JCalc: A curry-Howard view of justification logic

Konstantinos Pouliasis

July 27, 2016

Abstract

This work consists of my dissertation proposal and it is stuctured in two parts: The first part is, essentially, my second examination paper with some revisions and additions. It introduces basic elemements of my research topic which rests in the intersection of Justification Logic, Constructive Modality and Type Theory. I will present the relevant systems syntactically and I will pause on the basic metatheoretic proof techniques which will be useful in the rest of the text. In the second part I will delinenate the current state of my research in the area. I will elaborate on a modal type system that enhances simple type theory with elements of justification logic. I will present its accompanying calculus obtained a la Curry-Howard and I will argue for its computational relevance. More specifically, I will show that the obtained calculus characterizes certain computational phenomena that abound in modern programming language semantics. I will ommit full metatheoretic results here and I will merely hint to proof methods tha can be adopted from the first part. Finally, I will propose certain directions for future work. Such developments together with the omitted metatheoretic results are expected to be the study of my dissertation thesis. This is a paper towards the dissertation proposal requirement for my Phd candidacy under the supervision of Distinguished Professor Sergei Artemov at the Department of Computer Science of the Graduate Center at City University of New York.

Contents

Chapter 1

Intuitionistic Logic

1.1 Intuitionism

In this Chapter, I will be presenting foundational work in the intersection of *Intuitionistic Logic* and *Type Theory*. The presentation is scaffolding following Prof. Robert Harper's lecture videos in *Homotopy Type Theory* [7] and the accompanying notes by students of the class [10]. I will often diverge to standard textbooks in the field [2], [6], [17] to present further important results.

1.1.1 A bird's eye view

In a nutshell, *Intuitionistic mathematics* is a program in foundations of mathematics that extends *Brouwer's program* [3]. Brouwer, in an almost Kantian fashion, viewed mathematical reasoning as a human faculty and mathematics as a language of the "creative subject' aiming to communicate mathematical concepts. The concept of *algorithm* as a step–by–step constructive process is brought in the foreground in Brouwer's program. As a result, intuitionistic theories adhere to computational interpretations. In the following I will be using the terms intutionistic and *constructive* interchangeably.

For the purposes of this paper, the main diverging point of Brouwer's program, later explicated by Heyting [11] and Kolmogorov [12] [1], lies in the treatment of proofs. In contrast to classical approaches to foundations that treat proof objects as external to theories, the constructive approach treats proofs as the fundamental forms of construction and hence, as first class citizens. As a result, the constructive view of logic draws heavily from proof theory and Gentzen's developments [5]. For the reader interested also in the philosophical implications of constructive foundations and *antirealism*, Dummet's treatment is a classic in the field [4].

It has to be emphasized that proofs in the intuitionistic approach are treated as stand–alone and are not bound to formal systems (i.e the notion of proof *precedes* that of a formal system). It is necessary, hence, to draw a distinction between the notion of *proof as construction* and the typical notion of *proof in a formal system* [9, 8].

A formal proof is a proof given in a fixed formal system, such as the axiomatic theory of sets, and arises from the application of the inductively defined rules in that system. Formal proofs can, thus, be viewed as gödelizations of textual derivation in some fixed system.

Although every formal proof (in a specific system) is also a proof (assuming soundness of the system) the converse is not true. This conforms with Gödel's Incompleteness Theorem, which precisely states that there exist true propositions (with a proof in *some* formal system), but for which there cannot be given a formal proof in the formal system that is at stake. This *openness* of the nature of proofs is necessary for a foundational treatment of proofs that respects Gödelian phenomena. This is often coined as "Axiomatic Freedom" of intuitionistic foundations.

Following the same line of thought, and adopting the doctrine of *proof relevance* for obtaining true judgments, leads to another main difference of the constructive approach and the classical one i.e the (default) absence of the *law of excluded middle*. Current developments in constructive foundations like *Homotopy Type Theory* and in general systems that rely on *Martin-Löf Type Theory* [14] do not necessarily rule out *LEM* but they might permit its usage locally, if needed, in a proof.

1.2 IPL

Intuitionistic Propositional Logic (IPL) can be viewed as "the logic of *proof relevance*" conforming with the intuitionistic view described in 1.1. To judge a fact as *true* one may provide a *proof* appropriate of the fact. *Proofs* can be synthesized to obtain proofs for more complex facts (*introduction rules*) and consumed to provide proofs relevant for other facts (*elimination rules*). The importance of the interplay between introduction and elimination rules was developed by Gentzen. A discussion on the meaning of the logical connectives that is prevalent in *MLTT* can be found in [13] Following the presentation style by Martin-Löf we split the notions of *judgment* and *proposition*. We have two main kinds of judgments:

- *Judgments* that are logical arguments about the truth(or, equivalently, proof) of a *proposition*. They might, optionally, involve assumptions on the truth (or, equivalently, proof) of other propositions. We might call these *logical judgments*.
- Judgments on *propositionality* or typeability. *Propositions* are the *subjects* of *logical judgments*. If something is judged to be a proposition then it belongs to the universe of discourse and can be mentioned in *logical judgments*.

In addition, since a *logical judgment* might involve a set Γ of assumptions (or a *context*), it is convenient to add a third kind of judgment of the form $\Gamma\ \mathsf{ctx}$ Thus, in IPL, we get the judgments $\phi\ \in\ \mathsf{Prop}$, $\phi\,\mathsf{true}$ and $\Gamma\,\mathsf{ctx}$:

$\phi \in\ \mathsf{Prop}$	ϕ is a (well-formed) proposition
$\phi\ \ \mathsf{true}$	Proposition ϕ is true i.e., has a proof.
$\Gamma\,\mathsf{ctx}$	Γ is a (well-formed) context of assumptions

The natural deduction system of IPL is given below:

Prop Formation

$$\frac{}{P_i \in \mathsf{Prop}}\ \textsc{Atom} \qquad \frac{}{\top \in \mathsf{Prop}}\ \textsc{Top} \qquad \frac{}{\bot \in \mathsf{Prop}}\ \textsc{Bottom}$$

$$\frac{\phi_1 \in \mathsf{Prop} \qquad \phi_2 \in \mathsf{Prop}}{\phi_1 \supset \phi_2 \in \mathsf{Prop}}\ \textsc{Arr} \qquad \frac{\phi_1 \in \mathsf{Prop} \qquad \phi_2 \in \mathsf{Prop}}{\phi_1 \wedge \phi_2 \in \mathsf{Prop}}\ \textsc{Conj}$$

$$\frac{\phi_1 \in \mathsf{Prop} \qquad \phi_2 \in \mathsf{Prop}}{\phi_1 \vee \phi_2 \in \mathsf{Prop}}\ \textsc{Disj}$$

Context Formation

$$\frac{}{\mathsf{nil}\ \ \mathsf{ctx}}\ \textsc{Nil} \qquad \frac{\Gamma\ \mathsf{ctx} \qquad \phi \in \mathsf{Prop}}{\Gamma, \phi\ \mathsf{true}\ \ \mathsf{ctx}}\ \Gamma\text{-}\textsc{Add}$$

Context Reflection

$$\frac{\Gamma\ \mathsf{ctx} \qquad \phi\ \mathsf{true} \in \Gamma}{\Gamma \vdash \phi\ \mathsf{true}}\ \Gamma\text{-}\textsc{Refl}$$

Top Introduction – Bottom Elimination

$$\frac{}{\Gamma \vdash \top\ \mathsf{true}}\ \top\mathrm{I} \qquad \frac{\Gamma \vdash \bot\ \mathsf{true}}{\Gamma \vdash \phi\ \mathsf{true}}\ \bot\mathrm{E}$$

Implication Introduction and Elimination

$$\frac{\Gamma, \phi_1\ \mathsf{true} \vdash \phi_2\ \mathsf{true}}{\Gamma \vdash \phi_1 \supset \phi_2\ \mathsf{true}}\ {\supset}\mathrm{I} \qquad \frac{\Gamma \vdash \phi_1 \supset \phi_2\ \mathsf{true} \qquad \Gamma \vdash \phi_1\ \mathsf{true}}{\Gamma \vdash \phi_2\ \mathsf{true}}\ {\supset}\mathrm{E}$$

Conjunction Introduction and Elimination

$$\frac{\Gamma \vdash \phi_1\ \mathsf{true} \qquad \Gamma \vdash \phi_2\ \mathsf{true}}{\Gamma \vdash \phi_1 \wedge \phi_2\ \mathsf{true}}\ \wedge\mathrm{I}$$

$$\frac{\Gamma \vdash \phi_1 \wedge \phi_2\ \mathsf{true}}{\Gamma \vdash \phi_1\ \mathsf{true}}\ \wedge\mathrm{E\scriptsize L} \qquad \frac{\Gamma \vdash \phi_1 \wedge \phi_2\ \mathsf{true}}{\Gamma \vdash \phi_2\ \mathsf{true}}\ \wedge\mathrm{E\scriptsize R}$$

Disjunction Introduction and Elimination

$$\frac{\Gamma \vdash \phi_1\ \mathsf{true}}{\Gamma \vdash \phi_1 \vee \phi_2\ \mathsf{true}}\ \vee\mathrm{I\scriptsize L} \qquad \frac{\Gamma \vdash \phi_2\ \mathsf{true}}{\Gamma \vdash \phi_1 \vee \phi_2\ \mathsf{true}}\ \vee\mathrm{I\scriptsize R}$$

$$\frac{\Gamma \vdash \phi_1 \vee \phi_2 \ \mathsf{true} \qquad \Gamma, \phi_1 \ \mathsf{true} \vdash \phi \ \mathsf{true} \qquad \Gamma, \phi_2 \ \mathsf{true} \vdash \phi \ \mathsf{true}}{\Gamma \vdash \phi \ \mathsf{true}} \vee\mathrm{E}$$

1.2.1 Basic Properties of Intuitionistic Entailment

Reflexivity

$$\frac{}{\Gamma, \phi \ \mathsf{true} \vdash \phi \ \mathsf{true}}$$

Transitivity

$$\frac{\Gamma \vdash \psi \ \mathsf{true} \qquad \Gamma, \psi \ \mathsf{true} \vdash \phi \ \mathsf{true}}{\Gamma, \phi \ \mathsf{true} \vdash \phi \ \mathsf{true}}$$

Contraction

$$\frac{\Gamma, \phi \ \mathsf{true}, \phi \ \mathsf{true} \vdash \psi \ \mathsf{true}}{\Gamma, \phi \ \mathsf{true} \vdash \psi \ \mathsf{true}}$$

Exchange

$$\frac{\Gamma \vdash \phi\ \mathsf{true}}{\pi(\Gamma) \vdash \phi\ \mathsf{true}}$$

1.3 Order Theoretic Semantics: *Hayting Algebras*

IPL viewed order theoretically gives rise to a *Hayting Algebra(HA)*. To define *HA* we need the notion of a *lattice*. For our purposes we define it as follows[1]:

Definition: A *lattice* is a *pre–order* with finite meets and joins.

In addition, we define *bounded lattice* as follows:

Definition: A *bounded lattice* $(L, \leq)$ is a lattice that additionally has a greatest element 1 and a least element 0, which satisfy

$0 \leq x \leq 1$ for every x in L

Finally, we can define *HA*:

Definition: A *HA* is a bounded lattice $(L, \leq, 0, 1)$ s.t. for every $a, b \in L$ there exists an x (we name it $a \to b$) with the properties:

1. $a \wedge x \leq b$

[1] One can take a lattice being a partial order. The same results hold with slight modifications.

2. x is the greatest such element

Axiomatization of HAs

We can axiomatize the meet (i.e. greatest lower bound)($\wedge$) of ϕ, ψ for any lower bound χ.

$$\frac{}{\phi \wedge \psi \leq \phi} \qquad \frac{}{\phi \wedge \psi \leq \psi}$$

$$\frac{\chi \leq \phi \quad \chi \leq \psi}{\chi \leq \phi \wedge \psi}$$

We can axiomatize the join ($\vee$)(i.e. the least upper bound) of ϕ, ψ for any upper bound χ as follows .

$$\frac{}{\phi \leq \phi \vee \psi} \qquad \frac{}{\psi \leq \phi \vee \psi}$$

$$\frac{\phi \leq \chi \quad \psi \leq \chi}{\phi \vee \psi \leq \chi}$$

We can axiomatize the existence of a greatest element as follows:

$$\frac{}{\chi \leq 1}$$

which says that 1 is the greatest element.

We can axiomatize the existence of a least element as follows:

$$\frac{}{0 \leq \chi}$$

which says that 0 is the least element.

Finally, to axiomatize *HAs* we require the existence of exponentials for every ϕ, ψ as follows:

$$\frac{}{\phi \wedge (\phi \supset \psi) \leq \psi} \qquad \frac{\phi \wedge \chi \leq \psi}{\chi \leq \phi \supset \psi}$$

Soundness and Completeness

Theorem. $\Gamma \vdash_{IPL} \phi\,\mathsf{true}$ iff for any *Heyting Algebra H* we have $\Gamma^+ \leq \phi^*$ where $*$ is defined as the lifting of any map of Props to elements of H and $(+)$ is defined inductively on the length of Γ as follows

$$\begin{array}{rcl} nil^+ & = & \top \\ (\Gamma, \phi)^+ & = & \Gamma^+ \wedge \phi * \end{array}$$

1.4 Proof trees and proof terms

IPL can be viewed as a declarative axiomatization of proof constructs. Take the introduction rule for conjunction as an example:

$$\frac{\Gamma \vdash \phi_1 \ \mathsf{true} \qquad \Gamma \vdash \phi_2 \ \mathsf{true}}{\Gamma \vdash \phi_1 \wedge \phi_2 \ \mathsf{true}} \wedge\mathrm{I}$$

The rule says, "given the existence a proof of ϕ and a proof of ψ from assumptions Γ, there exists a proof of $\phi \wedge \psi$ from assumptions Γ at hand ".

We used the description "declarative" because in this format *IPL* sequents $\Gamma \vdash \ \mathsf{true}$ do not describe how such existentials are realized. It is in essence a logic of "proof relevant truth" but it does not involve the proofs themselves as first class objects.

An alternative presentation is to explicate proof constructs by directly providing a system of "proof trees". Such, an approach was actually championed in Gentzen's natural deduction systems and is the necessary move to obtain proof calculi. Once we have proof explicit proof objects (either as trees, or as we will, see as terms) the system is enriched with equality principles involving such objects. Such rules give computational value ("proof dynamics") to the constructs and are the driver idea in the "Curry- Howard Isomorphism" and its extensions.

Here we provide such a formulation in proof trees of judgments together with the equality rules on trees, essentially following Gentzen. Proof trees of judgments have the following shape:

$$\begin{array}{c} J_1, \ldots, J_i \\ \vdots \\ \\ J \end{array}$$

We focus one judgments J of the form $A\,\mathsf{true}$. Here are the the rules for constructing proof trees. First, the deductions using reflection on hypothesis are valid:

$$\begin{array}{c} A_1\ \mathsf{true}, \ldots, A_i\ \mathsf{true} \\ \vdots \\ A_{j\in 1\ldots i}\ \mathsf{true} \end{array}$$

$$\textsc{$\wedge$ Intro}\quad \dfrac{\begin{array}{c}\mathcal{D}\\ A\ \mathsf{true}\end{array} \qquad \begin{array}{c}\mathcal{E}\\ B\ \mathsf{true}\end{array}}{A \wedge B\ \mathsf{true}}$$

$$\wedge\ \textsc{Elim1}\quad \dfrac{\begin{array}{c}\mathcal{D}\\ A \wedge B\ \mathsf{true}\end{array}}{A\ \mathsf{true}} \qquad\qquad \wedge\ \textsc{Elim2}\quad \dfrac{\begin{array}{c}\mathcal{D}\\ A \wedge B\ \mathsf{true}\end{array}}{B\ \mathsf{true}}$$

1.4.1 Linear representation of trees with proof terms

IPL can be viewed as a logic for proof constructs. Take the introduction rule for conjunction as an example:

$$\frac{\Gamma \vdash \phi_1\ \mathsf{true} \qquad \Gamma \vdash \phi_2\ \mathsf{true}}{\Gamma \vdash \phi_1 \wedge \phi_2\ \mathsf{true}}\ \wedge\mathrm{I}$$

The rule says, "given a proof of ϕ and a proof of ψ from assumptions Γ, we should have a proof of $\phi \wedge \psi$ from assumptions Γ at hand". The simply typed lambda calculus makes this proof-tracking explicit by adding constructors and destructors on proof terms. Hypothetical assumptions are assigned unique variables as proof terms. The judgments of *IPL* are then transformed into judgments on proof terms. This is the well–known Curry–Howard Correspondence.

$$\phi_1\ \ \mathsf{true}, \ldots, \phi_n\ \ \mathsf{true} \vdash \phi\ \ \mathsf{true} \quad \rightleftharpoons \quad x_1 : \phi_1, x_2 : \phi_2, \ldots x_n : \phi_n \vdash M : \phi$$

Simply typed lambda calculus

Type Formation

$$\frac{}{P_i \in \mathsf{Type}}\ \textsc{Atom} \qquad \frac{}{\top \in \mathsf{Type}}\ \textsc{Top} \qquad \frac{}{\bot \in \mathsf{Type}}\ \textsc{Bottom}$$

$$\frac{\phi_1 \in \mathsf{Type} \qquad \phi_2 \in \mathsf{Type}}{\phi_1 \to \phi_2 \in \mathsf{Type}}\ \textsc{Arr} \qquad \frac{\phi_1 \in \mathsf{Type} \qquad \phi_2 \in \mathsf{Type}}{\phi_1 \times \phi_2 \in \mathsf{Type}}\ \textsc{Prod}$$

$$\frac{\phi_1 \in \mathsf{Type} \qquad \phi_2 \in \mathsf{Type}}{\phi_1 + \phi_2 \in \mathsf{Type}}\ \textsc{Union}$$

Context Formation

$$\frac{}{\mathsf{nil}\ \ \mathsf{ctx}}\ \textsc{Nil} \qquad \frac{\Gamma\ \mathsf{ctx} \qquad \phi \in \mathsf{Type} \qquad x \text{ fresh in } \Gamma}{\Gamma,\ x : \phi\ \ \mathsf{ctx}}\ \Gamma\text{-}\textsc{Add}$$

Context Reflection

$$\frac{\Gamma\ \mathsf{ctx} \qquad x : \phi \in \Gamma}{\Gamma \vdash x : \phi}\ \Gamma\text{-}\textsc{Refl}$$

Top Introduction – Bottom Elimination

$$\frac{}{\Gamma \vdash \langle\rangle : \top}\ \top\mathrm{I} \qquad \frac{\Gamma \vdash M : \bot}{\Gamma \vdash abort[\phi](M) : \phi}\ \bot\mathrm{E}$$

Function Construction and Application

$$\frac{\Gamma, x : \phi_1 \vdash M : \phi_2}{\Gamma \vdash \lambda x.M : \phi_1 \to \phi_2}\ \lambda\text{–}\textsc{Abs} \qquad \frac{\Gamma \vdash M : \phi_1 \to \phi_2 \qquad \Gamma \vdash M^{'} : \phi_1}{\Gamma \vdash (MM^{'}) : \phi_2}\ \textsc{App}$$

Tuple Construction and Projections

$$\frac{\Gamma \vdash M : \phi_1 \qquad \Gamma \vdash M^{'} : \phi_2}{\Gamma \vdash \langle M, M^{'}\rangle : \phi_1 \times \phi_2}\ \textsc{Tup}$$

$$\frac{\Gamma \vdash M : \phi_1 \times \phi_2}{\Gamma \vdash \mathrm{fst}(M) : \phi_1}\ \textsc{LPrj} \qquad \frac{\Gamma \vdash M : \phi_1 \times \phi_2}{\Gamma \vdash \mathrm{snd}(M) : \phi_2}\ \textsc{RPrj}$$

Union Construction and Elimination

$$\frac{\Gamma \vdash M : \phi_1}{\Gamma \vdash inj_l[\phi_2](M) : \phi_1 + \phi_2}\ \textsc{InjL} \qquad \frac{\Gamma \vdash M : \phi_2}{\Gamma \vdash inj_r[\phi_1](M) : \phi_1 + \phi_2}\ \textsc{InjR}$$

$$\frac{\Gamma \vdash M : \phi_1 + \phi_2 \qquad \Gamma, x : \phi_1 \vdash N : \phi \qquad \Gamma, y : \phi_2 \vdash O : \phi}{\Gamma \vdash \text{case } M \text{ of } inj_l(x) \longmapsto N |\ inj_r(y) \longmapsto O : \phi} \vee\text{E}$$

1.5 The computational view

Given the formulation of the λ-calculus we can think of *formulae-as-types* and *proof terms-as-programs*. This enriches logic with a computational aspect(*proof dynamics*) that is absent from other formulations. Dynamics stems from Gentzen's insight to give an equational system for proofs equipped with $\beta\eta$ rules for proof-tree conversion. These insights, give rise to λ-calculus dynamics if we devise an execution strategy (an *operational semantics*) for program reduction that respects these rules. The correspondence between computations and Gentzen equational principles for proof terms is enlightened by specific metatheoretic results.

1.5.1 Definitional Equality

We want to think about when two proofs $M : A$ and $M' : A$ are the same. In the following we elaborate Gentzen's principles introducing an equivalence relation called *definitional equality* that respects these principles, denoted $M \equiv M' : A$. We want definitional equality $\equiv$ to be the least congruence closed under the β rules. We will define what this means:

A *congruence* is an equivalence relation (i.e. reflexive transitive and antisymmetric) that respects our operators as formulated below.

$$\frac{}{\Gamma \vdash M \equiv M : \phi}\ \mathrm{R} \qquad \frac{\Gamma \vdash M \equiv M' : \phi}{\Gamma \vdash M' \equiv M : \phi}\ \mathrm{S}$$

$$\frac{\Gamma \vdash M \equiv M' : \phi \qquad \Gamma \vdash M' \equiv M'' : \phi}{\Gamma \vdash M \equiv M'' : \phi}\ \mathrm{T}$$

For the equivalence relation to respect operators we means that if $M \equiv M' : A$, then that their image under any operator should be equivalent. In other words, we should be able to replace M with M' everywhere.For example:

Congruence over fst

$$\frac{\Gamma \vdash M \equiv M' : A \wedge B}{\Gamma \vdash \mathrm{fst}(M) \equiv \mathrm{fst}(M') : A}$$

Inversion Principle

Gentzen's Inversion Principle captures the idea that "elim is post-inverse to intro," which is the informal notion that the elimination rules should cancel the introduction rules.

The β rules are as follows:

$$\frac{\Gamma \vdash M : \phi_1 \qquad \Gamma \vdash N : \phi_2}{\Gamma \vdash \text{fst}(\langle M, N \rangle) \equiv M : \phi_1} \beta\wedge_1$$

$$\frac{\Gamma \vdash M : \phi_1 \qquad \Gamma \vdash N : \phi_2}{\Gamma \vdash \text{snd}(\langle M, N \rangle) \equiv N : \phi_2} \beta\wedge_2 \qquad \frac{\Gamma, x : A \vdash M : B \qquad \Gamma \vdash N : A}{\Gamma \vdash (\lambda x.M)(N) \equiv [N/x]M : B} \beta \supset$$

$$\frac{\Gamma, x : \phi_1 \vdash N : \psi \qquad \Gamma, y : \phi_2 \vdash O : \psi \qquad \Gamma \vdash P : \phi_1}{\Gamma \vdash (\text{case } inj_l(P) \text{ of } inj_l(x) \longmapsto N |\ inj_r(y) \longmapsto O) \equiv [P/x]N : \psi} \beta\vee_1$$

$$\frac{\Gamma, x : \phi_1 \vdash N : \psi \qquad \Gamma, y : \phi_2 \vdash O : \psi \qquad \Gamma \vdash Q : \phi_2}{\Gamma \vdash (\text{case } inr_r(Q) \text{ of } inj_l(x) \longmapsto N |\ inj_r(y) \longmapsto O) \equiv [Q/y]O : \psi} \beta\vee_2$$

Unicity Principle

Gentzen's Unicity Principles on the other hand capture the idea that "intro is post-inverse to elim". There should be only one way – modulo definitional equivalence – to prove something. The "β" rules give rise to computational interpretation. The "η" rules impose properties that the computational model should satisfy (e.g. Church-Rosser property).

The η rules are given below:

$$\frac{\Gamma \vdash M : \top}{\Gamma \vdash M \equiv \langle\ \rangle : \top}\ \eta\top \qquad \frac{\Gamma \vdash M \equiv \langle \mathrm{fst}(M), \mathrm{snd}(M) \rangle : A \wedge B}{\Gamma \vdash M : A \wedge B}\ \eta\wedge$$

$$\frac{\Gamma \vdash M : \phi \supset \psi}{\Gamma \vdash M \equiv \lambda x.Mx : \phi \supset \psi}\ \eta\supset$$

$$\frac{\Gamma, z : \phi_1 + \phi_2 \vdash M : \psi \qquad \Gamma \vdash N : \phi_1 + \phi_2}{\begin{array}{ll}\Gamma \vdash M[N/z] \equiv \text{case } N \text{ of} & |inj_l(x) \mapsto M[inj_l(x)/z] \\ & inj_r(y) \mapsto M[inj_r(y)/z]) : \psi\end{array}}\ \eta\vee$$

1.5.2 Propositions as Types

There is a correspondence between propositions and types:

Propositions	Types
$\top$	1
$A \wedge B$	$A \times B$
$A \supset B$	function $A \to B$ or B^A
$\perp$	0
$A \vee B$	$A + B$

1.6 Categories for *IPL*

In a Heyting Algebra, we have a preorder (or, partial order in the "textbook" definition) $\phi \leq \psi$ when ϕ implies ψ. *HAs* are insufficient, however, for the treatment of proof objects (there can be at most one instance of $\phi \leq \psi$ for specific ϕ,ψ). We can keep track of proofs, so if M is a proof from Γ to ψ, we want to think of it as a map $M : \Gamma+ \to \psi+$. The category theoretic analog of a Heyting Algebra is a Bi–Cartesian Closed Category ($BiCCC$). That is a category with all finite products, co–products and exponentials. The axiomatization of a category (in general), finite (and nullary) products and co-products and exponentials is given in this section.

1.6.1 Definitions and Axioms of a Category

A category has *objects* $\phi, \psi, \ldots$ and *arrows* $f, g, h \ldots$ Each arrow goes from an object to an object. To say that g goes from ϕ to ψ we write $g : \phi \to \psi$, or say that ϕ is the domain of g, and ψ the *co–domain*. We write $Dom(g) = \phi$ and $Cod(g) = \psi$. We say that two arrows f and g are *composable* with $Dom(f) = Cod(g)$. If f and g are composable, they have a *composite*, an arrow called $f \circ g$. There is an *identity* for every object ϕ.

$$\frac{}{\mathrm{id} : \phi \to \phi}\ \mathrm{ID}_{ex} \qquad \frac{f : \phi \to \psi \qquad g : \psi \to \chi}{g \circ f : \phi \to \chi}\ \textsc{Comp}$$

$$\frac{f : \phi \to \psi}{id_{\psi} \circ f = f : \phi \to \psi}\ \mathrm{ID}_{l} \qquad \frac{f : \phi \to \psi}{f \circ id_{\phi} = f : \phi \to \psi}\ \textsc{idr}$$

$$\frac{f : \phi \to \psi \qquad g : \psi \to \chi \qquad h : \chi \to \omega}{h \circ (g \circ f) = (h \circ g) \circ f : \phi \to \omega}\ \textsc{idr}$$

1.6.2 Terminal, Co-Terminal objects, Products and Co-Products

Now we can think about objects in the category that correspond to propositions given in the correspondence.

Terminal Object 1 is the terminal object, also called the final object, which corresponds to $\top$. For any object Γ there is a unique map $\Gamma \to 1$.

$$\frac{}{\langle\ \rangle : \phi \to 1}\ \textsc{Existence} \qquad \frac{M : \Gamma \to 1}{M = \langle\ \rangle : \Gamma \to 1}\ \textsc{Unicity}(\eta)$$

Product For any objects ϕ and ψ there is an object $\chi = \phi \times \psi$ equipped with arrows fst : $\phi \times \psi \to \phi$ and snd : $\phi \times \psi \to \psi$ that is the *product* of ϕ and ψ, which corresponds to the join $\phi \wedge \psi$. For any other object Γ with arrows $M : \Gamma \to \phi$ and $\Gamma \to \psi$ there exists *unique* arrow, $\langle M, N\rangle$ s.t. $\text{fst} \circ \langle M, N\rangle = M(\beta\times_1)$ and $\text{snd} \circ \langle M, N\rangle = N(\beta\times_2)$.

$$\frac{M : \Gamma \to \phi \qquad N : \Gamma \to \psi}{\langle M, N\rangle : \Gamma \to \phi \times \psi}\ \textsc{Exist}_1$$

$$\frac{M : \Gamma \to \phi \qquad N : \Gamma \to \psi}{\text{fst} \circ \langle M, N\rangle : \Gamma \to \phi}\ \textsc{Exist}_2(\beta_1)$$

$$\frac{M : \Gamma \to \phi \qquad N : \Gamma \to \psi}{\text{snd} \circ \langle M, N\rangle : \Gamma \to \phi}\ \textsc{Exist}_3(\beta_2)$$

$$\frac{P : \Gamma \to \phi \times \psi \qquad \text{fst} \circ P = M : \Gamma \to \phi \qquad \text{snd} \circ P = N : \Gamma \to \psi}{P = \langle M, N\rangle : \Gamma \to \phi \times \psi}\ \textsc{Un}(\eta)$$

Diagrammatically:

$$\begin{tikzcd}
 & \Gamma \arrow[d, dashed, "\langle M,N\rangle"] \arrow[dl, bend right, "M"'] \arrow[dr, bend left, "N"] & \\
\phi & \phi \times \psi \arrow[l, "\text{fst}"] \arrow[r, "\text{snd}"'] & \psi
\end{tikzcd}$$

Exponentials Given objects A and B, an exponential B^A (which corresponds to $A \supset B$) is an object with the following universal property:

$$\begin{tikzcd}
C \arrow[d, dashed, "\lambda(h)"'] & C \times A \arrow[d, dashed, "\lambda(h)\times \mathrm{id}_A"'] \arrow[dr, "h"] & \\
B^A & B^A \times A \arrow[r, "\mathrm{ap}"'] & B
\end{tikzcd}$$

such that the diagram commutes.

This means that there exists a map $\mathrm{ap} : B^A \times A \to B$ (application map) that corresponds to implication elimination.

The universal property is that for all objects C that have a map $h : C \times A \to B$, there exists a unique map $\lambda(h) : C \to B^A$ such that

$$\mathrm{ap} \circ (\lambda(h) \times \mathrm{id}_A) = h : C \times A \to B$$

This means that the diagram commutes. Another way to express the induced map is $\lambda(h) \times \mathrm{id}_A = \langle \lambda(h) \circ \mathrm{fst}, \mathrm{snd} \rangle$.

The map $\lambda(h) : C \to B^A$ is unique, meaning that

$$\frac{\mathrm{ap} \circ (g \times \mathrm{id_A}) = h : C \times A \to B}{g = \lambda(h) : C \to B^A}$$

Co–Products For any objects ϕ and ψ there is an object $\chi = \phi + \psi$ equipped with arrows $\mathrm{inl} : \phi \to \phi + \psi$ and $\mathrm{inr} : \psi \to \phi + \psi$ that is the *co-product* of ϕ and ψ, which corresponds to the meet $\phi \wedge \psi$. For any other object ω with arrows $M : \omega \rightarrow \phi \vee \psi$ and $N : \omega \rightarrow \phi \vee \psi$ there exists *unique* arrow, M, N s.t. $\{M, N\} \circ \mathrm{inl} = M$ and $\{M, N\} \circ \mathrm{inr} = N$.

$$\frac{O : \Gamma \rightarrow \phi}{\mathrm{inl} \circ O : \Gamma \rightarrow \phi + \psi}\ \textsc{Exist}_1 \qquad \frac{P : \Gamma \rightarrow \psi}{\mathrm{inr} \circ P : \Gamma \rightarrow \phi + \psi}\ \textsc{Exist}_2$$

$$\frac{O : \Gamma \rightarrow \phi \qquad M : \phi \rightarrow \omega \qquad N : \psi \rightarrow \omega}{\{M, N\} \circ \mathrm{inl} \circ O = M \circ O : \Gamma \rightarrow \omega}\ \textsc{Exist}_3(\beta_1)$$

$$\frac{P : \Gamma \rightarrow \psi \qquad M : \phi \rightarrow \omega \qquad N : \psi \rightarrow \omega}{\{M, N\} \circ \mathrm{inr} \circ P = N \circ P : \Gamma \rightarrow \omega}\ \textsc{Exist}_3(\beta_2)$$

$$\frac{M : \Gamma \rightarrow \phi \qquad N : \Gamma \rightarrow \psi}{\mathrm{snd} \circ \langle M, N \rangle : \Gamma \rightarrow \phi}\ \textsc{Exist}_3(\beta_2)$$

$$\frac{\begin{array}{c} O : \Gamma \rightarrow \phi \qquad P : \Gamma \rightarrow \psi \qquad U : \phi + \psi \rightarrow \omega \qquad M : \phi \rightarrow \omega \\ N : \psi \rightarrow \omega \qquad U \circ \mathrm{inl} \circ O = M \qquad U \circ \mathrm{inr} \circ N = M \end{array}}{U = \{M, N\}}\ \textsc{Un}(\eta)$$

Diagrammatically:

$$\begin{tikzcd}
\phi \arrow[dr, "\mathsf{inl}"] \arrow[ddr, bend right, "M"'] & & \psi \arrow[dl, "\mathsf{inr}"'] \arrow[ddl, bend left, "N"] \\
{} & \phi + \psi \arrow[l, phantom, "(\beta)"] \arrow[r, phantom, "(\beta)"] \arrow[d, bend right, "M"'] \arrow[d, bend left, "\{M,N\}"] \arrow[d, phantom, "(\eta)"] & {} \\
 & C &
\end{tikzcd}$$

Bibliography

[1] Sergei Nikolaevich Artemov. Kolmogorov and gödel's approach to intuitionistic logic: current developments. *Russian Mathematical Surveys*, 59(2):203, 2004.

[2] H. P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics*. Sole Distributors for the U.S.A. And Canada, Elsevier Science Pub. Co., 1984.

[3] Luitzen Egbertus Jan Brouwer and A Heyting. *Collected Works: Vol.: 1.: Philosophy and Foundations of Mathematics*. North-Holland Publishing Company, American Elsevier Publishing Company, Incorporated, 1975.

[4] Michael AE Dummett. *Elements of intuitionism*, volume 39. Oxford University Press, 2000.

[5] Gerhard Gentzen. The collected papers of gerhard gentzen. 1970.

[6] Jean-Yves Girard, Paul Taylor, and Yves Lafont. *Proofs and types*, volume 7. Cambridge University Press Cambridge, 1989.

[7] Robert Harper. Homotopy type theory seminar. `http://www.cs.cmu.edu/~rwh/courses/hott/notes/notes_week1.pdf`.

[8] Robert Harper. Extensionality, intensionality, and Brouwer's dictum. `http://existentialtype.wordpress.com/2012/08/11/extensionality-intensionality-and-brouwers-dictum/`, August 2012.

[9] Robert Harper. Constructive mathematics is not metamathematics. `http://existentialtype.wordpress.com/2013/07/10/constructive-mathematics-is-not-meta-mathematics/`, July 2013.

[10] S.Balzer H.DeYoung. Homotopy type theory seminar. `http://http://www.cs.cmu.edu/~rwh/courses/hott/`.

[11] Arend Heyting. *Intuitionism: an introduction*, volume 41. Elsevier, 1966.

[12] Andrey Kolmogorov. O principe tertium non datur. mathematicheskij sbornik 32: 646–667. *English trans. in van Heijenoort [1967, 414-437]*, 1925.

[13] Per Martin-Löf. On the meanings of the logical constants and the justifications of the logical laws. *Nordic Journal of Philosophical Logic*, 1(1):11–60.

[14] Per Martin-Lof and Giovanni Sambin. *Intuitionistic type theory*, volume 17. Bibliopolis Naples, 1984.

[15] Frank Pfenning. Lecture notes on harmony. `http://www.cs.cmu.edu/~fp/courses/15317-f09/lectures/03-harmony.pdf`, September 2009.

[16] Frank Pfenning. Lecture notes on natural deduction. `http://www.cs.cmu.edu/~fp/courses/15317-f09/lectures/02-natded.pdf`, August 2009.

[17] Benjamin C. Pierce. *Types and Programming Languages.* MIT Press, Cambridge, MA, USA, 2002.