circ + \beta \cdot T^\circ \equiv (\alpha + \beta) \cdot T^\circ = ((\alpha + \beta) \cdot T)^\circ$.
   - $(T + R)^\circ = T^\circ + R^\circ \equiv R^\circ + T^\circ = (R + T)^\circ$.
   - $(T + (R + S))^\circ = T^\circ + (R^\circ + S^\circ) \equiv (T^\circ + R^\circ) + S^\circ = ((T + R) + S)^\circ$.
2. Structural induction on U.
   - $U = \mathcal{X}$. Then $(\mathcal{X}[V/\mathcal{X}])^\circ = V^\circ = \mathcal{X}[V^\circ/\mathcal{X}] = \mathcal{X}^\circ[V^\circ/\mathcal{X}]$.
   - $U = \mathcal{Y}$. Then $(\mathcal{Y}[A/X])^\circ = \mathcal{Y} = \mathcal{Y}^\circ[A/X]$.
   - $U = V \to T$. Then $((V \to T)[A/X])^\circ = (V[A/X] \to T[A/X])^\circ = V[A/X] \to T[A/X] = (V \to T)[A/X] = (V \to T)^\circ[A/X]$.
   - $U = \forall Y.V$. Then $((\forall Y.V)[A/X])^\circ = (\forall Y.V[A/X])^\circ = (V[A/X])^\circ$, which by the induction hypothesis is equivalent to $V^\circ[B/X] = (\forall Y.V)^\circ[B/X]$.
3. It suffices to show this for $V \succ^{\mathbf{t}}_{X,\Gamma} \forall \vec{X}.U$. Cases:
   - $\forall \vec{X}.U \equiv \forall Y.V$, then notice that $(\forall \vec{X}.U)^\circ \equiv_{(1)} (\forall Y.V)^\circ = V^\circ$.
   - $V \equiv \forall Y.W$ and $\forall \vec{X}.U \equiv W[A/X]$, then $(\forall \vec{X}.U)^\circ \equiv_{(1)} (W[A/X])^\circ \equiv_{(2)} W^\circ[B/X] = (\forall Y.W)^\circ[B/X] \equiv_{(1)} V^\circ[B/X]$.

Proof of the lemma. $U \to T \equiv (U \to T)^\circ$, by the intermediate result 3, this is equivalent to $(V \to R)^\circ[\vec{A}/\vec{X}] = (V \to R)[\vec{A}/\vec{X}]$. $\square$

Lemma 4.10 (Scalars)**.** For any context Γ, term $\mathbf{t}$, type T and scalar α, if $\Gamma \vdash \alpha \cdot \mathbf{t} : T$, then there exists a type R such that $T \equiv \alpha \cdot R$ and $\Gamma \vdash \mathbf{t} : R$. Moreover, if the minimum size of the derivation of $\Gamma \vdash \alpha \cdot \mathbf{t} : T$ is s, then if $T = \alpha \cdot R$, the minimum size of the derivation of $\Gamma \vdash \mathbf{t} : R$ is at most $s-1$, in other case, its minimum size is at most $s-2$.

Proof. We proceed by induction on the typing derivation.

$$\dfrac{\Gamma \vdash \alpha \cdot \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i}{\Gamma \vdash \alpha \cdot \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i}\ \forall_I$$

By the induction hypothesis $\sum_{i=1}^n \alpha_i \cdot U_i \equiv \alpha \cdot R$, and by Lemma 4.2, $R \equiv \sum_{j=1}^m \beta_j \cdot V_j + \sum_{k=1}^h \gamma_k \cdot \mathbb{X}_k$. So it is easy to see that $h = 0$ and so $R \equiv \sum_{j=1}^m \beta_j \cdot V_j$. Hence $\sum_{i=1}^n \alpha_i \cdot U_i \equiv \sum_{j=1}^m \alpha \times \beta_j \cdot V_j$. Then by Lemma 4.5, $\sum_{i=1}^n \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^m \alpha \times \beta_j \cdot \forall X.V_j \equiv \alpha \cdot \sum_{j=1}^m \beta_j \cdot \forall X.V_j$. In addition, by the induction hypothesis, $\Gamma \vdash \mathbf{t} : R$ with a derivation of size $s-3$ (or $s-2$ if $n=1$), so by rules $\forall_I$ and $\equiv$ (not needed if $n=1$), $\Gamma \vdash \mathbf{t} : \sum_{j=1}^m \beta_j \cdot \forall X.V_j$ in size $s-2$ (or $s-1$ in the case $n=1$).

$$\dfrac{\Gamma \vdash \alpha \cdot \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i}{\Gamma \vdash \alpha \cdot \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i[A/X]}\ \forall_E$$

By the induction hypothesis $\sum_{i=1}^n \alpha_i \cdot \forall X.U_i \equiv \alpha \cdot R$, and by Lemma 4.2, $R \equiv \sum_{j=1}^m \beta_j \cdot V_j + \sum_{k=1}^h \gamma_k \cdot \mathbb{X}_k$. So it is easy to see that $h = 0$ and so $R \equiv \sum_{j=1}^m \beta_j \cdot V_j$. Hence $\sum_{i=1}^n \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^m \alpha \times \beta_j \cdot V_j$. Then by Lemma 4.5, for each V_j, there exists W_j such that $V_j \equiv \forall X.W_j$, so $\sum_{i=1}^n \alpha_i \cdot \forall X.U_i \equiv \sum_{j=1}^m \alpha \times \beta_j \cdot \forall X.W_j$. Then by the same lemma, $\sum_{i=1}^n \alpha_i \cdot U_i[A/X] \equiv \sum_{j=1}^m \alpha \times \beta_j \cdot W_j[A/X] \equiv \alpha \cdot \sum_{j=1}^m \beta_j \cdot W_j[A/X]$. In addition, by the induction hypothesis, $\Gamma \vdash \mathbf{t} : R$ with a derivation of size $s-3$ (or $s-2$ if $n=1$), so by rules $\forall_E$ and $\equiv$ (not needed if $n=1$), $\Gamma \vdash \mathbf{t} : \sum_{j=1}^m \beta_j \cdot W_j[A/X]$ in size $s-2$ (or $s-1$ in the case $n=1$).

$$\dfrac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \alpha \cdot \mathbf{t} : \alpha \cdot T}\ \alpha_I$$

Trivial case.

$$\dfrac{\Gamma \vdash \alpha \cdot \mathbf{t} : T \qquad T \equiv R}{\Gamma \vdash \alpha \cdot \mathbf{t} : R}\ \equiv$$

By the induction hypothesis $T \equiv \alpha \cdot S$, and $\Gamma \vdash \mathbf{t} : S$. Notice that $R \equiv T \equiv \alpha \cdot S$. If $T = \alpha \cdot S$, then it is derived with a minimum size of at most $s-2$. If $T = R$, then the minimum size remains because the last $\equiv$ rule is redundant. In other case, the sequent can be derived with minimum size at most $s-1$. □

Lemma 4.11 (Type for zero)**.** Let $\mathbf{t} = \mathbf{0}$ or $\mathbf{t} = \alpha \cdot \mathbf{0}$, then $\Gamma \vdash \mathbf{t} : T$ implies $T \equiv 0 \cdot R$.

Proof. We proceed by induction on the typing derivation.

$$\dfrac{\Gamma \vdash \mathbf{0} : T}{\Gamma \vdash \alpha \cdot \mathbf{0} : 0 \cdot T}\ \alpha_I \quad \text{and} \quad \dfrac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \mathbf{0} : 0 \cdot T}\ 0_I$$

Trivial cases

$$\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot V_i}\ \forall$$

$\forall$-rules ($\forall_I$ and $\forall_E$) have both the same structure as shown on the left. In both cases, by the induction hypothesis $\sum_{i=1}^{n} \alpha_i \cdot U_i \equiv 0 \cdot R$, and by Lemma 4.2, $R \equiv \sum_{j=1}^{m} \beta_j \cdot W_j + \sum_{k=1}^{h} \gamma_k \cdot \mathbb{X}_k$. It is easy to check that $h = 0$, so $\sum_{i=1}^{n} \alpha_i \cdot U_i \equiv 0 \cdot \sum_{j=1}^{m} \beta_j \cdot W_j \equiv \sum_{j=1}^{m} 0 \cdot W_j$. Hence, using the same $\forall$-rule, we can derive $\Gamma \vdash \mathbf{t} : \sum_{j=1}^{m} 0 \cdot W'_j$, and by Lemma 4.5 we can ensure that $\sum_{i=1}^{n} \alpha_i \cdot V_i \equiv 0 \cdot \sum_{j=1}^{m} W'_j$.

$$\frac{\Gamma \vdash \mathbf{t} : T \qquad T \equiv R}{\Gamma \vdash \mathbf{t} : R}\ \equiv$$

By the induction hypothesis $R \equiv T \equiv 0 \cdot S$. □

Lemma 4.12 (Sums)**.** If $\Gamma \vdash \mathbf{t} + \mathbf{r} : S$, then $S \equiv T + R$ with $\Gamma \vdash \mathbf{t} : T$ and $\Gamma \vdash \mathbf{r} : R$. Moreover, if the size of the derivation of $\Gamma \vdash \mathbf{t} + \mathbf{r} : S$ is s, then if $S = T + R$, the minimum sizes of the derivations of $\Gamma \vdash \mathbf{t} : T$ and $\Gamma \vdash \mathbf{r} : R$ are at most $s - 1$, and if $S \neq T + R$, the minimum sizes of these derivations are at most $s - 2$.

Proof. We proceed by induction on the typing derivation.

$$\frac{\Gamma \vdash \mathbf{t} + \mathbf{r} : \sum_{i=1}^{n} \alpha_i \cdot U_i}{\Gamma \vdash \mathbf{t} + \mathbf{r} : \sum_{i=1}^{n} \alpha_i \cdot V_i}\ \forall$$

Rules $\forall_I$ and $\forall_E$ have both the same structure as shown on the left. In any case, by the induction hypothesis $\Gamma \vdash \mathbf{t} : T$ and $\Gamma \vdash \mathbf{r} : R$ with $T + R \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i$, and derivations of minimum size at most $s - 2$ if the equality is true, or $s - 3$ if these types are not equal.

In the second case (when the types are not equal), there exists $N, M \subseteq \{1, \dots, n\}$ with $N \cup M = \{1, \dots, n\}$ such that

$$T \equiv \sum_{i \in N \setminus M} \alpha_i \cdot U_i + \sum_{i \in N \cap M} \alpha'_i \cdot U_i \qquad \text{and}$$
$$R \equiv \sum_{i \in M \setminus N} \alpha_i \cdot U_i + \sum_{i \in N \cap M} \alpha''_i \cdot U_i$$

where $\forall i \in N \cap M$, $\alpha'_i + \alpha''_i = \alpha_i$. Therefore, using $\equiv$ (if needed) and the same $\forall$-rule, we get $\Gamma \vdash \mathbf{t} : \sum_{i \in N \setminus M} \alpha_i \cdot V_i + \sum_{i \in N \cap M} \alpha'_i \cdot V_i$ and $\Gamma \vdash \mathbf{r} : \sum_{i \in M \setminus N} \alpha_i \cdot V_i + \sum_{i \in N \cap M} \alpha''_i \cdot V_i$, with derivations of minimum size at most $s - 1$.

$$\frac{\Gamma \vdash \mathbf{t} + \mathbf{r} : S' \qquad S' \equiv S}{\Gamma \vdash \mathbf{t} + \mathbf{r} : S}\ \equiv$$

By the induction hypothesis, $S \equiv S' \equiv T + R$ and we can derive $\Gamma \vdash \mathbf{t} : T$ and $\Gamma \vdash \mathbf{r} : R$ with a minimum size of at most $s - 2$.

$$\frac{\Gamma \vdash \mathbf{t} : T \qquad \Gamma \vdash \mathbf{r} : R}{\Gamma \vdash \mathbf{t} + \mathbf{r} : T + R}\ +_I$$

This is the trivial case. □

Lemma 4.13 (Applications)**.** If $\Gamma \vdash (\mathbf{t})\ \mathbf{r} : T$, then $\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i)$ and $\Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$ where $\sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t})\mathbf{r}} T$ for some $\mathcal{V}$.

Proof. We proceed by induction on the typing derivation.

$$\frac{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{k=1}^{o} \gamma_k \cdot V_k}{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{k=1}^{o} \gamma_k \cdot W_k}\ \forall$$

Rules $\forall_I$ and $\forall_E$ have both the same structure as shown on the left. In any case, by the induction hypothesis $\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i)$, $\Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$ and $\sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t})\mathbf{r}} \sum_{k=1}^{o} \gamma_k \cdot V_k \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t})\mathbf{r}} \sum_{k=1}^{o} \gamma_k \cdot W_k$.

$$\frac{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : S \qquad S \equiv R}{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : R}\ \equiv$$

By the induction hypothesis $\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i)$, $\Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$ and $\sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t})\mathbf{r}} S \equiv R$.

$$\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i) \quad \Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]}{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]}\ \to_E$$

This is the trivial case. □

Lemma 4.14 (Abstractions)**.** If $\Gamma \vdash \lambda x.\mathbf{t} : T$, then $\Gamma, x : U \vdash \mathbf{t} : R$ where $U \to R \succeq_{\mathcal{V},\Gamma}^{\lambda x.\mathbf{t}} T$ for some $\mathcal{V}$.

Proof. We proceed by induction on the typing derivation.

$$\frac{\Gamma \vdash \lambda x.\mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i}{\Gamma \vdash \lambda x.\mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot V_i}\ \forall$$

Rules $\forall_I$ and $\forall_E$ have both the same structure as shown on the left. In any case, by the induction hypothesis $\Gamma, x : U \vdash \mathbf{t} : R$, where $U \to R \succeq_{\mathcal{V},\Gamma}^{\lambda x.\mathbf{t}} \sum_{i=1}^{n} \alpha_i \cdot U_i \succeq_{\mathcal{V},\Gamma}^{\lambda x.\mathbf{t}} \sum_{i=1}^{n} \alpha_i \cdot V_i$.

$$\frac{\Gamma \vdash \lambda x.\mathbf{t} : R \qquad R \equiv T}{\Gamma \vdash \lambda x.\mathbf{t} : T}\ \equiv$$

By the induction hypothesis $\Gamma, x : U \vdash \mathbf{t} : S$ where $U \to S \succeq_{\mathcal{V},\Gamma}^{\lambda x.\mathbf{t}} R \equiv T$.

$$\frac{\Gamma, x : U \vdash \mathbf{t} : T}{\Gamma \vdash \lambda x.\mathbf{t} : U \to T}\ \to_I$$

This is the trivial case. □

Lemma 4.15 (Basis terms)**.** For any context Γ, type T and basis term $\mathbf{b}$, if $\Gamma \vdash \mathbf{b} : T$ then there exists a unit type U such that $T \equiv U$.

Proof. By induction on the typing derivation.

$$\frac{\Gamma \vdash \mathbf{b} : \sum_{i=1}^{n} \alpha_i \cdot U_i}{\Gamma \vdash \mathbf{b} : \sum_{i=1}^{n} \alpha_i \cdot V_i}\ \forall$$

Rules $\forall_I$ and $\forall_E$ have both the same structure as shown on the left. In any case, by the induction hypothesis $U \equiv \sum_{i=1}^{n} \alpha_i \cdot U_i \succeq_{\mathcal{V},\Gamma}^{\mathbf{b}} \sum_{i=1}^{n} \alpha_i \cdot V_i$, then by a straightforward case analysis, we can check that $\sum_{i=1}^{n} \alpha_i \cdot V_i \equiv U'$.

$$\frac{\Gamma \vdash \mathbf{b} : R \qquad R \equiv T}{\Gamma \vdash \mathbf{b} : T}\ \equiv$$

By the induction hypothesis $U \equiv R \equiv T$.

$$\frac{}{\Gamma, x : U \vdash x : U}\, ax \quad \text{or} \quad \frac{\Gamma, x : U \vdash \mathbf{t} : T}{\Gamma \vdash \lambda x.\mathbf{t} : U \to T} \to_I$$

These two are the trivial cases. ∎

Lemma 4.16 (Substitution lemma)**.** For any term $\mathbf{t}$, basis term $\mathbf{b}$, term variable x, context Γ, types T, U, type variable X and type A, where A is a unit type if X is a unit variables, otherwise A is a general type, we have,

1. if $\Gamma \vdash \mathbf{t} : T$, then $\Gamma[A/X] \vdash \mathbf{t} : T[A/X]$;
2. if $\Gamma, x : U \vdash \mathbf{t} : T$, $\Gamma \vdash \mathbf{b} : U$ then $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : T$.

Proof.

1. Induction on the typing derivation.

$$\frac{}{\Gamma, x : U \vdash x : U}\, ax$$

Notice that $\Gamma[A/X], x : U[A/X] \vdash x : U[A/X]$ can also be derived with the same rule.

$$\frac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \mathbf{0} : 0 \cdot T}\, 0_I$$

By the induction hypothesis $\Gamma[A/X] \vdash \mathbf{t} : T[A/X]$, so by rule 0_I, $\Gamma[A/X] \vdash \mathbf{0} : 0 \cdot T[A/X] = (0 \cdot T)[A/X]$.

$$\frac{\Gamma, x : U \vdash \mathbf{t} : T}{\Gamma \vdash \lambda x.\mathbf{t} : U \to T} \to_I$$

By the induction hypothesis $\Gamma[A/X], x : U[A/X] \vdash \mathbf{t} : T[A/X]$, so by rule $\to_I$, $\Gamma[A/X] \vdash \lambda x.\mathbf{t} : U[A/X] \to T[A/X] = (U \to T)[A/X]$.

$$\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{Y}.(U \to T_i) \qquad \Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{B}_j/\vec{Y}]}{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{B}_j/\vec{Y}]} \to_E$$

By the induction hypothesis $\Gamma[A/X] \vdash \mathbf{t} : (\sum_{i=1}^{n} \alpha_i \cdot \forall \vec{Y}.(U \to T_i))[A/X]$ and this type is equal to $\sum_{i=1}^{n} \alpha_i \cdot \forall \vec{Y}.(U[A/X] \to T_i[A/X])$. Also $\Gamma[A/X] \vdash \mathbf{r} : (\sum_{j=1}^{m} \beta_j \cdot U[\vec{B}_j/\vec{Y}])[A/X] = \sum_{j=1}^{m} \beta_j \cdot U[\vec{B}_j/\vec{Y}][A/X]$. Since $\vec{Y}$ is bound, we can consider it is not in A. Hence $U[\vec{B}_j/\vec{Y}][A/X] = U[A/X][\vec{B}_j[A/X]/\vec{Y}]$, and so, by rule $\to_E$,

$$\begin{aligned}\Gamma[A/X] \vdash (\mathbf{t})\ \mathbf{r} &: \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[A/X][\vec{B}_j[A/X]/\vec{Y}] \\ &= (\sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{B}_j/\vec{Y}])[A/X]\ .\end{aligned}$$

$$\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i \qquad Y \notin FV(\Gamma)}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall Y.U_i} \forall_I$$

By the induction hypothesis, $\Gamma[A/X] \vdash \mathbf{t} : (\sum_{i=1}^{n} \alpha_i \cdot U_i)[A/X] = \sum_{i=1}^{n} \alpha_i \cdot U_i[A/X]$. Then, by rule $\forall_I$, $\Gamma[A/X] \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall Y.U_i[A/X] = (\sum_{i=1}^{n} \alpha_i \cdot \forall Y.U_i)[A/X]$ (in the case $Y \in FV(A)$, we can rename the free variable).

$$\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall Y.U_i}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i[B/Y]}\ \forall_E$$

Since Y is bounded, we can consider $Y \notin FV(A)$. By the induction hypothesis $\Gamma[A/X] \vdash \mathbf{t} : (\sum_{i=1}^{n} \alpha_i \cdot \forall Y.U_i)[A/X] = \sum_{i=1}^{n} \alpha_i \cdot \forall Y.U_i[A/X]$. Then by rule $\forall_E$, $\Gamma[A/X] \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i[A/X][B/Y]$. We can consider $X \notin FV(B)$ (in other case, just take $B[A/X]$ in the $\forall$-elimination), hence $\sum_{i=1}^{n} \alpha_i \cdot U_i[A/X][B/Y] = \sum_{i=1}^{n} \alpha_i \cdot U_i[B/Y][A/X]$.

$$\frac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \alpha \cdot \mathbf{t} : \alpha \cdot \mathbf{T}}\ \alpha_I$$

By the induction hypothesis $\Gamma[A/X] \vdash \mathbf{t} : T[A/X]$, so by rule α_I, $\Gamma[A/X] \vdash \alpha \cdot \mathbf{t} : \alpha \cdot T[A/X] = (\alpha \cdot T)[A/X]$.

$$\frac{\Gamma \vdash \mathbf{t} : T \qquad \Gamma \vdash \mathbf{r} : R}{\Gamma \vdash \mathbf{t} + \mathbf{r} : T + R}\ +_I$$

By the induction hypothesis $\Gamma[A/X] \vdash \mathbf{t} : T[A/X]$ and $\Gamma[A/X] \vdash \mathbf{r} : R[A/X]$, so by rule $+_I$, $\Gamma[A/X] \vdash \mathbf{t} + \mathbf{r} : T[A/X] + R[A/X] = (T + R)[A/X]$.

$$\frac{\Gamma \vdash \mathbf{t} : T \qquad T \equiv R}{\Gamma \vdash \mathbf{t} : R}\ \equiv$$

By the induction hypothesis $\Gamma[A/X] \vdash \mathbf{t} : T[A/X]$, and since $T \equiv R$, then $T[A/X] \equiv R[A/X]$, so by rule $\equiv$, $\Gamma[A/X] \vdash \mathbf{t} : R[A/X]$.

2. We proceed by induction on the typing derivation of $\Gamma, x : U \vdash \mathbf{t} : T$.
   (a) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule ax. Cases:
      - $\mathbf{t} = x$, then $T = U$, and so $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : T$ and $\Gamma \vdash \mathbf{b} : U$ are the same sequent.
      - $\mathbf{t} = y$. Notice that $y[\mathbf{b}/x] = y$. By Lemma 4.3 $\Gamma, x : U \vdash y : T$ implies $\Gamma \vdash y : T$.

   (b) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule 0_I, then $\mathbf{t} = \mathbf{0}$ and $T = 0 \cdot R$, with $\Gamma, x : U \vdash \mathbf{r} : R$ for some $\mathbf{r}$. By the induction hypothesis, $\Gamma \vdash \mathbf{r}[\mathbf{b}/x] : R$. Hence, by rule 0_I, $\Gamma \vdash \mathbf{0} : 0 \cdot R$.

   (c) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule $\to_I$, then $\mathbf{t} = \lambda y.\mathbf{r}$ and $T = V \to R$, with $\Gamma, x : U, y : V \vdash \mathbf{r} : R$. Since our system admits weakening (Lemma 4.3), the sequent $\Gamma, y : V \vdash \mathbf{b} : U$ is derivable. Then by the induction hypothesis, $\Gamma, y : V \vdash \mathbf{r}[\mathbf{b}/x] : R$, from where, by rule $\to_I$, we obtain $\Gamma \vdash \lambda y.\mathbf{r}[\mathbf{b}/x] : V \to R$. We are done since $\lambda y.\mathbf{r}[\mathbf{b}/x] = (\lambda y.\mathbf{r})[\mathbf{b}/x]$.

   (d) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule $\to_E$, then $\mathbf{t} = (\mathbf{r})\ \mathbf{u}$ and $T = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot R_i[\vec{B}/\vec{Y}]$, with $\Gamma, x : U \vdash \mathbf{r} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{Y}.(V \to T_i)$ and $\Gamma, x : U \vdash \mathbf{u} : \sum_{j=1}^{m} \beta_j \cdot V[\vec{B}/\vec{Y}]$. By the induction hypothesis, $\Gamma \vdash \mathbf{r}[\mathbf{b}/x] : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{Y}.(V \to R_i)$ and $\Gamma \vdash \mathbf{u}[\mathbf{b}/x] : \sum_{j=1}^{m} \beta_j \cdot V[\vec{B}/\vec{Y}]$. Then, by rule $\to_E$, $\Gamma \vdash \mathbf{r}[\mathbf{b}/x])\ \mathbf{u}[\mathbf{b}/x] : \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot R_i[\vec{B}/\vec{Y}]$.

   (e) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule $\forall_I$. Then $T = \sum_{i=1}^{n} \alpha_i \cdot \forall Y.V_i$, with $\Gamma, x : U \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot V_i$ and $Y \notin FV(\Gamma) \cup FV(U)$. By the induction hypothesis, $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : \sum_{i=1}^{n} \alpha_i \cdot V_i$. Then by rule $\forall_I$, $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : \sum_{i=1}^{n} \alpha_i \cdot \forall Y.V_i$.

   (f) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule $\forall_E$, then $T = \sum_{i=1}^{n} \alpha_i \cdot V_i[B/Y]$, with $\Gamma, x : U \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall Y.V_i$. By the induction hypothesis, $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : \sum_{i=1}^{n} \alpha_i \cdot \forall Y.V_i$. By rule $\forall_E$, $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : \sum_{i=1}^{n} \alpha_i \cdot V_i[B/Y]$.

(g) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule α_I. Then $T = \alpha \cdot R$ and $\mathbf{t} = \alpha \cdot \mathbf{r}$, with $\Gamma, x : U \vdash \mathbf{r} : R$. By the induction hypothesis $\Gamma \vdash \mathbf{r}[\mathbf{b}/x] : R$. Hence by rule α_I, $\Gamma \vdash \alpha \cdot \mathbf{r}[\mathbf{b}/x] : \alpha \cdot R$. Notice that $\alpha \cdot \mathbf{r}[\mathbf{b}/x] = (\alpha \cdot \mathbf{r})[\mathbf{b}/x]$.

(h) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule $+_I$. Then $\mathbf{t} = \mathbf{r} + \mathbf{u}$ and $T = R + S$, with $\Gamma, x : U \vdash \mathbf{r} : R$ and $\Gamma, x : U \vdash \mathbf{u} : S$. By the induction hypothesis, $\Gamma \vdash \mathbf{r}[\mathbf{b}/x] : R$ and $\Gamma \vdash \mathbf{u}[\mathbf{b}/x] : S$. Then by rule $+_I$, $\Gamma \vdash \mathbf{r}[\mathbf{b}/x] + \mathbf{u}[\mathbf{b}/x] : R + S$. Notice that $\mathbf{r}[\mathbf{b}/x] + \mathbf{u}[\mathbf{b}/x] = (\mathbf{r} + \mathbf{u})[\mathbf{b}/x]$.

(i) Let $\Gamma, x : U \vdash \mathbf{t} : T$ as a consequence of rule $\equiv$. Then $T \equiv R$ and $\Gamma, x : U \vdash \mathbf{t} : R$. By the induction hypothesis, $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : R$. Hence, by rule $\equiv$, $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : T$. □

Appendix A.2. Proof of Theorem 4.1

Theorem 4.1 (Weak subject reduction)**.** For any terms $\mathbf{t}$, $\mathbf{t}'$, any context Γ and any type T, if $\mathbf{t} \to_R \mathbf{t}'$ and $\Gamma \vdash \mathbf{t} : T$, then:

1. if $R \notin$ Group F, then $\Gamma \vdash \mathbf{t}' : T$;
2. if $R \in$ Group F, then $\exists S \sqsupseteq T$ such that $\Gamma \vdash \mathbf{t}' : S$ and $\Gamma \vdash \mathbf{t} : S$.

Proof. Let $\mathbf{t} \to_R \mathbf{t}'$ and $\Gamma \vdash \mathbf{t} : T$. We proceed by induction on the rewrite relation.

Group E.

$0 \cdot \mathbf{t} \to \mathbf{0}$ Consider $\Gamma \vdash 0 \cdot \mathbf{t} : T$. By Lemma 4.10, we have that $T \equiv 0 \cdot R$ and $\Gamma \vdash \mathbf{t} : R$. Then, by rule 0_I, $\Gamma \vdash \mathbf{0} : 0 \cdot R$. We conclude using rule $\equiv$.

$1 \cdot \mathbf{t} \to \mathbf{t}$ Consider $\Gamma \vdash 1 \cdot \mathbf{t} : T$, then by Lemma 4.10, $T \equiv 1 \cdot R$ and $\Gamma \vdash \mathbf{t} : R$. Notice that $R \equiv T$, so we conclude using rule $\equiv$.

$\alpha \cdot \mathbf{0} \to \mathbf{0}$ Consider $\Gamma \vdash \alpha \cdot \mathbf{0} : T$, then by Lemma 4.11, $T \equiv 0 \cdot R$. Hence by rules $\equiv$ and 0_I, $\Gamma \vdash \mathbf{0} : 0 \cdot 0 \cdot R$ and so we conclude using rule $\equiv$.

$\alpha \cdot (\beta \cdot \mathbf{t}) \to (\alpha \times \beta) \cdot \mathbf{t}$ Consider $\Gamma \vdash \alpha \cdot (\beta \cdot \mathbf{t}) : T$. By Lemma 4.10, $T \equiv \alpha \cdot R$ and $\Gamma \vdash \beta \cdot \mathbf{t} : R$. By Lemma 4.10 again, $R \equiv \beta \cdot S$ with $\Gamma \vdash \mathbf{t} : S$. Notice that $(\alpha \times \beta) \cdot S \equiv \alpha \cdot (\beta \cdot S) \equiv T$, hence by rules α_I and $\equiv$, we obtain $\Gamma \vdash (\alpha \times \beta) \cdot \mathbf{t} : T$.

$\alpha \cdot (\mathbf{t} + \mathbf{r}) \to \alpha \cdot \mathbf{t} + \alpha \cdot \mathbf{r}$ Consider $\Gamma \vdash \alpha \cdot (\mathbf{t} + \mathbf{r}) : T$. By Lemma 4.10, $T \equiv \alpha \cdot R$ and $\Gamma \vdash \mathbf{t} + \mathbf{r} : R$. By Lemma 4.12 $\Gamma \vdash \mathbf{t} : R_1$ and $\Gamma \vdash \mathbf{r} : R_2$, with $R_1 + R_2 \equiv R$. Then by rules α_I and $+_I$, $\Gamma \vdash \alpha \cdot \mathbf{t} + \alpha \cdot \mathbf{r} : \alpha \cdot R_1 + \alpha \cdot R_2$. Notice that $\alpha \cdot R_1 + \alpha \cdot R_2 \equiv \alpha \cdot (R_1 + R_2) \equiv \alpha \cdot R \equiv T$. We conclude by rule $\equiv$.

Group F.

$\alpha \cdot \mathbf{t} + \beta \cdot \mathbf{t} \to (\alpha + \beta) \cdot \mathbf{t}$ Consider $\Gamma \vdash \alpha \cdot \mathbf{t} + \beta \cdot \mathbf{t} : T$, then by Lemma 4.12, $\Gamma \vdash \alpha \cdot \mathbf{t} : T_1$ and $\Gamma \vdash \beta \cdot \mathbf{t} : T_2$ with $T_1 + T_2 \equiv T$. Then by Lemma 4.10, $T_1 \equiv \alpha \cdot R$ and $\Gamma \vdash \mathbf{t} : R$ and $T_2 \equiv \beta \cdot S$. By rule α_I, $\Gamma \vdash (\alpha + \beta) \cdot \mathbf{t} : (\alpha + \beta) \cdot R$. Notice that $(\alpha + \beta) \cdot R \sqsupseteq \alpha \cdot R + \beta \cdot S \equiv T_1 + T_2 \equiv T$.

$\alpha \cdot \mathbf{t} + \mathbf{t} \to (\alpha + 1) \cdot \mathbf{t}$ **and** $R = \mathbf{t} + \mathbf{t} \to (1 + 1) \cdot \mathbf{t}$ The proofs of these two cases are simplified versions of the previous case.

$\mathbf{t} + \mathbf{0} \to \mathbf{t}$ Consider $\Gamma \vdash \mathbf{t} + \mathbf{0} : T$. By Lemma 4.12, $\Gamma \vdash \mathbf{t} : R$ and $\Gamma \vdash \mathbf{0} : S$ with $R + S \equiv T$. In addition, by Lemma 4.11, $S \equiv 0 \cdot S'$. Notice that $R + 0 \cdot R \equiv R \sqsupseteq R + 0 \cdot S' \equiv R + S \equiv T$.

Group B.

$(\lambda x.\mathbf{t})\ \mathbf{b} \to \mathbf{t}[\mathbf{b}/x]$ Consider $\Gamma \vdash (\lambda x.\mathbf{t})\ \mathbf{b} : T$, then by Lemma 4.13, we have $\Gamma \vdash \lambda x.\mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to R_i)$ and $\Gamma \vdash \mathbf{b} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$ where $\sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot R_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\lambda x.\mathbf{t})\mathbf{b}} T$. However, we can simplify these types using Lemma 4.15, and so we have $\Gamma \vdash \lambda x.\mathbf{t} : \forall \vec{X}.(U \to R)$ and $\Gamma \vdash \mathbf{b} : U[\vec{A}/\vec{X}]$ with $R[\vec{A}/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\lambda x.\mathbf{t})\mathbf{b}} T$. Note that $\vec{X} \notin FV(\Gamma)$ (from the arrow introduction rule). Hence, by Lemma 4.14, $\Gamma, x : V \vdash \mathbf{t} : S$, with $V \to S \succeq_{\mathcal{V},\Gamma}^{\lambda x.\mathbf{t}} \forall \vec{X}.(U \to R)$. Hence, by Lemma 4.9, $U \equiv V[\vec{B}/\vec{Y}]$ and $R \equiv S[\vec{B}/\vec{Y}]$ with $\vec{Y} \notin FV(\Gamma)$, so by Lemma 4.16(1), $\Gamma, x : U \vdash \mathbf{t} : R$. Applying Lemma 4.16(1) once more, we have $\Gamma[\vec{A}/\vec{X}, x : U[\vec{A}/\vec{X}] \vdash \mathbf{t}[\mathbf{b}/x] : R[\vec{A}/\vec{X}]$. Since $\vec{X} \notin FV(\Gamma)$, $\Gamma[\vec{A}/\vec{X}] = \Gamma$ and we can apply Lemma 4.16(2) to get $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : R[\vec{A}/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\lambda x.\mathbf{t})\mathbf{b}} T$. So, by Lemma 4.8, $R[\vec{A}/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{\mathbf{t}[\mathbf{b}/x]} T$, which implies $\Gamma \vdash \mathbf{t}[\mathbf{b}/x] : T$.

Group A.

$(\mathbf{t}+\mathbf{r})\ \mathbf{u} \to (\mathbf{t})\ \mathbf{u} + (\mathbf{r})\ \mathbf{u}$ Consider $\Gamma \vdash (\mathbf{t}+\mathbf{r})\ \mathbf{u} : T$. Then by Lemma 4.13, $\Gamma \vdash \mathbf{t}+\mathbf{r} : \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i)$ and $\Gamma \vdash \mathbf{u} : \sum_{j=1}^{m} \beta_j . U[\vec{A}_j/\vec{X}]$ where $\sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t}+\mathbf{r})\mathbf{u}} T$. Then by Lemma 4.12, $\Gamma \vdash \mathbf{t} : R_1$ and $\Gamma \vdash \mathbf{r} : R_2$, with $R_1 + R_2 \equiv \sum_{i=1}^{n} \alpha_i \cdot \forall \vec{X}.(U \to T_i)$. Hence, there exists $N_1, N_2 \subseteq \{1, \ldots, n\}$ with $N_1 \cup N_2 = \{1, \ldots, n\}$ such that

$$R_1 \equiv \sum_{i \in N_1 \setminus N_2} \alpha_i \cdot \forall \vec{X}.(U \to T_i) + \sum_{i \in N_1 \cap N_2} \alpha'_i \cdot \forall \vec{X}.(U \to T_i) \quad \text{and}$$
$$R_2 \equiv \sum_{i \in N_2 \setminus N_1} \alpha_i \cdot \forall \vec{X}.(U \to T_i) + \sum_{i \in N_1 \cap N_2} \alpha''_i \cdot \forall \vec{X}.(U \to T_i)$$

where $\forall i \in N_1 \cap N_2$, $\alpha'_i + \alpha''_i = \alpha_i$. Therefore, using $\equiv$ we get

$$\Gamma \vdash \mathbf{t} : \sum_{i \in N_1 \setminus N_2} \alpha_i \cdot \forall \vec{X}.(U \to T_i) + \sum_{i \in N_1 \cap N_2} \alpha'_i \cdot \forall \vec{X}.(U \to T_i) \quad \text{and}$$
$$\Gamma \vdash \mathbf{r} : \sum_{i \in N_2 \setminus N_1} \alpha_i \cdot \forall \vec{X}.(U \to T_i) + \sum_{i \in N_1 \cap N_2} \alpha''_i \cdot \forall \vec{X}.(U \to T_i)$$

So, using rule $\to_E$, we get

$$\Gamma \vdash (\mathbf{t})\ \mathbf{u} : \sum_{i \in N_1 \setminus N_2} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] + \sum_{i \in N_1 \cap N_2} \sum_{j=1}^{m} \alpha'_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \quad \text{and}$$
$$\Gamma \vdash (\mathbf{r})\ \mathbf{u} : \sum_{i \in N_2 \setminus N_1} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] + \sum_{i \in N_1 \cap N_2} \sum_{j=1}^{m} \alpha''_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$$

Finally, by rule $+_I$ we can conclude $\Gamma \vdash (\mathbf{t})\ \mathbf{u} + (\mathbf{r})\ \mathbf{u} : \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t}+\mathbf{r})\mathbf{u}} T$. Then by Lemma 4.8, : $\sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t})\mathbf{u}+(\mathbf{r})\mathbf{u}} T$, so $\Gamma \vdash (\mathbf{t})\ \mathbf{u} + (\mathbf{r})\ \mathbf{u} : T$.

$(\mathbf{t})\ (\mathbf{r}+\mathbf{u}) \to (\mathbf{t})\ \mathbf{r} + (\mathbf{t})\ \mathbf{u}$ Consider $\Gamma \vdash (\mathbf{t})\ (\mathbf{r}+\mathbf{u}) : T$. By Lemma 4.13, $\Gamma \vdash \mathbf{t} : \sum_{i=1}^n \alpha_i \cdot \forall \vec{X}.(U \to T_i)$ and $\Gamma \vdash \mathbf{r}+\mathbf{u} : \sum_{j=1}^m \beta_j.U[\vec{A}_j/\vec{X}]$ where $\sum_{i=1}^n \sum_{j=1}^m \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\mathbf{t})(\mathbf{r}+\mathbf{u})} T$. Then by Lemma 4.12, $\Gamma \vdash \mathbf{r} : R_1$ and $\Gamma \vdash \mathbf{u} : R_2$, with $R_1 + R_2 \equiv \sum_{j=1}^m \beta_j.U[\vec{A}_j/\vec{X}]$. Hence, there exists $M_1, M_2 \subseteq \{1,\dots,m\}$ with $M_1 \cup M_2 = \{1,\dots,m\}$ such that

$$R_1 \equiv \sum_{j\in M_1\setminus M_2} \beta_j.U[\vec{A}_j/\vec{X}] + \sum_{j\in M_1\cap M_2} \beta_j'.U[\vec{A}_j/\vec{X}] \quad \text{and}$$
$$R_2 \equiv \sum_{j\in M_2\setminus M_1} \beta_j.U[\vec{A}_j/\vec{X}] + \sum_{j\in M_1\cap M_2} \beta_j''.U[\vec{A}_j/\vec{X}]$$

where $\forall j \in M_1 \cap M_2$, $\beta_j' + \beta_j'' = \beta_j$. Therefore, using $\equiv$ we get

$$\Gamma \vdash \mathbf{r} : \sum_{j\in M_1\setminus M_2} \beta_j.U[\vec{A}_j/\vec{X}] + \sum_{j\in M_1\cap M_2} \beta_j'.U[\vec{A}_j/\vec{X}] \quad \text{and}$$
$$\Gamma \vdash \mathbf{u} : \sum_{j\in M_2\setminus M_1} \beta_j.U[\vec{A}_j/\vec{X}] + \sum_{j\in M_1\cap M_2} \beta_j''.U[\vec{A}_j/\vec{X}]$$

So, using rule $\to_E$, we get

$$\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^n \sum_{j\in M_1\setminus M_2} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] + \sum_{i=1}^n \sum_{j\in M_1\cap M_2} \alpha_i \times \beta_j' \cdot T_i[\vec{A}_j/\vec{X}] \quad \text{and}$$
$$\Gamma \vdash (\mathbf{t})\ \mathbf{u} : \sum_{i=1}^n \sum_{j\in M_2\setminus M_1} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] + \sum_{i=1}^n \sum_{j\in M_1\cap M_2} \alpha_i \times \beta_j'' \cdot T_i[\vec{A}_j/\vec{X}]$$

Finally, by rule $+_I$ we can conclude $\Gamma \vdash (\mathbf{t})\ \mathbf{r} + (\mathbf{t})\ \mathbf{u} : \sum_{i=1}^n \sum_{j=1}^m \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$. We finish the case with Lemma 4.8.

$(\alpha\cdot\mathbf{t})\ \mathbf{r} \to \alpha\cdot(\mathbf{t})\ \mathbf{r}$ Consider $\Gamma \vdash (\alpha\cdot\mathbf{t})\ \mathbf{r} : T$. Then by Lemma 4.13, $\Gamma \vdash \alpha\cdot\mathbf{t} : \sum_{i=1}^n \alpha_i \cdot \forall \vec{X}.(U \to T_i)$ and $\Gamma \vdash \mathbf{r} : \sum_{j=1}^m \beta_j \cdot U[\vec{A}_j/\vec{X}]$, where $\sum_{i=1}^n \sum_{j=1}^m \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq_{\mathcal{V},\Gamma}^{(\alpha\cdot\mathbf{t})\mathbf{r}} T$. Then by Lemma 4.10, $\sum_{i=1}^n \alpha_i \cdot \forall \vec{X}.(U \to T_i) \equiv \alpha \cdot R$ and $\Gamma \vdash \mathbf{t} : R$. By Lemma 4.2, $R \equiv \sum_{i=1}^{n'} \gamma_i \cdot V_i + \sum_{k=1}^h \eta_k \cdot \mathbb{X}_k$, however it is easy to see that $h=0$ because R is equivalent to a sum of terms, where none of them is $\mathbb{X}$. So $R \equiv \sum_{i=1}^{n'} \gamma_i \cdot V_i$. Without lost of generality (cf. previous case), take $T_i \neq T_k$ for all $i \neq k$ and $h=0$, and notice that $\sum_{i=1}^n \alpha_i \cdot \forall \vec{X}.(U \to T_i) \equiv \sum_{i=1}^{n'} \alpha \times \gamma_i \cdot V_i$. Then by Lemma 4.4, there exists a permutation p such that $\alpha_i = \alpha \times \gamma_{p(i)}$ and $\forall \vec{X}.(U \to T_i) \equiv V_{p(i)}$. Without lost of generality let p be the trivial permutation, and so $\Gamma \vdash \mathbf{t} : \sum_{i=1}^n \gamma_i \cdot \forall \vec{X}.(U \to T_i)$. Hence, using rule $\to_E$, $\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^n \sum_{j=1}^m \gamma_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$. Therefore, by rule α_I, $\Gamma \vdash \alpha\cdot(\mathbf{t})\ \mathbf{r} : \alpha \cdot \sum_{i=1}^n \sum_{j=1}^m \gamma_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$. Notice that $\alpha \cdot \sum_{i=1}^n \sum_{j=1}^m \gamma_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \equiv \sum_{i=1}^n \sum_{j=1}^m \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$. We finish the case with Lemma 4.8.

$(\mathbf{t})\ (\alpha\cdot\mathbf{r}) \to \alpha\cdot(\mathbf{t})\ \mathbf{r}$ Consider $\Gamma \vdash (\mathbf{t})\ (\alpha\cdot\mathbf{r}) : T$. Then by Lemma 4.13, $\Gamma \vdash \mathbf{t} : \sum_{i=1}^n \alpha_i \cdot \forall \vec{X}.(U \to T_i)$ and $\Gamma \vdash \alpha\cdot\mathbf{r} : \sum_{j=1}^m \beta_j \cdot U[\vec{A}_j/\vec{X}]$, where $\sum_{i=1}^n \sum_{j=1}^m \alpha_i \times$

$\beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{(\mathbf{t})(\alpha\cdot\mathbf{r})}_{\mathcal{V},\Gamma} T$. Then by Lemma 4.10, $\sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}] \equiv \alpha \cdot R$ and $\Gamma \vdash \mathbf{r} : R$. By Lemma 4.2, $R \equiv \sum_{j=1}^{m'} \gamma_j \cdot V_j + \sum_{k=1}^{h} \eta_k \cdot \mathbb{X}_k$, however it is easy to see that $h = 0$ because R is equivalent to a sum of terms, where none of them is $\mathbb{X}$. So $R \equiv \sum_{j=1}^{m'} \gamma_j \cdot V_j$. Without lost of generality (cf. previous case), take $A_j \neq A_k$ for all $j \neq k$, and notice that $\sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}] \equiv \sum_{j=1}^{m'} \alpha \times \gamma_j \cdot V_j$. Then by Lemma 4.4, there exists a permutation p such that $\beta_j = \alpha \times \gamma_{p(j)}$ and $U[\vec{A}_j/\vec{X}] \equiv V_{p(j)}$. Without lost of generality let p be the trivial permutation, and so $\Gamma \vdash \mathbf{r} : \sum_{j=1}^{m} \gamma_i \cdot U[\vec{A}_j/\vec{X}]$. Hence, using rule $\to_E$, $\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \gamma_j \cdot T_i[\vec{A}_j/\vec{X}]$. Therefore, by rule α_I, $\Gamma \vdash \alpha \cdot (\mathbf{t})\ \mathbf{r} : \alpha \cdot \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \gamma_j \cdot T_i[\vec{A}_j/\vec{X}]$. Notice that $\alpha \cdot \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \gamma_j \cdot T_i[\vec{A}_j/\vec{X}] \equiv \sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$. We finish the case with Lemma 4.8.

$(\mathbf{0})\ \mathbf{t} \to \mathbf{0}$ Consider $\Gamma \vdash (\mathbf{0})\ \mathbf{t} : T$. By Lemma 4.13, $\Gamma \vdash \mathbf{0} : \sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i)$ and $\Gamma \vdash \mathbf{t} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$, where $\sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{(\mathbf{0})\mathbf{t}}_{\mathcal{V},\Gamma} T$. Then by Lemma 4.11, $\sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i) \equiv 0 \cdot R$. By Lemma 4.2, $R \equiv \sum_{i=1}^{n'} \gamma_i \cdot V_i + \sum_{k=1}^{h} \eta_k \cdot \mathbb{X}_k$, however, it is easy to see that $h = 0$ and so $R \equiv \sum_{i=1}^{n'} \gamma_i \cdot V_i$. Without lost of generality, take $T_i \neq T_k$ for all $i \neq k$, and notice that $\sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i) \equiv \sum_{i=1}^{n'} 0 \cdot V_i$. By Lemma 4.4, $\alpha_i = 0$. Notice that by rule $\to_E$, $\Gamma \vdash (\mathbf{0})\ \mathbf{t} : \sum_{i=1}^{n}\sum_{j=1}^{m} 0 \cdot T_i[\vec{A}_j/\vec{X}]$, hence by rules 0_I and $\equiv$, $\Gamma \vdash \mathbf{0} : \sum_{i=1}^{n}\sum_{j=1}^{m} 0 \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{(\mathbf{0})\mathbf{t}}_{\mathcal{V},\Gamma} T$. By Lemma 4.8, $\sum_{i=1}^{n}\sum_{j=1}^{m} 0 \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{\mathbf{0}}_{\mathcal{V},\Gamma} T$, so $\Gamma \vdash \mathbf{0} : T$.

$(\mathbf{t})\ \mathbf{0} \to \mathbf{0}$ Consider $\Gamma \vdash (\mathbf{t})\ \mathbf{0} : T$. By Lemma 4.13, $\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i)$ and $\Gamma \vdash \mathbf{0} : \sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}]$, where $\sum_{i=1}^{n}\sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{(\mathbf{t})\mathbf{0}}_{\mathcal{V},\Gamma} T$. Then by Lemma 4.11, $\sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}] \equiv 0 \cdot R$. By Lemma 4.2, $R \equiv \sum_{j=1}^{m'} \gamma_j \cdot V_j + \sum_{k=1}^{h} \eta_k \cdot \mathbb{X}_k$, however, it is easy to see that $h = 0$ and so $R \equiv \sum_{j=1}^{m'} \gamma_j \cdot V_j$. Without lost of generality, take $A_j \neq A_k$ for all $j \neq k$, and notice that $\sum_{j=1}^{m} \beta_j \cdot U[\vec{A}_j/\vec{X}] \equiv \sum_{j=1}^{m'} 0 \cdot V_j$. By Lemma 4.4, $\beta_j = 0$. Notice that by rule $\to_E$, $\Gamma \vdash (\mathbf{t})\ \mathbf{0} : \sum_{i=1}^{n}\sum_{j=1}^{m} 0 \cdot T_i[\vec{A}_j/\vec{X}]$, hence by rules 0_I and $\equiv$, $\Gamma \vdash \mathbf{0} : \sum_{i=1}^{n}\sum_{j=1}^{m} 0 \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{(\mathbf{t})\mathbf{0}}_{\mathcal{V},\Gamma} T$. By Lemma 4.8, $\sum_{i=1}^{n}\sum_{j=1}^{m} 0 \cdot T_i[\vec{A}_j/\vec{X}] \succeq^{\mathbf{0}} T$. Hence, $\Gamma \vdash \mathbf{0} : T$.

Contextual rules. Follows from the generation lemmas, the induction hypothesis and the fact that $\sqsupseteq$ is congruent. ☐

Appendix B. Detailed proofs of lemmas and theorems in Section 5

Appendix B.1. First lemmas

Lemma 5.3 If A, B and all the A_i's are in RC, then so are $\mathsf{A} \to \mathsf{B}$, $\sum_i \mathsf{A}_i$ and $\cap_i \mathsf{A}_i$.

Proof. Before proving that these operators define reducibility candidates, we need the following result which simplifies its proof: a linear combination of strongly normalising terms, is strongly normalising. That is

Auxiliary Lemma (AL)**.** If $\{\mathbf{t}_i\}_i$ are strongly normalising, then so is $F(\vec{\mathbf{t}})$ for any algebraic context F.

Proof. Let $\vec{\mathbf{t}} = \mathbf{t}_1, \ldots, \mathbf{t}_n$. We define two notions.

- A measure s on $\vec{\mathbf{t}}$ defined as the the sum over i of the sum of the lengths of all the possible rewrite sequences starting with $\mathbf{t}_i$.
- An algebraic measure a over algebraic contexts $F(.)$ defined inductively by $a(\mathbf{t}_i) = 1$, $a(F(\vec{\mathbf{t}}) + G(\vec{\mathbf{t}'})) = 2 + a(F(\vec{\mathbf{t}})) + a(G(\vec{\mathbf{t}'}))$, $a(\alpha \cdot F(\vec{\mathbf{t}})) = 1 + 2 \cdot a(F(\vec{\mathbf{t}}))$, $a(\mathbf{0}) = 0$.

We claim that for all linear algebraic contexts $F(\cdot)$ (in the sense of Remark 5.2) and all strongly normalising terms $\mathbf{t}_i$ that are not linear combinations (that is, of the form x, $\lambda x.\mathbf{r}$ or $(\mathbf{s})\ \mathbf{r}$), the term $F(\vec{\mathbf{t}})$ is also strongly normalising.

The claim is proven by induction on $s(\vec{\mathbf{t}})$ (the size is finite because $\mathbf{t}$ is SN, and because the rewrite system is finitely branching).

- If $s(\vec{\mathbf{t}}) = 0$. Then none of the $\mathbf{t}_i$ reduces. We show by induction on $a(F(\vec{\mathbf{t}}))$ that $F(\vec{\mathbf{t}})$ is SN.
  - If $a(F(\vec{\mathbf{t}})) = 0$, then $F(\vec{\mathbf{t}}) = \mathbf{0}$ which is SN.
  - Suppose it is true for all $F(\vec{\mathbf{t}})$ of algebraic measure less or equal to m, and consider $F(\vec{\mathbf{t}})$ such that $a(F(\vec{\mathbf{t}})) = m + 1$. Since the $\mathbf{t}_i$ are not linear combinations and they are in normal form, because $s(\vec{\mathbf{t}}) = 0$, then $F(\vec{\mathbf{t}})$ can only reduce with a rule from Group E or a rule from group F. We show that those reductions are strictly decreasing on the algebraic measure, by a rule by rule analysis, and so, we can conclude by induction hypothesis.
    - $0 \cdot F(\vec{\mathbf{t}}) \to \mathbf{0}$. Note that $a(0 \cdot F(\vec{\mathbf{t}})) = 1 > 0 = a(\mathbf{0})$.
    - $1 \cdot F(\vec{\mathbf{t}}) \to F(\vec{\mathbf{t}})$. Note that $a(1 \cdot F(\vec{\mathbf{t}})) = 1 + 2 \cdot a(F(\vec{\mathbf{t}})) > a(F(\vec{\mathbf{t}}))$.
    - $\alpha \cdot \mathbf{0} \to \mathbf{0}$. Note that $a(\alpha \cdot \mathbf{0}) = 1 > 0 = a(\mathbf{0})$.
    - $\alpha \cdot (\beta \cdot F(\vec{\mathbf{t}})) \to (\alpha \times \beta) \cdot F(\vec{\mathbf{t}})$. Note that $a(\alpha \cdot (\beta \cdot F(\vec{\mathbf{t}}))) = 1 + 2 \cdot (1 + 2 \cdot a(F(\vec{\mathbf{t}}))) > 1 + 2 \cdot a(F(\vec{\mathbf{t}})) = a((\alpha \times \beta) \cdot F(\vec{\mathbf{t}}))$.
    - $\alpha \cdot (F(\vec{\mathbf{t}}) + G(\vec{\mathbf{t}'})) \to \alpha \cdot F(\vec{\mathbf{t}}) + \alpha \cdot G(\vec{\mathbf{t}'})$. Note that $a(\alpha \cdot (F(\vec{\mathbf{t}}) + G(\vec{\mathbf{t}'}))) = 5 + 2 \cdot a(F(\vec{\mathbf{t}})) + 2 \cdot a(G(\vec{\mathbf{t}'})) > 4 + 2 \cdot a(F(\vec{\mathbf{t}})) + 2 \cdot a(G(\vec{\mathbf{t}'})) = a(\alpha \cdot F(\vec{\mathbf{t}}) + \alpha \cdot G(\vec{\mathbf{t}'}))$.
    - $\alpha \cdot F(\vec{\mathbf{t}}) + \beta \cdot F(\vec{\mathbf{t}}) \to (\alpha + \beta) \cdot F(\vec{\mathbf{t}})$. Note that $a(\alpha \cdot F(\vec{\mathbf{t}}) + \beta \cdot F(\vec{\mathbf{t}})) = 4 + 4 \cdot a(F(\vec{\mathbf{t}})) > 1 + 2 \cdot a(F(\vec{\mathbf{t}})) = a((\alpha + \beta) \cdot F(\vec{\mathbf{t}}))$.
    - $\alpha \cdot F(\vec{\mathbf{t}}) + F(\vec{\mathbf{t}}) \to (\alpha + 1) \cdot F(\vec{\mathbf{t}})$. Note that $a(\alpha \cdot F(\vec{\mathbf{t}}) + F(\vec{\mathbf{t}})) = 3 + 3 \cdot a(F(\vec{\mathbf{t}})) > 1 + 2 \cdot a(F(\vec{\mathbf{t}})) = a \cdot ((\alpha + 1) \cdot F(\vec{\mathbf{t}}))$.
    - $F(\vec{\mathbf{t}}) + F(\vec{\mathbf{t}}) \to (1 + 1) \cdot F(\vec{\mathbf{t}})$. Note that $a \cdot (F(\vec{\mathbf{t}}) + F(\vec{\mathbf{t}})) = 2 + 2 \cdot a(F(\vec{\mathbf{t}})) > 1 + 2 \cdot a(F(\vec{\mathbf{t}})) = a \cdot ((1 + 1) \cdot F(\vec{\mathbf{t}}))$.
    - $F(\vec{\mathbf{t}}) + \mathbf{0} \to F(\vec{\mathbf{t}})$. Note that $a \cdot (F(\vec{\mathbf{t}}) + \mathbf{0}) = 2 + a(F(\vec{\mathbf{t}})) > a(F(\vec{\mathbf{t}}))$.
    - Contextual rules are trivial.
- Suppose it is true for n, then consider $\vec{\mathbf{t}}$ such that $s(\vec{\mathbf{t}}) = n + 1$. Again, we show that $F(\vec{\mathbf{t}})$ is SN by induction on $a(F(\vec{\mathbf{t}}))$.
  - If $a(F(\vec{\mathbf{t}})) = 0$, then $F(\vec{\mathbf{t}}) = \mathbf{0}$ which is SN.

– Suppose it is true for all $F(\vec{\mathbf{t}})$ of algebraic measure less or equal to m, and consider $F(\vec{\mathbf{t}})$ such that $a(F(\vec{\mathbf{t}})) = m+1$. Since the $\mathbf{t}_i$ are not linear combinations, $F(\vec{\mathbf{t}})$ can reduce in two ways:

  * $F(\mathbf{t}_1, \ldots \mathbf{t}_i, \ldots \mathbf{t}_k) \to F(\mathbf{t}_1, \ldots \mathbf{t}'_i, \ldots \mathbf{t}_k)$ with $\mathbf{t}_i \to \mathbf{t}'_i$. Then $\mathbf{t}'_i$ can be written as $H(\mathbf{r}_1, \ldots \mathbf{r}_l)$ for some algebraic context H, where the $\mathbf{r}_j$'s are not linear combinations. Note that

  $$\sum_{j=1}^{l} s(\mathbf{r}_j) \leq s(\mathbf{t}'_i) < s(\mathbf{t}_i).$$

  Define the context

  $$G(\mathbf{t}_1, \ldots, \mathbf{t}_{i-1}, \mathbf{u}_1, \ldots \mathbf{u}_l, \mathbf{t}_{i+1}, \ldots \mathbf{t}_k) = \\ F(\mathbf{t}_1, \ldots, \mathbf{t}_{i-1}, H(\mathbf{u}_1, \ldots \mathbf{u}_l), \mathbf{t}_{i+1}, \ldots \mathbf{t}_k).$$

  The term $F(\vec{\mathbf{t}})$ then reduces to the term

  $$G(\mathbf{t}_1, \ldots, \mathbf{t}_{i-1}, \mathbf{r}_1, \ldots \mathbf{r}_l, \mathbf{t}_{i+1} \ldots \mathbf{t}_k),$$

  where

  $$s(\mathbf{t}_1, \ldots, \mathbf{t}_{i-1}, \mathbf{r}_1, \ldots \mathbf{r}_l, \mathbf{t}_{i+1} \ldots \mathbf{t}_k) < s(\vec{\mathbf{t}}).$$

  Using the top induction hypothesis, we conclude that $F(\mathbf{t}_1, \ldots \mathbf{t}'_i, \ldots \mathbf{t}_k)$ is SN.

  * $F(\vec{\mathbf{t}}) \to G(\vec{\mathbf{t}})$, with $a(G(\vec{\mathbf{t}})) < a(F(\vec{\mathbf{t}}))$. Using the second induction hypothesis, we conclude that $G(\vec{\mathbf{t}})$ is SN

  All the possible reducts of $F(\vec{\mathbf{t}})$ are SN: so is $F(\vec{\mathbf{t}})$.

This closes the proof of the claim. Now, consider any SN terms $\{\mathbf{t}_i\}_i$ and any algebraic context $G(\vec{\mathbf{t}})$. Each $\mathbf{t}_i$ can be written as an algebraic sum of x's, $\lambda x.\mathbf{s}$'s and $(\mathbf{r})\,\mathbf{s}$'s. The context $G(\vec{\mathbf{t}})$ can then be written as $F(\vec{\mathbf{t}'})$ where none of the $\mathbf{t}'_i$ is a linear combination. From Remark 5.2, there exists a linear algebraic context $F'(\vec{\mathbf{t}}';)$ where $\vec{\mathbf{t}}''$ is $\vec{\mathbf{t}}'$ with possibly some repetitions, and where $F'(\vec{\mathbf{t}}'') = F(\vec{\mathbf{t}}')$, when considered as terms. The hypotheses of the claim are satisfied: $F'(\vec{\mathbf{t}}'')$ is therefore SN. Since it is ultimely equal to $G(\vec{\mathbf{t}})$, $G(\vec{\mathbf{t}})$ is also SN: the Auxiliary Lemma (AL) is valid.

Now, we can prove Lemma 5.3 First, we consider the case $\mathsf{A} \to \mathsf{B}$.

RC$_1$ We must show that all $\mathbf{t} \in \mathsf{A} \to \mathsf{B}$ are in SN_0. We proceed by induction on the definition of $\mathsf{A} \to \mathsf{B}$.

- Assume that $\mathbf{t}$ is such that for $\mathbf{r} = \mathbf{0}$ and $\mathbf{r} = \mathbf{b}$, with $\mathbf{b} \in A$, then $(\mathbf{t})\,\mathbf{r} \in \mathsf{B}$. Hence by **RC**$_1$ in B, $\mathbf{t} \in SN_0$.
- Assume that $\mathbf{t}$ is closed neutral and that $\mathrm{Red}(\mathbf{t}) \subseteq \mathsf{A} \to \mathsf{B}$. By induction hypothesis, all the elements of $\mathrm{Red}(\mathbf{t})$ are strongly normalising: so is $\mathbf{t}$.
- The last case is immediate: if $\mathbf{t}$ is the term $\mathbf{0}$, it is strongly normalising.

RC$_2$ We must show that if $\mathbf{t} \to \mathbf{t}'$ and $\mathbf{t} \in \mathsf{A} \to \mathsf{B}$, then $\mathbf{t}' \in \mathsf{A} \to \mathsf{B}$. We again proceed by induction on the definition of $\mathsf{A} \to \mathsf{B}$.

- Let $\mathbf{t}$ such that $(\mathbf{t})\,\mathbf{0} \in \mathsf{B}$ and such that for all $\mathbf{b} \in \mathsf{A}$, $(\mathbf{t})\ \mathbf{b} \in \mathsf{B}$. Then by $\mathbf{RC}_2$ in B, $(\mathbf{t}')\ \mathbf{0} \in \mathsf{B}$ and $(\mathbf{t}')\ \mathbf{b} \in \mathsf{B}$, and so $\mathbf{t}' \in \mathsf{A} \to \mathsf{B}$.
- If $\mathbf{t}$ is closed neutral and $\mathrm{Red}(\mathbf{t}) \subseteq \mathsf{A} \to \mathsf{B}$, then $\mathbf{t}' \in \mathsf{A} \to \mathsf{B}$ since $\mathbf{t}' \in \mathrm{Red}(\mathbf{t})$.
- If $\mathbf{t} = \mathbf{0}$, it does not reduce.

$\mathbf{RC}_3$ and $\mathbf{RC}_4$ Trivially true by definition.

Then we analyze the case $\sum_i \mathsf{A}_i$.

$\mathbf{RC}_1$ We show that "if $\mathbf{t} \in \sum_i \mathsf{A}_i$ then $\mathbf{t}$ is stongly normalizing" by structural induction on the justification of $\mathbf{t} \in \sum_i \mathsf{A}_i$.

- Base case. $\mathbf{t}$ belongs to one of the A_i: it is then SN by $\mathbf{RC}_1$.
- $\mathbf{CC}_1$. $\mathbf{t}$ is part of a list $\vec{\mathbf{t}'}$ where $F(\vec{\mathbf{t}'}) \in \sum_i \mathsf{A}_i$ for some alg. context F. The induction hypothesis says that $F(\vec{\mathbf{t}'})$ is SN. This implies that $\mathbf{t}$ is also SN.
- $\mathbf{CC}_2$. $\mathbf{t} = F(\vec{\mathbf{t}'})$ where F is an alg. context and $\mathbf{t}'_i \in \mathsf{A}_i$. The result is obtained using the auxiliary lemma (AL) and $\mathbf{RC}_1$ on the A_i's.
- $\mathbf{RC}_2$. $\mathbf{t}$ is such that $\mathbf{s} \to \mathbf{t}$ where $\mathbf{s}$ is SN. This implies that $\mathbf{t}$ is also SN.
- $\mathbf{RC}_3$. $\mathbf{t}$ is closed neutral and $\mathrm{Red}(\mathbf{t}) \subseteq \sum_i \mathsf{A}_i$, then $\mathbf{t}$ is strongly normalising since all elements of $\mathrm{Red}(\mathbf{t})$ are strongly normalising.

$\mathbf{RC}_2$ and $\mathbf{RC}_3$ Trivially true by definition.

$\mathbf{RC}_4$ Since $\mathbf{0}$ is an algebraic context, it is also in the set by $\mathbf{CC}_2$.

Finally, we prove the case $\cap_i \mathsf{A}_i$.

$\mathbf{RC}_1$ Trivial since for all i, $\mathsf{A}_i \subseteq SN_0$.

$\mathbf{RC}_2$ Let $\mathbf{t} \in \cap_i \mathsf{A}_i$, then $\forall i$, $\mathbf{t} \in \mathsf{A}_i$ and so by $\mathbf{RC}_2$ in A_i, $\mathrm{Red}(\mathbf{t}) \subseteq \mathsf{A}_i$. Thus $\mathrm{Red}(\mathbf{t}) \subseteq \cap_i \mathsf{A}_i$.

$\mathbf{RC}_3$ Let $\mathbf{t} \in \mathcal{N}$ and $\mathrm{Red}(\mathbf{t}) \subseteq \cap_i \mathsf{A}$. Then $\forall_i$, $\mathrm{Red}(\mathbf{t}) \subseteq A_i$, and thus, by $\mathbf{RC}_3$ in A_i, $\mathbf{t} \in \mathsf{A}_i$, which implies $\mathbf{t} \in \cap_i \mathsf{A}_i$.

$\mathbf{RC}_4$ By $\mathbf{RC}_4$, for all i, $\mathbf{0} \in \mathsf{A}_i$. Therefore, $\mathbf{0} \in \cap_i \mathsf{A}_i$.

This concludes the proof of Lemma 5.3. □

Lemma 5.4 Any type T, has a unique (up to $\equiv$) canonical decomposition $T \equiv \sum_{i=1}^n \alpha_i \cdot \mathbb{U}_i$ such that for all l, k, $\mathbb{U}_l \not\equiv \mathbb{U}_k$.

Proof. By Lemma 4.2, $T \equiv \sum_{i=1}^n \alpha_i \cdot U_i + \sum_{j=1}^m \beta_j \cdot \mathbb{X}_j$. Suppose that there exist l, k such that $U_l \equiv U_k$. Then notice that $T \equiv (\alpha_l + \alpha_k) \cdot U_l + \sum_{i \neq l,k} \alpha_i \cdot U_i$. Repeat the process until there is no more l, k such that $U_l \not\equiv U_k$. Proceed in the analogously to obtain a linear combination of different $\mathbb{X}_j$. □

Lemma 5.5 For any types T and A, variable X and valuation ρ, we have $[\![T[A/X]]\!]_\rho = [\![T]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar\rho},[\![A]\!]_\rho)}$ and $[\![T[A/X]]\!]_{\bar\rho} = [\![T]\!]_{\bar\rho,(X_-,X_+)\mapsto([\![A]\!]_\rho,[\![A]\!]_{\bar\rho})}$.

Proof. We proceed by structural induction on T. On each case we only show the case of ρ since the $\bar{\rho}$ case follows analogously.

- $T = X$. Then $[\![X[A/X]]\!]_\rho = [\![A]\!]_\rho = [\![X]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)}$.
- $T = Y$. Then $[\![Y[A/X]]\!]_\rho = [\![Y]\!]_\rho = \rho_+(Y) = [\![Y]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)}$.
- $Y = U \to R$. Then $[\![(U \to R)[A/X]]\!]_\rho = [\![U[A/X]]\!]_{\bar{\rho}} \to [\![R[A/X]]\!]_\rho$. By the induction hypothesis, we have $[\![U[A/X]]\!]_{\bar{\rho}} \to [\![R[A/X]]\!]_\rho = [\![U]\!]_{\bar{\rho},(X_-,X_+)\mapsto([\![A]\!]_\rho,[\![A]\!]_{\bar{\rho}})} \to [\![R]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)} = [\![U \to R]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)}$.
- $U = \forall Y.V$. Then $[\![(\forall Y.V)[A/X]]\!]_\rho = [\![\forall Y.V[A/X]]\!]_\rho$ which by definition is equal to $\bigcap_{\mathsf{B}\in\mathsf{RC}}[\![V[A/X]]\!]_{\rho,(Y_+,Y_-)\mapsto(\mathsf{B},\mathsf{B})}$ and this, by the induction hypothesis, is equal to $\bigcap_{\mathsf{B}\in\mathsf{RC}}[\![V]\!]_{\rho,(Y_+,Y_-)\mapsto(\mathsf{B},\mathsf{B}),(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)} = [\![\forall Y.V]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)}$.
- T of canonical decomposition $\sum_i \alpha_i \cdot \mathbb{U}_i$. Then $[\![T[A/X]]\!]_\rho = \sum_i [\![\mathbb{U}_i[A/X]]\!]_\rho$, which by induction hypothesis is $\sum_i [\![\mathbb{U}_i]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)} = [\![T]\!]_{\rho,(X_+,X_-)\mapsto([\![A]\!]_{\bar{\rho}},[\![A]\!]_\rho)}$.

□

Appendix B.2. Proof of the Adequacy Lemma (5.6)

We need the following results first.

Lemma Appendix B.1. *For any type T, if $\rho = (\rho_+, \rho_-)$ and $\rho' = (\rho'_+, \rho'_-)$ are two valid valuations over $FV(T)$ such that $\forall X$, $\rho'_-(X) \subseteq \rho_-(X)$ and $\rho_+(X) \subseteq \rho'_+(X)$, then we have $[\![T]\!]_\rho \subseteq [\![T]\!]_{\rho'}$ and $[\![T]\!]_{\bar{\rho}'} \subseteq [\![T]\!]_{\bar{\rho}}$.*

Proof. Structural induction on T.

- $T = X$. Then $[\![X]\!]_\rho = \rho_+(X) \subseteq \rho'_+(X) = [\![X]\!]_{\rho'}$ and $[\![X]\!]_{\bar{\rho}'} = \rho'_-(X) \subseteq \rho_-(X) = [\![X]\!]_{\bar{\rho}}$.
- $T = U \to R$. Then $[\![U \to R]\!]_\rho = [\![U]\!]_{\bar{\rho}} \to [\![R]\!]_\rho$ and $[\![U \to R]\!]_{\bar{\rho}'} = [\![U]\!]_{\rho'} \to [\![R]\!]_{\bar{\rho}'}$. By the induction hypothesis $[\![U]\!]_{\bar{\rho}'} \subseteq [\![U]\!]_{\bar{\rho}}$, $[\![U]\!]_\rho \subseteq [\![U]\!]_{\rho'}$, $[\![R]\!]_\rho \subseteq [\![R]\!]_{\rho'}$ and $[\![R]\!]_{\bar{\rho}'} \subseteq [\![R]\!]_{\bar{\rho}}$. We proceed by induction on the definition of $\to$ to show that $\forall \mathbf{t} \in [\![U]\!]_{\bar{\rho}} \to [\![R]\!]_\rho$, then $\mathbf{t} \in [\![U]\!]_{\bar{\rho}'} \to [\![R]\!]_{\rho'} = [\![U \to R]\!]_{\rho'}$

  - Let $\mathbf{t} \in \{\mathbf{t}\,|(\mathbf{t})\ \mathbf{0} \in [\![R]\!]_\rho$ and $\forall \mathbf{b} \in [\![U]\!]_{\bar{\rho}}, (\mathbf{r})\ \mathbf{b} \in [\![R]\!]_\rho\}$. Notice that $(\mathbf{t})\ \mathbf{0} \in [\![R]\!]_\rho \subseteq [\![R]\!]_{\rho'}$. Also, $\forall \mathbf{b} \in [\![U]\!]_{\bar{\rho}'}$, $\mathbf{b} \in [\![U]\!]_{\bar{\rho}}$ and then $(\mathbf{t})\ \mathbf{b} \in [\![R]\!]_\rho \subseteq [\![R]\!]_{\rho'}$.
  - Let $\mathrm{Red}(\mathbf{t}) \in [\![U \to R]\!]_\rho$ and $\mathbf{t} \in \mathcal{N}$. By the induction hypothesis $\mathrm{Red}(\mathbf{t}) \in [\![U \to R]\!]_{\rho'}$ and so, by $\mathbf{RC}_3$, $\mathbf{t} \in [\![U \to R]\!]_{\rho'}$.
  - Let $\mathbf{t} = \mathbf{0}$. By $\mathbf{RC}_4$, $\mathbf{0}$ is in any reducibility candidate, in particular it is in $[\![U \to R]\!]_{\rho'}$.

  Analogously, $\forall \mathbf{t} \in [\![U]\!]_{\rho'} \to [\![R]\!]_{\bar{\rho}'}$, $\mathbf{t} \in [\![U]\!]_\rho \to [\![R]\!]_{\bar{\rho}} = [\![U \to R]\!]_\rho$.
- $T = \forall X.U$. Then $[\![\forall X.U]\!]_\rho = \bigcap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$. By the induction hypothesis we have $[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} \subseteq [\![U]\!]_{\rho',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$, Hence we have that $\bigcap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} \subseteq \bigcap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} = [\![\forall X.U]\!]_{\rho'}$. The proof for the case $[\![\forall X.U]\!]_{\bar{\rho}'} \subseteq [\![\forall X.U]\!]_{\bar{\rho}}$ is analogous.

- $T \equiv \sum_i \alpha_i \cdot \mathbb{U}_i$ and $T \not\equiv \mathbb{U}$. Then $[\![T]\!]_\rho = \sum_i [\![\mathbb{U}_i]\!]_\rho$. By the induction hypothesis $[\![\mathbb{U}_i]\!]_\rho \subseteq [\![\mathbb{U}_i]\!]_{\rho'}$. We proceed by induction on the justification of $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_\rho$ to show that if $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_\rho$ then $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_{\rho'}$.
  - Base case. $\mathbf{t}$ belongs to one of the $[\![\mathbb{U}_i]\!]_\rho$. We conclude using the fact that $[\![\mathbb{U}_i]\!]_\rho \subseteq [\![\mathbb{U}_i]\!]_{\rho'}$: $\mathbf{t}$ then belongs to $[\![\mathbb{U}_i]\!]_{\rho'} \subseteq \sum_i [\![\mathbb{U}_i]\!]_{\rho'}$.
  - $\mathbf{CC}_1$. $\mathbf{t}$ belongs to $\sum_i [\![\mathbb{U}_i]\!]_\rho$ because it is in a list $\vec{\mathbf{t'}}$ where $F(\vec{\mathbf{t'}}) \in \sum_i [\![\mathbb{U}_i]\!]_\rho$ for some alg. context F. Induction hypothesis says that $F(\vec{\mathbf{t'}}) \in \sum_i [\![\mathbb{U}_i]\!]_{\rho'}$. We then get $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_{\rho'}$ using $\mathbf{CC}_1$.
  - $\mathbf{CC}_2$. Let $\mathbf{t} = F(\vec{\mathbf{r}})$ where F is an algebraic context and $\mathbf{r}_i \in [\![\mathbb{U}_i]\!]_{\bar\rho}$. Note that by induction hypothesis $\forall \mathbf{r}_i \in [\![\mathbb{U}_i]\!]_\rho$, $\mathbf{r}_i \in [\![\mathbb{U}_i]\!]_{\rho'}$ and so $F(\vec{\mathbf{r}}) \in \sum_i [\![\mathbb{U}_i]\!]_{\rho'}$.
  - $\mathbf{RC}_2$. $\mathbf{t}' \to \mathbf{t}$ and $\mathbf{t}' \in \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho'}$. Invoking $\mathbf{RC}_2$, $\mathbf{t} \in \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho'}$.
  - $\mathbf{RC}_3$. $\mathrm{Red}(\mathbf{t}) \subseteq [\![T]\!]_\rho$ and $\mathbf{t} \in \mathcal{N}$. By the induction hypothesis $\mathrm{Red}(\mathbf{t}) \subseteq [\![T]\!]_{\rho'}$ and so, by $\mathbf{RC}_3$, $\mathbf{t} \in [\![T]\!]_{\rho'}$.

  The case $[\![T]\!]_{\bar\rho'} \subseteq [\![T]\!]_{\bar\rho}$ is analogous. □

Lemma Appendix B.2. *For any type T, if $\rho = (\rho_+, \rho_-)$ is a valid valuation over $FV(T)$, then we have $[\![T]\!]_{\bar\rho} \subseteq [\![T]\!]_\rho$.*

Proof. Structural induction on T.

- $T = X$. Then $[\![T]\!]_{\bar\rho} = \rho_-(X) \subseteq \rho_+(X) = [\![T]\!]_\rho$.
- $T = U \to R$. Then $[\![U \to R]\!]_{\bar\rho} = [\![U]\!]_\rho \to [\![R]\!]_{\bar\rho}$. By the induction hypothesis $[\![U]\!]_{\bar\rho} \subseteq [\![U]\!]_\rho$ and $[\![R]\!]_{\bar\rho} \subseteq [\![R]\!]_\rho$. We must show that $\forall \mathbf{t} \in [\![U \to R]\!]_{\bar\rho}$, $\mathbf{t} \in [\![U \to R]\!]_\rho$. Let $\mathbf{t} \in [\![U \to R]\!]_{\bar\rho} = [\![U]\!]_\rho \to [\![R]\!]_{\bar\rho}$. We proceed by induction on the definition of $\to$.
  - Let $\mathbf{t} \in \{\mathbf{t} \,|\, (\mathbf{t})\, \mathbf{0} \in [\![R]\!]_{\bar\rho}$ and $\forall \mathbf{b} \in [\![U]\!]_\rho, (\mathbf{t})\ \mathbf{b} \in [\![R]\!]_{\bar\rho}\}$. Notice that $(\mathbf{t})\, \mathbf{0} \in [\![R]\!]_{\bar\rho} \subseteq [\![R]\!]_\rho$ and forall $\mathbf{b} \in [\![U]\!]_{\bar\rho}$, $\mathbf{b} \in [\![U]\!]_\rho$, and so $(\mathbf{t})\ \mathbf{b} \in [\![R]\!]_{\bar\rho} \subseteq [\![R]\!]_\rho$. Thus $\mathbf{t} \in [\![U]\!]_{\bar\rho} \to [\![R]\!]_\rho = [\![U \to R]\!]_\rho$.
  - Let $\mathrm{Red}(\mathbf{t}) \in [\![U \to R]\!]_{\bar\rho}$ and $\mathbf{t} \in \mathcal{N}$. By the induction hypothesis $\mathrm{Red}(\mathbf{t}) \in [\![U \to R]\!]_\rho$ and so, by $\mathbf{RC}_3$, $\mathbf{t} \in [\![U \to R]\!]_\rho$.
  - Let $\mathbf{t} = \mathbf{0}$. By $\mathbf{RC}_4$, $\mathbf{0}$ is in any reducibility candidate, in particular it is in $[\![U \to R]\!]_\rho$.
- $T = \forall X.U$. Then $[\![\forall X.U]\!]_{\bar\rho} = \cap_{\mathsf{A} \in \mathsf{RC}} [\![U]\!]_{\bar\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$. By the induction hypothesis
$$[\![U]\!]_{\bar\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} \subseteq [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}.$$
So $\cap_{\mathsf{A} \in \mathsf{RC}} [\![U]\!]_{\bar\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} \subseteq \cap_{\mathsf{A} \in \mathsf{RC}} [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$ which is $[\![\forall X.U]\!]_\rho$ by definition.
- $T \equiv \sum_i \alpha_i \cdot \mathbb{U}_i$ and $T \not\equiv \mathbb{U}$. Then $[\![T]\!]_{\bar\rho} = \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$. By the induction hypothesis $[\![\mathbb{U}_i]\!]_{\bar\rho} \subseteq [\![\mathbb{U}_i]\!]_\rho$. We proceed by induction on the justification of $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$ to show that $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$ imples $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_\rho$.

- Base case. $\mathbf{t} \in [\![\mathbb{U}_i]\!]_{\bar\rho}$ for some i: by induction hypothesis $\mathbf{t} \in [\![\mathbb{U}_i]\!]_\rho$, which is included in $\sum_i [\![\mathbb{U}_i]\!]_\rho$.
- $\mathbf{CC}_1$. $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$ because it is in a list $\vec{\mathbf{t'}}$ where $F(\vec{\mathbf{t'}}) \in \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$ for some alg. context F. Induction hypothesis says that $F(\vec{\mathbf{t'}}) \in \sum_i [\![\mathbb{U}_i]\!]_\rho$. We then get $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_\rho$ using $\mathbf{CC}_1$.
- $\mathbf{CC}_2$. Let $\mathbf{t} = F(\vec{\mathbf{r}})$ where F is an algebraic context and $\mathbf{r}_i \in [\![\mathbb{U}_i]\!]_{\bar\rho}$. By induction hypothesis $\forall \mathbf{r} \in [\![\mathbb{U}_i]\!]_{\bar\rho}$, $\mathbf{r} \in [\![\mathbb{U}_i]\!]_\rho$ and so the result holds by $\mathbf{CC}_2$.
- $\mathbf{RC}_2$. Let $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$ and $\mathbf{t} \to \mathbf{t}'$. By the induction hypothesis $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_\rho$, hence by $\mathbf{RC}_2$, $\mathbf{t}' \in \sum_i [\![\mathbb{U}_i]\!]_\rho$.
- $\mathbf{RC}_3$. Let $\mathrm{Red}(\mathbf{t}) \in \sum_i [\![\mathbb{U}_i]\!]_{\bar\rho}$ and $\mathbf{t} \in \mathcal{N}$. By the induction hypothesis $\mathrm{Red}(\mathbf{t}) \in \sum_i [\![\mathbb{U}_i]\!]_\rho$ and so, by $\mathbf{RC}_3$, $\mathbf{t} \in \sum_i [\![\mathbb{U}_i]\!]_\rho$. □

Lemma Appendix B.3. *Let $\{\mathsf{A}_i\}_{i=1\cdots n}$ be a family of reducibility candidates. If $\mathbf{s}$ and $\mathbf{t}$ both belongs to $\sum_{i=1}^n \mathsf{A}_i$, then so does $\mathbf{s}+\mathbf{t}$. Similarly, if $\mathbf{t} \in \sum_{i=1}^n \mathsf{A}_i$, then for any α, $\alpha \cdot \mathbf{t} \in \sum_{i=1}^n \mathsf{A}_i$.*

Proof. Direct corollary of the closure under $\mathbf{CC}_2$. □

Lemma Appendix B.4. *Suppose that $\lambda x.\mathbf{s} \in \mathsf{A} \to \mathsf{B}$ and $\mathbf{b} \in \mathsf{A}$, then $(\lambda x.\mathbf{s})\,\mathbf{b} \in \mathsf{B}$.*

Proof. Induction on the definition of $\mathsf{A} \to \mathsf{B}$.

- If $\lambda x.\mathbf{s}$ is in $\{\mathbf{t} \mid (\mathbf{t})\ \mathbf{0} \in \mathsf{B}$ and $\forall \mathbf{b} \in \mathsf{A}, (\mathbf{t})\,\mathbf{b} \in \mathsf{B}\}$, then it is trivial
- $\lambda x.\mathbf{s}$ cannot be in $\mathsf{A} \to \mathsf{B}$ by the closure under $\mathbf{RC}_3$, because it is not neutral, neither by the closure under $\mathbf{RC}_4$, because it is not the term $\mathbf{0}$. □

Remark Appendix B.5. For the proof of adequacy, we show in the following lemma that $[\![\forall \vec{X}.U]\!]_\rho$ can be equivalently defined as a more general intersection, provided that ρ is valid.

Lemma Appendix B.6. *Suppose that $\rho = (\rho_+, \rho_-)$ is a valid valuation. Then*

$$\cap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})} = \cap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}.$$

Proof. Suppose that $\mathbf{t} \in \cap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}$, and pick any $\mathsf{A} \in \mathsf{RC}$. Let $\mathsf{B} := \mathsf{A}$: we have $\mathsf{B} \subseteq \mathsf{A}$, so $\mathbf{t} \in [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}$, and then $\mathbf{t} \in [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$. Since this is the case for all $\mathsf{A} \in \mathsf{RC}$, we conclude that

$$\cap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})} \subseteq \cap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}.$$

Now, suppose that $\mathbf{t} \in \cap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$. Pick any pair $\mathsf{B} \subseteq \mathsf{A} \in \mathsf{RC}$. Then $\mathbf{t} \in [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})}$. From Lemma Appendix B.1 and because $\mathsf{B} \subseteq \mathsf{A}$,

$$[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} \subseteq [\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}.$$

Since this is true for any pair $\mathsf{B} \subseteq \mathsf{A} \in \mathsf{RC}$, we deduce that

$$\cap_{\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{A})} \subseteq \cap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U]\!]_{\rho,(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}.$$

We therefore have the required equality. □

Remark Appendix B.7. Now, we can prove the Adequacy Lemma. In the proof, we replace the definition of $[\![\forall\vec{X}.U]\!]_\rho$ with the more precise intersection proposed in Lemma Appendix B.6.

Lemma 5.6 (Adequacy Lemma)**.** Every derivable typing judgement is valid: For every valid sequent $\Gamma \vdash \mathbf{t} : T$, we have $\Gamma \models \mathbf{t} : T$.

Proof. The proof of the adequacy lemma is made by induction on the size of the typing derivation of $\Gamma \vdash \mathbf{t} : T$. We look at the last typing rule that is used, and show in each case that $\Gamma \models \mathbf{t} : T$, i.e. if $T \equiv \mathbb{U}$, then $\mathbf{t}_\sigma \in [\![\mathbb{U}]\!]_\rho$ or if $T \equiv \sum_{i=1}^n \alpha_i.\mathbb{U}_i$ in the sense of Lemma 5.4, then $\mathbf{t}_\sigma \in \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho,\rho_i}$, for every valid valuation ρ, set of valid valuations $\{\rho_i\}_n$, and substitution $\sigma \in [\![\Gamma]\!]_\rho$ (i.e. substitution σ such that $(x : V) \in \Gamma$ implies $x_\sigma \in [\![V]\!]_{\bar{\rho}}$).

$$\frac{}{\Gamma, x : U \vdash x : U}\ ax$$

Then for any ρ, $\forall\sigma \in [\![\Gamma, x : U]\!]_\rho$ by definition we have $x_\sigma \in [\![U]\!]_{\bar{\rho}}$.From Lemma Appendix B.2, we deduce that $x_\sigma \in [\![U]\!]_\rho$.

$$\frac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \mathbf{0} : 0 \cdot T}\ 0_I$$

Note that $\forall\sigma$, $\mathbf{0}_\sigma = \mathbf{0}$, and $\mathbf{0}$ is in any reducibility candidate by $\mathbf{RC}_4$.

$$\frac{\Gamma, x : U \vdash \mathbf{t} : T}{\Gamma \vdash \lambda x.\mathbf{t} : U \to T}\ \to_I$$

Let $T \equiv \mathbb{V}$ or $T \equiv \sum_{i=1}^n \alpha_i \cdot \mathbb{U}_i$ with $n > 1$. Then by the induction hypothesis, for any ρ, set $\{\rho_i\}_n$ not acting on $FV(\Gamma) \cup FV(U)$, and $\forall\sigma \in [\![\Gamma, x : U]\!]_\rho$, we have $\mathbf{t}_\sigma \in \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho,\rho_i}$, or simply $\mathbf{t}_\sigma \in [\![\mathbb{V}]\!]_\rho$ if $T \equiv \mathbb{V}$.

In any case, we must prove that $\forall\sigma \in [\![\Gamma]\!]_\rho$, $(\lambda x.\mathbf{t})_\sigma \in [\![U \to T]\!]_{\rho,\rho'}$, or what is the same $\lambda x.\mathbf{t}_\sigma \in [\![U]\!]_{\bar{\rho},\bar{\rho}'} \to [\![T]\!]_{\rho,\rho'}$, where ρ' does not act on $FV(\Gamma)$. If we can show that $\mathbf{b} \in [\![U]\!]_{\bar{\rho},\bar{\rho}'}$ implies $(\lambda x.\mathbf{t}_\sigma)\ \mathbf{b} \in [\![T]\!]_{\rho,\rho'}$, then we are done. Notice that $[\![T]\!]_{\rho,\rho'} = \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho,\rho'}$, or $[\![T]\!]_{\rho,\rho'} = [\![\mathbb{V}]\!]_{\rho,\rho'}$ Since $(\lambda x.\mathbf{t}_\sigma)\ \mathbf{b}$ is a neutral term, we just need to prove that every one-step reduction of it is in $[\![T]\!]_\rho$, which by $\mathbf{RC}_3$ closes the case. By $\mathbf{RC}_1$, $\mathbf{t}_\sigma$ and $\mathbf{b}$ are strongly normalising, and so is $\lambda x.\mathbf{t}_\sigma$. Then we proceed by induction on the sum of the lengths of all the reduction paths starting from $(\lambda x.\mathbf{t}_\sigma)$ plus the same sum starting from $\mathbf{b}$:

$(\lambda x.\mathbf{t}_\sigma)\ \mathbf{b} \to (\lambda x.\mathbf{t}_\sigma)\ \mathbf{b}'$ with $\mathbf{b} \to \mathbf{b}'$. Then $\mathbf{b}' \in [\![U]\!]_{\bar{\rho},\bar{\rho}'}$ and we close by induction hypothesis.

$(\lambda x.\mathbf{t}_\sigma)\ \mathbf{b} \to (\lambda x.\mathbf{t}')\ \mathbf{b}$ with $\mathbf{t}_\sigma \to \mathbf{t}'$. If $T \equiv \mathbb{V}$, then $\mathbf{t}_\sigma \in [\![\mathbb{V}]\!]_{\rho,\rho'}$, and by $\mathbf{RC}_2$ so is $\mathbf{t}'$. In other case $\mathbf{t}_\sigma \in \sum_{i=1}^n [\![\mathbb{U}_i]\!]_{\rho,\rho_i}$ for any $\{\rho_i\}_n$ not acting on $FV(\Gamma)$, take $\forall i$, $\rho_i = \rho'$, so $\mathbf{t}_\sigma \in [\![T]\!]_{\rho,\rho'}$ and so are its reducts, such as $\mathbf{t}'$. We close by induction hypothesis.

$(\lambda x.\mathbf{t}_\sigma)\ \mathbf{b} \to \mathbf{t}_\sigma[\mathbf{b}/x]$ Let $\sigma' = \sigma; x \mapsto \mathbf{b}$. Then $\sigma' \in [\![\Gamma, x : U]\!]_{\rho,\rho'}$, so $\mathbf{t}_{\sigma'} \in [\![T]\!]_{\rho,\rho_i}$. Notice that $\mathbf{t}_\sigma[\mathbf{b}/x] = \mathbf{t}_{\sigma'}$.

$$\frac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^n \alpha_i \cdot \forall\vec{X}.(U \to T_i) \qquad \Gamma \vdash \mathbf{r} : \sum_{j=1}^m \beta_j \cdot U[\vec{A}_j/\vec{X}]}{\Gamma \vdash (\mathbf{t})\ \mathbf{r} : \sum_{i=1}^n \sum_{j=1}^m \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]}\ \to_E$$

Without loss of generality, assume that the T_i's are different from each other (similarly for $\vec{A}_j$). By the induction hypothesis, for any ρ, $\{\rho_{i,j}\}_{n,m}$ not acting on $FV(\Gamma)$, and $\forall\sigma \in [\![\Gamma]\!]_\rho$ we have $\mathbf{t}_\sigma \in \sum_{i=1}^{n} \cap_{\vec{\mathsf{A}}\subseteq\vec{\mathsf{B}}\in\mathsf{RC}} [\![(U \to T_i)]\!]_{\rho,\rho_i,(\vec{X}_+,\vec{X}_-)\mapsto(\vec{\mathsf{A}},\vec{\mathsf{B}})}$ and $\mathbf{r}_\sigma \in \sum_{j=1}^{m} [\![U[\vec{A}_j/\vec{X}]]\!]_{\rho,\rho_j}$, or if $n = \alpha_1 = 1$, $\mathbf{t}_\sigma \in \cap_{\vec{\mathsf{A}}\subseteq\vec{\mathsf{B}}\in\mathsf{RC}} [\![(U \to T_1)]\!]_{\rho,(\vec{X}_+,\vec{X}_-)\mapsto(\vec{\mathsf{A}},\vec{\mathsf{B}})}$ and if $m = 1$ and $\beta_1 = 1$, $\mathbf{r}_\sigma \in [\![U[\vec{A}_j/\vec{X}]]\!]_\rho$. Notice that for any $\vec{A}_j$, if U is a unit type, $U[\vec{A}_j/\vec{X}]$ is still unit.

For every i, j, let $T_i[\vec{A}_j/\vec{X}] \equiv \sum_{k=1}^{r^{ij}} \delta_k^{ij} \cdot \mathbb{W}_k^{ij}$. We must show that for any ρ, sets $\{\rho'_{i,j,k}\}_{r_{i,j}}$ not acting on $FV(\Gamma)$ and $\forall\sigma \in [\![\Gamma]\!]_\rho$, the term $((\mathbf{t})\ \mathbf{r})_\sigma$ is in the set $\sum_{i=1\cdots n, j=1\cdots m, k=1\cdots r^{ij}} [\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}}$, or in case of $n = m = \alpha_1 = \beta_1 = r^{11} = 1$, $((\mathbf{t})\ \mathbf{r})_\sigma \in [\![\mathbb{W}_1^{11}]\!]_\rho$.

Since both $\mathbf{t}_\sigma$ and $\mathbf{r}_\sigma$ are strongly normalising, we proceed by induction on the sum of the lengths of their rewrite sequence. The set $\mathrm{Red}(((\mathbf{t})\ \mathbf{r})_\sigma)$ contains:

- $(\mathbf{t}_\sigma)\ \mathbf{r}'$ or $(\mathbf{t}')\ \mathbf{r}_\sigma$ when $\mathbf{t}_\sigma \to \mathbf{t}'$ or $\mathbf{r}_\sigma \to \mathbf{r}'$. By $\mathbf{RC}_2$, the term $\mathbf{t}'$ is in the set $\sum_{i=1}^{n} \cap_{\vec{\mathsf{A}}\subseteq\vec{\mathsf{B}}\in\mathsf{RC}} [\![(U \to T_i)]\!]_{\rho,\rho_i,(\vec{X}_+,\vec{X}_-)\mapsto(\vec{\mathsf{A}},\vec{\mathsf{B}})}$ (or if $n = \alpha_1 = 1$, the term $\mathbf{t}'$ is in $\cap_{\vec{\mathsf{A}}\subseteq\vec{\mathsf{B}}\in\mathsf{RC}} [\![(U \to T_1)]\!]_{\rho,(\vec{X}_+,\vec{X}_-)\mapsto(\vec{\mathsf{A}},\vec{\mathsf{B}})}$), and $\mathbf{r}' \in \sum_{j=1}^{m} [\![U[\vec{A}_j/\vec{X}]]\!]_{\rho,\rho_j}$ (or in $[\![U[\vec{A}_1/\vec{X}]]\!]_\rho$ if $m = \beta_1 = 1$). In any case, we conclude by the induction hypothesis.
- $(\mathbf{t}_{1\sigma})\ \mathbf{r}_\sigma + (\mathbf{t}_{2\sigma})\ \mathbf{r}_\sigma$ with $\mathbf{t}_\sigma = \mathbf{t}_{1\sigma} + \mathbf{t}_{2\sigma}$, where, $\mathbf{t} = \mathbf{t}_1 + \mathbf{t}_2$. Let s be the size of the derivation of $\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i)$. By Lemma 4.12, there exists $R_1 + R_2 \equiv \sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i)$ such that $\Gamma \vdash \mathbf{t}_{1\sigma} : R_1$ and $\Gamma \vdash \mathbf{t}_{2\sigma} : R_2$ can be derived with a derivation tree of size $s-1$ if $R_1 + R_2 = \sum_{i=1}^{n} \alpha_i \cdot \forall\vec{X}.(U \to T_i)$, or of size $s-2$ in other case. In such case, there exists $N_1, N_2 \subseteq \{1, \ldots, n\}$ with $N_1 \cup N_2 = \{1, \ldots, n\}$ such that

$$R_1 \equiv \sum_{i\in N_1\setminus N_2} \alpha_i \cdot \forall\vec{X}.(U \to T_i) + \sum_{i\in N_1\cap N_2} \alpha'_i \cdot \forall\vec{X}.(U \to T_i) \quad \text{and}$$
$$R_2 \equiv \sum_{i\in N_2\setminus N_1} \alpha_i \cdot \forall\vec{X}.(U \to T_i) + \sum_{i\in N_1\cap N_2} \alpha''_i \cdot \forall\vec{X}.(U \to T_i)$$

where $\forall i \in N_1 \cap N_2$, $\alpha'_i + \alpha''_i = \alpha_i$. Therefore, using $\equiv$ we get

$$\Gamma \vdash \mathbf{t}_1 : \sum_{i\in N_1\setminus N_2} \alpha_i \cdot \forall\vec{X}.(U \to T_i) + \sum_{i\in N_1\cap N_2} \alpha'_i \cdot \forall\vec{X}.(U \to T_i) \quad \text{and}$$
$$\Gamma \vdash \mathbf{t}_2 : \sum_{i\in N_2\setminus N_1} \alpha_i \cdot \forall\vec{X}.(U \to T_i) + \sum_{i\in N_1\cap N_2} \alpha''_i \cdot \forall\vec{X}.(U \to T_i)$$

with a derivation three of size $s-1$. So, using rule $\to_E$, we get

$$\Gamma \vdash (\mathbf{t}_1)\ \mathbf{r} : \sum_{i\in N_1\setminus N_2} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] + \sum_{i\in N_1\cap N_2} \sum_{j=1}^{m} \alpha'_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] \qquad \text{and}$$
$$\Gamma \vdash (\mathbf{t}_2)\ \mathbf{r} : \sum_{i\in N_2\setminus N_1} \sum_{j=1}^{m} \alpha_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}] + \sum_{i\in N_1\cap N_2} \sum_{j=1}^{m} \alpha''_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$$

with a derivation three of size s. Hence, by the induction hypothesis the term $(\mathbf{t}_{1\sigma})\ \mathbf{r}_\sigma$ is in the set $\sum_{i=N_1, j=1\cdots m, k=1\cdots r^{ij}} [\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}}$, and the term $(\mathbf{t}_{2\sigma})\ \mathbf{r}_\sigma$ is in $\sum_{i=N_2,j=1\cdots m,k=1\cdots r^{ij}}[\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}}$. Hence, by Lemma Appendix B.3 the term $(\mathbf{t}_{1\sigma})\ \mathbf{r}_\sigma + (\mathbf{t}_{2\sigma})\ \mathbf{r}_\sigma$ is in the set $\sum_{i=1,\dots,n,j=1\cdots m,k=1\cdots r^{ij}}[\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}}$. The case where $m = \alpha_1 = \beta_1 = r^{11} = 1$, and $card(N_1)$ or $card(N_2)$ is equal to 1 follows analogously.

- $(\mathbf{t}_\sigma)\ \mathbf{r}_{1\sigma} + (\mathbf{t}_\sigma)\ \mathbf{r}_{2\sigma}$ with $\mathbf{r}_\sigma = \mathbf{r}_{1\sigma} + \mathbf{r}_{2\sigma}$. Analogous to previous case.

- $\gamma \cdot (\mathbf{t}'_\sigma)\ \mathbf{r}_\sigma$ with $\mathbf{t}_\sigma = \gamma \cdot \mathbf{t}'_\sigma$, where $\mathbf{t} = \gamma \cdot \mathbf{t}'$. Let s be the size of the derivation of $\Gamma \vdash \gamma \cdot \mathbf{t}' : \sum_{i=1}^n \alpha_i \cdot \forall\vec{X}.(U \to T_i)$. Then by Lemma 4.10, $\sum_{i=1}^n \alpha_i \cdot \forall\vec{X}.(U \to T_i) \equiv \alpha \cdot R$ and $\Gamma \vdash \mathbf{t}' : R$. If $\sum_{i=1}^n \alpha_i \cdot \forall\vec{X}.(U \to T_i) = \alpha \cdot R$, such a derivation is obtained with size $s-1$, in other case it is obtained in size $s-2$ and by Lemma 4.2, $R \equiv \sum_{i=1}^{n'} \gamma_i \cdot V_i + \sum_{k=1}^{h} \eta_k \cdot \mathbb{X}_k$, however it is easy to see that $h = 0$ because R is equivalent to a sum of terms, where none of them is $\mathbb{X}$. So $R \equiv \sum_{i=1}^{n'} \gamma_i \cdot V_i$. Notice that $\sum_{i=1}^n \alpha_i \cdot \forall\vec{X}.(U \to T_i) \equiv \sum_{i=1}^{n'} \alpha \times \gamma_i \cdot V_i$. Then by Lemma 4.4, there exists a permutation p such that $\alpha_i = \alpha \times \gamma_{p(i)}$ and $\forall\vec{X}.(U \to T_i) \equiv V_{p(i)}$. Then by rule $\equiv$, in size $s-1$ we can derive $\Gamma \vdash \mathbf{t}' : \sum_{i=1}^n \gamma_i \cdot \forall\vec{X}.(U \to T_i)$. Using rule $\to_E$, we get $\Gamma \vdash (\mathbf{t}')\ \mathbf{r} : \sum_{i=1}^n \sum_{j=1}^m \gamma_i \times \beta_j \cdot T_i[\vec{A}_j/\vec{X}]$ in size s. Therefore, by the induction hypothesis, $(\mathbf{t}'_\sigma)\ \mathbf{r}_\sigma$ is in the set $\sum_{i=1,\dots,n,j=1\cdots m,k=1\cdots r^{ij}}[\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}}$. We conclude with Lemma Appendix B.3.

- $\gamma \cdot (\mathbf{t}_\sigma)\ \mathbf{r}'_\sigma$ with $\mathbf{r}_\sigma = \gamma \cdot \mathbf{r}'_\sigma$. Analogous to previous case.

- $\mathbf{0}$ with $\mathbf{t}_\sigma = \mathbf{0}$, or $\mathbf{r}_\sigma = \mathbf{0}$. By $\mathbf{RC}_4$, $\mathbf{0}$ is in every candidate.

- The term $\mathbf{t}'_\sigma[\mathbf{r}_\sigma/x]$, when $\mathbf{t}_\sigma = \lambda x.\mathbf{t}'$ and $\mathbf{r}$ is a base term. Note that this term is of the form $\mathbf{t}'_{\sigma'}$ where $\sigma' = \sigma; x \mapsto \mathbf{r}$. We are in the situation where the types of $\mathbf{t}$ and $\mathbf{r}$ are respectively $\forall\vec{X}.(U \to T)$ and $U[\vec{A}/\vec{X}]$, and so $\sum_{i,j,k}[\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}} = \sum_{k=1}^r [\![\mathbb{W}_k]\!]_{\rho,\rho_k}$, where we omit the index "11" (or directly $[\![\mathbb{W}]\!]_\rho$ if $r = 1$). Note that

$$\lambda x.\mathbf{t}'_\sigma \in [\![\forall\vec{X}.(U \to T)]\!]_{\rho,\rho'} = \cap_{\vec{\mathsf{A}} \subseteq \vec{\mathsf{B}} \in \mathsf{RC}} [\![U \to T]\!]_{\rho,\rho',(\vec{X}_+,\vec{X}_-)\mapsto(\vec{A},\vec{B})}$$

for all possible ρ' such that $|\rho'|$ does not intersect $FV(\Gamma)$. Choose $\vec{\mathsf{A}}$ and $\vec{\mathsf{B}}$ equal to $[\![\vec{A}]\!]_{\rho,\rho'}$ and choose ρ'_- to send every X in its domain to $\cap_k \rho_{k-}(X)$ and ρ'_+ to send all the X in its domain to $\sum_k \rho_{k+}(X)$. Then by definition of $\to$ and Lemma 5.5,

$$\begin{aligned}\lambda x.\mathbf{t}'_\sigma &\in [\![U \to T]\!]_{\rho,\rho',(\vec{X}_+,\vec{X}_-)\mapsto([\![\vec{A}]\!]_{\bar\rho,\bar\rho'},[\![\vec{A}]\!]_{\rho,\rho'})} \\ &= [\![U[\vec{A}/\vec{X}]]\!]_{\bar\rho,\bar\rho'} \to [\![T]\!]_{\rho,\rho',(\vec{X}_+,\vec{X}_-)\mapsto([\![\vec{A}]\!]_{\bar\rho,\bar\rho'},[\![\vec{A}]\!]_{\rho,\rho'})}.\end{aligned}$$

Since $\mathbf{r} \in [\![U[\vec{A}/\vec{X}]]\!]_{\bar\rho,\bar\rho'}$, using Lemmas Appendix B.4 and 5.5,

$$\begin{aligned}(\lambda x.\mathbf{t}_\sigma)\ \mathbf{r} &\in [\![T]\!]_{\rho,\rho',(\vec{X}_+,\vec{X}_-)\mapsto([\![\vec{A}]\!]_{\bar\rho,\bar\rho'},[\![\vec{A}]\!]_{\rho,\rho'})} \\ &= [\![T[\vec{A}/\vec{X}]]\!]_{\rho,\rho'} \\ &= \sum_{k=1}^{n} [\![\mathbb{W}_k]\!]_{\rho,\rho'} \quad \text{or just} \quad [\![\mathbb{W}_1]\!]_{\rho,\rho'} \text{ if } n = 1.\end{aligned}$$

Now, from Lemma Appendix B.1, for all k we have $[\![\mathbb{W}_k]\!]_{\rho,\rho'} \subseteq [\![\mathbb{W}_k]\!]_{\rho,\rho_k}$. Therefore

$$(\lambda x.\mathbf{t}_\sigma)\ \mathbf{r} \in \sum_{k=1}^{n} [\![\mathbb{W}_k]\!]_{\rho,\rho_k} \quad .$$

Since the set $\mathrm{Red}(((\mathbf{t})\ \mathbf{r})_\sigma) \subseteq \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{r^{ij}} [\![\mathbb{W}_k^{ij}]\!]_{\rho,\rho_{ijk}}$, we can conclude by $\mathbf{RC}_3$.

$$\dfrac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i \qquad X \notin FV(\Gamma)}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i}\ \forall_I$$

By the induction hypothesis, for any ρ, set $\{\rho_i\}_n$ not acting on $FV(\Gamma)$, we have $\forall \sigma \in [\![\Gamma]\!]_\rho$, $\mathbf{t}_\sigma \in \sum_{i=1}^{n}[\![U_i]\!]_{\rho,\rho_i}$ (or $\mathbf{t}_\sigma \in [\![U_1]\!]_{\rho,\rho_1}$ if $n = \alpha_1 = 1$). Since $X \notin FV(\Gamma)$, we can take $\rho_i = \rho_i', (X_+, X_-) \mapsto (\mathsf{A},\mathsf{B})$, then for any $\mathsf{B} \subseteq \mathsf{A}$, we have $\mathbf{t}_\sigma \in \sum_{i=1}^{n}[\![U_i]\!]_{\rho,\rho_i',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}$ (or $\mathbf{t}_\sigma \in [\![U_1]\!]_{\rho,\rho_1',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}$ if $n = \alpha_1 = 1$).

Since it is valid for any $\mathsf{B} \subseteq \mathsf{A}$, we can take all the intersections, thus we have $\mathbf{t}_\sigma \in \sum_{i=1}^{n}\bigcap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U_i]\!]_{\rho,\rho_i',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})} = \sum_{i=1}^{n}[\![\forall X.U_i]\!]_{\rho,\rho_i'}$ (or if $n = \alpha_1 = 1$ simply $\mathbf{t}_\sigma \in \bigcap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U_1]\!]_{\rho,\rho_1',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})} = [\![\forall X.U_1]\!]_{\rho,\rho_1'}$).

$$\dfrac{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot \forall X.U_i}{\Gamma \vdash \mathbf{t} : \sum_{i=1}^{n} \alpha_i \cdot U_i[A/X]}\ \forall_E$$

By the induction hypothesis, for any ρ and for all families $\{\rho_i\}_n$, we have for all σ in $[\![\Gamma]\!]_\rho$ that the term $\mathbf{t}_\sigma$ is in $\sum_{i=1}^{n}[\![\forall X.U_i]\!]_{\rho,\rho_i} = \sum_{i=1}^{n}\bigcap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U_i]\!]_{\rho,\rho_i',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}$ (or if $n = \alpha_1 = 1$, $\mathbf{t}_\sigma$ is in the set $[\![\forall X.U_1]\!]_{\rho,\rho_1} = \bigcap_{\mathsf{B}\subseteq\mathsf{A}\in\mathsf{RC}}[\![U_1]\!]_{\rho,\rho_1',(X_+,X_-)\mapsto(\mathsf{A},\mathsf{B})}$). Since it is in the intersections, we can chose $\mathsf{A} = [\![A]\!]_{\bar\rho,\bar\rho_i}$ and $\mathsf{B} = [\![A]\!]_{\rho,\rho_i}$, and then $\mathbf{t}_\sigma \in \sum_{i=1}^{n}[\![U_i]\!]_{\rho,\rho_i',X\mapsto A} = \sum_{i=1}^{n}[\![U_i[A/X]]\!]_{\rho,\rho_i'}$ (or $\mathbf{t}_\sigma \in [\![U_1]\!]_{\rho,\rho_1',X\mapsto A} = [\![U_i[A/X]]\!]_{\rho,\rho_1'}$, if $n = \alpha_1 = 1$).

$$\dfrac{\Gamma \vdash \mathbf{t} : T}{\Gamma \vdash \alpha\cdot\mathbf{t} : \alpha\cdot T}\ \alpha_I$$

Let $T \equiv \sum_{i=1}^{n}\beta_i\cdot\mathbb{U}_i$, so $\alpha\cdot T \equiv \sum_{i=1}^{n}\alpha\times\beta_i\cdot\mathbb{U}_i$. By the induction hypothesis, for any ρ, we have $\forall\sigma \in [\![\Gamma]\!]_\rho$, $\mathbf{t}_\sigma \in \sum_{i=1}^{n}[\![\mathbb{U}_i]\!]_{\rho,\rho_i}$. By Lemma Appendix B.3, $(\alpha\cdot\mathbf{t})_\sigma = \alpha\cdot\mathbf{t}_\sigma \in \sum_{i=1}^{n}[\![\mathbb{U}_i]\!]_{\rho,\rho_i}$. Analogous if $n = \beta_1 = 1$.

$$\dfrac{\Gamma \vdash \mathbf{t} : T \qquad \Gamma \vdash \mathbf{r} : R}{\Gamma \vdash \mathbf{t}+\mathbf{r} : T+R}\ +_I$$

Let $T \equiv \sum_{i=1}^{n}\alpha_i\cdot\mathbb{U}_{i1}$ and $R \equiv \sum_{j=1}^{m}\beta_j\cdot\mathbb{U}_{j2}$. By the induction hypothesis, for any ρ, $\{\rho_i\}_n$, $\{\rho_j'\}_m$, we have $\forall\sigma \in [\![\Gamma]\!]_\rho$, $\mathbf{t}_\sigma \in \sum_{i=1}^{n}[\![\mathbb{U}_{i1}]\!]_{\rho,\rho_i}$ and $\mathbf{r}_\sigma \in \sum_{j=1}^{m}[\![\mathbb{U}_{j2}]\!]_{\rho,\rho'_j}$. Then by Lemma Appendix B.3, $(\mathbf{t}+\mathbf{r})_\sigma = \mathbf{t}_\sigma + \mathbf{r}_\sigma \in \sum_{i,k}[\![\mathbb{U}_{ik}]\!]_{\rho,\rho_i}$. Analogous if $n = \beta_1 = 1$ and/or $m = \beta_1 = 1$.

$$\dfrac{\Gamma \vdash \mathbf{t} : T \qquad T \equiv R}{\Gamma \vdash \mathbf{t} : R}\ \equiv$$

Let $T \equiv \sum_{i=1}^{n}\alpha_i\cdot\mathbb{U}_i$ in the sense of Lemma 5.4, then since $T \equiv R$, R is also equivalent to $\sum_{i=1}^{n}\alpha_i\cdot\mathbb{U}_i$, so $\Gamma \vDash \mathbf{t} : T \Rightarrow \Gamma \vDash \mathbf{t} : R$. ☐

Appendix C. Detailed proofs of lemmas and theorems in Section 6

Theorem 6.1 (Characterisation of terms)**.** Let T be a generic type with canonical decomposition $\sum_{i=1}^{n}\alpha_i.\mathbb{U}_i$, in the sense of Lemma 5.4. If $\vdash \mathbf{t} : T$, then $\mathbf{t} \to^* \sum_{i=1}^{n}\sum_{j=1}^{m_i}\beta_{ij}\cdot\mathbf{b}_{ij}$, where for all i, $\vdash \mathbf{b}_{ij} : \mathbb{U}_i$ and $\sum_{j=1}^{m_i}\beta_{ij} = \alpha_i$, and with the convention that $\sum_{j=1}^{0}\beta_{ij} = 0$ and $\sum_{j=1}^{0}\beta_{ij}\cdot\mathbf{b}_{ij} = \mathbf{0}$.

Proof. We proceed by induction on the maximal length of reduction from $\mathbf{t}$.

- Let $\mathbf{t} = \mathbf{b}$ or $\mathbf{t} = \mathbf{0}$. Trivial using Lemma 4.15 or 4.11, and Lemma 5.4.

- Let $\mathbf{t} = (\mathbf{t}_1)\ \mathbf{t}_2$. Then by Lemma 4.13, $\vdash \mathbf{t}_1 : \sum_{k=1}^{o} \gamma_k \cdot \forall\vec{X}.(U \to T_k)$ and $\vdash \mathbf{t}_2 : \sum_{l=1}^{p} \delta_l \cdot U[\vec{A}_l/\vec{X}]$, where $\sum_{k=1}^{o}\sum_{l=1}^{p} \gamma_k \times \delta_l \cdot T_k[\vec{A}_l/\vec{X}] \succeq_{\mathcal{V},\emptyset}^{(\mathbf{t}_1)\mathbf{t}_2} T$, for some $\mathcal{V}$. Without loss of generality, consider these two types to be already canonical decompositions, that is, for all k_1, k_2, $T_{k_1} \not\equiv T_{k_2}$ and for all l_1, l_2, $U[\vec{A}_{l_1}/\vec{X}] \not\equiv U[\vec{A}_{l_2}/\vec{X}]$ (in other case, it suffices to sum up the equal types). Hence, by the induction hypothesis, $\mathbf{t}_1 \to^* \sum_{k=1}^{o}\sum_{s=1}^{q_k} \psi_{ks} \cdot \mathbf{b}_{ks}$ and $\mathbf{t}_2 \to^* \sum_{l=1}^{p}\sum_{r=1}^{t_l} \phi_{lr} \cdot \mathbf{b}'_{lr}$, where for all k, $\vdash \mathbf{b}_{ks} : \forall\vec{X}.(U \to T_k)$ and $\sum_{s=1}^{q_k} \psi_{ks} = \gamma_k$, and for all l, $\vdash \mathbf{b}'_{lr} : U[\vec{A}_l/\vec{X}]$ and $\sum_{r=1}^{t_l} \phi_{lr} = \delta_l$. By rule $\to_E$, for each k, s, l, r we have $\vdash (\mathbf{b}_{ks})\ \mathbf{b}'_{lr} : T_k[\vec{A}_l/\vec{X}]$, where the induction hypothesis also apply, and notice that $(\mathbf{t}_1)\ \mathbf{t}_2 \to^* (\sum_{k=1}^{o}\sum_{s=1}^{q_k} \psi_{ks} \cdot \mathbf{b}_{ks})\ \sum_{l=1}^{p}\sum_{r=1}^{t_l} \phi_{lr} \cdot \mathbf{b}'_{lr} \to^* \sum_{k=1}^{o}\sum_{s=1}^{q_k}\sum_{l=1}^{p}\sum_{r=1}^{t_l} \psi_{ks} \times \phi_{lr} \cdot (\mathbf{b}_{ks})\ \mathbf{b}'_{lr}$. Therefore, we conclude with the induction hypothesis.

- Let $\mathbf{t} = \alpha \cdot \mathbf{r}$. Then by Lemma 4.10, $\vdash \mathbf{r} : R$, with $\alpha \cdot R \equiv T$. Hence, using Lemmas 5.4 and 4.4, R has a type decomposition $\sum_{i=1}^{n} \gamma_i \cdot \mathbb{U}_i$, where $\alpha \times \gamma_i = \alpha_i$. Hence, by the induction hypothesis, $\mathbf{r} \to^* \sum_{i=1}^{n}\sum_{j=1}^{m_i} \beta_{ij} \cdot \mathbf{b}_{ij}$, where for all i, $\vdash \mathbf{b}_{ij} : \mathbb{U}_i$ and $\sum_{j=1}^{m_i} \beta_{ij} = \gamma_i$. Notice that $\mathbf{t} = \alpha \cdot \mathbf{r} \to^* \alpha \cdot \sum_{i=1}^{n}\sum_{j=1}^{m_i} \beta_{ij} \cdot \mathbf{b}_{ij} \to^* \sum_{i=1}^{n}\sum_{j=1}^{m_i} \alpha \times \beta_{ij} \cdot \mathbf{b}_{ij}$, and $\alpha \cdot \sum_{j=1}^{m_i} \beta_{ij} = \sum_{j=1}^{m_i} \alpha \times \beta_{ij} = \alpha \times \gamma_i = \alpha_i$.

- Let $\mathbf{t} = \mathbf{t}_1 + \mathbf{t}_2$. Then by Lemma 4.12, $\vdash \mathbf{t}_1 : T_1$ and $\vdash \mathbf{t}_2 : T_2$, with $T_1 + T_2 \equiv T$. By Lemma 5.4, T_1 has canonical decomposition $\sum_{j=1}^{m} \beta_j \cdot \mathbb{V}_j$ and T_2 has canonical decomposition $\sum_{k=1}^{o} \gamma_k \cdot \mathbb{W}_k$. Hence by the induction hypothesis $\mathbf{t}_1 \to^* \sum_{j=1}^{m}\sum_{l=1}^{p_j} \delta_{jl} \cdot \mathbf{b}_{jl}$ and $\mathbf{t}_2 \to^* \sum_{k=1}^{o}\sum_{s=1}^{q_k} \epsilon_{ks} \cdot \mathbf{b}'_{ks}$, where for all j, $\vdash \mathbf{b}_{jl} : \mathbb{V}_j$ and $\sum_{l=1}^{p_j} \delta_{jl} = \beta_j$, and for all k, $\vdash \mathbf{b}'_{ks} : \mathbb{W}_k$ and $\sum_{s=1}^{q_k} \epsilon_{ks} = \gamma_k$. In for all j, k we have $\mathbb{V}_j \neq \mathbb{W}_k$, then we are done since the canonical decomposition of T is $\sum_{j=1}^{m} \beta_j \cdot \mathbb{V}_j + \sum_{k=1}^{o} \gamma_k \cdot \mathbb{W}_k$. In other case, suppose there exists j', k' such that $\mathbb{V}_{j'} = \mathbb{W}_{k'}$, then the canonical decomposition of T would be $\sum_{j=1, j\neq j'}^{m} \beta_j \cdot \mathbb{V}_j + \sum_{k=1, k\neq k'}^{o} \gamma_k \cdot \mathbb{W}_k + (\beta_{j'} + \gamma_{k'}) \cdot \mathbb{V}_{j'}$. Notice that $\sum_{l=1}^{p_{j'}} \delta_{j'l} + \sum_{s=1}^{q_{k'}} \epsilon_{k's} = \beta_{j'} + \gamma_{k'}$. $\square$