Normalization by Completeness with Heyting Algebras

Gaëtan Gilbert, Olivier Hermant

Normalisation by Completeness with Heyting Algebras

Gaëtan Gilbert[1][2] and Olivier Hermant[3][4]

[1] ENS Lyon, France
gaetan.gilbert@ens-lyon.fr
[2] Inria Paris, France
[3] MINES ParisTech, PSL Research University, France
[4] Wesleyan University, Middletown, CT, USA
olivier.hermant@mines-paristech.fr

Abstract. Usual normalization by evaluation techniques have a strong relationship with completeness with respect to Kripke structures. But Kripke structures is not the only semantics that fits intuitionistic logic: Heyting algebras are a more algebraic alternative.
In this paper, we focus on this less investigated area: how completeness with respect to Heyting algebras generate a normalization algorithm for a natural deduction calculus, in the propositional fragment. Our main contributions is that we prove in a direct way completeness of natural deduction with respect to Heyting algebras, that the underlying algorithm natively deals with disjunction, that we formalized those proofs in Coq, and give an extracted algorithm.

1 Introduction

In logic, a restriction to cut-free proofs makes analysis of a theory and proof-search significantly simpler. Evaluating programs boils down to finding efficient ways to reach a normal form, in order to produce a result.

Through the proof-as-programs paradigms, those two processes can be reduced to a single one: reduction steps of lambda-terms, such as β reduction, can be seen as a way to remove cuts from proofs expressed in natural deduction. Under this correspondence, a proof is cut-free when the associated proof-term is in normal form.

But there exist other, semantic, ways to eliminate cuts from proofs [15], through a completeness theorem that produces cut-free proofs, hereafter strong completeness, in combination with soundness. When those proofs can be made constructive [17,10,3], a natural question arises: what is the computational content of such proofs ?

A link has already been exhibited. A line of research in program normalization, dubbed normalisation by evaluation, aims at evaluating a program in a type-directed fashion, by reusing the reduction mechanisms at

hand at the meta level[5] through a pair of reflection/reification functions [2]. Soon after, Coquand noticed a strong similarity with completeness proofs [4].

This seminal work has been extended to more complex types [5,1], and also studied from the point of view of the completeness theorem for intuitionistic natural deduction with respect to Kripke-like structures [9,6,8]. But when it comes to incorporating disjunction, one must be very careful, in particular because Kripke structures require worlds to *decide* between both members of the conjunction - from a pure normalization by evaluation point of view, dealing with sum types also requires special care.

In this paper, we follow this line, relating constructive completeness proofs and normalization procedures. But, instead of considering Kripke semantics, as has been done in the works described above, we consider *Heyting algebras*:

- completeness theorems for the cut-free system (strong completeness), and therefore cut elimination [11,7] can be proved constructively;
- handling disjunction is straightforward, and hence we get cut elimination for sum types.

An adaptation of existing completeness proofs with respect to Heyting algebras is required, since all the known proofs, starting from Okada's contribution to linear logic [12], use sequent calculus.

To support these claims, we have formalized the proofs of this paper in Coq, and used extraction to get an executable interpreter. To keep the complexity of the formalization reasonable, we remained in the propositional fragment. The Coq sources are available at `https://github.com/SkySkimmer/NormalisationByCompleteness`.

The organization of this paper is the following: in Sec. 2 we recall natural deduction, in particular the notion of cut, and show basic lemmas. In Sec. 3, we develop the strong completeness proof, and discuss its Coq formalization in the next Sec. 4.1, where we also devise the behavior of the extracted algorithm on examples. Sec. 5 concludes the paper.

2 Natural Deduction

Definition 1 (Terms and formulas). *Let $\mathcal{V}$ be an infinite set of variables, $\mathcal{S}$ be a set of function symbols along with an arity and $\mathcal{P}$ be a set of predicate symbols along with an arity. The set of terms $\mathcal{T}$ is defined by:*

$$t ::= x \mid f(t_1, ..., t_n)$$

[5] namely, the programming language in which the evaluation function is written

where $x \in \mathcal{V}$ and $f \in \mathcal{S}$ has arity n. The set of formulas $\mathcal{F}$ is defined by:

$$A, B ::= P(t_1, ..., t_n) \mid A \wedge B \mid A \vee B \mid A \Rightarrow B \mid \top \mid \bot \mid \forall x.A \mid \exists x.A$$

where $P \in \mathcal{P}$ has arity n.

Definition 2 (Substitutions). *A substitution σ is a partial function from variables to terms, with finite domain.*
We expand it inductively to a function from terms to terms and formulas to formulas, letting $\sigma(x) = x$ for $x \notin dom(\sigma)$.

Notably for $\mathcal{Q} \in \{\forall, \exists\}, \sigma(Q\ x.A) \coloneqq \mathcal{Q}x.\sigma(A)$, assuming x fresh w.r.t. the image of σ by α-conversion. This is always possible since $dom(\sigma)$ is finite, and so the image of σ is also finite.

Definition 3 (Updated Substitution). *Let σ be a substitution, $x \in \mathcal{V}$ and $t \in \mathcal{T}$, $\sigma[x \mapsto t]$ is the substitution with domain $dom(\sigma) \cup \{x\}$ such that for all $y \neq x$, $\sigma[x \mapsto t](y) = \sigma(y)$ and $\sigma[x \mapsto t](x) = t$.*

The substitution with the empty set as domain is denoted $\emptyset$. For t a term (resp. A a formula), x a variable and u a term, we abbreviate $\emptyset[x \mapsto u](t)$ (resp. $\emptyset[x \mapsto u](A)$) as $t[u/x]$ (resp. $A[t/x]$).

Definition 4 (Contexts). *A context Γ is a list of formulas $[A_1, ..., A_n]$. We let Γ, A be the concatenation of A and Γ. Membership is denoted $B \in \Gamma$. Inclusion, denoted $\Gamma \subseteq \Sigma$, holds when any $B \in \Gamma$ is also in Σ.*

Remark 1. The relation $\subseteq$ is a preorder, but not an order. Indeed, it strictly subsumes contraction ($\Gamma, A, A \subseteq \Gamma, A$) as well as reordering of premises.

Definition 5 (Cut-Free Proofs). *Figure 1 defines the relations $\vdash_{ne}$ (neutral proof) and $\vdash^*$ (cut-free proof) by mutual induction.*

In Fig. 1, rules on the left are introduction rules and produce cut-free proofs, while rules on the right are elimination rules and produce neutral proofs. FV denotes the set of free variables. The usual natural deduction calculus NJ is a merge of both relations. For two contexts Γ, Σ and any relation $\vdash'$, $\Sigma \vdash' \Gamma$ denotes $\Sigma \vdash' A$ for all $A \in \Gamma$.

Definition 6 (Natural Deduction). *The judgment $\Gamma \vdash A$ has the same rules as both $\Gamma \vdash^* A$ and $\Gamma \vdash_{ne} A$.*

Therefore, if $\Gamma \vdash^* A$ or $\Gamma \vdash_{ne} A$, then $\Gamma \vdash A$.

$$\frac{\Gamma \vdash_{ne} A}{\Gamma \vdash^* A}\ coerce \qquad \frac{A \in \Gamma}{\Gamma \vdash_{ne} A}\ ax$$

$$\frac{\Gamma \vdash^* A \qquad \Gamma \vdash^* B}{\Gamma \vdash^* A \wedge B}\ \wedge_I \qquad \frac{\Gamma \vdash_{ne} A \wedge B}{\Gamma \vdash_{ne} A}\ \wedge_{E_l} \qquad \frac{\Gamma \vdash_{ne} A \wedge B}{\Gamma \vdash_{ne} B}\ \wedge_{E_r}$$

$$\frac{\Gamma \vdash^* A}{\Gamma \vdash^* A \vee B}\ \vee_{I_l} \quad \frac{\Gamma \vdash^* B}{\Gamma \vdash^* A \vee B}\ \vee_{I_r} \qquad \frac{\Gamma \vdash_{ne} A \vee B \qquad A, \Gamma \vdash^* C \qquad B, \Gamma \vdash^* C}{\Gamma \vdash_{ne} C}\ \vee_E$$

$$\frac{\Gamma, A \vdash^* B}{\Gamma \vdash^* A \Rightarrow B}\ \Rightarrow_I \qquad \frac{\Gamma \vdash_{ne} A \Rightarrow B \qquad \Gamma \vdash^* A}{\Gamma \vdash_{ne} B}\ \Rightarrow_E$$

$$\frac{}{\Gamma \vdash^* \top}\ \top_I \qquad \frac{\Gamma \vdash_{ne} \bot}{\Gamma \vdash_{ne} A}\ \bot_E$$

$$\frac{\Gamma \vdash^* A \qquad x \notin FV(\Gamma)}{\Gamma \vdash^* \forall x.A}\ \forall_I \qquad \frac{\Gamma \vdash_{ne} \forall x.A}{\Gamma \vdash_{ne} A[t/x]}\ \forall_E$$

$$\frac{\Gamma \vdash^* A[t/x]}{\Gamma \vdash^* \exists x.A}\ \exists_I \qquad \frac{\Gamma \vdash_{ne} \exists x.A \qquad A, \Gamma \vdash^* C \qquad x \notin FV(C, \Gamma)}{\Gamma \vdash_{ne} C}\ \exists_E$$

Fig. 1. Rules of Natural Deduction

Lemma 1 (Weakening). *Let Γ, Σ be contexts such that $\Gamma \subseteq \Sigma$. Let A be a formula. The three following rules are admissible:*

$$\frac{\Gamma \vdash^* A}{\Sigma \vdash^* A} \qquad \frac{\Gamma \vdash_{ne} A}{\Sigma \vdash_{ne} A} \qquad \frac{\Gamma \vdash A}{\Sigma \vdash A}$$

Proof. By mutual induction on $\Gamma \vdash^* A$ and $\Gamma \vdash_{ne} A$, and by induction on $\Gamma \vdash A$. □

Corollary 1 (Contraction). *For any context Γ and any formula B, if $\Gamma, A, A \vdash B$ then $\Gamma, A \vdash B$.*

Neutral proofs are such that they can replace axioms in cut-free proofs without introducing any cut.

Lemma 2 (Axiom Replacement). *Let Γ, Σ be contexts and A be a formula. The three following rules are admissible:*

$$\frac{\Sigma \vdash_{ne} \Gamma \qquad \Gamma \vdash^* A}{\Sigma \vdash^* A} \qquad \frac{\Sigma \vdash_{ne} \Gamma \qquad \Gamma \vdash_{ne} A}{\Sigma \vdash_{ne} A} \qquad \frac{\Sigma \vdash \Gamma \qquad \Gamma \vdash A}{\Sigma \vdash A}$$

Proof. By mutual induction on $\Gamma \vdash^* A$ and $\Gamma \vdash_{ne} A$, and by induction on $\Gamma \vdash A$. Note that we need the weakening lemma (Lem. 1) when the context is modified in a premise of a rule.

Consider for instance the $\Rightarrow_I$ case of Fig. 1. $\Gamma, A \vdash^* B$ is derivable. $\Sigma, A \vdash_{ne} \Gamma, A$ holds, by weakening for Γ and by ax for A. By induction hypothesis, $\Sigma, A \vdash^* B$ and by $\Rightarrow_I$ we conclude $\Sigma \vdash^* A \Rightarrow B$. □

Lemma 3 (Kleene's Inversion Lemma). *Let Γ be a context, A and B be formulas.*
If $\Gamma \vdash_{ne} A \Rightarrow B$ (resp. $\Gamma \vdash^* A \Rightarrow B$) then $\Gamma, A \vdash_{ne} B$ (resp. $\Gamma, A \vdash^* B$).

Proof. If $\Gamma \vdash_{ne} A \Rightarrow B$, then by weakening $\Gamma, A \vdash_{ne} A \Rightarrow B$. By *ax* and *coerce* we have $\Gamma, A \vdash^* A$. Then by $\Rightarrow_E$, $\Gamma, A \vdash_{ne} B$.

If $\Gamma \vdash^* A \Rightarrow B$, we analyze the last rule of the derivation:

- it is *coerce*: the premiss is $\Gamma \vdash_{ne} A \Rightarrow B$, then $\Gamma, A \vdash_{ne} B$ and by *coerce*, $\Gamma, A \vdash^* B$.
- otherwise it is $\Rightarrow_I$: the premiss is $\Gamma, A \vdash^* B$. □

3 Strong completeness by Heyting Algebras

3.1 Heyting Algebras

Definition 7 (Complete Lattice). *A complete lattice is a tuple*

$$\mathcal{A} = (A, \leq, \bigwedge, \bigvee)$$

such that $(A, \leq)$ is a partial order with arbitrary meet $\bigwedge$ and join $\bigvee$.

In the sequel, we distinguish the binary meet $\wedge$, join $\vee$ and the global maximum $\top$ (empty meet) and minimum $\bot$ (empty join).

Definition 8 (Complete Heyting Algebra). *A Heyting algebra is a structure $\mathcal{H} = (H, \leq, \wedge, \vee, \Rightarrow, \top, \bot, \bigwedge, \bigvee)$ such that $(H, \leq, \bigwedge, \bigvee)$ is a complete lattice and verifies the implication property*

$$\forall a\ b\ c, a \leq b \Rightarrow c \text{ if and only if } a \wedge b \leq c$$

Lemma 4. *In a Heyting algebra, binary meet and join distribute over each other.*

Proof. Let $a, b, c \in H$

- $a \wedge (b \vee c) \leq (a \wedge b) \vee (a \wedge c)$: we have $a \wedge b \leq (a \wedge b) \vee (a \wedge c)$ and $a \wedge c \leq (a \wedge b) \vee (a \wedge c)$. By the implication property,

  $$b \leq a \Rightarrow ((a \wedge b) \vee (a \wedge c)) \quad \text{and} \quad c \leq a \Rightarrow ((a \wedge b) \vee (a \wedge c))$$

  Then $b \vee c \leq a \Rightarrow ((a \wedge b) \vee (a \wedge c))$ and we conclude by the implication property.
- $(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c)$: holds in all lattices

- $a \vee (b \wedge c) \leq (a \vee b) \wedge (a \vee c)$: holds in all lattices
- $(a \vee b) \wedge (a \vee c) \leq a \vee (b \wedge c)$: By the implication property, this is equivalent to $a \vee b \leq (a \vee c) \Rightarrow (a \vee (b \wedge c))$
  $\iff a \leq (a \vee c) \Rightarrow (a \vee (b \wedge c))$ and $b \leq (a \vee c) \Rightarrow (a \vee (b \wedge c))$
  $\iff a \wedge (a \vee c) \leq a \vee (b \wedge c)$ (trivial) and $b \wedge (a \vee c) \leq a \vee (b \wedge c)$
  $\iff a \vee c \leq b \Rightarrow (a \vee (b \wedge c))$
  $\iff a \leq b \Rightarrow (a \vee (b \wedge c))$ and $c \leq b \Rightarrow (a \vee (b \wedge c))$
  $\iff a \wedge b \leq a \vee (b \wedge c)$ (trivial) and $c \wedge b \leq a \vee (b \wedge c)$ (trivial) □

Definition 9 (Interpretation). *A valuation on a set $\mathcal{D}$, called the domain, is a partial function $\varphi : \mathcal{V} \to \mathcal{D}$ with finite support. The syntax for updating valuations is the same as in Def. 3.*

A model on a Heyting algebra $\mathcal{H}$ is given by a domain $\mathcal{D}$, and for each function symbol $f \in \mathcal{S}$ (resp. predicate symbol $P \in \mathcal{P}$) of arity n a function $[\![f]\!] : \mathcal{D}^n \to \mathcal{D}$ (resp. a function $[\![P]\!] : \mathcal{D}^n \to H$).

Let φ be a valuation, t a term and A a formula, such that $FV(t) \cup FV(A) \subseteq dom(\varphi)$. The interpretations $[\![t]\!]_\varphi \in \mathcal{D}$ and $[\![A]\!]_\varphi \in \mathcal{H}$ are defined in the usual inductive way.
We define interpretation for contexts to be $[\![\Gamma]\!]_\varphi \coloneqq \bigwedge_{C \in \Gamma} [\![C]\!]_\varphi$.

Notably:

$$\begin{aligned} [\![P(t_1, ..., t_k)]\!]_\varphi &\coloneqq [\![P]\!]([\![t_1]\!]_\varphi, ..., [\![t_k]\!]_\varphi) \\ [\![\forall x.A]\!]_\varphi &\coloneqq \textstyle\bigwedge_{v \in \mathcal{D}} \{[\![A]\!]_{\varphi[x \mapsto v]}\} \\ [\![\exists x.A]\!]_\varphi &\coloneqq \textstyle\bigvee_{v \in \mathcal{D}} \{[\![A]\!]_{\varphi[x \mapsto v]}\} \end{aligned}$$

Theorem 1 (Soundness). *Let Γ be a context and A be a formula. If $\Gamma \vdash A$ is derivable, then for any Heyting algebra $\mathcal{H}$, for any model on $\mathcal{H}$ and valuation φ, $[\![\Gamma]\!]_\varphi \leq [\![A]\!]_\varphi$*

Proof. Standard induction [14]. □

3.2 Completeness

We now proceed to the construction of a universal Heyting algebra, that is suitable for *cut-free*, or *strong*, completeness, that is to say, that produces cut-free proofs [13]. This contrasts with more usual Lindenbaum algebras [14], formed with (provability-)equivalence classes of formulas.

Definition 10 (Extraction). *Let A be a formula. We define $\lfloor A \rfloor$ (the extraction of A) to be $\{\Gamma, \Gamma \vdash^* A\}$.*

$\lfloor A \rfloor$ is the set of contexts that prove A without cut, and will represent an upper bound for the interpretation of A, and as well the basis of our Heyting algebra below.

Definition 11 (Universal Heyting Algebra). *The underlying set of the universal Heyting algebra (aka the context algebra) is:*

$$\Omega \coloneqq \{\bigcap \lfloor A_i \rfloor, (A_i)_{i\in I} \textit{ family of formulas}\}$$

That is to say, the closure by arbitrary intersections of formula extractions. The partial order is inclusion and the operations are:

$$\begin{array}{rcl} a \leq b & \coloneqq & a \subseteq b \\ a \wedge b & \coloneqq & a \cap b \\ \bigwedge A & \coloneqq & \bigcap A \\ a \vee b & \coloneqq & \bigcap\{\omega \in \Omega, a \cup b \leq \omega\} \\ \bigvee A & \coloneqq & \bigcap\{\omega \in \Omega, \bigcup A \leq \omega\} \\ a \Rightarrow b & \coloneqq & \bigvee\{c \in \Omega, a \wedge c \leq b\} \\ \top & \coloneqq & \{\Gamma, \Gamma \textit{ context}\} = \lfloor \top \rfloor \\ \bot & \coloneqq & \{\Gamma, \forall A, \Gamma \vdash A\} = \lfloor \bot \rfloor \end{array}$$

By abuse of notation, we also denote this algebra as Ω. $\bigwedge$ and $\bigvee$ are clearly greatest lower and lowest upper bounds, respectively. We can also simplify a bit lowest upper bounds, thanks to the following lemma:

Lemma 5. *The following identities are verified:*

$$\begin{array}{rcl} a \vee b & = & \bigcap\{\lfloor D \rfloor, a \cup b \leq \lfloor D \rfloor, D \in \mathcal{F}\} \\ \bigvee A & = & \bigcap\{\lfloor D \rfloor, \bigcup A \leq \lfloor D \rfloor, D \in \mathcal{F}\} \\ a \Rightarrow b & = & \bigcap\{\lfloor D \rfloor, \bigcup\{c \in \Omega, a \wedge c \leq b\} \leq \lfloor D \rfloor, D \in \mathcal{F}\} \end{array}$$

Proof. We focus on the first identity. The two other have a similar proof, as $a \vee b$, $\bigvee A$ and $a \Rightarrow b$ are all defined as lowest upper bound.

By definition of $\vee$, $a \vee b \leq \bigcap\{\lfloor D \rfloor, a \cup b \leq \lfloor D \rfloor, D \in \mathcal{F}\}$. Conversely, let ω such that $a \cup b \leq \omega$. Since $\omega \in \Omega$, $\omega = \bigcap_{i\in I} \lfloor C_i \rfloor$ for some $(C_i)_{i\in I}$. For all $i \in I$, $a \vee b \leq \lfloor C_i \rfloor$, and therefore $\omega \leq \bigcap\{\lfloor D \rfloor, a \cup b \leq \lfloor D \rfloor\}$.

Lemma 6. *Let $\omega \in \Omega$, and $\Gamma \in \omega$. Then, for any context Δ, $\Delta, \Gamma \in \omega$.*

Proof. By applying Lem. 1 to Def. 10 and Def. 11. □

Lemma 7. *Ω forms a Heyting algebra.*

Proof. Ω is closed by arbitrary intersection and for all A, $\lfloor A \rfloor \in \Omega$, so the operations produce values in Ω. As already said, $\leq$ is an order for which $\wedge$ and $\bigwedge$ are greatest lower bounds, and $\vee$ and $\bigvee$ are lowest upper bounds. $\top$ and $\bot$ are trivially is the greatest and least element, respectively. It remains to check the implication property:

– Assume $a \leq b \Rightarrow c$, with $c = \bigcap_{k \in K} \lfloor C_k \rfloor$. Let $\Gamma \in a \wedge b$ and $k \in K$, we want to show $\Gamma \in \lfloor C_k \rfloor$, that is to say $\Gamma \vdash^* C_k$.
$\Gamma \in a$ so $\Gamma \in b \Rightarrow c$ and we have for any D, if $\bigcup \{e \in \Omega, b \wedge e \leq c\} \leq \lfloor D \rfloor$ then $\Gamma \in \lfloor D \rfloor$.
Let us show that $D \coloneqq \Gamma \Rightarrow C_k$ verifies this hypothesis, where $\Gamma \Rightarrow B \coloneqq A_1 \Rightarrow ... \Rightarrow A_n \Rightarrow B$ (with $\Gamma = A_1, ..., A_n$ and B formula).
Let $e \in \Omega$ with $b \wedge e \leq c$. Let $\Delta \in e$, $\{\Delta, \Gamma\} \in b \wedge e$ by Lem. 1, then $\Delta, \Gamma \in c$, and $\Delta, \Gamma \vdash^* C_k$.
By $\Rightarrow_I$, $\Delta \vdash^* \Gamma \Rightarrow C_k$, that is to say $\Delta \vdash^* D$. This holds for any such Δ, so $e \leq \lfloor D \rfloor$, and D verifies the desired hypothesis.
Therefore $\Gamma \vdash^* \Gamma \Rightarrow C_k$ and by repeated application of Lem. 3 and Lem. 1 $\Gamma \vdash^* C_k$.
Finally, $\Gamma \in c$.
– Conversely, assume $a \wedge b \leq c$, then $a \leq \bigvee \{e, e \wedge b \leq c\} = b \Rightarrow c$. □

Definition 12 (Interpretation in the Context Algebra). *The domain $\mathcal{D}$ of the model on Ω is defined as the set of terms. If f is a function symbol of arity n, P is a predicate symbol of arity n, we let:*

$$[\![f]\!] \coloneqq (t_1, ..., t_n) \mapsto f(t_1, ..., t_n)$$
$$[\![P]\!] \coloneqq (t_1, ..., t_n) \mapsto \lfloor P(t_1, ..., t_n) \rfloor$$

A consequence of this lemma is the following, where we implicitly coerce valuations with their underlying substitution.

Lemma 8. *For any t and valuation φ, $[\![t]\!]_\varphi = \varphi(t)$.*

Proof. By induction. □

Definition 13 (Closure). *Let A be a formula. We define the closure of A to be*

$$cl(A) \coloneqq \bigcap \{d \in \Omega, [A] \in d\}$$

Remind that $[A]$ is the one-formula context, containing only A (Def. 4).

Lemma 9. *For any A, $cl(A) \in \Omega$.*

Proof. Ω is stable by arbitrary intersection. □

Lemma 10. $[A] \in cl(A)$

Proof. $cl(A)$ is the greatest lower bound of all d containing $[A]$. □

Lemma 11. *For any A, $cl(A) = \bigcap \{\lfloor D \rfloor, [A] \in \lfloor D \rfloor\}$.*

Proof. Similar to the proof of Lem. 5. □

Then $\Gamma \in cl(A)$ means for all formulas D, if $[A] \vdash^* D$ then $\Gamma \vdash^* D$. In a sense, the members of $cl(A)$ verify the axiom replacement lemma, except that this new operation does not necessarily preserve the structure of the derivation. $\Gamma \in cl(A)$ is a weaker statement than $\Gamma \vdash_{ne} A$:

Lemma 12. *For Γ context and A formula, if $\Gamma \vdash_{ne} A$ then $\Gamma \in cl(A)$.*

Proof. By Lem. 2, considering the previous Lem. 11. □

Theorem 2 (Key theorem). *For any formula A and valuation σ into Ω, σ is also a substitution and*

$$cl(\sigma(A)) \leq [\![A]\!]_\sigma \leq \lfloor \sigma(A) \rfloor$$

Proof. For clarity, we omit the valuation/substitution σ when it plays no role. The proof is done by induction on A:

- A is atomic: $[\![A]\!] = \lfloor A \rfloor$, so we only need to check $cl(A) \leq \lfloor A \rfloor$. Let $\Gamma \in cl(A)$. as we have $A \vdash^* A$, by definition of $cl(A)$, we have $\Gamma \vdash^* A$ and therefore $\Gamma \in \lfloor A \rfloor$.
- $cl(A \wedge B) \leq [\![A \wedge B]\!]$: by induction hypothesis we only need to show $cl(A \wedge B) \leq cl(A) \cap cl(B)$.
  Let $\Gamma \in cl(A \wedge B)$ and D such that $A \vdash^* D$ (resp. $B \vdash^* D$). Since $A \wedge B \vdash_{ne} A$ (resp. $A \wedge B \vdash_{ne} B$), by Lem. 2 we have $\Gamma \vdash^* D$ and $\Gamma \in cl(A)$ (resp. $\Gamma \in cl(B)$).

  $[\![A \wedge B]\!] \leq \lfloor A \wedge B \rfloor$: by the induction hypothesis we have $[\![A \wedge B]\!] \leq \lfloor A \rfloor \cap \lfloor B \rfloor$. The $\wedge_I$ rule concludes the proof.
- $cl(A \vee B) \leq [\![A \vee B]\!]$: consider C such that $[\![A]\!] \cup [\![B]\!] \leq \lfloor C \rfloor$. We have to show $[A \vee B] \in \lfloor C \rfloor$.
  Since, by Lem. 10 and induction hypothesis, $[A] \in cl(A) \leq \lfloor C \rfloor$ (resp. $[B] \in cl(B) \leq \lfloor C \rfloor$), we have $A \vdash^* C$ (resp. $B \vdash^* C$). Then by $\vee_E$ and *coerce* we have $A \vee B \vdash^* C$.

  $[\![A \vee B]\!] \leq \lfloor A \vee B \rfloor$: by definition of $[\![A]\!] \vee [\![B]\!]$, we need to show that $[\![A]\!] \cup [\![B]\!] \leq \lfloor A \vee B \rfloor$.
  By induction hypothesis, $[\![A]\!] \cup [\![B]\!] \leq \lfloor A \rfloor \cup \lfloor B \rfloor$, then the $\vee_I$ rule concludes.
- $cl(A \Rightarrow B) \leq [\![A \Rightarrow B]\!]$: by the implication rule we need $cl(A \Rightarrow B) \wedge [\![A]\!] \leq [\![B]\!]$, and by induction hypothesis, it is sufficient to show

$cl(A \Rightarrow B) \wedge \lfloor A \rfloor \leq cl(B)$.
Let $\Gamma \in cl(A \Rightarrow B) \wedge \lfloor A \rfloor$, that is to say:

$$\Gamma \vdash^* A \text{ and for any } C, \text{ if } A \Rightarrow B \vdash^* C \text{ then } \Gamma \vdash^* C$$

Let D such that $B \vdash^* D$. To show $\Gamma \vdash^* D$, we first show that $\Gamma \vdash^* \Gamma \Rightarrow D$. We have, by hypothesis and Lem. 1, the following proof:

$$\frac{\overline{A \Rightarrow B, \Gamma \vdash_{ne} A \Rightarrow B} \qquad A \Rightarrow B, \Gamma \vdash^* A}{A \Rightarrow B, \Gamma \vdash_{ne} B}$$

So, by Lem. 2, $A \Rightarrow B, \Gamma \vdash^* D$, and by repeated $\Rightarrow_I$, $A \Rightarrow B \vdash^* \Gamma \Rightarrow D$. By hypothesis on Γ, $\Gamma \vdash^* \Gamma \Rightarrow D$. By a repeated application of Lem. 3 and Lem. 1, we get $\Gamma \vdash^* D$.

$[\![A \Rightarrow B]\!] \leq \lfloor A \Rightarrow B \rfloor$: by induction hypothesis, $[\![B]\!] \leq \lfloor B \rfloor$, so $[\![A]\!] \Rightarrow [\![B]\!] \leq [\![A]\!] \Rightarrow \lfloor B \rfloor$ by the intersection (with $[\![A]\!]$) and the implication properties. By induction hypothesis also, $cl(A) \leq [\![A]\!]$, and therefore $cl(A) \wedge ([\![A]\!] \Rightarrow \lfloor B \rfloor) \leq [\![A]\!] \wedge ([\![A]\!] \Rightarrow \lfloor B \rfloor) \leq \lfloor B \rfloor$, that is to say $[\![A]\!] \Rightarrow \lfloor B \rfloor \leq cl(A) \Rightarrow \lfloor B \rfloor$.
All in all, $[\![A \Rightarrow B]\!] \leq cl(A) \Rightarrow \lfloor B \rfloor$, and showing $cl(A) \Rightarrow \lfloor B \rfloor \leq \lfloor A \Rightarrow B \rfloor$ suffices.
Let c such that $cl(A) \wedge c \leq \lfloor B \rfloor$, we show that $\lfloor A \Rightarrow B \rfloor$ is an upper bound for c, so let $\Gamma \in c$. By Lem. 1 $A, \Gamma \in cl(A) \wedge c$, and $A, \Gamma \vdash^* B$, so by $\Rightarrow_I$, $\Gamma \vdash^* A \Rightarrow B$. This holds for any c, so $cl(A) \Rightarrow \lfloor B \rfloor \leq \lfloor A \Rightarrow B \rfloor$.

- $\top$ and $\bot$ are trivial cases.
- $cl(\sigma(\forall x.A)) \leq [\![\forall x.A]\!]_\sigma$:
  Without loss of generality, we assume $\sigma(\forall x.A) = \forall x.\sigma(A)$ (see Def. 2).
  Let $\Gamma \in cl(\sigma(\forall x.A))$.
  We need to prove that for any term d, $\Gamma \in [\![A]\!]_{\sigma[x \mapsto d]}$. Let d a term, showing $\Gamma \in cl(\sigma[x \mapsto d](A))$ suffices by induction hypothesis.
  Let D such that $\sigma[x \mapsto d](A) \vdash^* D$. As x does not appear in the image of σ, $\sigma[x \mapsto d](A) = (\sigma(A))[d/x]$, and we have:

$$\frac{\overline{[\forall x.\sigma(A)] \vdash_{ne} \forall x.\sigma(A)}\, ax}{[\forall x.\sigma(A)] \vdash_{ne} \sigma[x \mapsto d](A)}\, \forall_E$$

  Then by Lem. 2, $[\forall x.\sigma(A)] \vdash^* D$. As we assumed $\Gamma \in cl(\forall x.\sigma(A))$, the claim follows.

  $[\![\forall x.A]\!]_\sigma \leq \lfloor \sigma(\forall x.A) \rfloor$:
  Let $\Gamma \in [\![\forall x.A]\!]_\sigma$, where by α-conversion we assume x fresh.

By Def. 9, $\Gamma \in [\![A]\!]_{\sigma[x\mapsto x]}$. By induction hypothesis we conclude $\Gamma \in \lfloor \sigma[x \mapsto x](A) \rfloor$.
Finally, by the $\forall_I$ rule $\Gamma \vdash^* \forall x.\sigma[x \mapsto x](A)$, and by freshness of x, $\Gamma \vdash^* \sigma(\forall x.A)$.

– $cl(\sigma(\exists x.A)) \leq [\![\exists x.A]\!]_\sigma$:
Let $\Gamma \in cl(\sigma(\exists x.A))$, assuming x fresh. By Lem. 5, $\Gamma \in [\![\exists x.A]\!]_\sigma$ if and only if for any D, such that for each term d $[\![A]\!]_{\sigma[x\mapsto d]} \leq \lfloor D \rfloor$, then $\Gamma \vdash^* D$. Let such a D, we give a derivation of $[\exists x.\sigma(A)] \vdash^* D$, which allows to conclude by assumption on Γ.
By induction hypothesis, $[\sigma(A)] \in cl(\sigma(A)) \leq [\![A]\!]_\sigma$, and by hypothesis on D, $[\sigma(A)] \in \lfloor D \rfloor$. With Lem. 1, we get a derivation of the sequent $\exists x.\sigma(A), \sigma(A) \vdash^* D$. As $\exists x.\sigma(A) \vdash_{ne} \exists x.\sigma(A)$ has a neutral proof, we can build the desired derivation:

$$\frac{\exists x.\sigma(A), \sigma(A) \vdash^* D \qquad \exists x.\sigma(A) \vdash_{ne} \exists x.\sigma'(A)}{\exists x.\sigma(A) \vdash_{ne} D}\ \exists_E$$

$[\![\exists x.A]\!]_\sigma \leq \lfloor \sigma(\exists x.A) \rfloor$, assuming x fresh in the image of σ:
We show that $\lfloor \sigma(\exists x.A) \rfloor$ is an upper bound for all $[\![A]\!]_{\sigma[x\mapsto d]}$, where d is any term. This allows to conclude.
Let d, Γ, such that $\Gamma \in [\![A]\!]_{\sigma[x\mapsto d]}$. By induction hypothesis $\Gamma \in \lfloor \sigma[x \mapsto d](A) \rfloor$.
$\sigma[x \mapsto d](A) = (\sigma(A))[d/x]$, so $\Gamma \vdash^* (\sigma(A))[d/x]$ and by the $\Rightarrow_I$ rule, $\Gamma \vdash^* \exists x.\sigma(A)$, i.e. $\Gamma \vdash^* \sigma(\exists x.A)$. □

Theorem 3 (Strong Completeness). *Let Γ be a context and A a formula. If for any Heyting algebra, any model and any valuation φ, $[\![\Gamma]\!]_\varphi \leq [\![A]\!]_\varphi$, then $\Gamma \vdash^* A$.*

Proof. We apply the hypothesis on the universal algebra of Def. 11, the interpretation of Def. 12 and the empty valuation/substitution.

Consider $C \in \Gamma$. $C \in cl(C)$ by Lem. 10. By Lem. 6 and Thm. 2, $\Gamma \in cl(C) \leq [\![C]\!]$. So $\Gamma \in [\![\Gamma]\!] \leq [\![A]\!]$. By Thm. 2, $[\![A]\!] \leq \lfloor A \rfloor$. Finally $\Gamma \vdash^* A$.

Theorem 4 (Cut Elimination). *Let Γ be a context and A a formula. If $\Gamma \vdash A$, then $\Gamma \vdash^* A$.*

Proof. By soundness (Thm. 1) and strong completeness (Thm. 3).

4 The algorithm in Practice

This work has been formalized in Coq for the propositional fragment, so as to focus on the core of the algorithm, without dealing with binders.

4.1 Formalization: the Algorithm

Ω contains *arbitrary intersections* of extractions. To define it, we need to range over index predicates for the formulas A_i, that have type form $\to$ $Prop$ and let Ω be $\{\{\Gamma : \text{ context}, \forall A, \mathcal{P}(A) \to \Gamma \vdash^* A\} \mid \mathcal{P} : \text{ form} \to Prop\}$. We cannot range over predicates of type form $\to Type$, because of the need for impredicativity.

As a consequence, the predicate $\Gamma \vdash^* A$ lives in $Prop$, which prevents us to extract a program due to proof irrelevance. Nevertheless, we can apply the theorem to a derivation and use *Eval compute* to observe the behavior of the algorithm. However, since formulas are processed by Thm. 2 which performs case analysis, computation stalls if the derivation involves formula variables.

To have both impredicativity and extraction, we considered using an impredicative Set type, but we were not able to extract a program due to internal limitations. As a last resort, we relaxed the universe constraint, deliberately making the system inconsistent, but gaining an impredicative $Type$ type and a (possibly unsound) algorithm.

Three difficulties obfuscate the investigation of the algorithm (see the proofs of Thm. 3 and Thm. 4):

- the $\Gamma \in cl(\Gamma)$ step involves a conjunction of formula closures, and calls technical lemmas. This step can be avoided by considering empty contexts, i.e. $\Gamma = []$ and $[\![\Gamma]\!] = \top$.
- the $[\![A]\!] \subseteq \lfloor A \rfloor$ and $cl(\Gamma) \subseteq [\![\Gamma]\!]$ steps, i.e. calling Thm. 2, the key theorem, that in many cases makes a very indirect use of the NJ rules, potentially appealing to inversion results (Lem. 3).
- the $[\![\Gamma]\!] \subseteq [\![A]\!]$ step, i.e. soundness of NJ with respect to Ω. It involves in particular the proof that Ω is a Heyting algebra, which is non-trivial especially for the $\Rightarrow$ operator, and then composes these properties somehow.

Simplifying those steps is necessary for a further analysis. For the time being, we are only able to investigate the behavior of the algorithm by observational means, applying it to specific derivations, as shown below.

4.2 Examples

Implication cut When applied to a simple implication cut, the algorithm does what expected.

Initial proof	Reduct
$\dfrac{\dfrac{\dfrac{}{A, A \vdash A}\,ax}{A \vdash A \Rightarrow A}\,\Rightarrow_I \quad \dfrac{}{A \vdash A}\,ax}{A \vdash A}\,\Rightarrow_E$	$\dfrac{}{A \vdash A}\,ax$

Disjunction cut A disjunction cut is also properly reduced:

Initial proof	Reduct
$\dfrac{\dfrac{\dfrac{}{A \vdash A}\,ax}{A \vdash A \vee A}\,\vee_{I_l} \quad \dfrac{\dfrac{}{A, A \vdash A}\,ax}{A, A \vdash A \vee A}\,\vee_{I_r} \quad \dfrac{\dfrac{}{A, A \vdash A}\,ax}{A, A \vdash A \vee A}\,\vee_{I_l}}{A \vdash A \vee A}\,\vee_E$	$\dfrac{\dfrac{}{A \vdash A}\,ax}{A \vdash A \vee A}\,\vee_{I_r}$

Eta expansion As the algorithm is type-directed, it is not neutral on elementary proofs, when formulas are not atomic:

Initial proof	Reduct
$\dfrac{}{A \Rightarrow B \vdash A \Rightarrow B}\,ax$	$\dfrac{\dfrac{\dfrac{}{A \Rightarrow B, A \vdash A \Rightarrow B}\,ax \quad \dfrac{}{A \Rightarrow B, A \vdash A}\,ax}{A \Rightarrow B, A \vdash B}\,\Rightarrow_E}{A \Rightarrow B \vdash A \Rightarrow B}\,\Rightarrow_I$
$\dfrac{}{A \wedge B \vdash A \wedge B}\,ax$	$\dfrac{\dfrac{\dfrac{}{A \wedge B \vdash A \wedge B}\,ax}{A \wedge B \vdash A}\,\wedge_{E_l} \quad \dfrac{\dfrac{}{A \wedge B \vdash A \wedge B}\,ax}{A \wedge B \vdash B}\,\wedge_{E_r}}{A \wedge B \vdash A \wedge B}\,\wedge_I$
$\dfrac{}{A \vee B \vdash A \vee B}\,ax$	$\dfrac{\dfrac{}{A \vee B \vdash A \vee B}\,ax \quad \dfrac{\dfrac{}{A \vee B, A \vdash A}\,ax}{A \vee B, A \vdash A \vee B}\,\vee_{I_l} \quad \dfrac{\dfrac{}{A \vee B, B \vdash B}\,ax}{A \vee B, B \vdash A \vee B}\,\vee_{I_r}}{A \vee B \vdash A \vee B}\,\vee_E$

When applying iteratively the theorem once again to those proofs, one can notice that we already have reached a fixed point in the $\Rightarrow$ and $\wedge$ cases, while we continue reducing in the disjunction case, where we can notice that the context is abstracted via a $\Rightarrow_I$ rule, that introduces a commutative cut:

$$\dfrac{\dfrac{\dfrac{}{A \vee B \vdash A \vee B}\,ax \quad \dfrac{\dfrac{\dfrac{}{A \vee B, A, A \vee B \vdash A}\,ax}{A \vee B, A, A \vee B \vdash A \vee B}\,\vee_{I_l}}{A \vee B, A \vdash (A \vee B) \Rightarrow (A \vee B)}\,\Rightarrow_I \quad \dfrac{\dfrac{\dfrac{}{A \vee B, B, A \vee B \vdash B}\,ax}{A \vee B, B, A \vee B \vdash A \vee B}\,\vee_{I_r}}{A \vee B, B \vdash (A \vee B) \Rightarrow (A \vee B)}\,\Rightarrow_I}{A \vee B \vdash (A \vee B) \Rightarrow (A \vee B)}\,\vee_E \quad \dfrac{}{A \vee B \vdash A \vee B}\,ax}{A \vee B \vdash A \vee B}\,\Rightarrow_E$$

5 Conclusion

Strong completeness with respect to Heyting algebras has a constructive proof. In this paper, we have applied this result to natural deduction, and formalized it in Coq, so as to produce an algorithm for proof normalization. This argument can also be lifted to classical logic, using Boolean algebras instead, although we would have to carefully choose a classical natural deduction calculus. Obviously, this also applies to sequent calculus, in an even more straightforward way.

Our algorithm can be studied by evaluating it on specific derivations and by *Print*ing the Coq function to review the generated code. However, simplifying Coq proofs, via more general inversion (Kleene) or weakening lemmas for instance, is still necessary for a more in-depth understanding. Moreover, we still have to show that the normal proof obtained is really a reduct of the original proof. This could be done by carrying the original proof along soundness and completeness, as a for of proof-relevant version of those theorems.

It would also be interesting to compare the algorithm that we obtain with the ones that come from completeness with respect to Kripke structure [9,6,8], and in particular the produced normal proofs. One of the interests of our methodology is that we deal with disjunction (sum types) without requiring any modification of the semantics.

Semantic transformations could help in the study of the relationship between both algorithms. In particular, turning a Heyting algebra into a Kripke structure is not purely constructive [16]. Applied to the particular universal Heyting algebra/Kripke structure, translations may also be more informative and constructive [8].

As for disjunction, we did not focus on commutative cuts, and more work is required in this direction. It theoretically possible, as we can always eliminate those cuts by translating back and forth natural deduction into sequent calculus, semantically normalizing there. But a direct study is much more preferable.

Strong completeness for higher-order logic is also within reach, which, besides giving a normalization algorithm for a powerful logic, would give another way of studying disjunction, through their higher-order encoding.

6 Acknowledgments

The authors would like to thanks the reviewers for their insightful and constructive comments and pointers. Unfortunately we lacked time to include them all.

References

1. Thorsten Altenkirch, Peter Dybjer, Martin Hofmann, and Phil Scott. Normalization by evaluation for typed lambda calculus with coproducts. In *16th Annual IEEE Symposium on Logic in Computer Science*, pages 303–310, 2001.
2. Ulrich Berger and Helmut Schwichtenberg. An inverse of the evaluation functional for typed λ–calculus. In R. Vemuri, editor, *Proceedings of the Sixth Annual IEEE Symposium on Logic in Computer Science*, pages 203–211. IEEE Computer Society Press, Los Alamitos, 1991.
3. Richard Bonichon and Olivier Hermant. On constructive cut admissibility in deduction modulo. In Thorsten Altenkirch and Conor McBride, editors, *TYPES for proofs and programs*, volume 4502 of *LNCS*, pages 33–47. Springer, 2006.
4. Catarina Coquand. From semantics to rules: A machine assisted analysis. In *CSL*, pages 91–105, 1993.
5. Olivier Danvy. Type-directed partial evaluation. In John Hatcliff, Torben Æ. Mogensen, and Peter Thiemann, editors, *Partial Evaluation - Practice and Theory, DIKU 1998 International Summer School, Copenhagen, Denmark, June 29 - July 10, 1998*, volume 1706 of *LNCS*, pages 367–411. Springer, 1998.
6. Hugo Herbelin and Gyesik Lee. Formalizing logical metatheory: Semantical cut-elimination using kripke models for first-order predicate logic. *http://formal.hknu.ac.kr/Kripke/*, 2014. [Online, accessed 2014-06-11].
7. Olivier Hermant and James Lipton. A constructive semantic approach to cut elimination in type theories with axioms. In Michael Kaminski and Simone Martini, editors, *CSL*, volume 5213 of *LNCS*, pages 169–183. Springer, 2008.
8. Danko Ilik. Continuation-passing Style Models Complete for Intuitionistic Logic. *Annals of Pure and Applied Logic*, May 2012.
9. Danko Ilik, Gyesik Lee, and Hugo Herbelin. Kripke Models for Classical Logic. *Annals of Pure and Applied Logic*, 161(11):1367–1378, August 2010.
10. Jean-Louis Krivine. Une preuve formelle et intuitionniste du théorème de complétude de la logique classique. *The Bulletin of Symbolic Logic*, 2:405–421, 1996.
11. Shoji Maehara. Lattice-valued representation of the cut-elimination theorem. *Tsukuba journal of mathematics*, 15(2):509–521, 1991.
12. Mitsuhiro Okada. *An Introduction to Linear Logic: Expressiveness and Phase Semantics*, volume Volume 2 of *MSJ Memoirs*, pages 255–295. The Mathematical Society of Japan, Tokyo, Japan, 1998.
13. Mitsuhiro Okada. Phase semantic cut-elimination and normalization proofs of first- and higher-order linear logic. *Theoretical Computer Science*, 227:333–396, 1999.
14. Helena Rasiowa and Roman Sikorski. *The mathematics of metamathematics*. PWN, Polish Scientific Publishers, Warszawa, 1963.
15. William W. Tait. A non constructive proof of gentzen's hauptsatz for second order logic. *Bulletin of the AMS*, 72:980–983, 1966.
16. Anne Sjerp Troelstra and Dirk van Dalen. *Constructivism in Mathematics, An Introduction*. North-Holland, 1988.
17. Wim Veldman. An intuitionistic completeness theorem for intuitionistic predicate logic. *Journal of Symbolic Logic*, 41:159–166, 1976.