Intersection Logic

Simona Ronchi Della Rocca and Luca Roversi

Dipartimento di Informatica
C.so Svizzera, n. 185
10149 Torino – ITALY
{ronchi,rover}@di.unito.it
http://www.di.unito.it/~{ronchi, rover}

Abstract. The intersection type assignment system IT uses the formulas of the negative fragment of the predicate calculus (LJ) as types for the λ-terms. However, the deductions of IT only correspond to the proper sub-set of the derivations of LJ, obtained by imposing a meta-theoretic condition about the use of the conjunction of LJ. This paper proposes a logical foundation for IT. This is done by introducing a logic IL. Intuitively, a derivation of IL is a set of derivations in LJ such that the derivations in the set can be thought of as writable in parallel. This way of looking at LJ, by means of IL, allows to transform the meta-theoretic condition, mentioned above, into a purely structural property of IL. The relation between IL and LJ surely has a first main benefit: the strong normalization of LJ directly implies the same property on IL, which translates in a very simple proof of the strong normalizability of the λ-terms typable with IT.

1 Introduction

The intersection type assignment system (IT) is a set of rules for assigning types to terms of the untyped λ-calculus. The types of IT are formulas of the predicate logic, built from the two connectives “implication” ($\rightarrow$) and “conjunction” ($\wedge$).

IT was introduced in the early Eighty by Mariangiola Dezani and Mario Coppo [6], in order to enhance the tipability power of the Curry type assignment system. The system characterizes important properties of the λ-terms, like the normalization and the strong normalization. Indeed, it has be proved that IT assigns types to all and only the strong normalizing terms [19]. Moreover, if the set of types is extended to contain a “universal type” ω, that can be given to all the λ-terms, then the normalizing terms are exactly those that can be given a type free of occurrences of ω [4].

Intersection types have been particularly useful in studying the semantics of various kinds of λ-calculi. This can be done by extending the system with suitable sub-typing relations. In this way the type assignment is a finitary tool to reason about the interpretation of the terms in the topological models of λ-calculus, like Scott domains, DI-domains, coherence spaces [1, 4, 7, 10, 14, 15, 22].

Unlike other type assignment systems (like Curry type assignment [8], or the type assignment version of the II order λ-calculus [12, 17]), IT has not been designed starting from a logic, and up to now, the relationship between IL and the implicational and conjunctive fragment of the predicate calculus (LJ) are far from being clear. This was firstly pointed out by Hindley [13]. The problem is the logical interpretation of the rule introducing the conjunction:

$$(\wedge I_{IT})\frac{\Gamma \vdash_{\wedge} M : \sigma \quad \Gamma \vdash_{\wedge} M : \tau}{\Gamma \vdash_{\wedge} M : \sigma \wedge \tau} \tag{1}$$

If we reason according to the Curry-Howard isomorphism, which is the standard relationship between logic and λ-calculus, then we can observe that in the rule (1) here above the λ-term M denotes two proofs. In particular, (1) says that an intersection type $\sigma \wedge \tau$ can be built from the two components σ and τ only when they can be proved by two "isomorphic" proofs, according to a notion of isomorphism that relates proofs encoded by the same λ-term. Our point is that the use of a λ-term to express the isomorphism of two derivations is a meta-theoretical restriction on the introduction of the conjunction, and for this reason LJ is not the logic which IT is based on.

In this paper we establish a logical foundation for IT, and clarify the relationship between IT and LJ. More precisely, we define the new logic IL, such that every of its deductions corresponds to a set of deductions in LJ, sharing some structural properties. IL is the desired "bridge" between LJ and IT, since a deduction in IT can be obtained as a partial decoration of a deduction in IL. Moreover, IL has all the good properties we ask for a logic. In particular it enjoys the strong normalization property, whose proof directly derives from the analogous property of LJ [20]. Moreover, thanks to the relation between IL and IT, we obtain for free that if a term is typable by an intersection type, then it is also strongly normalizable. As a side result, a typed version of λ-terms with intersection type can be obtained through a full decoration of deductions in IL. But this is subject of a forthcoming paper.

The literature presents other proof-theoretical investigations of the intersection type assignment. Barbanera and Alessi [2], refining a previous attempt of Mints [18], proved that IT, equipped with both β and η-reduction, gives a complete realizability semantics of the predicate logic with implication and "strong conjunction". This result has been further extended to other connectives in [3].

The first attempt to give a logical foundation to intersection types was by Venneri [23]. She proposed an Hilbert style logic corresponding to a system which assigns intersection types to the terms of Combinatory Logic. A further extension with union types is in [9]. Our feeling is that the approach in [23, 9] is not suitable for the λ-calculus.

Moreover, in [5] there is the proof that a typed version of IT can be obtained, through the Curry-Howard isomorphism, from a logic, where formulas are sequence of formulas of LJ. The untyped version of such a language is quite similar to the language in [16], which has been defined to highlight the intrinsic parallelism of the β-reduction. Other typed versions of IT have been defined by

Reynolds [21] and Wells [24], but they did not follow a logical approach, so are unrelated with the topic of the present paper.

To conclude, we believe that our approach to a logical foundation of IT could evolve by adopting the principles at the base of Ludics, the project initiated by J.-Y. Girard to re-found logic [11].

The paper is organized as follows. In Section 2 and 3 the systems LJ and IT are briefly recalled. In Section 4 the intersection logic IL is defined, and its relation with LJ is stated. In Section 5 the strong normalization of IL is proved. Section 6 contains the main theorem, which formalizes the relation between IL and IT.

2 Intuitionistic Logic: Implicative and Conjunctive Fragment

We start by recalling the natural deduction of the implicative and conjunctive fragment of Intuitionistic Logic that, somewhat abusing the name, we call LJ.

Definition 1. *i) The* formulas *of LJ belong to the language generated by the grammar:*

$$\sigma ::= \alpha \mid (\sigma \to \sigma) \mid (\sigma \wedge \sigma)$$

where $\alpha \in V$ (a denumerable set of constants). The formulas are denoted by ρ, σ, τ. We assume the associativity to the right for $\to$ and the one to the left for $\wedge$.

ii) A context *is a* finite sequence *$\sigma_1, \ldots, \sigma_m$ of formulas, denoted by Γ, Δ, possibly indexed.*

iii) The natural *deduction system LJ proves statements $\Gamma \vdash_{LJ} \sigma$, where Γ is a context and σ a formula. It consists of the following rules:*

$$(A_{LJ})\frac{}{\sigma \vdash_{LJ} \sigma} \qquad (W_{LJ})\frac{\Gamma \vdash_{LJ} \sigma}{\Gamma, \tau \vdash_{LJ} \sigma}$$

$$(X_{LJ})\frac{\Gamma, \sigma, \tau, \Delta \vdash_{LJ} \rho}{\Gamma, \tau, \sigma, \Delta \vdash_{LJ} \rho} \qquad (\wedge I_{LJ})\frac{\Gamma \vdash_{LJ} \sigma \quad \Gamma \vdash_{LJ} \tau}{\Gamma \vdash_{LJ} \sigma \wedge \tau}$$

$$(\wedge E^l_{LJ})\frac{\Gamma \vdash_{LJ} \sigma \wedge \tau}{\Gamma \vdash_{LJ} \sigma} \qquad (\wedge E^r_{LJ})\frac{\Gamma \vdash_{LJ} \sigma \wedge \tau}{\Gamma \vdash_{LJ} \tau}$$

$$(\to I_{LJ})\frac{\Gamma, \sigma \vdash_{LJ} \tau}{\Gamma \vdash_{LJ} \sigma \to \tau} \qquad (\to E_{LJ})\frac{\Gamma \vdash_{LJ} \sigma \to \tau \quad \Gamma \vdash_{LJ} \sigma}{\Gamma \vdash_{LJ} \tau}$$

The deductions of LJ are denoted by $\Pi, \Pi_1, \ldots$. Moreover, $\Pi : \Gamma \vdash_{LJ} \sigma$ means that Π concludes, proving $\Gamma \vdash_{LJ} \sigma$.

Example 1. Let σ denote the formula $(\alpha \wedge \beta) \wedge \gamma$. Define the following three deductions:

$$\Pi_0 : (\wedge E^l_{LJ})\frac{(\wedge E^l_{LJ})\dfrac{(A_{LJ})\dfrac{}{\sigma \vdash_{LJ} (\alpha \wedge \beta) \wedge \gamma}}{\sigma \vdash_{LJ} \alpha \wedge \beta}}{\sigma \vdash_{LJ} \alpha}$$

$$\Pi_1 : (\wedge E^r_{LJ})\frac{(A_{LJ})\dfrac{}{\sigma \vdash_{LJ} (\alpha \wedge \beta) \wedge \gamma}}{\sigma \vdash_{LJ} \gamma}$$

$$\Pi_2 : (\to I_{LJ}) \dfrac{(A_{LJ})\dfrac{}{\alpha \vdash_{LJ} \alpha}}{\vdash_{LJ} \alpha \to \alpha}$$

Then, Π_3 is:

$$(\wedge I_{LJ}) \dfrac{(\to I_{LJ}) \dfrac{(\wedge I_{LJ}) \dfrac{\Pi_0 : \sigma \vdash_{LJ} \alpha \quad \Pi_1 : \sigma \vdash_{LJ} \gamma}{\sigma \vdash_{LJ} \alpha \wedge \gamma}}{\vdash_{LJ} (\alpha \wedge \beta) \wedge \gamma \to (\alpha \wedge \gamma)} \qquad \Pi_2 : \vdash_{LJ} \alpha \to \alpha}{\vdash_{LJ} ((\alpha \wedge \beta) \wedge \gamma \to (\alpha \wedge \gamma)) \wedge (\alpha \to \alpha)}$$

Let us recall the strong normalization property of the deductions in LJ.

Definition 2. *Let Π be a deduction of LJ.*

- *i) A $\wedge$-LJ-*redex *of Π is a sequence of* two *rules, formed by an instance of $(\wedge I_{LJ})$, followed by an instance of either $(\wedge E^l_{LJ})$ or $(\wedge E^r_{LJ})$;*
- *ii) A $\to$-LJ-*redex *of Π is a sequence of* two *rules, formed by an instance of $(\to I_{LJ})$, followed by an instance of $(\to E_{LJ})$;*
- *iii) Π is* normal *if it does not contain neither $\wedge$-LJ-redexes nor $\to$-LJ-redexes.*

Lemma 1. *Consider two derivations $\Pi : \Gamma \vdash_{LJ} \sigma$ and $\Pi' : \Gamma, \sigma \vdash_{LJ} \tau$. Call $S(\Pi, \Pi')$ the deductive structure obtained by replacing the conclusion of Π for every occurrence (A_{LJ}) deriving $\sigma \vdash_{LJ} \sigma$, and such that σ to left of $\vdash_{LJ}$ is free in the conclusion of Π'. Then, $S(\Pi, \Pi') : \Gamma \vdash_{LJ} \tau$.*

Definition 3. *The rewriting relation $\rightsquigarrow_{LJ}$ between derivations is defined as follows:*

- $(\wedge E^s_{LJ}) \dfrac{(\wedge I_{LJ}) \dfrac{\Pi_l : \Gamma \vdash_{LJ} \sigma_l \quad \Pi_r : \Gamma \vdash_{LJ} \sigma_r}{\Gamma \vdash_{LJ} \sigma_l \wedge \sigma_r}}{\Gamma \vdash_{LJ} \sigma_s} \rightsquigarrow_{LJ} \Pi_s : \Gamma \vdash_{LJ} \sigma_s,$
  where $s \in \{l, r\}$.
- $(\to E_{LJ}) \dfrac{(\to I_{LJ}) \dfrac{\Pi_0 : \Gamma, \tau \vdash_{LJ} \sigma}{\Gamma \vdash_{LJ} \tau \to \sigma} \quad \Pi_1 : \Gamma \vdash_{LJ} \tau}{\Gamma \vdash_{LJ} \sigma} \rightsquigarrow_{LJ} S(\Pi_1, \Pi_0) : \Gamma \vdash_{LJ} \sigma$

Theorem 1 (Prawitz). *The rewriting relation $\rightsquigarrow_{LJ}$ is strongly normalizing.*

3 Intersection Types

We briefly recall the system of Intersection Types (IT), which works as a type assignment system for the untyped λ-calculus.

Definition 4. *i) The set of* types *of IT coincides to the formulas of LJ.*

- *ii) An IT-context is a finite set of pairs $\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\}$ that assigns types to λ-variables so that $i \neq j$ implies $x_i \not\equiv x_j$. By abusing the notation, Γ and Δ denote IT-contexts.*
- *iii) IT is a deductive system that derives judgments $\Gamma \vdash_{IT} M : \sigma$ where M is a λ-term, Γ is an IT-context, and σ is a type. IT consists of the following rules:*

$$(A_{IT})\frac{x:\sigma \in \Gamma}{\Gamma \vdash_{IT} x:\sigma} \qquad (\wedge I_{IT})\frac{\Gamma \vdash_{IT} M:\sigma \quad \Gamma \vdash_{IT} M:\tau}{\Gamma \vdash_{IT} M:\sigma \wedge \tau}$$

$$(\wedge E^{l}_{IT})\frac{\Gamma \vdash_{IT} M:\sigma \wedge \tau}{\Gamma \vdash_{IT} M:\sigma} \qquad (\wedge E^{r}_{IT})\frac{\Gamma \vdash_{IT} M:\sigma \wedge \tau}{\Gamma \vdash_{IT} M:\tau}$$

$$(\rightarrow I_{IT})\frac{\Gamma \cup \{x:\sigma\} \vdash_{IT} M:\tau}{\Gamma \vdash_{IT} \lambda x.M:\sigma \rightarrow \tau} \qquad (\rightarrow E_{IT})\frac{\Gamma \vdash_{IT} M:\sigma \rightarrow \tau \quad \Gamma \vdash_{IT} N:\sigma}{\Gamma \vdash_{IT} MN:\tau}$$

We keep using $\Pi, \Pi_1, \dots$ to denote the deductions of IT. Moreover, the meaning of $\Pi : \Gamma \vdash_{IT} M : \sigma$ is analogous to the one of $\Pi : \Gamma \vdash_{LJ} \sigma$.

Example 2. Let σ denote the type $(\alpha \wedge \beta) \wedge \gamma$ and let Π_4 be the following deduction:

$$(\rightarrow I_{IT})\dfrac{(\wedge I_{IT})\dfrac{(\wedge E^{l}_{IT})\dfrac{(\wedge E^{l}_{IT})\dfrac{(A_{IT})\dfrac{}{x:\sigma \vdash_{IT} x:\sigma}}{x:\sigma \vdash_{IT} x:\alpha \wedge \beta}}{x:\sigma \vdash_{IT} x:\alpha} \quad (\wedge E^{r}_{IT})\dfrac{(A_{IT})\dfrac{}{x:\sigma \vdash_{IT} x:\sigma}}{x:\sigma \vdash_{IT} x:\gamma}}{x:\sigma \vdash_{IT} x:\alpha \wedge \gamma}}{\vdash_{IT} \lambda x.x:(\alpha \wedge \beta) \wedge \gamma \rightarrow (\alpha \wedge \gamma)}$$

Let Π_5 be the following deduction:

$$(\rightarrow I_{IT})\dfrac{(A_{IT})\dfrac{}{x:\alpha \vdash_{IT} x:\alpha}}{\vdash_{IT} \lambda x.x:\alpha \rightarrow \alpha}$$

And finally let Π_6 be:

$$(\wedge I_{IT})\frac{\Pi_4 :\vdash_{IT} \lambda x.x:(\alpha \wedge \beta) \wedge \gamma \rightarrow (\alpha \wedge \gamma) \quad \Pi_2 :\vdash_{IT} \lambda x.x:\alpha \rightarrow \alpha}{\vdash_{IT} \lambda x.x:((\alpha \wedge \beta) \wedge \gamma \rightarrow (\alpha \wedge \gamma)) \wedge (\alpha \rightarrow \alpha)}$$

4 Intersection Logic

In this section we introduce *Intersection Logic* (IL), whose derivations correspond to sets of derivations in LJ, sharing some similarity in the structure. The formulas of IL are binary trees, whose leaves are labeled by formulas of LJ. The relation between IL and LJ can be informally described as follows: a derivation Π of IL groups a set, say $LJ(\Pi)$, of derivations of LJ. Every derivation $\Pi' \in LJ(\Pi)$ can be obtained by taking the leaf of a given path in every tree of Π. In particular, the elements of $LJ(\Pi)$ share both the number of instances and the order of application of the rules introducing and eliminating the connective $\rightarrow$.

We need to introduce some preliminary notions.

Definition 5. *i) A* kit *is a binary tree in the language generated by the following grammar: $K ::= \sigma \mid [K, K]$. The leaves of any kit, which we call also* atoms*, are* formulas of LJ*. The kits are denoted by H, K.*

ii) Two kits overlap *if they are two trees with exactly the same structure, but which may differ only on the name of their atoms; $H \simeq K$ denotes two overlapping kits H and K. For example, $\sigma \simeq \tau$.*

iii) *Two overlapping kits map to a kit with only arrows as its leaves, by the function* $(\ )^+$*, defined as follows:*

$$[\sigma, \tau]^+ \mapsto \sigma \to \tau$$
$$[[H_1, H_2], [K_1, K_2]]^+ \mapsto [[H_1, K_1]^+, [H_2, K_2]^+]$$

$H \to K$ *denotes the result of* $[H, K]^+$*, with* $H \simeq K$*. Otherwise,* $H \to K$ *is undefined.*

iv) *A* path *is a string built over the set* $\{l, r\}$*;* p, q*, possibly indexed, denote paths, and* ϵ *denotes the empty path. The subtree of a kit* H *at the path* p *is inductively defined as follows:*

$$H^\epsilon = H, \qquad [H_1, H_2]^{lp} = H_1^p, \qquad [H_1, H_2]^{rp} = H_2^p$$

Otherwise, σ^p *is undefined.*
A path p *is* defined in H *if* H^p *is defined, and it is* terminal in H *if* H^p *is an atom; the set of terminal paths of a kit* H *is denoted by* $P_T(H)$*.*
$H[p := K]$ *denotes the kit resulting from the replacement of* K *for* H^p *in* H*.*

v) *Let* $s \in \{l, r\}$*, and let* ps *be a path, defined in* H*. The* pruning *of* H *at path* ps*, is defined as follows:* $H\backslash^{ps} = H[p := H^{ps}]$*.*

vi) $\equiv$ *is the syntactical identity of both atoms, kits and paths.*

By definition, we have:

Fact 1 *1.* $P_T(H \to K) = P_T(H) = P_T(K)$*;*
2. $H\backslash^p \to K\backslash^p \equiv (H \to K)\backslash^p$*;*
3. If p *and* q *are two different paths, then* $(H\backslash^p)\backslash^q \equiv (H\backslash^q)\backslash^p$*.*

The definition here below is about the deductive system $\vdash_{pIL}$ on which we shall define IL. The key feature of $\vdash_{pIL}$ is that every of its judgments exclusively contains overlapping kits. Intuitively, this invariance on the judgment form formalizes that every derivation of $\vdash_{pIL}$, being introduced, stands for a set of deductions of LJ, which share structural properties.

Definition 6 (Natural deduction $\vdash_{pIL}$). *The natural deduction system* $\vdash_{pIL}$*, that we call* pre Intersection Logic*, derives judgments* $\Gamma \vdash_{pIL} K$*, where* Γ *is a sequence of kits and* K *is a kit. It consists of the following rules:*

$$(A_{pIL})\frac{}{H \vdash_{pIL} H} \qquad (X_{pIL})\frac{\Gamma, H, H', \Delta \vdash_{pIL} K}{\Gamma, H', H, \Delta \vdash_{pIL} K}$$

$$(W_{pIL})\frac{H_1, \ldots, H_n \vdash_{pIL} H \quad H_i \simeq H' \quad (1 \le i \le n)}{H_1, \ldots, H_n, H' \vdash_{pIL} H}$$

$$(P_{pIL})\frac{\Gamma \vdash_{pIL} K \quad s \in \{l, r\} \quad ps \in P_T(K)}{\Gamma\backslash^{ps} \vdash_{pIL} K\backslash^{ps}}$$

$$(\to E_{pIL})\frac{\Gamma \vdash_{pIL} H \to K \quad \Gamma \vdash_{pIL} H}{\Gamma \vdash_{pIL} K} \qquad (\to I_{pIL})\frac{\Gamma, H \vdash_{pIL} K}{\Gamma \vdash_{pIL} H \to K}$$

$$(\wedge E^l_{pIL})\frac{\Gamma \vdash_{pIL} K[p := \sigma \wedge \tau]}{\Gamma \vdash_{pIL} K[p := \sigma]} \qquad (\wedge E^r_{pIL})\frac{\Gamma \vdash_{pIL} K[p := \sigma \wedge \tau]}{\Gamma \vdash_{pIL} K[p := \tau]}$$

$$(\wedge I_{pIL})\frac{H_1[p := [\sigma_1, \sigma_1]], ..., H_n[p := [\sigma_n, \sigma_n]] \vdash_{pIL} K[p := [\sigma, \tau]]}{H_1[p := \sigma_1], ..., H_n[p := \sigma_n] \vdash_{pIL} K[p := \sigma \wedge \tau]}$$

where, in rule (P_{pIL}), the notation $\Gamma \backslash^{ps}$ stands for the distribution of the pruning to the components of Γ.

The judgments of $\vdash_{pIL}$ enjoy an invariant:

Lemma 2. *If $H_1, \ldots, H_n \vdash_{pIL} K$, then $H_i \simeq K$ $(1 \leq i \leq n)$.*

Proof. By structural induction on the deduction of $H_1, \ldots, H_n \vdash_{pIL} K$.

In the following definition we introduce a "decoration" of all the systems previously defined, inspired to the so called "Curry-Howard isomorphism": every deduction Π is associated to a λ-term to keep track of some structural properties of Π. Note that this decoration is not standard: the λ-term associated to Π is *untyped*, and does not encode *the whole* structure of Π, but just the order of the occurrences of the rules which introduce and eliminate $\rightarrow$. Moreover, the decoration is a partial function when applied to a derivation of LJ. Indeed, a decoration of a proof whose last rule is $(\wedge I_{LJ})$ is defined only if the derivations of its two premises are decorated by the same term.

Definition 7. *Let $\vdash_* \in \{LJ, pIL\}$, and $A, B, C, ...$ be meta-variables for denoting either atoms or kits. Also, let Δ denote a sequence built over $\{A, B, C, ...\}$.*

i) Every Π proving $\Delta \vdash_* A$ can be decorated by a λ-term $T_{dom(\Delta^*)}(\Pi)$, where Δ^* is a decoration of Δ, and, if $\Delta^* \equiv x_1 : A_1, \ldots, x_n : A_n$, then $dom(\Delta^*)$ is the sequence $x_1, \ldots, x_n$. The decoration of $\vdash_*$ is denoted by $\vdash^+_*$ and is inductively defined as:

- $\Pi : (A_*)\dfrac{}{A \vdash_* A} \Rightarrow (A^+_*)\dfrac{}{x : A \vdash^+_* x : A}$
  and $T_x(\Pi) \equiv x$;
- $\Pi : (W_*)\dfrac{\Pi_1 : \Delta \vdash_* A}{\Delta, B \vdash_* A} \Rightarrow (W^+_*)\dfrac{\Delta^* \vdash^+_* T_{dom(\Delta^*)}(\Pi_1) : A}{\Delta, x : B \vdash^+_* T_{dom(\Delta^*),x}(\Pi_1) : A}$ *where Δ^* is the decoration of Δ, x is fresh, and $T_{dom(\Delta^*),x}(\Pi) \equiv T_{dom(\Delta^*)}(\Pi_1)$;*
- $\Pi : (\rightarrow I_*)\dfrac{\Pi_1 : \Delta, A \vdash_* B}{\Delta \vdash_* A \rightarrow B} \Rightarrow$

  $$(\rightarrow I^+_*)\frac{\Delta^*, x : A \vdash^+_* T_{dom(\Delta^*),x}(\Pi_1) : B}{\Delta^* \vdash^+_* \lambda x.T_{dom(\Delta^*),x}(\Pi_1) : A \rightarrow B}$$
  and $T_{dom(\Delta^*)}(\Pi) \equiv \lambda x.T_{dom(\Delta^*),x}(\Pi_1)$;
- $\Pi : (\rightarrow E_*)\dfrac{\Pi_1 : \Delta \vdash_* A \rightarrow B \quad \Pi_2 : \Delta \vdash_* A}{\Delta \vdash_* B} \Rightarrow$

  $$(\rightarrow E^+_*)\frac{\Delta^* \vdash^+_* T_{dom(\Delta^*)}(\Pi_1) : A \rightarrow B \quad \Delta^* \vdash^+_* T_{dom(\Delta^*)}(\Pi_2) : A}{\Delta^* \vdash^+_* T_{dom(\Delta^*)}(\Pi_1)T_{dom(\Delta^*)}(\Pi_2) : B}$$
  and $T_{dom(\Delta^*)}(\Pi) \equiv T_{dom(\Delta^*)}(\Pi_1)T_{dom(\Delta^*)}(\Pi_2)$;

- $\Pi : (P_{pIL})\dfrac{\Pi_1 : K_1, \ldots, K_n \vdash_{pIL} H}{K_1\backslash^{ps}, \ldots, K_n\backslash^{ps} \vdash_{pIL} H\backslash^{ps}} \Rightarrow$

  $(P^+_{pIL})\dfrac{x_1 : K_1, \ldots, x_n : K_n \vdash^+_{pIL} T_{x_1,\ldots,x_n}(\Pi_1) : H}{x_1 : K_1\backslash^{ps}, \ldots, x_n : K_n\backslash^{ps} \vdash^+_{pIL} T_{x_1,\ldots,x_n}(\Pi) : H\backslash^{ps}}$

  and $T_{x_1,\ldots,x_n}(\Pi) \equiv T_{x_1,\ldots,x_n}(\Pi_1)$*;*
- $\Pi : (\wedge I_{pIL})\dfrac{\Pi_1 : H_1[p := [\sigma_1, \sigma_1]], ..., H_n[p := [\sigma_n, \sigma_n]] \vdash_{pIL} K[p := [\sigma, \tau]]}{H_1[p := \sigma_1], ..., H_n[p := \sigma_n] \vdash_{pIL} K[p := \sigma \wedge \tau]} \Rightarrow$

  $(\wedge I^+_{pIL})\dfrac{x_1 : H_1[p := [\sigma_1, \sigma_1]], ..., x_n : H_n[p := [\sigma_n, \sigma_n]] \vdash^+_{pIL} T_{x_1,\ldots,x_n}(\Pi_1) : K[p := [\sigma, \tau]]}{x_1 : H_1[p := \sigma_1], ..., x_n : H_n[p := \sigma_n] \vdash^+_{pIL} T_{x_1,\ldots,x_n}(\Pi_1) : K[p := \sigma \wedge \tau]}$

  and $T_{x_1,\ldots,x_n}(\Pi) \equiv T_{x_1,\ldots,x_n}(\Pi_1)$*;*
- *with*

  $$\Pi : (\wedge I_{LJ})\dfrac{\Pi_1 : \Delta \vdash_{LJ} \sigma \quad \Pi_2 : \Delta \vdash_{LJ} \tau}{\Delta \vdash_{LJ} \sigma \wedge \tau}$$

  let $\Pi^+_1 : \Delta^* \vdash^+_{LJ} T_{dom(\Delta^*)}(\Pi_1) : \sigma$ *and* $\Pi^+_2 : \Delta^* \vdash^+_{LJ} T_{dom(\Delta^*)}(\Pi_2) : \tau$*.* *If* $T_{dom(\Delta^*)}(\Pi_1) \equiv T_{dom(\Delta^*)}(\Pi_2)$*, then the decoration is:*

  $$(\wedge I^+_{LJ})\dfrac{\Delta^* \vdash_{LJ} T_{dom(\Delta^*)}(\Pi_1) : \sigma \quad \Delta^* \vdash_{LJ} T_{dom(\Delta^*)}(\Pi_2) : \tau}{\Delta^* \vdash_{LJ} T_{dom(\Delta^*)}(\Pi) : \sigma \wedge \tau}$$

  where $T_{dom(\Delta^*)}(\Pi_1) \equiv T_{dom(\Delta^*)}(\Pi)$*.*
  Otherwise, if $T_{dom(\Delta^*)}(\Pi_1) \not\equiv T_{dom(\Delta^*)}(\Pi_2)$*, then* $T_{dom(\Delta^*)}(\Pi)$ *does not exist.*
- $\Pi : (\circ)\dfrac{\Pi_1 : \Delta \vdash_* A}{\Delta \vdash_* B} \Rightarrow (\circ^+)\dfrac{\Delta^* \vdash^+_* T_{dom(\Delta^*)}(\Pi_1) : A}{\Delta^* \vdash^+_* T_{dom(\Delta^*)}(\Pi_1) : B}$

  and $T_{dom(\Delta^*)}(\Pi) \equiv T_{dom(\Delta^*)}(\Pi_1)$*, for all rules* $(\circ)$ *not in the set:* $\{(\rightarrow I_*), (\rightarrow E_*), (A_*), (W_*), (P_{pIL}), (\wedge I_*)\}$*.*

ii) Let Π *be a deduction in the system* $\vdash_*$*. Then,* $U(\Pi)$ *is the set:*

$$\{T_{dom(\Delta^*)}(\Pi) \mid \Pi : \Delta \vdash_* A, T_{dom(\Delta^*)}(\Pi) \text{ exists and } \Delta^* \text{ is a decoration of } \Delta\}$$

Notice that, by construction, $M, N \in U(\Pi)$ implies M and N can differ each other by renaming of both free and bound variables. We will call $U(\Pi)$ the *form* of Π.

The next theorem shows that every derivation of $\vdash_{pIL}$ corresponds to a set of derivations in LJ with the same form.

Theorem 2 (From $\vdash_{pIL}$ to LJ). *Let* $\Pi : K_1, \ldots, K_n \vdash_{pIL} H$*. For all path* p *terminal in* H*, it is* $\Pi^p : K^p_1, \ldots, K^p_n \vdash_{LJ} H^p$*, and* $U(\Pi^p)$ *is defined, and* $U(\Pi^p) = U(\Pi)$*.*

Proof. By induction on Π.

We conclude this section by a definition that eliminates unnecessary differentiations among the deductions of $\vdash_{pIL}$. In particular, it allows to consider the deductions of $\vdash_{pIL}$ up to the order of applications of the rules, involving the manipulations of the kits. Equivalently, the definition here below introduces a set of commuting equivalences. We could get rid of them, simply by introducing new versions of the rules $(\wedge E^l_{pIL}), (\wedge E^r_{pIL}), (\wedge I_{pIL})$, working in parallel on disjoint paths of the same kit. Our opinion is that such a solution would have obscured the clarity of the logical system $\vdash_{pIL}$.

Definition 8 (Intersection Logic). Intersection Logic, *abbreviated as IL, is the set* $^{\vdash_{pIL}}/_{\sim}$ *of all the deductions of* $\vdash_{pIL}$*, quotiented by the congruence* $\sim$*, defined as:*

$$(\wedge E^{s'}_{pIL})\frac{(\wedge E^{s}_{pIL})\dfrac{\Gamma \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r][q := \tau_l \wedge \tau_r]}{\Gamma \vdash_{pIL} K[p := \sigma_s][q := \tau_l \wedge \tau_r]}}{\Gamma \vdash_{pIL} K[p := \sigma_s][q := \tau_{s'}]} \sim$$

$$(\wedge E^{s}_{pIL})\frac{(\wedge E^{s'}_{pIL})\dfrac{\Gamma \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r][q := \tau_l \wedge \tau_r]}{\Gamma \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r][q := \tau_{s'}]}}{\Gamma \vdash_{pIL} K[p := \sigma_s][q := \tau_{s'}]}$$

$$(\wedge E^{s}_{pIL})\frac{(\wedge I_{pIL})\dfrac{\Gamma \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r][q := [\tau_l, \tau_r]]}{\Gamma\backslash^{qs'} \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r][q := \tau_l \wedge \tau_r]}}{\Gamma\backslash^{qs'} \vdash_{pIL} K[p := \sigma_s][q := \tau_l \wedge \tau_r]} \sim$$

$$(\wedge I_{pIL})\frac{(\wedge E^{s}_{pIL})\dfrac{\Gamma \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r][q := [\tau_l, \tau_r]]}{\Gamma \vdash_{pIL} K[p := \sigma_s][q := \tau_l \wedge \tau_r]}}{\Gamma\backslash^{qs'} \vdash_{pIL} K[p := \sigma_s][q := \tau_l \wedge \tau_r]}$$

being $s, s' \in \{l, r\}$*, and* p, q *two different paths.*

An equivalence class in IL whose deductions prove $H_1,\ldots, H_n \vdash_{pIL} K$*, is denoted by* $H_1,\ldots, H_n \vdash_{IL} K$*, or* $\pi : H_1,\ldots, H_n \vdash_{IL} K$*.*

Definition 8 also assures that the term decorating the conclusion of two deductions of the same equivalence class of IL are the same:

Fact 2 $\pi : H_1,\ldots, H_n \vdash_{IL} K$ *implies* $T_{x_1,..,x_n}(\Pi) \equiv T_{x_1,..,x_n}(\Pi')$*, for every* $\Pi, \Pi' \in \pi$*.*

So, we can safely identify $T_{x_1,..,x_n}(\pi)$ and $T_{x_1,..,x_n}(\Pi)$, if $\Pi \in \pi$. Moreover, we can extend our terminology to to say that $U(\Pi)$ is the form of π as well.

Example 3. Let σ denote the formula $(\alpha\wedge\beta)\wedge\gamma$. Let also the following deductions be given:

$$\Pi_7 : (\to I_{pIL})\frac{(\wedge I_{pIL})\dfrac{(\wedge E^r_{pIL})\dfrac{(\wedge E^l_{pIL})\dfrac{(\wedge E^l_{pIL})\dfrac{(A_{pIL})\dfrac{}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\sigma,\sigma],\alpha]}}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\alpha\wedge\beta,\sigma],\alpha]}}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\alpha,\sigma],\alpha]}}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\alpha,\gamma],\alpha]}}{[\sigma,\alpha] \vdash_{pIL} [\alpha\wedge\gamma,\alpha]}}{\vdash_{pIL} [\sigma \to (\alpha\wedge\gamma), \alpha\to\alpha]}$$

$$\Pi_8 : (\wedge I_{pIL}) \frac{\Pi_7 : \vdash_{pIL} [\sigma \to (\alpha \wedge \gamma), \alpha \to \alpha]}{\vdash_{pIL} (\sigma \to (\alpha \wedge \gamma)) \wedge (\alpha \to \alpha)}$$

$$\Pi_9 : (\to I_{pIL}) \frac{(\wedge I_{pIL}) \frac{(\wedge E^l_{pIL}) \frac{(\wedge E^r_{pIL}) \frac{(\wedge E^l_{pIL}) \frac{(A_{pIL}) \overline{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\sigma,\sigma],\alpha]}}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\alpha \wedge \beta,\sigma],\alpha]}}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\alpha \wedge \beta,\gamma],\alpha]}}{[[\sigma,\sigma],\alpha] \vdash_{pIL} [[\alpha,\gamma],\alpha]}}{[\sigma,\alpha] \vdash_{pIL} [\alpha \wedge \gamma,\alpha]}}{\vdash_{pIL} [\sigma \to (\alpha \wedge \gamma), \alpha \to \alpha]}$$

$$\Pi_{10} : (\wedge I_{pIL}) \frac{\Pi_9 : \vdash_{pIL} [\sigma \to (\alpha \wedge \gamma), \alpha \to \alpha]}{\vdash_{pIL} (\sigma \to (\alpha \wedge \gamma)) \wedge (\alpha \to \alpha)}$$

As an exercise, verify that: i) $\Pi_7 \sim \Pi_9$, ii) if π is the equivalence class of Π_7, then $T_x[\pi] \equiv \lambda x.x$, and iii) the deductions corresponding to Π_7 and Π_{10} in the system LJ, according to Theorem 2, are respectively: $(\Pi_7)^l \equiv \Pi_1$, $(\Pi_7)^r \equiv \Pi_2$, $(\Pi_{10})^\epsilon \equiv \Pi_3$, being Π_1, Π_2 and Π_3 defined in Example 1.

5 Strong Normalization

In this section we shall prove that IL is strongly normalizable. This property follows from the strong normalizability of $\vdash_{pIL}$, proved by reducing to the analogous property of LJ.

Definition 9. *Let $s \in \{l, r\}$.*

*i) The P-*commuting conversions *on the sequent calculus $\vdash_{pIL}$ are the following rewriting rules:*

$$(P_{pIL}) \frac{(A_{pIL}) \overline{\Gamma, H \vdash_{pIL} H}}{\Gamma\backslash^p, H\backslash^p \vdash_{pIL} H\backslash^p} \rightsquigarrow_P (A_{pIL}) \overline{\Gamma\backslash^p, H\backslash^p \vdash_{pIL} H\backslash^p}$$

$$(P_{pIL}) \frac{(X_{pIL}) \frac{\Gamma, H, H', \Delta \vdash_{pIL} K}{\Gamma, H', H, \Delta \vdash_{pIL} K}}{\Gamma\backslash^p, H'\backslash^p, H\backslash^p, \Delta\backslash^p \vdash_{pIL} K\backslash^p} \rightsquigarrow_P$$

$$(X_{pIL}) \frac{(P_{pIL}) \frac{\Gamma, H, H', \Delta \vdash_{pIL} K}{\Gamma\backslash^p, H\backslash^p, H'\backslash^p, \Delta\backslash^p \vdash_{pIL} K\backslash^p}}{\Gamma\backslash^p, H'\backslash^p, H\backslash^p, \Delta\backslash^p \vdash_{pIL} K\backslash^p}$$

$$(P_{pIL}) \frac{(W_{pIL}) \frac{\Gamma \vdash_{pIL} K}{\Gamma, H \vdash_{pIL} K}}{\Gamma\backslash^p, H\backslash^p \vdash_{pIL} K\backslash^p} \rightsquigarrow_P (W_{pIL}) \frac{(P_{pIL}) \frac{\Gamma \vdash_{pIL} K}{\Gamma\backslash^p \vdash_{pIL} K\backslash^p}}{\Gamma\backslash^p, H\backslash^p \vdash_{pIL} K\backslash^p}$$

$$(P_{pIL}) \frac{(\to I_{pIL}) \frac{\Gamma, H \vdash_{pIL} K}{\Gamma \vdash_{pIL} H \to K}}{\Gamma\backslash^p \vdash_{pIL} (H \to K)\backslash^p} \rightsquigarrow_P (\to I_{pIL}) \frac{(P_{pIL}) \frac{\Gamma, H \vdash_{pIL} K}{\Gamma\backslash^p, H\backslash^p \vdash_{pIL} K\backslash^p}}{\Gamma\backslash^p \vdash_{pIL} (H \to K)\backslash^p}$$

$$(P_{pIL}) \frac{(\to E_{pIL}) \frac{\Gamma \vdash_{pIL} H \to K \quad \Gamma \vdash_{pIL} H}{\Gamma \vdash_{pIL} K}}{\Gamma\backslash^p \vdash_{pIL} K\backslash^p} \rightsquigarrow_P$$

$$(\rightarrow E_{pIL})\dfrac{(P_{pIL})\dfrac{\Gamma \vdash_{pIL} H \rightarrow K}{\Gamma\backslash^{p} \vdash_{pIL} (H \rightarrow K)\backslash^{p}} \qquad (P_{pIL})\dfrac{\Gamma \vdash_{pIL} H}{\Gamma\backslash^{p} \vdash_{pIL} H\backslash^{p}}}{\Gamma\backslash^{p} \vdash_{pIL} K\backslash^{p}}$$

$$(P_{pIL})\dfrac{(\wedge I_{pIL})\dfrac{\Gamma \vdash_{pIL} K[q := [\sigma, \tau]]}{\Gamma\backslash^{q} \vdash_{pIL} K[q := \sigma \wedge \tau]}}{(\Gamma\backslash^{q})\backslash^{p} \vdash_{pIL} (K[q := \sigma \wedge \tau])\backslash^{p}} \rightsquigarrow_{P}$$

$$(\wedge I_{pIL})\dfrac{(P_{pIL})\dfrac{\Gamma \vdash_{pIL} K[q := [\sigma, \tau]]}{\Gamma\backslash^{p} \vdash_{pIL} (K[q := [\sigma, \tau]])\backslash^{p}}}{(\Gamma\backslash^{p})\backslash^{q} \vdash_{pIL} ((K[q := \sigma \wedge \tau])\backslash^{p})\backslash^{q}}$$

$$(P_{pIL})\dfrac{(\wedge E^{s}_{pIL})\dfrac{\Gamma \vdash_{pIL} K[q := \sigma_l \wedge \sigma_r]}{\Gamma \vdash_{pIL} K[q := \sigma_s]}}{\Gamma\backslash^{p} \vdash_{pIL} (K[q := \sigma_s])\backslash^{p}} \rightsquigarrow_{P}$$

$$(\wedge E^{s}_{pIL})\dfrac{(P_{pIL})\dfrac{\Gamma \vdash_{pIL} K[q := \sigma_l \wedge \sigma_r]}{\Gamma\backslash^{p} \vdash_{pIL} (K[q := \sigma_l \wedge \sigma_r])\backslash^{p}}}{\Gamma\backslash^{p} \vdash_{pIL} (K[q := \sigma_s])\backslash^{p}}$$

Under the standard terminology, every sequence of rules to the left of $\rightsquigarrow$ *is a* P*-redex, while that one to its right is a* P*-reduct.*

ii) A derivation free of occurrences of (P_{pIL}) *is* P*-normal.*

iii) We say that a class π *of IL reduces to another class* π' *of IL, under* $\rightsquigarrow_P$*, and we write* $\pi \rightsquigarrow_P \pi'$*, if there are* $\Pi \in \pi$ *and* $\Pi' \in \pi'$ *such that* $\Pi \rightsquigarrow_P \Pi'$*.*

Remark that the fourth and sixth P-commuting conversions exploits Fact 1.

Lemma 3 (P-strong normalization). *Every* $\Pi : \Gamma \vdash_{pIL} H$ *can be reduced to a* P*-normal* $\Pi' : \Gamma \vdash_{pIL} H$*, under any strategy.*

Proof. Observe that the commuting conversions shift every occurrence of (P_{pIL}) upwards, which, eventually, gets erased.

Definition 10. *Let* $s \in \{l, r\}$ *and* $\Pi \in \pi$*.*

i) A ∧-IL-redex *of* Π *is the sequence:*

$$(\wedge E^{s}_{pIL})\dfrac{(\wedge I_{pIL})\dfrac{H_1[p := [\sigma_1, \sigma_1]], \ldots, H_n[p := [\sigma_n, \sigma_n]] \vdash_{pIL} K[p := [\sigma_l, \sigma_r]]}{H_1[p := \sigma_1], \ldots, H_n[p := \sigma_n] \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r]}}{H_1[p := \sigma_1], \ldots, H_n[p := \sigma_n] \vdash_{pIL} K[p := \sigma_s]}$$

ii) A ∧-IL-rewriting step *on* Π *is:*

$$(\wedge E^{s}_{pIL})\dfrac{(\wedge I_{pIL})\dfrac{H_1[p := [\sigma_1, \sigma_1]], \ldots, H_n[p := [\sigma_n, \sigma_n]] \vdash_{pIL} K[p := [\sigma_l, \sigma_r]]}{H_1[p := \sigma_1], \ldots, H_n[p := \sigma_n] \vdash_{pIL} K[p := \sigma_l \wedge \sigma_r]}}{H_1[p := \sigma_1], \ldots, H_n[p := \sigma_n] \vdash_{pIL} K[p := \sigma_s]} \rightsquigarrow_{\wedge}$$

$$(P_{pIL})\dfrac{H_1[p := [\sigma_1, \sigma_1]], \ldots, H_n[p := [\sigma_n, \sigma_n]] \vdash_{pIL} K[p := [\sigma_l, \sigma_r]]}{(H_1[p := [\sigma_1, \sigma_1]])\backslash^{ps}, \ldots, (H_n[p := [\sigma_n, \sigma_n]])\backslash^{ps} \vdash_{pIL} K\backslash^{ps}}$$

where $(H_i[p := [\sigma_i, \sigma_i]])\backslash^{ps} \equiv H_i[p := \sigma_i]$*, with* $1 \leq i \leq n$*, and* $K\backslash^{ps} \equiv K[p := \sigma_s]$*.*

iii) We say that a class π *of IL reduces to another class* π' *of IL, under a* ∧*-IL-rewriting step, and we write* $\pi \rightsquigarrow_{\wedge} \pi'$*, if there are* $\Pi \in \pi$ *and* $\Pi' \in \pi'$ *such that* $\Pi \rightsquigarrow_{\wedge} \Pi'$*.*

Lemma 4. *Consider $\Pi : \Gamma \vdash_{pIL} H$ and $\Pi' : \Gamma, H \vdash_{pIL} K$. Call $S(\Pi, \Pi')$ the deductive structure, obtained by replacing the conclusion of Π for every occurrence (A_{pIL}) deriving $H \vdash_{pIL} H$, and such that H to the left of $\vdash_{pIL}$ is free in Π'. Then, $S(\Pi, \Pi') : \Gamma \vdash_{pIL} K$.*

Proof. By structural induction on the deduction of $\Gamma, H \vdash_{pIL} K$.

Definition 11. *Let $\Pi \in \pi$.*

i) A $\rightarrow$-IL-redex *of Π is the sequence:*

$$(\rightarrow E_{pIL})\frac{(\rightarrow I_{pIL})\dfrac{\Gamma, H \vdash_{pIL} K}{\Gamma \vdash_{pIL} H \rightarrow K} \qquad \Gamma \vdash_{pIL} H}{\Gamma \vdash_{pIL} K}$$

ii) A $\rightarrow$-IL-rewriting *step on Π is:*

$$(\rightarrow E_{pIL})\frac{(\rightarrow I_{pIL})\dfrac{\Pi_0 : \Gamma, H \vdash_{pIL} K}{\Gamma \vdash_{pIL} H \rightarrow K} \qquad \Pi_1 : \Gamma \vdash_{pIL} H}{\Gamma \vdash_{pIL} K} \rightsquigarrow_{\rightarrow}$$

$$S(\Pi_1, \Pi_0) : \Gamma \vdash K \ .$$

iii) We say that a class π of IL reduces to another class π' of IL, under a $\rightarrow$-IL-rewriting step, and we write $\pi \rightsquigarrow_{\rightarrow} \pi'$, if there are $\Pi \in \pi$ and $\Pi' \in \pi'$ such that $\Pi \rightsquigarrow_{\rightarrow} \Pi'$.

Definition 12. *A deduction of IL is* normal *if it is free of (P), $\wedge$, and $\rightarrow$-IL-redexes.*

Definition 13. *$\rightsquigarrow$ is the smallest contextual, reflexive and transitive closure of $\rightsquigarrow_P \cup \rightsquigarrow_\wedge \cup \rightsquigarrow_\rightarrow$.*

Lemma 5. *$\rightsquigarrow$ is strongly normalizing.*

Proof. The proof proceeds by exploiting the embedding of IL into LJ, which allows to show the number of both the $\rightarrow$-IL-redexes and the $\wedge$-IL-redexes, of any derivation Π of IL, can be bound by the number of the analogous redexes of any projection of Π into LJ. The existence of the P-commuting conversions in $\rightsquigarrow$, which are strongly normalizable, is completely transparent to the embedding.

Theorem 3. *IL is strongly normalizable.*

Proof. From Definition 13 and Lemma 5.

6 Intersection Types and Intersection Logic

In this section, we trace the relationship between IL and the intersection type system IT. On one side, every deduction $\pi : \Gamma \vdash_{IL} H$ corresponds to a set of type assignments. Every of such type assignments gives a type to the form of π, which, we recall, is a λ-term. The type is one of the leaves of H. On the other side, every type assignment $\{x_1 : \sigma_1, ..., x_n : \sigma_n\} \vdash_\wedge M : \sigma$ corresponds to a deduction $\pi : \sigma_1, ..., x : \sigma_n \vdash_{IL} \sigma$ such that M is the form of π.

Lemma 6. *$\pi : H_1,\ldots,H_n \vdash_{IL} K$ implies $\{x_1 : H_1^p, .., x_n : H_n^p\} \vdash_\wedge T_{x_1\ldots x_n}(\pi) : K^p$, for every $p \in P_T(K)$.*

Proof. The proof is reduced to proving the statement: "$\Pi : H_1,\ldots,H_n \vdash_{pIL} K$ implies $\{x_1 : H_1^p, .., x_n : H_n^p\} \vdash_\wedge T_{x_1\ldots x_n}(\Pi) : K^p$, for every $p \in P_T(K)$, and $\Pi \in \pi$" by induction on Π. Then, the final statement is a corollary of Fact 2.

In order to study the opposite direction of the correspondence between $\vdash_{IL}$ and $\vdash_\wedge$, we need an auxiliary lemma.

Lemma 7. *Let assume the notations:*

$$\Delta_1 \equiv H_1,\ldots,H_n,\ \ \Delta_2 \equiv H'_1,\ldots,H'_n,\ \ \Delta \equiv [H_1, H'_1],\ldots,[H_n, H'_n]\ .$$

Take any $\Pi_1 : \Delta_1 \vdash_{pIL} K_1$, and $\Pi_2 : \Delta_2 \vdash_{pIL} K_2$ such that $T_{dom(\Delta_1^*)}(\Pi_1) \equiv T_{dom(\Delta_2^*)}(\Pi_2)$, for every decoration Δ_1^*, Δ_2^*, such that $dom(\Delta_1^*) \equiv dom(\Delta_2^*)$. There is $\Pi : \Delta \vdash_{pIL} [K_1, K_2]$ such that $T_{dom(\Delta^*)}(\Pi) \equiv T_{dom(\Delta_1^*)}(\Pi_1)$, whenever $dom(\Delta^*) \equiv dom(\Delta_1^*)$.

Proof. The proof can proceed by structural induction on Π_1. As an example, we show the details about an instance of one of the more interesting cases.

Let:

$$(\to I_{pIL})\frac{\Delta_1, K'_1 \vdash_{pIL} K''_1}{\Delta_1 \vdash_{pIL} K'_1 \to K''_1}$$

be the last rule of Π_1. For any decorations Δ_1^* and Δ_2^*, such that $dom(\Delta_1^*) \equiv dom(\Delta_2^*)$, the assumption $T_{dom(\Delta_1^*)}(\Pi_1) \equiv T_{dom(\Delta_2^*)}(\Pi_2)$, assures that the last rules of Π_2 can only be an instance of $(\to I_{pIL})$, followed by a, possibly empty, sequence $R_1,\ldots,R_m$ of rules, each belonging to $\{(P_{pIL}), (\wedge I_{pIL}), (\wedge E_{pIL}), (X_{pIL}), (W_{pIL})\}$, and such that they apply to the paths $p_1,\ldots,p_m$ of the conclusion of Π_2. Assume:

$$(\to I_{pIL})\frac{\hat{\Delta}_2, K'_2 \vdash_{pIL} K''_2}{\hat{\Delta}_2 \vdash_{pIL} K'_2 \to K''_2}$$

be the last $(\to I_{pIL})$ instance of Π_2, where $\hat{\Delta}_2 \equiv \hat{H}'_1,\ldots,\hat{H}'_n$. Tanks to the α-equivalence on the λ-terms, we can always end up with decorations such that $dom(\Delta_1^*, x : K'_1) \equiv dom(\hat{\Delta}_2^*, x : K'_2)$, for some suitable x. So, by induction, there exists a deduction $\bar{\Pi} : \bar{\Delta}, [K'_1, K'_2] \vdash_{pIL} [K''_1, K''_2]$, such that $\bar{\Delta} \equiv [H_1, \hat{H}'_1],\ldots,[H_n, \hat{H}'_n]$, and whose decoration is

$$T_{dom(\bar{\Delta}^*, x:K'_1)}(\bar{\Pi}) \equiv T_{dom(\Delta_1^*, x:K'_1)}(\Pi_1).$$

Now, we can firstly extend $\bar{\Pi}$ to $\check{\Pi}$ by a $(\to I_{pIL})$, as follows:

$$\check{\Pi} : (\to I)\frac{\bar{\Pi} : \bar{\Delta}, [K'_1, K'_2] \vdash [K''_1, K''_2]}{\bar{\Delta} \vdash_{pIL} [K'_1 \to K''_1, K'_2 \to K''_2]}$$

By Definition 7, it must be $\lambda x.T_{dom(\bar{\Delta}^*, x:K'_1)}(\bar{\Pi}) \equiv \lambda x.T_{dom(\Delta_1^*, x:K'_1)}(\Pi_1)$. To conclude, it is enough to apply $R_1,\ldots,R_m$ to the paths $rp_1,\ldots,rp_m$ of $\check{\Pi}$.

The other case, which requires some work to be proved, has $(\to E_{pIL})$ as last rule of Π_1.

All the remaining cases, instead, exploit the induction in the simplest way.

Lemma 8. *$\Pi : \{x_1 : \sigma_1, \ldots, x_n : \sigma_n\} \vdash_\wedge M : \tau$ implies $\pi : \sigma_1, \ldots, \sigma_n \vdash_{IL} \tau$ such that $M \equiv T_{x_1 \ldots x_n}(\pi)$.*

Proof. The proof is by induction on Π. Here we limit to sketch only the not obvious case. Assume to prove $\Pi : \{x_1 : \sigma_1, \ldots, x_n : \sigma_n\} \vdash_\wedge M : \sigma \wedge \tau$ from the assumptions $\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\} \vdash_\wedge M : \sigma$ and $\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\} \vdash_\wedge M : \tau$. By induction, we get both $\sigma_1, \ldots, \sigma_n \vdash_{IL} \sigma$ and $\sigma_1, \ldots, \sigma_n \vdash_{IL} \tau$. Then, Lemma 7 implies $[\sigma_1, \sigma_1], \ldots, [\sigma_n, \sigma_n] \vdash_{IL} [\sigma, \tau]$, to which we can apply $(\wedge I_{IL})$ to conclude.

Definition 14. *A judgment $K_1, \ldots, K_n \vdash_{IL} H$ is* proper *if, and only if, H and every K_i is an atom.*

We are finally in the position to relate IL and IT:

Theorem 4. *$\pi : \sigma_1, \ldots, \sigma_n \vdash_{IL} \tau$ if, and only if, $x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash_\wedge T_{x_1 \ldots x_n}(\pi) : \tau$, for every $\pi : \sigma_1, \ldots, \sigma_n \vdash_{IL} \tau$ proper.*

Proof. Directly from Lemma 6 and Lemma 8.

Example 4. Let π be the equivalence class which Π_8 (or Π_{10}) in Example 3 belongs to. The corresponding deduction of $\vdash_\wedge$ is Π_6 of Example 2.

The correspondence between IL and IT allows to derive for free the property of strong normalization of the λ-terms, typable in IT, with respect to the β-reduction. This property has been first proved in [19].

Theorem 5. *Let $\Gamma \vdash_\wedge M : \sigma$. Then, M is strongly normalizable.*

Proof. The proof proceeds in two steps. Firstly, we embed the derivation of $\Gamma \vdash_\wedge M : \sigma$ into LJ, getting a derivation Π. Secondly, we assume the existence of a redex of M not present in the normal form of Π, getting a contradiction.

References

1. Samson Abramsky. Domain theory in logical form. *Ann. Pure Appl. Logic*, 51(1-2):1–77, 1991.
2. F. Alessi and F. Barbanera. Strong conjunction and intersection types. In *16h International Symposium on Mathematical Foundation of Computer Science (MFCS91)*, volume Lecture Notes in Computer Science 520. Springer-Verlag, 1991.
3. F. Barbanera and S. Martini. Proof-functional connectives and realizability. *Archive for Mathematical Logic*, 33:189–211, 1994.
4. Henk Barendregt, Mario Coppo, and Mariangiola Dezani-Ciancaglini. A filter lambda model and the completeness of type assignment. *J. Symbolic Logic*, 48(4):931–940 (1984), 1983.
5. B. Capitani, M. Loreti, and Venneri B. Hyperformulae, parallel deductions and intersection types. To appear in "Workshop on Bohm Theorem", ICALP 2001, Creta (Greece), 2001.

6. Mario Coppo and Mariangiola Dezani-Ciancaglini. An extension of the basic functionality theory for the λ-calculus. *Notre Dame J. Formal Logic*, 21(4):685–693, 1980.
7. Mario Coppo, Mariangiola Dezani-Ciancaglini, Furio Honsell, and Giuseppe Longo. Extended type structures and filter lambda models. In *Logic colloquium '82 (Florence, 1982)*, pages 241–262. North-Holland, Amsterdam, 1984.
8. H. Curry, R. Feys, and W. Craig. *Combinatory Logic*, volume 1. North Holland, 1958.
9. M. Dezani-Ciancaglini, S. Ghilezan, and B. Venneri. The "relevance" of intersection and union types. *Notre Dame J. Formal Logic*, 38(2):246–269, 1997.
10. Lavinia Egidi, Furio Honsell, and Simona Ronchi Della Rocca. Operational, denotational and logical descriptions: a case study. *Fund. Inform.*, 16(2):149–169, 1992. Mathematical foundations of computer science '91 (Kazimierz Dolny, 1991).
11. Jean-Yves Girard. Locus solum: From the rules of logic to the logic of rules. Internal Report, IML, Marseille, 2001.
12. J.Y. Girard. *Interpretation Fonctionelle et Elimination des Coupures de l'Arithmetique d'Ordre Superieur*. PhD thesis, Université Paris VII, 1972.
13. J. Roger Hindley. Coppo Dezani types do not correspond to propositional logic. *Theoret. Comput. Sci.*, 28(1-2):235–236, 1984.
14. Furio Honsell and Simona Ronchi Della Rocca. Reasoning about interpretations in qualitative lambda-models. In *Programming Concepts and Methods*, pages 505–522. North Holland, 1990.
15. Furio Honsell and Simona Ronchi Della Rocca. An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus. *J. Comput. System Sci.*, 45(1):49–75, 1992.
16. Assaf Kfoury. Beta-reduction as unification. In *Logic Algebra and Computer Science*, pages 241–262. Polish Academy of Science, Warsaw, 1999.
17. Daniel Leivant. Polymorphic type inference. *Symposium on Principles of Programming Languages*, 1983.
18. G. E. Mints. The completeness of provable realizability. *Notre Dame J. Formal Logic*, 30(3):420–441, 1989.
19. Garrel Pottinger. A type assignment for the strongly normalizable λ-terms. In *To H. B. Curry: essays on combinatory logic, lambda calculus and formalism*, pages 561–577. Academic Press, London, 1980.
20. Dag Prawitz. *Natural Deduction*. Almquist & Wiksell.
21. J. C. Reynolds. Design of the programming language Forsythe. In P. O'Hearn and R.D. Tennent, editors, *Algol-like Languages*. Birkhauser, 1996.
22. L. Roversi. a Type-Free Resource-Aware λ-Calculus. In *Fifth Annual Conference of the EACSL (CSL'96)*, volume 1258 of *Lecture Notes in Computer Science*, pages 399 – 413, Utrecht (The Nederland), September 1996. Springer-Verlag.
23. Betti Venneri. Intersection types as logical formulae. *J. Logic Comput.*, 4(2):109–124, 1994.
24. J. B. Wells, Allyn Dimock, Robert Muller, and Franklyn Turbak. A typed intermediate language for flow-directed compilation. In *7th International Joint Conference on Theory and Practice of Software Development (TAPSOFT97)*, pages 757–771.