Electronic Notes in Theoretical Computer Science 1 (1995) *to appear*

Subtypes and bounded quantification from a fibred perspective

B.P.F. Jacobs

CWI
Kruislaan 413
1098 SJ Amsterdam
The Netherlands

Abstract

A general categorical description of subtyping $\sigma <: \sigma'$ and of bounded quantification $\forall\alpha <: \sigma.\, \tau$ and $\exists\alpha <: \sigma.\, \tau$ is presented in terms of fibrations. In fact, we shall generalize these bounded quantifiers to "constrained quantifiers" $\forall\alpha[\sigma <: \sigma'].\, \tau$ and $\exists\alpha[\sigma <: \sigma'].\, \tau$. In these cases one quantifies over those type variables α for which $\sigma(\alpha) <: \sigma'(\alpha)$ holds. Semantically we distinguish three levels: types τ, which are fibred over (depend on) subtypings $\sigma <: \sigma'$, which in turn are fibred over (depend on) kinds K. In this setting we can describe constrained quantification $\forall\alpha[\sigma <: \sigma'].\,(-)$ and $\exists\alpha[\sigma <: \sigma'].\,(-)$ as right and left adjoints to the weakening functor which adds the (dummy) hypothesis $\sigma <: \sigma'$ to an appropriate context. This shows that, like ordinary quantifiers, these constrained (and hence especially bounded) quantifiers are adjoints.

1 Introduction

One of the features of object-oriented programming is subtyping: for types $\sigma, \tau\colon \mathsf{Type}$, the relation $\sigma <: \tau$ expresses that σ is a subtype of τ. It means that every time a term of type τ is expected, one may also use a term of type σ. Typical examples are $\mathbb{N} <: \mathbb{Q}$ and $\mathbb{Q} <: \mathbb{R}$, but also $\sigma <: \mathrm{Top}$, where Top is the type which contains all types. For record types (labelled cartesian products) $\{\ell_1\colon\sigma_1, \ldots, \ell_n\colon\sigma_n\}$ one usually has a subtyping rule

$$\frac{\sigma_1 <: \tau_1 \qquad \cdots \qquad \sigma_n <: \tau_n}{\{\ell_1\colon\sigma_1, \ldots, \ell_n\colon\sigma_n, \ell_{n+1}\colon\rho_1, \ldots, \ell_{n+m}\colon\rho_m\} <: \{\ell_1\colon\sigma_1, \ldots, \ell_n\colon\sigma_n\}}$$

where the ℓ_i are labels.

In higher order object-oriented programming one may have higher order quantification

$$\forall\alpha\colon \mathsf{Type}.\, \tau(\alpha)\colon \mathsf{Type}.$$

This type $\forall\alpha\colon \mathsf{Type}.\, \tau(\alpha)$ is obtained by impredicative quantification over all types. One may also wish to quantify over type variables $\alpha\colon \mathsf{Type}$ which are restricted to be subtypes of a given type σ, as in

$$\forall\alpha <: \sigma.\, \tau(\alpha)\colon \mathsf{Type}.$$

This is called bounded quantification. It is called "F-bounded" quantification (see [CCHOM89]) if the type variable α is allowed to occur free in σ. This may be written explicitly as $\forall\alpha <: \sigma(\alpha).\, \tau$. One can have $\forall\alpha\colon \mathsf{Type}.\, \tau(\alpha)$ as special case, namely as $\forall\alpha <: \mathrm{Top}.\, \tau(\alpha)$. Of course, one may also have $\exists$ instead of $\forall$, see [CW85, 6.4] for use of bounded $\exists$'s in "partial data abstraction"; this is data abstraction with $\exists$ as in [MP88], but where some subtyping information about the hidden state is available.

Instead of quantifying over $\alpha\colon \mathsf{Type}$ with $\alpha <: \sigma$ one may wish to quantify over α with $\sigma <: \alpha$, i.e. over those α that are supertypes of σ. More generally, one may wish to restrict α to satisfy $\sigma(\alpha) <: \sigma'(\alpha)$, for given types $\sigma(\alpha), \sigma'(\alpha)\colon \mathsf{Type}$ possibly containing α. The above subtype and supertype cases are then special instances. We write such "constrained quantification" as $\forall\alpha[\sigma <: \sigma'].\, \tau$ and $\exists\alpha[\sigma <: \sigma'].\, \tau$. The intuitive meaning of the type $\forall\alpha[\sigma <: \sigma'].\, \tau$ is the collection of maps which take a type variable α for which $\sigma(\alpha) <: \sigma'(\alpha)$ holds to a value in $\tau(\alpha)$. Similarly, $\exists\alpha[\sigma <: \sigma'].\, \tau$ is the collection of pairs $\langle\rho, M\rangle$ where $\rho\colon \mathsf{Type}$ is a type with $\sigma[\rho/\alpha] <\text{''}\sigma'[\rho/\alpha]$ and $M\colon \tau[\rho/\alpha]$ is a term.

In [BL90] a subtyping statement $\sigma <: \sigma'$ is called a *type constraint*. And a subtyping $\alpha <: \sigma$ where α is a type variable is a *simple* type constraint. A derivation system is described for sequents of the form $\alpha_1 <: \sigma_1, \ldots, \alpha_n <: \sigma_n \vdash \tau <: \tau'$ with simple type constraints as assumptions, but arbitrary constraints as conclusion. It is both proof-theoretically and categorically more natural to have arbitrary constraints as assumptions, like in $\sigma_1 <: \sigma_1', \ldots, \sigma_n <: \sigma_n' \vdash \tau <: \tau'$, so that one has, for example, a cut rule. Once this step has been taken, it becomes natural also to quantify over these more general constraints, like with the above constrained quantifiers $\forall\alpha[\sigma <: \sigma'].\, \tau$ and $\exists\alpha[\sigma <: \sigma'].\, \tau$ [1]. One may wish to generalize these even further to multiple constraints $\forall\alpha[\sigma_1 <: \sigma_1', \ldots, \sigma_n <: \sigma_n'].\, \tau$ so that one can have quantification over intervals, like in $\forall\alpha[\sigma <: \alpha, \alpha <: \sigma'].\, \tau$.

Bruce, Cardelli and Longo [CL91,BL90] present a model of subtyping and bounded quantification (and of some other features as well) using partial equivalence relations and omega-sets. The model is described by giving the interpretation $[\![-]\!]$ of all expressions. There is thus a concrete model, but there is no general *notion of model* of subtyping or of bounded quantification. According to Cardelli and Longo: "...the invention of a general categorical meaning of subtyping and subkinds would be a relevant contribution" [CL91, top of

[1] At MFPS Luca Cardelli pushed me to investigate also these constrained quantifiers. They did not occur in the conference version of this paper, but the general subtyping sequents $\sigma_1 <: \sigma_1', \ldots, \sigma_n <: \sigma_n' \vdash \tau <: \tau'$ were already there.

page 435]). Phoa [Phoa92] goes a step further and formulates in categorical terms what a model should be. His emphasis is mostly on subsumption, by giving an axiomatization of coercion maps as certain terms. Our approach below is based on a "logic of subtyping" (via an explicit fibration of subtypings) and not on subsumption. In fact, it does not play a role at all, see Remark 4.4 (ii). Phoa uses his coercion maps to explain powerkinds $\downarrow\sigma\colon \mathsf{Kind}$ for a type $\sigma\colon \mathsf{Type}$ in terms of Bénabou's notion of definability. Bounded quantification is described via these (auxiliary) power kinds, since quantification over kinds is present in Phoa's structures. This, by the way, is also how Cardelli and Longo proceed.

Here we present a more intrinsic description of subtyping and constrained (and bounded) quantification. It also applies to situations where there are no powerkinds $\downarrow\sigma$. We have nothing against powerkinds, but we think that an explanation of subtypes and bounded quantification should not rely on them. The presentation is based on the categorical analysis of type theory given in the authors thesis [Jac91]. Fibrations play an important role in structuring the various dependencies that one may have. The key aspect is to separate contexts, notationally via vertical bars '|' like in statements

$$\alpha\colon K \mid \sigma(\alpha) <: \sigma'(\alpha) \mid x\colon\tau(\alpha) \vdash M(\alpha, x)\colon \rho(\alpha).$$

Here K is a kind, and α is a variable inhabiting this $K\colon\mathsf{Kind}$. Next, $\sigma(\alpha), \sigma'(\alpha)$ are types, (possibly) containing the variable $\alpha\colon K$. Thus

$$\alpha\colon K \vdash \sigma(\alpha)\colon \mathsf{Type} \qquad \text{and} \qquad \alpha\colon K \vdash \sigma'(\alpha)\colon \mathsf{Type}.$$

Also $\tau(\alpha), \rho(\alpha)$ are types, but in another context, namely

$$\alpha\colon K \mid \sigma(\alpha) <: \sigma'(\alpha) \vdash \tau(\alpha)\colon \mathsf{Type}$$
$$\alpha\colon K \mid \sigma(\alpha) <: \sigma'(\alpha) \vdash \rho(\alpha)\colon \mathsf{Type}.$$

We thus consider calculi where type formation may depend on subtyping statements, see Remark 4.4 (iii). So there are three levels, which will be captured by three different categories $\mathbb{B}$, $\mathbb{C}$ and $\mathbb{D}$ in two fibrations,

$$\begin{array}{l}\begin{array}{c}\mathbb{C}\\ \downarrow\\ \mathbb{B}\end{array} \ \text{subtypings } \sigma <: \sigma' \text{ over kinds } K\text{, and}\\ \quad\begin{array}{c}\mathbb{D}\\ \downarrow\\ \mathbb{C}\end{array} \ \text{types } \tau \text{ over subtypings } \sigma <: \sigma'\end{array}$$

In a next step we incorporate constrained quantification. In [Jac91] quantification was described by left ($\exists$) and right ($\forall$) adjoints to weakening functors (which is a minor adaptation of Lawvere's presentation of quantifiers as adjoints to arbitrary substitution functors). A weakening functor adds a dummy assumption A by moving from a context Γ to an extended context Γ, A. Quantification acts in the other direction: from contexts Γ, A to Γ by binding the as-

sumption A. We show how constrained (and bounded) quantification fits this pattern by taking for the assumption A a subtyping statement $\sigma(\alpha) <: \sigma'(\alpha)$, where α is a type variable, and σ, σ' are types, possibly containing α.

(To prevent circularity, we restrict the types σ, σ' occurring in subtypings $\sigma <: \sigma'$ and $\forall\alpha[\sigma <: \sigma'].\,\tau$ (or $\exists\alpha[\sigma <: \sigma'].\,\tau$) to those whose type formation does not depend on other subtypings. Categorically this can be expressed quite naturally via a change-of-base situation, as in Definition 4.3 (b).)

In this paper we describe in parallel a categorical set-up for subtyping and bounded quantification, and a concrete model, namely partial equivalence relations over omega-sets (as also used in [BL90,CL91,Phoa92]). In the end, we also show how to build a (term) model from syntax. We should emphasize that we do not introduce any new mathematical models for subtyping, but we only investigate some of the categorical aspects involved. We hope this clarifies both the syntax and semantics.

2 PERs and ω-sets

The category ω-**Sets** of omega-sets combines the set theoretic with the recursion theoretic. Its objects will be written as $I = (I, E_I)$, or simply as (I, E), where I is a set and E is an 'existence' predicate $E\colon I \to P\mathbb{N}$, such that for each $i \in I$ the set $E(i) \subseteq \mathbb{N}$ is non-empty. A morphism $u\colon (I, E_I) \to (J, E_J)$ is a function $u\colon I \to J$ between the underlying sets which is 'tracked': for some $e \in \mathbb{N}$ one has $\forall i \in I. \forall n \in E_I(i).\, e\cdot n \in E_J(u(i))$, where $e \cdot n$ is Kleene's application of the e-th recursive function to n. It is not hard to verify that ω-**Sets** is cartesian closed. There is a functor $\nabla\colon$ **Sets** $\to$ ω-**Sets** which maps a set I to I with the existence predicate $I \to P\mathbb{N}$ which is constantly $\mathbb{N}$. This ∇ is right adjoint to the forgetful functor ω-**Sets** $\to$ **Sets**.

A **partial equivalence relation** (PER) on $\mathbb{N}$ is a subset $R \subseteq \mathbb{N}\times\mathbb{N}$ which is symmetric and transitive. We write

$$\begin{aligned} |R| &= \{n \in \mathbb{N} \mid nRn\} && \text{for the domain of } R\\ [n]_R &= \{m \in \mathbb{N} \mid mRn\} && \text{for the } R\text{-class of } n \in \mathbb{N}\\ \mathbb{N}/R &= \{[n]_R \mid n \in |R|\} && \text{for the quotient of } R\\ \mathrm{PER} &= \{R \subseteq \mathbb{N}\times\mathbb{N} \mid R \text{ is symmetric and transitive}\}. \end{aligned}$$

(Notice that R is an equivalence relation on its domain $|R|$, so we should write $|R|/R$ instead of $\mathbb{N}/R$. But the latter is clearer.)

Every PER R yields an ω-set $(\mathbb{N}/R, \in)$, with $\in([n]_R) = [n]_R$. We write **PER** for the category with PERs as objects, and maps $(\mathbb{N}/R, \in) \to (\mathbb{N}/S, \in)$ in ω-**Sets** as morphisms $R \to S$. By construction there is a full and faithful functor **PER** $\hookrightarrow$ ω-**Sets**. One can identify the PERs inside ω-**Sets** as those (I, E) where $E(i) \cap E(j) = \emptyset$ for $i \neq j$. Forcing images of existence predicates to be disjoint yields a left adjoint to **PER** $\hookrightarrow$ ω-**Sets**.

We shall mostly be interested in PERs indexed by ω-sets. For an ω-set (I, E) there is a 'fibre' category of (I, E)-indexed PERs with

objects I-indexed collections $(R_i)_{i \in I}$ of PERs R_i. These may be described as maps $R: (I, E) \to \nabla\mathrm{PER}$ in ω-**Sets**.

morphisms $(R_i)_{i \in I} \to (S_i)_{i \in I}$ are I-indexed collections $f = (f_i)_{i \in I}$ of functions $f_i: \mathbb{N}/R_i \to \mathbb{N}/S_i$ which are 'tracked uniformly': for some $e \in \mathbb{N}$,

$$\forall i \in I. \forall n \in E_I(i).\, e \cdot n \text{ tracks } f_i$$

i.e.

$$\forall i \in I. \forall n \in E_I(i). \forall m \in |R_i|.\, f_i([m]_{R_i}) = [e \cdot n \cdot m]_{S_i}.$$

Mapping an object (I, E) in ω-**Sets** to this fibre category of (I, E)-indexed PERs yields a functor $\omega\text{-}\mathbf{Sets}^{\mathrm{op}} \to \mathbf{Cat}$, with for a morphism $u: (I, E) \to (J, E)$ in ω-**Sets** a 'substitution' functor u^* given by composition. Applying the Grothendieck construction yields a split fibration, which we write as $\begin{array}{c}\mathrm{UFam}(\mathbf{PER})\\ \downarrow\\ \omega\text{-}\mathbf{Sets}\end{array}$, where 'UFam' stands for 'uniform families'. This fibration is a standard model of higher order polymorphic λ-calculus ('$\lambda\omega$' or '$F\omega$'); it is due to Moggi and Hyland, see [Hyl88]. One thinks of ω-sets $I = (I, E)$ as kinds $I: \mathsf{Kind}$, and of (I, E)-indexed PERs $(R_i)_{i \in I}$ as types in kind context I, i.e. as $i: I \vdash R_i: \mathsf{Type}$. Morphisms $f: R \to S$ over I are terms $i: I \mid x: R_i \vdash f_i(x): S_i$ involving variables $i: I$ in a kind and $x: R_i$ in a type.

3 Subtypes

The standard way to describe subtyping for PERs $R, S \in \mathrm{PER}$, considered as types, is via inclusion $R \subseteq S$ as relations on $\mathbb{N}$. We formulate this in the following category of conditional subtypings.

Definition 3.1 *Let* $\mathbf{PER}_{<:}$ *be the category of "indexed PERs and inclusions". It has*

objects *triples* (I, R, R') *where* I *is an* ω*-set and* R, R' *are* I*-indexed PERs. Hence we may describe such an object as a pair of parallel maps* $R, R': (I, E) \rightrightarrows \nabla\mathrm{PER}$ *in* ω-**Sets**.

morphisms $(I, R, R') \to (J, S, S')$ *are morphisms* $u: (I, E) \to (J, E)$ *of* ω*-sets for which one has*

$$R_i \subseteq R'_i \;\Rightarrow\; S_{u(i)} \subseteq S'_{u(i)}$$

for all $i \in I$.

There is an obvious projection functor $\mathbf{PER}_{<:} \to \omega\text{-}\mathbf{Sets}$, namely $(I, R, R') \mapsto I$. This is a fibration, which we write as $\begin{array}{c}\mathbf{PER}_{<:}\\ \downarrow{\scriptstyle\mathrm{Fst}}\\ \omega\text{-}\mathbf{Sets}\end{array}$. Every fibre is a poset. We think of $(R, R') \leq (S, S')$ in the fibre over I as an entailment $i: I \mid R_i <: R'_i \vdash S_i <: S'_i$.

Notice that the inclusion relation $\subseteq$ on PERs may be described as a (regular) mono in ω-**Sets**,

$$\subseteq \rightarrowtail \nabla\mathrm{PER} \times \nabla\mathrm{PER}.$$

Using this mono we can describe a morphism $u: (I, R, R') \to (J, S, S')$ alternatively as a map $u: (I, E) \to (J, E)$ in ω-**Sets** for which one has a necessarily unique map $\dashrightarrow$ in

$$\begin{tikzcd}
\subseteq \arrow[d, tail] & \{R <: R'\} \arrow[l] \arrow[r, dashed] \arrow[d, tail, "\iota(R, R')"] & \{S <: S'\} \arrow[r] \arrow[d, tail, "\iota(S, S')"'] & \subseteq \arrow[d, tail] \\
\nabla\mathrm{PER}^2 & (I, E) \arrow[l, "\langle R, R' \rangle"] \arrow[r, "u"'] & (J, E) \arrow[r, "\langle S, S' \rangle"'] & \nabla\mathrm{PER}^2
\end{tikzcd}$$

Here we have suggestively written

$\{R <: R'\}$ for the ω-set $\{i \in I \mid R_i \subseteq R'_i\}$, with E as on I.

Notice that the assignment

$$(I, R, R') \mapsto \begin{pmatrix} \{R <: R'\} \\ \downarrow{\scriptstyle\iota(R,R')} \\ (I, E) \end{pmatrix}$$

yields a full and faithful fibred functor in a situation

$$\begin{tikzcd}
\mathbf{PER}_{<:} \arrow[rr, "\iota"] \arrow[dr, "\mathrm{Fst}"'] & & \omega\text{-}\mathbf{Sets}^{\to} \arrow[dl, "cod"] \\
& \omega\text{-}\mathbf{Sets} &
\end{tikzcd}$$

(It may be described as a 'full comprehension category with unit', in the terminology of [Jac91]. This functor ι actually restricts to a functor

$$\begin{tikzcd}
\mathbf{PER}_{<:} \arrow[rr, "\iota"] \arrow[dr, "\mathrm{Fst}"'] & & \mathrm{RegSub}(\omega\text{-}\mathbf{Sets}) \arrow[dl, "cod"] \\
& \omega\text{-}\mathbf{Sets} &
\end{tikzcd}$$

which tells that subtypings form a "sublogic" of the (classical) logic of regular subobjects over ω-**Sets**. Indeed, the fibration Fst on the left captures a logic of subtypings for indexed PERs.)

We write $\{-\}: \mathbf{PER}_{<:} \to \omega\text{-}\mathbf{Sets}$ for the functor $(I, R, R') \mapsto \{R <: R'\}$.

Definition 3.2 *Write* $\mathrm{Top} = \mathbb{N} \times \mathbb{N}$ *for the maximal PER, with respect to* $\subseteq$, *and* $\mathrm{Top}_I = \mathrm{Top}_{(I,E)} = (\mathrm{Top})_{i \in I}$ *for the* I*-indexed* Top. *Put* $\top_I = \top_{(I,E)} = (I, \mathrm{Top}_I, \mathrm{Top}_I) \in \mathbf{PER}_{<:}$ *over* I.

For PERs R, S *let* $R \dot\cup S$ *be the PER*

$$R \dot\cup S = \{(\langle n, 0\rangle, \langle m, 0\rangle) \mid nRm\} \cup \{(\langle n, 1\rangle, \langle m, 1\rangle) \mid nSm\}$$

where $\langle -,- \rangle$ *is a recursive coding* $\mathbb{N}\times\mathbb{N} \xrightarrow{\sim} \mathbb{N}$*. We use* $\dot\cup$ *to define a meet* $\wedge$ *in the fibres of* $\mathbf{PER}_{<:}$ *by*

$$\big(I, R, R'\big) \wedge \big(I, S, S'\big) = \big(I, (R_i \mathbin{\dot\cup} S_i)_{i\in I}, (R'_i \mathbin{\dot\cup} S'_i)_{i\in I}\big).$$

Lemma 3.3 *The operations* $\top$, $\wedge$ *yield fibred finite products (meets) in the fibration* $\begin{array}{c}\mathbf{PER}_{<:}\\ \downarrow\\ \omega\text{-}\mathbf{Sets}\end{array}$*: one has meets* $(\top, \wedge)$ *in every fibre, and these are preserved by substitution functors* u^*.

With these we can express reflexivity and transitivity of subtypings (in context $I \in \omega$-**Sets***) as morphisms over* I*:*

$$\top_I \leq (I, R, R) \qquad \text{and} \qquad (I, R, R') \wedge (I, R', R'') \leq (I, R, R'').$$

Proof. One has $(R, R') \leq \top$ over I, since $\mathrm{Top} \subseteq \mathrm{Top}$ always holds. Further one has over I,

$$\begin{array}{l}
(Q, Q') \leq (R, R') \text{ and } (Q, Q') \leq (S, S') \\
\quad \text{iff} \;\; \forall i \in I.\; Q_i \subseteq Q'_i \;\Rightarrow\; (R_i \subseteq R'_i \text{ and } S_i \subseteq S'_i) \\
\quad \text{iff} \;\; \forall i \in I.\; Q_i \subseteq Q'_i \;\Rightarrow\; R_i \mathbin{\dot\cup} S_i \subseteq R'_i \mathbin{\dot\cup} S'_i \\
\quad \text{iff} \;\; (Q, Q') \leq (R, R') \wedge (S, S').
\end{array}$$

By the pointwise definition of $\top$ and $\wedge$, substitution functors preserve these meets. □

There is one further aspect of $\begin{array}{c}\mathbf{PER}_{<:}\\ \downarrow\\ \omega\text{-}\mathbf{Sets}\end{array}$ that we wish to axiomatize. Recall that a split fibration $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ has a **split generic object** if there is an object $\Omega \in \mathbb{B}$ with an isomorphism

$$\mathbb{B}\big(I,\ \Omega\big) \xrightarrow[\cong]{\varphi_I} \mathrm{Obj}\,\mathbb{E}_I$$

which is natural in I: for $u\colon J \to I$ one has $\varphi_J(v \circ u) = u^*\varphi_I(v)$. Note that the object $\nabla\mathrm{PER} \times \nabla\mathrm{PER} \in \omega$-**Sets** is a split generic object for the fibration $\begin{array}{c}\mathbf{PER}_{<:}\\ \downarrow\\ \omega\text{-}\mathbf{Sets}\end{array}$ of PER-inclusions: the required isomorphisms φ_I are simply identities. And the object $\nabla\mathrm{PER} \in \omega$-**Sets** is split generic object for the fibration $\begin{array}{c}\mathrm{UFam}(\mathbf{PER})\\ \downarrow\\ \omega\text{-}\mathbf{Sets}\end{array}$ of PERs over ω-sets.

Each fibre category of this latter fibration is cartesian closed, via definitions of $\mathrm{Top}, \times, \Rightarrow$ which are pointwise as on **PER**. This cartesian closed structure is related to the finite meets $(\top, \wedge)$ between PER inclusions via the following three inequalities.

$$\begin{aligned}
\top &\leq (R, \mathrm{Top}) \\
(R, R') \wedge (S, S') &\leq (R \times S, R' \times S') \\
(R', R) \wedge (S, S') &\leq (R \Rightarrow S, R' \Rightarrow S').
\end{aligned}$$

They correspond to the familiar axioms in the logic of subtyping:

$$\begin{array}{c}
i\colon I \mid \emptyset \;\vdash\; R_i <: \mathrm{Top} \\
i\colon I \mid R_i <: S_i, R'_i <: S'_i \;\vdash\; (R \times S)_i <: (R' \times S')_i \\
i\colon I \mid R'_i <: R_i, S_i <: S'_i \;\vdash\; (R \Rightarrow S)_i <: (R' \Rightarrow S')_i.
\end{array}$$

We summarize the structure that we have found in the following definition. For convenience we restrict ourselves to split fibrations with split structure, without always saying so explicitly. What we call a $\lambda{\to}$**-fibration** is a fibration $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ which is a fibred CCC, has finite products in its base category $\mathbb{B}$, and has a generic object. As we have seen, $\begin{array}{c}\mathrm{UFam}(\mathbf{PER})\\ \downarrow\\ \omega\text{-}\mathbf{Sets}\end{array}$ is such a $\lambda{\to}$-fibration.

Definition 3.4 *A* **subtyping fibration** *for a* $\lambda{\to}$*-fibration* $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ *consists of another fibration* $\begin{array}{c}\mathbb{C}\\ \downarrow q\\ \mathbb{B}\end{array}$ *on* $\mathbb{B}$ *which*

- *is a preorder fibration, i.e. has preorder fibre categories;*
- *has fibred finite meets* $(\top, \wedge)$*;*
- *has a generic object* $\Omega \times \Omega \in \mathbb{B}$*—where* Ω *is the generic object of* p*;*
- *satisfies for objects* $X, X', X'', Y, Y' \in \mathbb{E}$ *in the same fibre*

$$\begin{aligned}
\top &\leq (X, X) \\
(X, X') \wedge (X', X'') &\leq (X, X'') \\
\top &\leq (X, \mathrm{Top}) \\
(X, X') \wedge (Y, Y') &\leq (X \times Y, X' \times Y') \\
(X', X) \wedge (Y, Y') &\leq (X \Rightarrow Y, X' \Rightarrow Y'),
\end{aligned}$$

where we have identified a pair of objects (X, X') *in* $\mathbb{E}$ *over* I *with an object of* $\mathbb{C}$ *over* I*. This can be done by the previous requirement.*

Remark 3.5 *In the PER model we have been able to capture multiple inclusions* $R \subseteq R'$ *and* $S \subseteq S'$ *as a single inclusion* $R \mathbin{\dot\cup} S \subseteq R' \mathbin{\dot\cup} S'$*. If such possibility does not exist, we have to modify the generic object requirement for* $\begin{array}{c}\mathbb{C}\\ \downarrow q\\ \mathbb{B}\end{array}$ *a bit: then we require that this fibration has a* **family of generic objects**

$$\textstyle\coprod_{n\in\mathbb{N}} \mathbb{B}\big(I,\ (\Omega\times\Omega)^n\big) \xrightarrow[\cong]{\varphi_I} \mathrm{Obj}\,\mathbb{C}_I$$

natural in I*. If we write* $G_n = \varphi_{(\Omega\times\Omega)^n}(\langle n, id\rangle) \in \mathbb{E}$ *over* $(\Omega \times \Omega)^n$*, then for each* $A \in \mathbb{C}$ *there is a unique* $n \in \mathbb{N}$ *and* $u\colon qA \to (\Omega\times\Omega)^n$ *such that* $A = u^*(G_n)$*.*

For a fibration $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ write the pullback of p against itself as

$$\begin{tikzcd} \mathbb{E}^2 = \mathbb{E}\times_{\mathbb{B}}\mathbb{E} \arrow[r] \arrow[d] \arrow[dr, "p^2"] & \mathbb{E} \arrow[d, "p"] \\ \mathbb{E} \arrow[r, "p"'] & \mathbb{B} \end{tikzcd}$$

The resulting fibration $\begin{array}{c}\mathbb{E}^2\\ \downarrow p^2\\ \mathbb{B}\end{array}$ is then the cartesian product $p \times p$ in the 2-category of split fibrations over $\mathbb{B}$. For a split fibration $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ we write $\mathrm{Split}(\mathbb{E}) \hookrightarrow \mathbb{E}$ for the category with the same objects as $\mathbb{E}$, but with only the splittings as morphisms.

Lemma 3.6 *Assume fibrations* $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ *and* $\begin{array}{c}\mathbb{C}\\ \downarrow q\\ \mathbb{B}\end{array}$ *as in the previous definition, with generic object isomorphisms*

$$\mathbb{B}\big(I,\ \Omega\big) \xrightarrow[\cong]{\varphi_I} \mathrm{Obj}\,\mathbb{E}_I \qquad\qquad \mathbb{B}\big(I,\ \Omega\times\Omega\big) \xrightarrow[\cong]{\psi_I} \mathrm{Obj}\,\mathbb{C}_I.$$

Then we can define a functor $\mathcal{I}$ in

$$\begin{tikzcd} \mathrm{Split}(\mathbb{E})^2 \arrow[rr, "\mathcal{I}"] \arrow[dr] & & \mathrm{Split}(\mathbb{C}) \arrow[dl] \\ & \mathbb{B} & \end{tikzcd}$$

by

$$(X, X') \mapsto \psi_I(\langle \varphi_I^{-1}(X), \varphi_I^{-1}(X')\rangle)$$

for $X, X' \in \mathbb{E}_I$.

Proof. We get a functor since for a morphism $(u^*(X), u^*(X')) \to (X, X')$ in $\mathrm{SPlit}(\mathbb{E})^2$ over $u\colon J \to I$ we have $\mathcal{I}(u^*(X), u^*(X')) = u^*\mathcal{I}(X, X')$ by naturality of φ and ψ. □

Definition 3.7 *Let* $\begin{array}{c}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{array}$ *and* $\begin{array}{c}\mathbb{C}\\ \downarrow q\\ \mathbb{B}\end{array}$ *be as in Definition 3.4. Form the fibration* $\begin{array}{c}\widehat{\mathbb{C}}\\ \downarrow\\ \mathbb{C}\end{array}$ *by change-of-base in*

$$\begin{tikzcd} \widehat{\mathbb{C}} \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & \bullet \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & \mathrm{Split}(\mathbb{E})^2 \arrow[d] \\ \mathbb{C} \arrow[r, "q"'] & \mathbb{B} \arrow[r, "(-)\times\Omega"'] & \mathbb{B} \end{tikzcd}$$

An object of $\widehat{\mathbb{C}}$ is thus a triple of objects $A \in \mathbb{C}$, $X \in \mathbb{E}$, $X' \in \mathbb{E}$ with $qA \times \Omega = pX = pX'$ in $\mathbb{B}$. We can define a functor $\mathcal{P}$ in

$$\begin{tikzcd} \widehat{\mathbb{C}} \arrow[rr, "\mathcal{P}"] \arrow[dr] & & \mathbb{C}^{\to} \arrow[dl] \\ & \mathbb{C} & \end{tikzcd}$$

by $(X, X, X') \mapsto \big($the composite $\pi^*(A) \wedge \mathcal{I}(X, X') \leq \pi^*(A) \to A\big)$, where the functor $\mathcal{I}$ comes from the previous lemma.

This functor $\mathcal{P}$ constitutes a "comprehension category", in the terminology of [Jac91]. It provides us with abstract projections $\mathcal{P}(A, X, X') = (\pi^*(A) \wedge \mathcal{I}(X, X') \to A)$ in $\mathbb{C}$, along which we can quantify. Intuitively, these projections are maps between subtyping contexts

$$\big(\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma(\alpha) <: \sigma'(\alpha)\big) \longrightarrow \big(\Gamma \mid \Theta\big)$$

see Section 5.

We close this section by describing the functor $\mathcal{P}$ for our running example of PERs over ω-**Sets**. It maps a subtyping $A = (S, S'\colon (I, E) \rightrightarrows \nabla\mathrm{PER})$ in $\mathbf{PER}_{<:}$ and two $(I, E) \times \nabla\mathrm{PER}$-indexed PERs R, R' to the projection

$$\begin{array}{c} \pi^*(A) \wedge \mathcal{I}(R, R') = \big((S_i \mathbin{\dot\cup} R_{i,X})_{i\in I, X\in\mathrm{PER}}, (S'_i \mathbin{\dot\cup} R'_{i,X})_{i\in I, X\in\mathrm{PER}}\big) \\ \Big\downarrow \mathcal{P}(A, R, R') \\ A \end{array}$$

in $\mathbf{PER}_{<:}$. It is given by the underlying projection $\pi\colon (I, E) \times \nabla\mathrm{PER} \to (I, E)$ in ω-**Sets**, since we have

$$S_i \mathbin{\dot\cup} R_{i,X} \subseteq S'_i \mathbin{\dot\cup} R'_{i,X} \;\Rightarrow\; S_i \subseteq S'_i.$$

4 Constrained quantification

In [Jac91] one finds how a fibration $\begin{array}{c}\mathbb{D}\\ \downarrow q\\ \mathbb{C}\end{array}$ may have quantification with respect to a comprehension category $\mathcal{P}\colon \mathbb{E} \to \mathbb{C}^{\to}$. This means that for each $X \in \mathbb{E}$, the "weakening functor" $\mathcal{P}X^*$ between the fibres of $\mathbb{D}$—induced by the projection map $\mathcal{P}X$ in $\mathbb{B}$—has a left/right adjoint (plus a Beck-Chevalley condition, which regulates the proper distrubution of substitution over the quantifiers). We shall show that this abstract set-up gives the right level of generality to describe constrained (and thus F-bounded) quantification in terms of adjunctions, by applying it to the comprehension category $\mathcal{P}\colon \widehat{\mathbb{C}} \to \mathbb{C}^{\to}$ that we introduced in the previous section.

We need the fibration of PERs indexed by subtyping statements. It is obtained by change-of-base:

$$\begin{tikzcd}
\mathrm{UFam}_{<:}(\mathbf{PER}) \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner"] & \mathrm{UFam}(\mathbf{PER}) \arrow[d] \\
\mathbf{PER}_{<:} \arrow[r, "\{-\}"'] & \omega\text{-}\mathbf{Sets}
\end{tikzcd}$$

Recall from Section 3 that the functor $\{-\}$ maps $(I, R, R') = (R, R' : I \rightrightarrows \mathrm{PER})$ to the ω-set $\{R <: R'\} = \{i \in I \mid R_i \subseteq R'_i\}$, with existence E as on I. An object of $\mathrm{UFam}_{<:}(\mathbf{PER})$ over $(I, E) \in \omega$-**Sets** thus consists of a 3-tuple (R, R', U), where $R, R' : (I, E) \rightrightarrows \nabla\mathrm{PER}$ are (I, E)-indexed PERs, and U is an $\{R <: R'\}$-indexed PER $U : \{i \in I \mid R_i \subseteq R'_i\} \to \mathrm{PER}$. It clearly depends on an inclusion.

Proposition 4.1 *The PER-model has both constrained products $\prod$ and constrained coproducts $\coprod$. More precisely, the fibration* $\begin{smallmatrix}\mathrm{UFam}_{<:}(\mathbf{PER})\\ \downarrow \\ \mathbf{PER}_{<:}\end{smallmatrix}$ *has products and coproducts with respect to the comprehension category*

$$\widehat{\mathbf{PER}}_{<:} \longrightarrow \mathbf{PER}_{<:}^{\to}.$$

Proof. Assume $A = (S, S' : (I, E) \rightrightarrows \nabla\mathrm{PER}) \in \mathbf{PER}_{<:}$, and $R, R' : I \rightrightarrows \mathrm{PER}$, as at the end of the previous section. For a family U of PERs in $\mathrm{UFam}_{<:}(\mathbf{PER})$ over $\mathrm{dom}\mathcal{P}(A, R, R') = \pi^*(A) \wedge \mathcal{I}(R, R') \in \mathbf{PER}_{<:}$, we have U as a map,

$$\{(i, X) \in I \times \mathrm{PER} \mid S_i \subseteq S'_i \text{ and } R_{i,X} \subseteq R'_{i,X}\} \xrightarrow{\;U\;} \mathrm{PER}$$

We have to define product $\prod_{(A,R,R')}(U)$ and coproduct $\coprod_{(A,R,R')}(U)$ objects in $\mathrm{UFam}_{<:}(\mathbf{PER})$ over $\mathrm{cod}\mathcal{P}(A, R, R') = A \in \mathbf{PER}_{<:}$. They thus must be maps

$$\begin{tikzcd}
\{i \in I \mid S_i \subseteq S'_i\} \arrow[r, bend left, "\prod_{(A,R,R')}(U)"] \arrow[r, bend right, "\coprod_{(A,R,R')}(U)"'] & \mathrm{PER}
\end{tikzcd}$$

They are defined as

$$\textstyle\prod_{(A,R,R')}(U)_i = \bigcap_{X \in \mathrm{PER}} \{U_{i,X} \mid R_{i,X} \subseteq R'_{i,X}\}$$

$$\textstyle\coprod_{(A,R,R')}(U)_i = \bigvee_{X \in \mathrm{PER}} \{U_{i,X} \mid R_{i,X} \subseteq R'_{i,X}\}$$

where $\bigcap$ and $\bigvee$ are the meet and join in the complete lattice $(\mathrm{PER}, \subseteq)$. The adjunctions $\mathcal{P}(A, R, R')^* \dashv \prod_{(A,R,R')}$ and $\coprod_{(A,R,R')} \dashv \mathcal{P}(A, R, R')^*$ involve bijective correspondences

$$\frac{\mathcal{P}(A, R, R')^*(V) \longrightarrow U \quad \text{over } \pi^*(A) \wedge \mathcal{I}(R, R')}{V \longrightarrow \prod_{(A,R,R')}(U) \quad \text{over } A}$$

$$\frac{U \longrightarrow \mathcal{P}(A, R, R')^*(V) \quad \text{over } \pi^*(A) \wedge \mathcal{I}(R, R')}{\coprod_{(A,R,R')}(U) \longrightarrow V \quad \text{over } A}$$

where the weakening functor $\mathcal{P}(A, R, R')^*$ moves V to a bigger context:

$$\Big(\{i \in I \mid S_i \subseteq S'_i\} \xrightarrow{\;V\;} \mathrm{PER}\Big)$$
$$\mapsto \Big(\{(i, X) \in I \times \mathrm{PER} \mid S_i \subseteq S'_i \text{ and } R_{i,X} \subseteq R'_{i,X}\} \xrightarrow{\;V \circ \pi\;} \mathrm{PER}\Big).$$

□

The constrained products of PERs are thus obtained by intersection. Not over all PERs, like in $\forall\alpha : \mathsf{Type}.\, \tau(\alpha)$, but over PERs appropriately restricted.

Example 4.2 *(i) [From [BL90]]. For a (closed) type σ, interpreted as $R \in$ PER, consider the type*

$$\forall\alpha <: \sigma.\, \alpha \to \alpha \equiv \forall\alpha[\alpha <: \sigma].\, \alpha \to \alpha.$$

It is interpreted as the intersection

$$S = \Big(\bigcap_{X \subseteq R} X \to X\Big) \in \mathrm{PER},$$

where $X \to X = \{(m, m') \mid \forall k, k' \in \mathbb{N}.\, kXk' \Rightarrow m \cdot kXm' \cdot k'\}$.

Assume e is an element of the domain $|S|$ of S. For each $n \in |R|$ we have a sub-PER,

$$X_n = \{(n, n)\} \subseteq R$$

so that

$$e \in |X_n \to X_n|.$$

But then $e \cdot n = n$. Hence e is a code for the identity map on $R \in$ **PER**. *The only term in $\forall\alpha <: \sigma.\, \alpha \to \alpha$ is thus the identity on σ.*

(ii) In the PER-model one has that quantification over a "singleton" interval in $\forall\alpha[\mathrm{Top} <: \alpha].\, \tau$ and $\exists\alpha[\mathrm{Top} <: \alpha].\, \tau$ yields $\tau[\mathrm{Top}/\alpha]$, since one takes the meet or join of the set

$$\{U_{i,X} \mid \mathrm{Top} \subseteq X\} = \{U_{i,\mathrm{Top}}\}.$$

The abstract structure that we recognize in the PER-model is axiomatized as follows.

Definition 4.3 *A setting for constrained quantification is given by two fibrations* $\begin{smallmatrix}\mathbb{D}\\ \downarrow r\\ \mathbb{C}\end{smallmatrix}$ *and* $\begin{smallmatrix}\mathbb{C}\\ \downarrow q\\ \mathbb{B}\end{smallmatrix}$ *where*

(a) r is a fibred CCC;

(b) q has a fibred terminal object $\top : \mathbb{B} \to \mathbb{C}$ such that q is a subtyping fibration for the fibration $p = \top^*(r)$ obtained the change-of-base situation

$$\begin{tikzcd}
\mathbb{D} \arrow[d, "r"'] & \mathbb{E} \arrow[l] \arrow[d, "p = \top^*(r)"] \arrow[dl, phantom, "\llcorner"] \\
\mathbb{C} & \mathbb{B} \arrow[l, "\top"']
\end{tikzcd}$$

This p is the fibration of types which do not depend on subtypings.

The fibration r then has constrained products / coproducts if it has products / coproducts with respect to the induced comprehension category $\mathcal{P}: \widehat{\mathbb{C}} \to \mathbb{C}^{\to}$ from Definition 3.7.

We note that this set-up indeed captures the PER models since there is a change-of-base sitution

$$\begin{tikzcd}
\mathrm{UFam}_{<:}(\mathbf{PER}) \arrow[d] & \mathrm{UFam}(\mathbf{PER}) \arrow[l] \arrow[d] \arrow[dl, phantom, "\lrcorner"] \\
\mathbf{PER}_{<:} & \omega\text{-}\mathbf{Sets} \arrow[l, "\top"']
\end{tikzcd}$$

because $\{-\} \circ \top \cong id: \omega\text{-}\mathbf{Sets} \to \omega\text{-}\mathbf{Sets}$.

A reader with experience in categorical type theory will (roughly) see how to interpret a polymorphically typed calculus with subtyping and constrained quantification in a structure as in the above definition. But actually carrying out such an interpretation may be complicated, due to coherence problems induced by the possibility of different derivations for a single term formation statement, see [BCGS91].

Remark 4.4 *(i) It is not hard to verify that in a situation as in the definition, a projection $\pi: I \times \Omega \to I$ in $\mathbb{B}$ is mapped by the terminal object functor $\top: \mathbb{B} \to \mathbb{C}$ to a projection $\mathcal{P}(\top(I), \mathrm{Top}_I, \mathrm{Top}_I): \pi^*(\top(I)) \wedge \mathcal{I}(\mathrm{Top}_I, \mathrm{Top}_I) \to \top(I)$ of the comprehension category. This yields that the fibration $\begin{smallmatrix}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{smallmatrix}$ has products and coproducts along the projections $I \times \Omega \to I$ in $\mathbb{B}$, and thus p becomes a $\lambda 2$-fibration. This is a categorical way of saying that with constrained quantification $\forall\alpha[\sigma <: \sigma'].\tau$ one also has second order quantification $\forall\alpha: \mathsf{Type}.\tau$ as $\forall\alpha[\mathrm{Top} <: \mathrm{Top}].\tau$.*

(ii) The **subsumption rule**

$$\frac{\Gamma \mid \Theta \vdash \sigma <: \sigma'}{\Gamma \mid \Theta \mid x: \sigma \vdash \mathrm{c}_{\sigma,\sigma'}(x): \sigma'}$$

where $\mathrm{c}_{\sigma,\sigma'}$ is a coercer map, does not play a role in the above exposition. The subtyping fibration gives a logic with certain relations between types (namely subtyping) which are used as restrictions in quantification. This is a distinctly logical approach. In the PER-model the subsumption rule is valid (as shown in [BL90,CL91]): if we have a map in $\mathbf{PER}_{<:}$ over $I \in \omega\text{-}\mathbf{Sets}$

$$(S, S') \le (R, R')$$

then there is a coercer map $R \to R'$ in $\mathrm{UFam}_{<:}(\mathbf{PER})$ over $\{S <: S'\}$, namely

$$[n]_{R_i} \mapsto [n]_{R'_i}$$

for $i \in I$ with $S_i \subseteq S'_i$.

(iii) In our categorical analysis we have explicitly included the possibility that type formation $\tau: \mathsf{Type}$ depends on subtyping $\sigma <: \sigma'$. For example, if we have a dependent type $n: \mathbb{N} \vdash \mathsf{List}(n): \mathsf{Type}$ of lists of length n (of some fixed type), then we can consider the type

$$\alpha: \mathsf{Type} \mid \alpha <: \mathbb{N} \mid n: \mathbb{N}, m: \alpha \vdash \mathsf{List}(n + \mathrm{c}_{\alpha,\mathbb{N}}(m)): \mathsf{Type}$$

depending on a subtyping. If this dependency of type formation is undesirable, then in the categorical set-up of Definition 4.3 one should require that the categories $\mathbb{D}$ and $\mathbb{E}$ have the same objects.

5 A term model

In this final section we sketch how to obtain a term model which fits the categorical setting described in the previous two sections. It is instructive in that it shows the importance of separating contexts according to the dependencies that one has.

We assume that we have some polymorphically typed calculus with subtyping and constrained quantification. Details of this language will become clear as we proceed. We form a base category $\mathbb{B}$ with

objects kind contexts $\Gamma = (\alpha_1: K_1, \ldots, \alpha_n: K_n)$. The kinds $K_i: \mathsf{Kind}$ are built up from constants—including $\mathsf{Type}: \mathsf{Kind}$—with as possible kind constructors $1, \times, \to, +, 0$, but powerkinds are not assumed.

morphisms $\Gamma \to \Delta$ where $\Delta = (\beta_1: L_1, \ldots, \beta_m: L_m)$ are sequences $(M_1, \ldots, M_m)$ of (equivalence classes of) terms $\Gamma \vdash M_i: L_i$.

Our base category is thus the category of (kind) contexts and context morphisms, as in simply typed λ-calculus. It has finite products by concatenation of contexts where the empty context serves as terminal object. What is special is that the types $\Gamma \vdash \sigma: \mathsf{Type}$ of our calculus appear as morphisms $\Gamma \to (\alpha: \mathsf{Type})$ in $\mathbb{B}$. Hence the singleton kind context $(\alpha: \mathsf{Type})$ plays the role of the generic object Ω.

Next there is a category $\mathbb{C}$ of type inclusions. It has

objects pairs $(\Gamma \mid \Theta)$ with Γ is a kind context and Θ is a subtyping context of the form $\sigma_1 <: \sigma'_1, \ldots, \sigma_n <: \sigma'_n$ where $\Gamma \vdash \sigma_i, \sigma'_i: \mathsf{Type}$. Thus Θ can be understood as an n-type $\sigma_i, \sigma'_i: \Gamma \rightrightarrows \Omega$ of parallel arrows in $\mathbb{B}$.

morphisms $(\Gamma \mid \Theta) \to (\Delta \mid \Xi)$ are context morphisms $\vec{M}: \Gamma \to \Delta$ in $\mathbb{B}$ such that for each inclusion $\tau_j <: \tau'_j$ in Ξ one can derive $\Gamma \mid \Theta \vdash \tau_j(\vec{M}) <: \tau'_j(\vec{M})$.

There is an obvious projection functor $(\Gamma \mid \Theta) \mapsto \Gamma$, which yields a fibration $\begin{smallmatrix}\mathbb{C}\\ \downarrow q\\ \mathbb{B}\end{smallmatrix}$. Each fibre, say over $\Gamma \in \mathbb{B}$, has finite products by concatenation of subtyping contexts. Notice that we have a *family of* generic objects as in Remark 3.5: the set of subtyping contexts over $\Gamma \in \mathbb{B}$ is the disjoint union $\coprod_{n \in \mathbb{N}} \mathbb{B}\big(\Gamma, (\Omega \times \Omega)^n\big)$ of n-tuples of pairs of types in kind context Γ.

There is a third category $\mathbb{D}$ of types, whose formation may depend on subtypings. (If the calculus does not have this dependency, then these are the ordinary types, i.e. the maps $\Gamma \to \Omega$ in $\mathbb{B}$. Certainly term formation will involve subtypings.) This category $\mathbb{D}$ has

objects types $(\Gamma \mid \Theta \vdash \sigma\colon \mathsf{Type})$ which are well-formed in kind context Γ and subtyping context Θ.

morphisms $(\Gamma \mid \Theta \vdash \sigma\colon \mathsf{Type}) \to (\Delta \mid \Xi \vdash \tau\colon \mathsf{Type})$ are pairs $(\vec{M}, N)$ where $\vec{M}\colon (\Gamma \mid \Theta) \to (\Delta \mid \Xi)$ is a morphism in $\mathbb{C}$, and N is a term $\Gamma \mid \Theta \mid x\colon\sigma \vdash N\colon \tau(\vec{M})$.

Again there is a projection functor $(\Gamma \mid \Theta \vdash \sigma\colon \mathsf{Type}) \mapsto (\Gamma \mid \Theta)$ which forms a fibration $\begin{smallmatrix}\mathbb{D}\\ \downarrow r\\ \mathbb{C}\end{smallmatrix}$. This fibration is cartesian closed if we assume finite products $1, \times$ and exponents $\to$ of types.

The terminal object functor $\top\colon \mathbb{B} \to \mathbb{C}$ maps a kind context Γ to the pair $(\Gamma \mid \emptyset) \in \mathbb{C}$ consisting of Γ and the empty typing context $\emptyset$. Pulling $\begin{smallmatrix}\mathbb{D}\\ \downarrow r\\ \mathbb{C}\end{smallmatrix}$ back along $\top$ yields the fibration $\begin{smallmatrix}\mathbb{E}\\ \downarrow p\\ \mathbb{B}\end{smallmatrix}$ of types (and terms) which do not depend on subtypings. This fibration p has $\Omega = (\alpha\colon \mathsf{Type}) \in \mathbb{B}$ as split generic object. Moreover, we have the subtyping axioms in our calculus,

$$\Gamma \mid \emptyset \vdash \sigma <: \mathrm{Top}, \qquad \Gamma \mid \sigma <: \sigma', \tau <: \tau' \vdash \sigma \times \tau <: \sigma' \times \tau',$$
$$\Gamma \mid \sigma' <: \sigma, \tau <: \tau' \vdash \sigma \to \tau <: \sigma' \to \tau'$$

so that $\begin{smallmatrix}\mathbb{C}\\ \downarrow\\ \mathbb{B}\end{smallmatrix}$ is a subtyping fibration for $\begin{smallmatrix}\mathbb{E}\\ \downarrow\\ \mathbb{B}\end{smallmatrix}$.

The induced functor $\mathcal{P}\colon \widehat{\mathbb{C}} \to \mathbb{C}$ from Definition 3.7 maps a subtyping context $(\Gamma \mid \Theta)$ and a pair of types $\Gamma, \alpha\colon \mathsf{Type} \vdash \sigma, \sigma'\colon \mathsf{Type}$ possibly containing an extra free type variable, to the projection map between subtyping contexts

$$\big(\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma <: \sigma'\big) \xrightarrow{\ \pi\ } \big(\Gamma \mid \Theta\big).$$

The construction of the category $\widehat{\mathbb{C}}$ in Definition 3.7 ensures that α may occur in σ, σ' but not in Θ. The induced weakening functor π^* in $\begin{smallmatrix}\mathbb{D}\\ \downarrow\\ \mathbb{C}\end{smallmatrix}$ maps

$$\big(\Gamma \mid \Theta \vdash \rho\colon \mathsf{Type}\big) \xmapsto{\ \pi^*\ } \big(\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma <: \sigma' \vdash \rho\colon \mathsf{Type}\big)$$

by adding the dummy assumption $\sigma <: \sigma'$.

We finish by showing that the constrained quantifiers $\forall\alpha[\sigma <: \sigma'].(-)$ and $\exists\alpha[\sigma <: \sigma'].(-)$ are right and left adjoints to this π^*. The adjunctions require bijective correspondences

$$\frac{\pi^*(\rho) \longrightarrow \tau \quad \text{over } (\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma <: \sigma')}{\rho \longrightarrow \forall\alpha[\sigma <: \sigma'].\,\tau \quad \text{over } (\Gamma \mid \Theta)}$$

$$\frac{\tau \longrightarrow \pi^*(\rho) \quad \text{over } (\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma <: \sigma')}{\exists\alpha[\sigma <: \sigma'].\,\tau \longrightarrow \rho \quad \text{over } (\Gamma \mid \Theta)}$$

i.e. correspondences between terms M and N in

$$\frac{\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma <: \sigma' \mid x\colon\rho \vdash M\colon\tau}{\Gamma \mid \Theta \mid x\colon\rho \vdash N\colon \forall\alpha[\sigma <: \sigma'].\,\tau}$$

$$\frac{\Gamma, \alpha\colon \mathsf{Type} \mid \Theta, \sigma <: \sigma' \mid y\colon\tau \vdash M\colon\rho}{\Gamma \mid \Theta \mid z\colon \exists\alpha[\sigma <: \sigma'].\,\tau \vdash N\colon\rho}$$

These adjoint correspondences are precisely the introduction and elimination rules for $\forall\alpha[\sigma <: \sigma'].\,\tau$ and $\exists\alpha[\sigma <: \sigma'].\,\tau$, plus the associated (β)- and (η)-conversions: for constrained products $\forall$: one takes

$$M \mapsto \lambda\alpha[\sigma < \sigma'].\,M \qquad \text{and} \qquad N \mapsto N\alpha.$$

And for constrained sums $\exists$:

$$M \mapsto M \ \mathsf{where}\ \langle\alpha, y\rangle := z \qquad \text{and} \qquad N \mapsto N[\langle\alpha, y\rangle/z].$$

References

[BCGS91] V. Breazu-Tannen, Th. Conquand, C. Gunter and A. Scedrov, 'Inheritance and explicit coercion', *Inform. & Comp.* **93** p. 172–221. Also in: C.A. Gunter and J.C. Mitchell (eds.), *Theoretical Aspects of Object-Oriented Programming*, The MIT Press, 1994, p. 197–245.

[BL90] K.B. Bruce and G. Longo, 'Modest models of records, inheritance and bounded quantification', *Inform. & Comp.* **87** (1990), p. 196–240. Also in: C.A. Gunter and J.C. Mitchell (eds.), *Theoretical Aspects of Object-Oriented Programming*, The MIT Press, 1994, p. 151–195.

[CCHOM89] P. Canning, W. Cook, W. Hill, W. Olthoff and J. Mitchell, 'F-Bounded Polymorphism for Object-Oriented Programming', *Funct. Progr. & Comp. Arch.* 1989, ACM Press, p. 273–280.

[CL91] L. Cardelli and G. Longo, 'A Semantic basis for Quest', *Journ. Funct. Progr.* **1** (1991) p. 417–458.

[CW85] L. Cardelli and P. Wegner, 'On Understanding Types, Data Abstraction and Polymorphism', *ACM Computing Surveys* **17** (1985) p. 471–522.

[Hyl88] J.M.E. Hyland, 'A small complete category', *Ann. Pure & Appl. Logic* **40** (1988) p. 135–165.

[Jac91] B. Jacobs, 'Categorical type theory', PhD. Thesis, Univ. of Nijmegen, 1991.

[MP88] J.C. Mitchell and G.D. Plotkin, 'Abstract types have existential type', *ACM Trans. on Progr. Lang. and Systems* **10(3)** (1988) p. 470–502.

[Ph92] W. Phoa, 'Using fibrations to understand subtypes', in: M.P. Fourman and P.T. Johnstone and A.M. Pitts (eds.), *Applications of Categories in Computer Science*, Cambridge Univ. Press, LMS 177, (1992), p. 239–257.