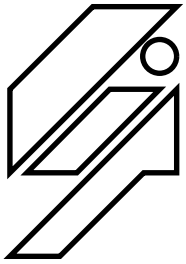
A decision procedure for Direct Predicate Calculus. Study and implementation in the system Coq.

Jean-Christophe Filliatre



Laboratoire de l'Informatique du Parallélisme

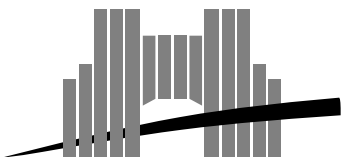
Ecole Normale Supérieure de Lyon
Unité de recherche associée au CNRS n°1398

A decision procedure for Direct Predicate Calculus Study and implementation in the system Coq

Jean-Christophe FILLIÂTRE

February 1995

Research Report N^{O} 96-25



Ecole Normale Supérieure de Lyon
46 Allée d'Italie, 69364 Lyon Cedex 07, France
Téléphone : (+33) 72.72.80.00 Télécopieur : (+33) 72.72.80.80
Adresse électronique : lip@lip.ens−lyon.fr

A decision procedure for Direct Predicate Calculus Study and implementation in the system Coq

Jean-Christophe FILLIÂTRE

February 1995

Abstract

The paper of J. Ketonen and R. Weyhrauch [6] defines a decidable fragment of first-order predicate logic — Direct Predicate Calculus — as the subset which is provable in Gentzen sequent calculus without the contraction rule, and gives an effective decision procedure for it. This report is a detailed study of this procedure. We extend the decidability to non-prenex formulas. We prove that the intuitionnistic fragment is still decidable, with a refinement of the same procedure. An intuitionnistic version has been implemented in the system Coq [2] using a translation into natural deduction.

Keywords: predicate calculus, sequent calculus, decision procedures, proof search, intuitionnistic logic.

Résumé

L'article de J. Ketonen et R. Weyhrauch [6] définit un fragment décidable du calcul des prédicats du premier ordre — le Calcul des Prédicats Direct — comme le sous-ensemble prouvable dans le calcul des séquents de Gentzen sans utiliser la règle de contraction, et en donne une procédure de décision effective. Ce rapport présente une étude détaillée de cette procédure. Nous étendons la décidabilité au cas des formules non nécessairement prénexes. Nous montrons que le fragment intuitionniste est également décidable, par un raffinement de la même procédure. Une version intuitionniste de cette algorithme a été implémentée dans le système Coq [2].

Mots-clés: calcul des prédicats, calcul des séquents, procédures de décision, recherche de preuves, logique intuitionniste.

A decision procedure for Direct Predicate Calculus Study and implementation in the system Coq

Jean-Christophe FILLIÂTRE *
LIP, URA CNRS 1398, ENS Lyon
46 Allée d'Italie, 69364 Lyon cedex 07, France
e-mail : jcfillia@lip.ens-lyon.fr

February 1995

Abstract

The paper of J. Ketonen and R. Weyhrauch [6] defines a decidable fragment of first-order predicate logic — Direct Predicate Calculus — as the subset which is provable in Gentzen sequent calculus without the contraction rule, and gives an effective decision procedure for it. This report is a detailed study of this procedure. We extend the decidability to non-prenex formulas. We prove that the intuitionnistic fragment is still decidable, with a refinement of the same procedure. An intuitionnistic version has been implemented in the system Coq [2] using a translation into natural deduction.

1 Introduction

First-order predicate logic is known to be undecidable. But some fragments are decidable, like propositional calculus, monadic predicates, or some classes of prenex formulas (Ackermann's class $\exists \ldots \exists \forall \exists \ldots \exists$, or Gödel's class $\exists \ldots \exists \forall \forall \exists \ldots \forall$ for instance). All those fragments are *syntactic* restrictions.

The paper of J. Ketonen and R. Weyhrauch [6] defines a decidable fragment of predicate logic, not in terms of syntactic restriction, but with a restriction on deduction rules. Indeed, Direct Predicate Calculus is defined as "the fragment of first-order predicate logic which is provable in Gentzen sequent calculus without the contraction rule".

Intuitively, it means that, for a given proof, each hypothesis (and each conclusion) can be used at most once during the proof. For instance, the hypothesis A is used once in a proof of

$$A \supset (A \supset B) \supset B$$

but necessarily twice in a proof of

$$A \supset ((A \supset B) \wedge (A \supset C)) \supset (B \wedge C)$$

and that's why the first formula is provable in Direct Predicate Calculus but not the second one.

In a more subtle way, it prevents proofs by case, like for instance the "drinkers' theorem"

$$\exists y. \forall x. (P(y) \supset P(x))$$

which is provable in Gentzen sequent calculus but not in Direct Predicate Calculus.

In [6], a decision procedure for Direct Predicate Calculus is explicitly given. It has been studied again in [1], which mentions a mistake in the original paper, carries out relations with linear logic and gives details about implementation of the decision procedure. The basic idea is simple: each atomic subformula

*This research was partly supported by ESPRIT Basic Research Action "Types" and by the GDR "Programmation" co-financed by MRE-PRC and CNRS.

can appear at most once in an axiom; therefore, we can see a derivation as the set of its axioms. The decision procedure consists of looking for such sets (called *paths*), which are finite and in finite number, then to construct derivation from paths. Quantification, in the case of prenex formulas, in handled through Herbrand functions and unification [1, 9].

The result is no longer true for non-prenex formulas. The skolemization does not assure the eigenvariable condition: it can now depend on the order of the quantifier rules, which was obvious and fixed in the prenex case. We extend the decision procedure to handle the case of non-prenex formulas; the construction of derivations from paths can now lead to a failure. We prove the completeness of this procedure. The ideas are closed to the framework presented in [9], but we do not perform proof search *bottom-up* exploiting the permutabilities of logical rules: we look for sets of axioms and re-construct *one particular proof* from these axioms. Thus the permutabilities of rules are still completely exploited.

At last, we are interested in Intuitionnistic Direct Predicate Calculus, that is Direct Predicate Calculus restricted to intuitionnistic sequents, or equivalently Gentzen intuitionnistic sequent calculus ($\mathcal{LJ}$) without contraction. Indeed, we want the decision procedure to be effective in Coq, which proof language is an intuitionnistic natural deduction. So we must know when a derivation corresponds to an intuitionnistic proof. We extend again the decision procedure to bring out intuitionnistic proofs, when they exist, and we prove its completeness with respect to intuitionnistic provability.

In section 2, we give notations and definitions. Then we present in section 3 the original main result and the decision procedure of [1, 6], but we give a slightly different proof. The extension we give for non-prenex formulas, and the case of intuitionnistic proofs is presented in section 4. Finally, we give details about implementation in the system Coq in section 5.

2 Direct Predicate Calculus

2.1 Notations and definitions

We assume the reader to be familiar with predicate calculus and sequent calculus. Our language is that of first-order predicate logic ($\mathcal{L}_0$): terms are built from variables and functions symbols applied to terms, formulas from atomic formulas applied to terms and the connectives $\neg, \supset, \wedge, \vee, \forall, \exists$, with the precedences $\supset < \vee < \wedge < \neg, \forall, \exists$. A sequent is a couple of sequences Γ and Δ of formulas, considered as multi-sets of formulas, and is written $\Gamma \vdash \Delta$.

In order to distinguish the occurrences of an atomic formula in a proof (or a formula, a sequent), for instance A in $A \supset (A \supset C) \supset D$, we extend the language with annotations on atomic formulas: A^i will denote an occurrence of A, where i is an integer. Therefore, $A^1 \supset (A^2 \supset C) \supset D$ represents the above formula, but in which we have syntactically distinguished the two occurrences of A. This language is denoted $\mathcal{L}$.

Definition 1 (separated formula) *A formula F of $\mathcal{L}$ is said to be* separated *if two occurrences of the same atomic formula of F are distinct, that is, annotated with different integers.*

If F is a formula, then $\pi(F)$ is the formula of $\mathcal{L}_0$ obtained by removing all annotations on atomic formulas. Two formulas F and G are called *similar* ($F \approx G$) if $\pi(F) = \pi(G)$, that is if they represent the same formula. From now on, we will assume that formulas and sequents are separated.

Definition 2 (occurrence) *We will write $u \prec t$ for "u occurs in t", u and t being terms or formulas.*

The notion of *positive* and *negative* occurrence is defined as usual: A formula A occurs positively in A, and if A occurs positively (resp. negatively) in B then A occurs positively (resp. negatively) in $C \supset B, B \wedge C, C \wedge B, B \vee C, C \vee B, \forall x.B, \exists x.B$ and negatively (resp. positively) in $\neg B, B \supset C$.

A *conjunctive subformula* is a positive occurrence of a conjunctive formula ($A \wedge B$) or a negative occurrence of a disjunctive formula ($A \vee B$, $A \supset B$), and a *disjunctive subformula* is a positive occurrence of a disjunctive formula or a negative occurrence of a conjunctive formula. In the following, we will sometimes write $A \circ B$ for a conjonctive or a disjunctive subformula, $\circ$ being one of the three connectives $\wedge$, $\vee$ or $\supset$.

A quantifier is called *essentially universal* if it is the outermost quantifier of a positive occurrence of $\forall x.A$ or a negative occurrence of $\exists x.A$, and *essentially existential* if it is the outermost quantifier of a positive occurrence of $\exists x.A$ or a negative occurrence of $\forall x.A$.

All those definitions are extended to sequents without any difficulty, interpreting $A_1, \ldots, A_n \vdash B_1, \ldots, B_m$ as the formula $A_1 \wedge \ldots \wedge A_n \supset B_1 \vee \ldots \vee B_m$.

2.2 Axioms and rules

Direct Predicate Calculus is the fragment of first-order predicate logic which can be proved in Gentzen sequent calculus ($\mathcal{LK}$) *without the contraction rules*:

$$\frac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}(\mathcal{L}-contract) \qquad \frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}(\mathcal{R}-contract)$$

Therefore, the rules for Direct Predicate Calculus are the following:

Axioms Axioms are

$$\frac{}{A(\vec{u}) \vdash B(\vec{u})}(Ax)$$

where A and B are two similar atomic formulas ($A \approx B$), and $\vec{u}$ a list of terms.

Structural rules Contraction being eliminated, and exchange rule being implicit, the only structural rules are weakening rules:

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}(\mathcal{L}-W) \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}(\mathcal{R}-W)$$

Logical rules Logical rules are exactly those of $\mathcal{LK}$. To each connective, $\supset, \wedge, \vee, \neg, \forall$, or $\exists$, correspond two introduction rules, one on the left side of the sequent, the other one on the right side:

$$\frac{\Gamma \vdash A, \Delta}{\neg A, \Gamma \vdash \Delta}(\mathcal{L}-\neg) \qquad \frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \neg A, \Delta}(\mathcal{R}-\neg)$$

$$\frac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}(\mathcal{L}-\wedge) \qquad \frac{\Gamma_1 \vdash \Delta_1, A \qquad \Gamma_2 \vdash \Delta_2, B}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, A \wedge B}(\mathcal{R}-\wedge)$$

$$\frac{A, \Gamma_1 \vdash \Delta_1 \qquad B, \Gamma_2 \vdash \Delta_2}{A \vee B, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}(\mathcal{L}-\vee) \qquad \frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B}(\mathcal{R}-\vee)$$

$$\frac{\Gamma_1 \vdash \Delta_1, A \qquad B, \Gamma_2 \vdash \Delta_2}{A \supset B, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}(\mathcal{L}-\supset) \qquad \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \supset B}(\mathcal{R}-\supset)$$

$$\frac{A(t), \Gamma \vdash \Delta}{\forall x.A(x), \Gamma \vdash \Delta}(\mathcal{L}-\forall) \qquad \frac{\Gamma \vdash \Delta, A(a)}{\Gamma \vdash \Delta, \forall x.A(x)}(\mathcal{R}-\forall)$$

$$\frac{A(a), \Gamma \vdash \Delta}{\exists x.A(x), \Gamma \vdash \Delta}(\mathcal{L}-\exists) \qquad \frac{\Gamma \vdash \Delta, A(t)}{\Gamma \vdash \Delta, \exists x.A(x)}(\mathcal{R}-\exists)$$

where $\Gamma, \Delta, \Gamma_1, \Gamma_2, \Delta_1, \Delta_2$ are formulas sequences, A, B are formulas, a is a variable which does not appear in $\Gamma \cup \Delta$, and t a term (called *witness* of the existential variable x).

In each previous rule, the formulas A and B are called *active formulas* and the formula appearing in the conclusion ($A \wedge B, A \vee B, \ldots$) is called the *principal formula* of the rule. Notice that positivity is preserved

by any rule, that is a positive (resp. negative) formula of the conclusion is also positive (resp. negative) in the premises and conversely.

One can notice here that rules for $\wedge, \vee$ and $\supset$ are given in their *multiplicative* way, that is formulas of the conclusion are split into the two premises, and not in their *additive* way where they would be duplicated in the two premises, like in the rule

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}(\mathcal{R}-\wedge)$$

This is consistent with the elimination of contraction.

The notion of *derivation* is defined as usual, from the rules above, and a formula F_0 of $\mathcal{L}_0$ is *provable in Direct Predicate Calculus* if there exists a derivation of $\vdash F$, where F is a formula of $\mathcal{L}$ such that $\pi(F) = F_0$.

2.3 Examples

We recall that $\mathcal{LK}$ denotes the first-order classical Gentzen sequent calculus (see for instance [3], page 44).

As a first example, let us consider the two formulas

$$A \supset (A \wedge (B \vee \neg B)) \quad \text{and} \quad A \supset ((A \wedge B) \vee (A \wedge \neg B))$$

They represent the same proposition, are both provable in $\mathcal{LK}$, but in the second one the $\wedge$ has been distributed on the $\vee$. Only the first one is provable is DPC:

Example 2.1 *$A \supset A \wedge (B \vee \neg B)$ is provable in DPC.*

Proof:

$$\dfrac{\dfrac{\dfrac{}{A \vdash A}(Ax) \qquad \dfrac{\dfrac{\dfrac{}{B \vdash B}(Ax)}{\vdash B, \neg B}(\mathcal{R}-\neg)}{\vdash B \vee \neg B}(\mathcal{R}-\vee)}{A \vdash A \wedge (B \vee \neg B)}(\mathcal{R}-\wedge)}{\vdash A \supset A \wedge (B \vee \neg B)}(\mathcal{R}-\supset)$$

□

On the other hand,

Example 2.2 *$A \supset ((A \wedge B) \vee (A \wedge \neg B))$ is not provable in DPC.*

Intuitively, every derivation of the sequent $\vdash A \supset ((A \wedge B) \vee (A \wedge \neg B))$ must use the hypothesis A twice, and that is exactly what is forbidden by the elimination of the contraction rule. A proof of $A \supset ((A \wedge B) \vee (A \wedge \neg B))$ in $\mathcal{LK}$ would be:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A}(Ax) \qquad \dfrac{\dfrac{}{A \vdash A}(Ax) \qquad \dfrac{\dfrac{}{B \vdash B}(Ax)}{\vdash B, \neg B}(\mathcal{R}-\neg)}{A \vdash B, A \wedge \neg B}(\mathcal{R}-\wedge)}{A, A \vdash A \wedge B, A \wedge \neg B}(\mathcal{R}-\wedge)}{A \vdash A \wedge B, A \wedge \neg B}(\mathcal{L}-contract)}{A \vdash (A \wedge B) \vee (A \wedge \neg B)}(\mathcal{R}-\vee)}{\vdash A \supset ((A \wedge B) \vee (A \wedge \neg B))}(\mathcal{R}-\supset)$$

One can be convinced that $\vdash A \supset ((A \wedge B) \vee (A \wedge \neg B))$ is not provable in DPC by trying to apply in a systematic way all inference rules on the sequent (see forthcoming examples).

□

Another example of formula provable in $\mathcal{LK}$ but not in DPC is the well-known "drinkers' theorem", $\exists y.A(y) \supset \forall x.A(x)$, whose name came from the interpretation "There exists a person y such that if y drinks then everybody drink". We choose here the prenex version of this formula, that is $\exists y.\forall x.(A(y) \supset A(x))$.

Example 2.3 *$\exists y.\forall x.(A(y) \supset A(x))$ is not provable in DPC.*

Proof: The only rule that can be applied to the sequent $\vdash \exists y.\forall x.(A(y) \supset A(x))$ is $\mathcal{R}-\exists$, and the resulting sequent is then:

$$\frac{\vdash \forall x.(A(t) \supset A(x))}{\vdash \exists y.\forall x.(A(y) \supset A(x))}(\mathcal{R}-\exists)$$

where t is a term in which x does not appear. Again, the only rule that can be applied is $\mathcal{R}-\forall$, which leads to:

$$\frac{\dfrac{\vdash A(t) \supset A(x)}{\vdash \forall x.(A(t) \supset A(x))}(\mathcal{R}-\forall)}{\vdash \exists y.\forall x.(A(y) \supset A(x))}(\mathcal{R}-\exists)$$

Once again, only the rule $\mathcal{R}-\supset$ can be used, and leads to the sequent $A(t) \vdash A(x)$, clearly not provable in DPC, seen the above eigenvariable condition on t.

□

On the other hand, if T denotes the formula $\exists y.\forall x.(A(y) \supset A(x))$, $T \vee T$ is provable in DPC:

Example 2.4 *$\exists y.\forall x.(A(y) \supset A(x)) \ \vee \ \exists y'.\forall x'.(A(y') \supset A(x'))$ is provable in DPC.*

Proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A(x) \vdash A(x)}(Ax)}{A(x) \vdash A(x), A(x')}(\mathcal{R}-W)}{\vdash A(x), A(x) \supset A(x')}(\mathcal{R}-\supset)}{A(y) \vdash A(x), A(x) \supset A(x')}(\mathcal{L}-W)}{\vdash A(y) \supset A(x), A(x) \supset A(x')}(\mathcal{R}-\supset)}{\vdash A(y) \supset A(x), \forall x'.(A(x) \supset A(x'))}(\mathcal{R}-\forall)}{\vdash A(y) \supset A(x), \exists y'.\forall x'.(A(y') \supset A(x'))}(\mathcal{R}-\exists)}{\vdash \forall x.(A(y) \supset A(x)), \exists y'.\forall x'.(A(y') \supset A(x'))}(\mathcal{R}-\forall)}{\vdash \exists y.\forall x.(A(y) \supset A(x)), \exists y'.\forall x'.(A(y') \supset A(x'))}(\mathcal{R}-\exists)}{\vdash \exists y.\forall x.(A(y) \supset A(x)) \ \vee \ \exists y'.\forall x'.(A(y') \supset A(x'))}(\mathcal{R}-\vee)$$

□

It shows that $\exists y.\forall x.(A(y) \supset A(x))$ is provable in $\mathcal{LK}$ (by first applying $\mathcal{R}-$contract, then the above proof).

3 A decision procedure

The decision procedure is based on the search for axioms. Axioms of a proof of F are pairs of atomic formulas appearing positively and negatively in F. We define the notion of *path* which is a set of such pairs satisfying some conditions, and show how paths and proofs are in correspondence, and how proofs are built from paths. Then the decision procedure will consist in looking for paths, which appears to be clearly decidable.

3.1 Definitions

Let S be a propositional sequent. Let $\mathcal{P}$ be a set of pairs of atomic subformulas of S.

Definition 3 ($\mathcal{P}$ satisfies A) *We say that $\mathcal{P}$* satisfies *a formula A (in symbols $\mathcal{P} \mapsto A$) if there is a pair (P, P') in $\mathcal{P}$ such that either $P \prec A$ or $P' \prec A$.*

Definition 4 (A and B connected) *For $A, B \prec S$, we say that A and B are* connected *(in symbols $A \|_{\mathcal{P}} B$) if there is a pair (P, P') in $\mathcal{P}$ such that $P \prec A$ and $P' \prec B$ (or vice versa).*

Definition 5 (conjunctive cycle) *We say that $\mathcal{P}$ has a* conjunctive cycle *if there exist distinct conjunctive subformulas of S, namely $A_0 \circ B_0, \ldots, A_n \circ B_n$ ($n \geq 1$), such that*

$$\forall i \in \{0, \ldots, n\} \qquad B_i \| _{\mathcal{P}} A_{i+1},$$

indexes being considered modulo n.

Definition 6 (path) *We say that $\mathcal{P}$ is a* path *for S if it satisfies the following conditions:*

(a) *$\mathcal{P} \neq \emptyset$;*

(b) *Atomic formulas in $\mathcal{P}$ are all distinct;*

(c) *If $(P, P') \in \mathcal{P}$ then P appears positively in S and P' negatively in S;*

(d) *If $A \circ B$ is a conjunctive subformula of S, and if $\mathcal{P} \mapsto A \circ B$, then $\mathcal{P} \mapsto A$ and $\mathcal{P} \mapsto B$;*

(e) *If $A \circ B$ is a conjunctive subformula of S, and if $\mathcal{P} \mapsto A \circ B$, then A and B are* not *connected;*

(f) *There is no conjunctive cycle in $\mathcal{P}$.*

3.2 The main theorem

Let S be a propositional sequent.

Theorem 1 *For any substitution σ, $S[\sigma]$ is provable in Direct Predicate Calculus if and only if there is a path $\mathcal{P}$ for S, minimal for inclusion, such that*

$$\forall (P, P') \in \mathcal{P} \qquad P[\sigma] \approx P'[\sigma] \tag{1}$$

Examples

- If $S = A^1 \supset (A^2 \supset B^1) \supset B^2$, then $\mathcal{P} = \{ (A^2, A^1), (B^2, B^1) \}$ is a path for S. It corresponds to the proof

$$\dfrac{\dfrac{\dfrac{\dfrac{}{A^1 \vdash A^2}(Ax) \quad \dfrac{}{B^1 \vdash B^2}(Ax)}{A^1, A^2 \supset B^1 \vdash B^2}(\mathcal{L}\text{-}\supset)}{A^1 \vdash (A^2 \supset B^1) \supset B^2}(\mathcal{R}\text{-}\supset)}{\vdash A^1 \supset (A^2 \supset B^1) \supset B^2}(\mathcal{R}\text{-}\supset)$$

  as we will show later.

- If S denotes the sequent

$$A^1(a) \supset (\forall x.(A^2(x) \supset B^1(x))) \supset \exists y.B^2(y)$$

  then its skolemized form $S_H(x, y)$ is

$$A^1(a) \supset (A^2(x) \supset B^1(x)) \supset B^2(y)$$

  and $\mathcal{P} = \{ (A^2(x), A^1(a)), (B^2(y), B^1(x)) \}$ is a path for $S_H(x, y)$ satisfying the condition (1) for the substitution $\sigma = \left[\begin{smallmatrix} x & y \\ a & a \end{smallmatrix}\right]$. It corresponds to the proof

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A^1(a) \vdash A^2(a)}(Ax) \quad \dfrac{}{B^1(a) \vdash B^2(a)}(Ax)}{A^1(a), A^2(a) \supset B^1(a) \vdash B^2(a)}(\mathcal{L}\text{-}\supset)}{A^1(a), \forall x.(A^2(x) \supset B^1(x)) \vdash B^2(a)}(\mathcal{L}\text{-}\forall)}{A^1(a), \forall x.(A^2(x) \supset B^1(x)) \vdash \exists y.B^2(y)}(\mathcal{R}\text{-}\exists)}{A^1(a) \vdash (\forall x.(A^2(x) \supset B^1(x))) \supset \exists y.B^2(y)}(\mathcal{R}\text{-}\supset)}{\vdash A^1(a) \supset (\forall x.(A^2(x) \supset B^1(x))) \supset \exists y.B^2(y)}(\mathcal{R}\text{-}\supset)$$

□

3.2.1 Proof of the theorem: only if part

The *only if* part is the most intuitive: the path corresponds exactly to the axioms of the derivation (proposition 2).

Definition 7 (path of a derivation) *The path of a derivation $\mathcal{D}$ (in symbols $\mathcal{P}(\mathcal{D})$) is defined as the set of the axioms of $\mathcal{D}$, that is*

$$\mathcal{P}(\mathcal{D}) = \{\, (A, B) \mid B \vdash A \text{ is an axiom of } \mathcal{D} \,\}$$

The notions of formulas satisfied and connected are extended to derivations through $\mathcal{P}(\mathcal{D})$.

Definition 8 (normal derivation) *A derivation $\mathcal{D}$ is said to be* normal *if it satisfies the following two conditions:*

- *if R is a rule of $\mathcal{D}$ with two premises, of active formulas A and B, then A and B are satisfied in $\mathcal{D}$;*
- *if R is a rule of $\mathcal{D}$ with one premise, which is not a weakening rule, then at least one of the active formula of R is satisfied in $\mathcal{D}$.*

Proposition 1 *If S is derivable in Direct Predicate Calculus, then there is a normal derivation of S in Direct Predicate Calculus.*

Proof: Let $\mathcal{D}$ be a derivation of S, and let us consider a weakening rule, for instance $\mathcal{R}-W$, in $\mathcal{D}$:

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma \vdash \Delta\end{array}}{\Gamma \vdash \Delta, A}(\mathcal{R}-W)}{\vdots}(\mathcal{X})$$

If A is not active in the rule $\mathcal{X}$ then we can exchange the application of $\mathcal{R}-W$ and $\mathcal{X}$. Let us assume this fact everywhere in the derivation $\mathcal{D}$. Then, only two cases can occur:

- The formula A is active in the inference $\mathcal{X}$. In that case, let us assume that $\mathcal{X}$ is a two premises rule, for instance $\mathcal{R}-\wedge$

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma_1 \vdash \Delta_1\end{array}}{\Gamma_1 \vdash \Delta_1, A}(\mathcal{R}-W) \qquad \begin{array}{c}\mathcal{D}''\\ \Gamma_2 \vdash \Delta_2, B\end{array}}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, A \wedge B}(\mathcal{R}-\wedge)$$

  Since $\mathcal{D}'$ is a derivation of $\Gamma_1 \vdash \Delta_1$, we can simplify $\mathcal{D}$ in this way

$$\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma_1 \vdash \Delta_1\end{array}}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, A \wedge B}(W^\star)$$

  where $W^\star$ represents a sequence of weakening rules.

  If $\mathcal{X}$ is a one premise rule, like for instance

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma \vdash \Delta\end{array}}{\Gamma \vdash \Delta, A}(\mathcal{R}-W)}{\neg A, \Gamma \vdash \Delta}(\mathcal{L}-\neg)}$$

  then we can simplify $\mathcal{D}$ in

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma \vdash \Delta\end{array}}{\neg A, \Gamma \vdash \Delta}(\mathcal{L}-W)}$$

- The rule $\mathcal{X}$ is also a weakening rule, on a formula B, and A and B are both active in the rule preceding $\mathcal{X}$, for instance $\mathcal{R} - \vee$

$$\dfrac{\dfrac{\dfrac{\mathcal{D}'}{\Gamma \vdash \Delta}(\mathcal{R}-W)}{\Gamma \vdash \Delta, A}(\mathcal{R}-W)}{\Gamma \vdash \Delta, A, B}(\mathcal{R}-\vee)$$

Then we can replace $\mathcal{D}$ by the simplified derivation

$$\dfrac{\dfrac{\mathcal{D}'}{\Gamma \vdash \Delta}(\mathcal{R}-W)}{\Gamma \vdash \Delta, A \vee B}(\mathcal{R}-W)$$

□

Proposition 2 *If $\mathcal{D}$ is a normal derivation of S then $\mathcal{P}(\mathcal{D})$ is a path for S, minimal for inclusion.*

Proof: Let $\mathcal{D}$ be a derivation of S. Let us show by induction on the length of $\mathcal{D}$ that $\mathcal{P}(\mathcal{D})$ is a path for S.

- If $\mathcal{D}$ is an axiom the result is clear.
- If the last rule of $\mathcal{D}$ has one premise

$$\dfrac{\dfrac{\mathcal{D}'}{S_1}(\mathcal{X})}{S}$$

then by induction hypothesis $\mathcal{P}(\mathcal{D}')$ is a path for S_1 (since unary rules do not generate new conjunctive formulas). But $\mathcal{P}(\mathcal{D}) = \mathcal{P}(\mathcal{D}')$ and, since $\mathcal{P}(\mathcal{D}')$ satisfies the conditions (a–f) then $\mathcal{P}(\mathcal{D})$ too.

- If the last rule of $\mathcal{D}$ has two premises, for instance

$$\dfrac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_2 \\ \Gamma_1 \vdash \Delta_1, A & \Gamma_2 \vdash \Delta_2, B\end{array}}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, A \wedge B}(\mathcal{R}-\wedge)$$

the induction hypothesis can be applied to $\mathcal{D}_1$ and $\mathcal{D}_2$, so $\mathcal{P}(\mathcal{D}_1)$ and $\mathcal{P}(\mathcal{D}_2)$ are paths for S_1 and S_2 respectively. Clearly, $\mathcal{P}(\mathcal{D}) = \mathcal{P}(\mathcal{D}_1) \cup \mathcal{P}(\mathcal{D}_2)$, and satisfies (a–c). Since $\mathcal{D}$ is assumed to be normal, then $\mathcal{P}(\mathcal{D})$ satisfies (d). $\mathcal{P}(\mathcal{D})$ is minimal because $\mathcal{P}(\mathcal{D}_1)$ and $\mathcal{P}(\mathcal{D}_2)$ are. Axioms of $\mathcal{P}(\mathcal{D}_1)$ and $\mathcal{P}(\mathcal{D}_2)$ are distinct, so $\mathcal{P}(\mathcal{D})$ satisfies (e).

It remains to verify the condition (f). Since $\mathcal{P}(\mathcal{D}_1)$ and $\mathcal{P}(\mathcal{D}_2)$ do not contain any conjunctive cycle by hypothesis, let us consider a conjunctive cycle of the form

$$F \circ C, \ldots, Y_i, \ldots, A \wedge B, \ldots, Z_i, \ldots, F \circ C$$

But A and B cannot be connected (condition (e)), so necessarily all Y_i must be subformulas of S_1 and all Z_i subformulas of S_2. But $F \circ C$ cannot belong to both S_1 and B_2. So $\mathcal{P}(\mathcal{D})$ satisfies (f).

□

Proof of the theorem (only if part): Let $\mathcal{D}$ be a proof of $S[\sigma]$. From Proposition 2, $\mathcal{P}(\mathcal{D})$ is a path for $S[\sigma]$, so also for S. And the condition (1) is clearly satisfied since axioms are of the form

$$\dfrac{}{A(\vec{u}) \vdash B(\vec{u})}(Ax)$$

with $(B, A) \in \mathcal{P}(\mathcal{D})$. □

3.2.2 Proof of the theorem: if part

The *if* part of the theorem is more subtle. Given a path for a sequent, we must re-construct a proof of this sequent. We prove it by induction on the sum of the sizes of the sequent and the path, and it gives us an algorithm for the decision procedure. The main difficulty appears when the sequent contains only conjunctive formulas, so that we have to choose one of them to apply the corresponding rule. We first establish some preliminaries to solve this critical point.

Preliminaries Let S be a sequent and $\mathcal{P}$ a path for S. Let $\mathcal{C}$ be the set of pairs (A_i, B_i) such that $A_i \circ B_i$ or $B_i \circ A_i$ is a conjunctive subformula of S. We define the oriented graph $G = (V, E)$ by

- $V = \mathcal{C}$
- $((A, B), (C, D)) \in E$ if and only if $\exists (P, P') \in \mathcal{P}$ with $P \prec B$ and $P' \prec C$, (*i.e.*) B and C are connected.

If $x = (A, B)$ is an vertex of G we denote $\bar{x}$ the vertex (B, A). We write $x \to y$ if $(x, y) \in E$. We denote $\stackrel{+}{\to}$ the transitive closure of $\to$, and $\stackrel{*}{\to}$ its reflexive transitive closure. If we have $x \stackrel{*}{\to} y$, we say that we have a *path* from x to y. If $c_0 \to c_1 \to \ldots \to c_n$ is a path, we say that it is *pure* if $i \neq j \implies c_i \neq c_j \wedge c_i \neq \bar{c_j}$. We write $x \stackrel{*}{\to}_p y$ in that case. When $y = x$ or $y = \bar{x}$ we still say that the path from x to y is pure if it is of the form $x \stackrel{*}{\to}_p z \to y$. Then we allow to write $x \stackrel{*}{\to}_p x$ or $x \stackrel{*}{\to}_p \bar{x}$.

We call *cycle* every pure path $x \stackrel{+}{\to}_p x$, and *loop* every pure path $x \stackrel{+}{\to}_p \bar{x}$.

The main property of this graph is the following:

Proposition 3 *$\mathcal{P}$ has a conjunctive cycle if and only if G has a cycle.*

Proof: Let

$$A_0 \circ B_0 - A_1 \circ B_1 - \cdots - A_n \circ B_n$$

be a conjunctive cycle of $\mathcal{P}$. Then it is clear that $(A_0, B_0) \to (A_1, B_1) \to \cdots \to (A_n, B_n) \to (A_0, B_0)$ is a cycle of G.

On the opposite, if $(A_0, B_0) \to (A_1, B_1) \to \cdots \to (A_n, B_n) \to (A_0, B_0)$ is a cycle of G, then we have $i \neq j \implies A_i \circ B_i \neq A_j \circ B_j$, and so

$$A_0 \circ B_0 - A_1 \circ B_1 - \cdots - A_n \circ B_n$$

is a conjunctive cycle of $\mathcal{P}$. □

Remarks

1. If $(x, y) \in E$ then $(\bar{y}, \bar{x}) \in E$, by definition of E. Consequently, if we have a path from u to v, then we have a path from $\bar{v}$ to $\bar{u}$ (more exactly, if we have the path $c_0 \to \cdots \to c_n$, we also have the path $\bar{c_n} \to \cdots \to \bar{c_0}$).

2. If the path $y \stackrel{+}{\to}_p z \stackrel{+}{\to}_p \bar{z}$ is not pure, then there exists a cycle in G.

   Indeed, if $y \stackrel{+}{\to}_p z \stackrel{+}{\to}_p \bar{z}$ is not pure, let us consider the smallest suffix of this path, $w \stackrel{+}{\to}_p z \stackrel{+}{\to}_p \bar{z}$, which is not pure. Two cases arise:

   - either $y \stackrel{+}{\to} w \stackrel{+}{\to} z \stackrel{+}{\to} w \stackrel{+}{\to} \bar{z}$, with $w \stackrel{+}{\to} z \stackrel{+}{\to} w$ pure, so we have a cycle;
   - or $y \stackrel{+}{\to} w \stackrel{+}{\to} z \stackrel{+}{\to} \bar{w} \stackrel{+}{\to} \bar{z}$, with $w \stackrel{+}{\to} z \stackrel{+}{\to} \bar{w}$ pure. We have $\bar{w} \stackrel{+}{\to}_p \bar{z}$ so by the previous remark we have $z \stackrel{+}{\to}_p w$, so $z \stackrel{+}{\to}_p w \stackrel{+}{\to}_p z$. If this path were not pure it would contradict the minimality of w. So we have a cycle $z \stackrel{+}{\to}_p z$.

We define on V the relation $\ll$ by

$$x \ll^1 y \stackrel{\text{def}}{\iff} \exists z\ x \to_p^\star z \to^+ y \to^+ \bar{z},\ \text{where}\ z \to^+ y \to^+ \bar{z}\ \text{is pure}$$

and

$$x \ll y \stackrel{\text{def}}{\iff} x \ll^1 y\ \wedge\ \neg(y \ll^1 x)$$

Lemma 1 *If G has no cycle then $\ll$ is a strict partial order.*

Proof: The relation $\ll$ is anti-reflexive by definition.

To show the transitivity of $\ll$, notice that is sufficient to show that

$$x \ll^1 y \wedge y \ll z \quad \Longrightarrow \quad x \ll^1 z$$

Indeed, assume that this fact is true. If $x \ll y$ and $y \ll z$, then clearly $x \ll^1 z$. If we had also $z \ll^1 x$, then, because $x \ll y$, we would have by the same result $z \ll^1 y$, which is not.

So assume that $x \ll^1 y$ and $y \ll z$, and let us show $x \ll^1 z$. We have the paths:

$$x \to_p^+ \underbrace{u \to^\star y \to^\star \bar{u}}_{\text{pure}} \qquad \text{and} \qquad y \to_p^+ \underbrace{v \to^\star z \to^\star \bar{v}}_{\text{pure}}$$

From remark 2 those two paths are necessarily pure. So we have

$$x \to_p^+ y \to_p^+ v \underbrace{\to^\star z \to^\star \bar{v}}_{\text{pure}}$$

and we aim at proving $x \ll^1 z$. Suppose $x \to_p^+ y \to_p^+ v$ is not pure, and let $w \to_p^+ y \to_p^+ v$ be the smallest suffix of this path which is not pure. Two cases arise:

- either we have $w \to_p^+ y \to_p^+ w \to_p^+ v$, and we have a cycle, contradiction;
- or we have $w \to_p^+ t \to_p^+ \bar{w} \to_p^+ v \to^\star z \to^\star \bar{v}$. we have $\bar{w} \to_p^+ v$, so $\bar{v} \to_p^+ w$ (Remark 1), so $z \to^\star \bar{v} \to_p^+ w$, which is a pure path because $\bar{w} \to_p^+ v \to^\star z \to^\star \bar{v}$ is. So we have

  $$z \to_p^+ w \to^+ y \to^+ \bar{w} \quad \text{with}\ w \to^+ y \to^+ \bar{w}\ \text{pure}$$

  that is $z \ll^1 y$, which cannot be, seen $y \ll z$.

In the end, we have the path

$$x \to_p^+ v \underbrace{\to^\star z \to^\star \bar{v}}_{\text{pure}}$$

that is $x \ll^1 z$. □

Lemma 2 *If G has no cycle, and if there is an infinite path*

$$x_0 \to x_1 \to \cdots \to x_n \to \cdots$$

with $x_{i+1} \neq \bar{x}_i$, then the relation $\ll$ has a minimal element.

Proof: Since G has no cycle, $\ll$ is a strict partial order, from the previous lemma. Therefore, seen that V is finite, $\ll$ has a minimal element if and only if $\ll$ is not empty.

Let $x_0 \to x_1 \to \cdots \to x_n \to \cdots$ be an infinite path in G, with $x_{i+1} \neq \bar{x}_i$. Let $x_0 \to \cdots \to x_k$ the longest pure prefix of this path. Two cases arise:

- either we have $x_0 \to^\star x_i = x_{k+1} \to_p^+ x_{k+1}$, then we have a cycle, which cannot be;

- or we have $x_0 \rightarrow^\star x_i = \overline{x_{k+1}} \rightarrow_p^+ x_{k+1}$, and then $x_0 \ll x_{i+1}$.

□

Proof of the theorem (if part): Let $\mathcal{P}$ be a path for $S[\sigma]$, for σ a substitution, such that the condition (1) is satisfied. Notice that $\mathcal{P}$ is also a path for S. The condition (1) only ensures that pairs in $\mathcal{P}$ could be considered as axioms in the following. So we won't mention σ anymore to clarify the proof.

Let us show by induction on the integer $t + w$ that $\mathcal{P}$ corresponds to a proof of S, where t is the size of S (the number of symbols) and w the number of atomic subformulas of S not satisfied in $\mathcal{P}$.

- If S has the shape $\Gamma, P' \vdash \Delta, P$ with $(P, P') \in \mathcal{P}$ then we can apply the rule *Axiom*, possibly preceded by weakenings.
- If S has the shape $\Gamma \vdash \Delta, F$ (or $\Gamma, F \vdash \Delta$) with F not satisfied in $\mathcal{P}$, then the induction hypothesis applies to $\Gamma \vdash \Delta$ and we use the weakening rule.
- If S contains a disjunctive formula, for instance $S = \Gamma \vdash \Delta, A \vee B$ then the induction hypothesis applies to $\Gamma \vdash \Delta, A, B$ for the same path $\mathcal{P}$, and we get a derivation of S by $\mathcal{R} - \vee$.

  Likewise if S contains a negation.
- Otherwise, the formulas of S can be split into two sets Π and Σ, where Π is a set of atoms (atomic formulas) and Σ a nonempty set of satisfied conjunctive formulas. Then we look for a conjunctive formulas X_0 of Σ (for instance $A_1 \wedge A_2$) on which we can apply the corresponding rule (here $\mathcal{R} - \wedge$), that is for which we can split S into two sequents containing A_1 and A_2, and find two paths for these sequents, to apply the induction hypothesis. We can distinguish two cases:

  - There exists a conjunctive formula X_0 of Σ, for instance $A_1 \wedge A_2$, such that A_1 is connected only to atoms (*i.e.*) if $(P, P') \in \mathcal{P}$ and $P \prec A_1$ (resp. $P' \prec A_1$) then $P' \in \Pi$ (resp. $P \in \Pi$).

    Then let S_1 and S_2 be the two sequents defined by $S_1 = A_1 \cup \Pi_1$ and $S_2 = (\Sigma \backslash \{X_0\}) \cup A_2 \cup \Pi_2$, where $\Pi_1 = \{ P \in \Pi \mid P \| _\mathcal{P} A_1 \}$ and $\Pi_2 = \Pi \backslash \Pi_1$. We restrict the path $\mathcal{P}$ to S_1 and S_2 by

    $$\mathcal{P}_i = \{ (P, P') \in \mathcal{P} \mid P \prec S_i \}$$

    Let us show that $\mathcal{P}_i$ is a path for S_i: $\mathcal{P}_i$ satisfies (a), because $\mathcal{P}$ satisfies (d) so $\mathcal{P} \mapsto A_1$ and $\mathcal{P} \mapsto A_2$. $\mathcal{P}_i$ satisfies (b–e) because $\mathcal{P}$ satisfies (b–e). $\mathcal{P}_i$ satisfies (f) because a conjunctive cycle of $\mathcal{P}_i$ would be a conjunctive cycle of $\mathcal{P}$. At last, $\mathcal{P}_i$ is minimal, because $\mathcal{P}$ is (if, for instance, $\mathcal{P}'_1 \subset \mathcal{P}_1$ would be a path for S_1 then $\mathcal{P}'_1 \cup \mathcal{P}_2$ would be a path for S, smaller than $\mathcal{P}$). So we can apply the induction hypothesis to $(S_1, \mathcal{P}_1)$ and $(S_2, \mathcal{P}_2)$: we get a derivation of S_1 and a derivation of S_2, and a derivation of S by $\mathcal{R} - \wedge$.
  - In the other case, every conjunctive formula of Σ is connected to another conjunctive formula of Σ.

    From the lemma 2 there exists an element X_0 of Σ minimal for $\ll$. Assume that $X_0 = A_1 \wedge A_2$, and let

    $$\Sigma_i = \{ Y \in \Sigma \backslash \{X_0\} \mid \text{there is a chain from } A_i \text{ to } Y \}$$

    Since there is no cycle and X_0 is minimal for $\ll$ we have $\Sigma_1 \cap \Sigma_2 = \emptyset$. Let $\Pi_i = \{ P \in \Pi \mid P \| _\mathcal{P} \Sigma_i \cup \{A_i\} \}$. $\mathcal{P}$ satisfies (b) so $\Pi_1 \cap \Pi_2 = \emptyset$.

    Suppose there exists C in Σ such that $C \notin \Sigma_i$, for $i = 1, 2$. Then let $\mathcal{P}_0 = \{ (P, P') \in \mathcal{P} \mid P \prec C \text{ or } P' \prec C \}$. S is satisfied so $\mathcal{P}_0 \neq \emptyset$. Moreover, $\mathcal{P}_0 \cap \mathcal{P}_i = \emptyset$ for $i = 1, 2$, otherwise C would be connected to A_1 or A_2. Then it's clear that $\mathcal{P} \backslash \mathcal{P}_0$ is a path for S, which contradict the minimality of $\mathcal{P}$.

    Likewise, suppose there exists $P \in \Pi$ such that $P \notin \Pi_i$ for $i = 1, 2$. The rule *Axiom* has not been applied, so P is connected to a conjunctive formula C of Σ. C cannot be X_0 by hypothesis. But, from the previous remark, such a formula C must belong to Σ_1 or Σ_2, which is a contradiction.

So we have, $\Sigma = \Sigma_1 \cup \Sigma_2 \cup \{X_0\}$ and $\Pi = \Pi_1 \cup \Pi_2$. Then let

$$S_i = \Pi_i \cup \Sigma_i \cup \{A_i\}$$

and

$$\mathcal{P}_i = \{\, (P, P') \in \mathcal{P} \mid P \text{ or } P' \text{ appears in } S_i \,\}$$

It's clear that $\mathcal{P}_i$ satisfies (b–f) since $\mathcal{P}$ satisfies these conditions. Moreover, $\mathcal{P}_i$ is minimal because $\mathcal{P}$ is. At last, $\mathcal{P}$ satisfies (d) and $\mathcal{P} \mapsto X_0$, so $\mathcal{P} \mapsto A_1$ and $\mathcal{P} \mapsto A_2$, and so $\mathcal{P}_i$ satisfies (a).

So we can apply the induction hypothesis to $(S_i, \mathcal{P}_i)$, and get a proof of S_1 and S_2, then a proof of S by $\mathcal{R} - \wedge$.

□

3.3 Skolemization

Definition 9 (Herbrand term) *Let F be a formula of $\mathcal{L}$ and $Qx.A$ an essentially universal subformula of F which lies in the scope of essentially existential quantifiers Q_1x_1,..., Q_nx_n of F. The* Herbrand term *associated with Qx is $f_x(x_1, \ldots, x_n)$ where f_x is a new symbol of function. f_x is called the* Herbrand function *associated with Qx.*

Definition 10 (Herbrand form) *Let F be a formula of $\mathcal{L}$. The* Herbrand form $F_H(x_1, \ldots, x_n)$ *of F is the result of erasing all quantifiers of F and replacing each essentially universal variable with the corresponding Herbrand term. Here $x_1, \ldots, x_n$ are all the essentially existential variables in F.*

In the following, we will consider free variables as (implicitly) universally quantified. It is the same as considering free variables as constants (since a free variable do not lie in the scope of any existential variable, so it is replaced by a new function symbol with no argument, that is a new constant symbol).

Examples

- If $F = \forall y.P(y) \supset \forall x.P(x)$, we have $F_H(y) = P(y) \supset P(f_x)$;
- If $F = \exists y.A(y) \vee B \supset \exists x.(A(x) \vee B)$, we have $F_H(x) = A(f_y) \vee B \supset A(x) \vee B$;
- If $F = \exists y.\forall x.(P(y) \supset P(x))$, we have $F_H(y) = P(y) \supset P(f_x(y))$;
- If $F = \forall x.\forall y.(A(x, y) \supset B(x)) \supset \exists t.(\forall z.A(u, z) \supset B(t))$,
  we have $F_H(x, t, z) = (A(x, f_y(x)) \supset B(x)) \supset (A(u, z) \supset B(t))$.

The interest of Herbrand form lies in the following result:

Theorem 2 (Skolem-Herbrand) *Let F be a* prenex *formula. Then F is provable in DPC if and only if there exist terms $t_1, \ldots, t_n$ such that $F_H(t_1, \ldots, t_n)$ is provable in DPC (more exactly in Direct Propositional Calculus, that is without the rules $\mathcal{R} - \forall$, $\mathcal{R} - \exists$, $\mathcal{L} - \forall$ and $\mathcal{L} - \exists$).*

Remark. Before giving the proof of this theorem let us notice that the result is no longer true for non-prenex formulas. For instance let F be the formula

$$\exists y.A(y) \vee B \supset \exists x.(A(x) \vee B)$$

Its Herbrand form is $F_H(x) = A(f_y) \vee B \supset A(x) \vee B$, and there exists a term $t = f_y$, such that $F_H(t)$ is provable in DPC.

But F is not provable in DPC. Indeed, the only rule that we can apply to the sequent $\vdash F$ is $\mathcal{R} - \supset$, and then we must prove the sequent $\exists y.A(y) \vee B \vdash \exists x.(A(x) \vee B)$. Two rules can be applied *a priori*:$\mathcal{R} - \exists$ and $\mathcal{L} - \vee$. The second one leads to a sequent of the form $\exists y.A(y) \vdash$ or $B \vdash$ which is not provable, so the

first one remains. So we look for a term t and a proof of $\exists y.A(y) \vee B \vdash A(t) \vee B$. Likewise, we must apply here $\mathcal{R}-\vee$, which leads to $\exists y.A(y) \vee B \vdash A(t), B$. At last, only the rule $\mathcal{L}-\vee$ can be applied and leads to $\exists y.A(y) \vdash A(t)$, which is not provable since t is a term which does not contain y (y is a bound variable when we introduce t with $\mathcal{R}-\exists$).

However, notice that F is provable in $\mathcal{LK}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A(y) \vdash A(y)}(Ax)}{A(y) \vdash A(y), B}(\mathcal{R}-W)}{A(y) \vdash A(y) \vee B}(\mathcal{R}-\vee)}{A(y) \vdash \exists x.(A(x) \vee B)}(\mathcal{R}-\exists)}{\exists y.A(y) \vdash \exists x.(A(x) \vee B)}(\mathcal{L}-\exists) \quad \dfrac{\dfrac{\dfrac{\dfrac{}{B \vdash B}(Ax)}{B \vdash A(x), B}(\mathcal{R}-W)}{B \vdash A(x) \vee B}(\mathcal{R}-\vee)}{B \vdash \exists x.(A(x) \vee B)}(\mathcal{R}-\exists)}{\exists y.A(y) \vee B \vdash \exists x.(A(x) \vee B), \exists x.(A(x) \vee B)}(\mathcal{L}-\vee)}{\exists y.A(y) \vee B \vdash \exists x.(A(x) \vee B)}(\mathcal{R}-contract)}{\vdash \exists y.A(y) \vee B \supset \exists x.(A(x) \vee B)}(\mathcal{R}-\supset)$$

□

Lemma 3 *Let F be a formula, σ a substitution and t a term. Let $(f_u)_{u \in \mathcal{U}}$ be function symbols appearing in F, and $(f'_u)_{u \in \mathcal{U}}$ new function symbols such that for all u of $\mathcal{U}$ we have $ar(f'_u) = ar(f_u) + 1$.*
$F[\sigma]$ is provable in DPC if and only if $F[f_u \leftarrow f'_u(t)][\sigma]$ is provable in DPC.

Proof: $F[f_u \leftarrow f'_u(t)][\sigma]$ is equal to $F[\sigma][f_u \leftarrow f'_u(\sigma(t))]$ so it's sufficient to show that if t is a term them F is provable if and only if $F[f_u \leftarrow f'_u(t)]$ is provable.

The proof is by induction on the length of the derivation, showing the more general result: for all sequent S, S is derivable if and only if $S[f_u \leftarrow f'_u(t)]$ is derivable. The reader will easily convinced himself of this result.

Notice that the derivation of $F[f_u \leftarrow f'_u(t)]$ is the derivation of F in which we apply everywhere the substitution $[f_u \leftarrow f'_u(t)]$. □

Lemma 4 *Let F be a formula, x a free variable in F, σ a substitution such that $\sigma(x) = x$ and f_x a function symbol which does not occur in F.*
$F[\sigma]$ is provable in DPC if and only if $F[x \leftarrow f_x][\sigma]$ is provable in DPC.

Proof: The proof is also by induction on the length of the derivation to show that for every sequent S such that x is free in S and f_x does not occur in S, $S[\sigma]$ is derivable if and only if $S[x \leftarrow f_x][\sigma]$ is.

The proof is trivial since we notice that the rules $\mathcal{R}-\forall, \mathcal{L}-\forall, \mathcal{R}-\exists$ and $\mathcal{L}-\exists$ cannot be applied on the variable x (because we assumed that x is free in S, and it's always possible to rename the bound variables of S with names other than x).

Notice that the derivation of $F[x \leftarrow f_x][\sigma]$ is the derivation of $F[\sigma]$ in which we apply everywhere the substitution $[x \leftarrow f_x]$. □

Proposition 4 *Let F be a prenex formula and σ a substitution.*
If $F_H[\sigma]$ is provable in DPC then $F[\sigma]$ is provable in DPC.

Proof: The proof is by induction on the number of quantifiers of F:

- If F has no quantifier then $F = F_H$, and the proposition is obvious.
- If $F = \exists x.F'$ then $F_H = F'_H[f_u \leftarrow f_u(x)]$ for every essentially universal variable u in F'. (We write f_u for the two function symbols, even if they are actually different symbols). By hypothesis $F_H[\sigma]$ (*i.e.*)

$F'_H[f_u \leftarrow f_u(x)][\sigma]$ is provable. So, from lemma 3, $F'_H[\sigma]$ is provable. But F' has one quantifier less than F, so the induction hypothesis applies to F', and $F'[\sigma]$ is provable (*i.e.*) we have a derivation

$$\frac{\mathcal{D}}{\vdash F'[\sigma]}$$

and so there is a derivation of $F[\sigma]$

$$\frac{\dfrac{\mathcal{D}}{\vdash F'[\sigma]}}{\vdash (\exists x.F')[\sigma]}(\mathcal{R}-\exists)$$

- If $F = \forall x.F'$ then $F_H = F'_H[x \leftarrow f_x]$.
  By hypothesis $F_H[\sigma]$ (*i.e.*) $F'_H[x \leftarrow f_x][\sigma]$ is provable so $F'_H[\sigma\backslash x][x \leftarrow f_x]$ is provable. So from lemma 4 $F'_H[\sigma\backslash x]$ is provable, and the induction hypothesis applies to F': $F'[\sigma\backslash x]$ is provable (*i.e.*) we have a derivation

  $$\frac{\mathcal{D}}{\vdash F'[\sigma\backslash x]}$$

  from which we get

  $$\frac{\dfrac{\mathcal{D}}{\vdash F'[\sigma\backslash x]}}{\vdash (\forall x.F')[\sigma]}(\mathcal{R}-\forall)$$

□

Proposition 5 *Let F be a prenex formula.*
If F is provable in DPC then there exist a substitution σ such that $F_H[\sigma]$ is provable in DPC.

Proof: The proof is by induction on the number of quantifiers of F:

- If F has no quantifier then $F = F_H$, and the result is obvious.
- If $F = \exists x.F'$ then $F_H = F'_H[f_u \leftarrow f_u(x)]$ for every essentially universal variable u in F'.
  By hypothesis F is provable so there exist a term t such that we have the derivation

  $$\frac{\dfrac{\mathcal{D}}{\vdash F'[x \leftarrow t]}}{\vdash \exists x.F'}(\mathcal{R}-\exists)$$

  $F'[x \leftarrow t]$ is derivable and by induction hypothesis there exist a substitution σ such that $(F'[x \leftarrow t])_H[\sigma]$ is provable. But $(F'[x \leftarrow t])_H = F'_H[x \leftarrow t]$ so $F'_H[x \leftarrow t][\sigma]$ is provable, and from lemma 3 $F'_H[f_u \leftarrow f_u(x)][x \leftarrow t][\sigma]$ is provable (*i.e.*) $F_H[\sigma']$ is provable, where σ' is the substitution defined by $\sigma'(x) = t$ and $\sigma'(y) = \sigma(y)$ if $y \neq x$.
- If $F = \forall x.F'$ then $F_H = F'_H[x \leftarrow f_x]$.
  By hypothesis F is provable so we have a derivation

  $$\frac{\dfrac{\mathcal{D}}{\vdash F'}}{\vdash \forall x.F'}(\mathcal{R}-\forall)$$

  Induction hypothesis applies to F' and so there exist a substitution σ such that $F'_H[\sigma]$ is provable. So from lemma 4, $F'_H[x \leftarrow f_x][\sigma]$ is provable (*i.e.*) $F_H[\sigma]$ is provable. □

Proof of the theorem: Let F be a prenex formula and $F_H(x_1, \ldots, x_n)$ its Herbrand form.

- If F is provable in DPC then, from proposition 5, there exist a substitution σ such that $F_H[\sigma]$ is provable (*i.e.*) $F_H(\sigma(x_1), \ldots, \sigma(x_n))$ is provable.
- Conversely, assume that there exist terms $t_1, \ldots, t_n$ such that $F_H(t_1, \ldots, t_n)$ is provable in DPC. Let σ be the substitution $(x_i \leftarrow t_i)_{i=1,\ldots,n}$. $F_H[\sigma]$ is provable so, from proposition 4, $F[\sigma]$ is provable, and $F[\sigma] = F$.

□

3.4 The decision procedure

Let $F = Q_1 x_1 . Q_2 x_2 \ldots Q_n x_n . G$ be a prenex formula, where G has no quantifier. We are going to apply the previous theorem to find derivations of F.

Let $F_H(x_{i_1}, \ldots, x_{i_p})$ be the Herbrand form of F. Let $\mathcal{A}^+$ (resp. $\mathcal{A}^-$) be the set of positive (resp. negative) atomic subformulas of F_H. First we consider the set

$$\mathcal{M} = \left\{ (P, P', u) \quad | \quad P \in \mathcal{A}^+ \wedge P' \in \mathcal{A}^- \wedge P[u] \approx P'[u] \right\}$$

where u is a principal solution of the unification problem $P \approx P'$. First-order unification is decidable, and has pseudo-linear solutions (see for instance [7, 10]).

Then we consider nonempty subsets $\mathcal{P}$ of $\mathcal{M}$ satisfying conditions (b) and (d), and such that the substitution

$$\sigma = \bigvee_{(P,P',u) \in \mathcal{P}} u$$

exists; therefore, conditions (a–d) and (1) are already satisfied. See [1] for an efficient algorithm to find such subsets.

Then we keep the subsets who also satisfied the last conditions (e) and (f). If $\mathcal{P}$ is such a subset (actually a path), and σ the above substitution, the proof of the theorem 1 gives a way to construct a derivation $\mathcal{D}$ of $F_H[\sigma]$. By replacing the Herbrand functions f_u by the corresponding variable u in σ we get a substitution σ', and by doing the same replacement in $\mathcal{D}$ we get a derivation $\mathcal{D}'$ of $G[\sigma']$. Then

$$\dfrac{\dfrac{\dfrac{\mathcal{D}'}{\vdash G[\sigma']}}{\vdash Q_n x_n . G[\sigma' \backslash x_n]}(\mathcal{R}-Q_n)}{\begin{array}{c}\vdots \\ \vdash Q_2 x_2 \ldots Q_n x_n . G[x_1 \leftarrow \sigma'(x_1)]\end{array}}$$
$$\vdash Q_1 x_1 \ldots Q_n x_n . G \quad (\mathcal{R}-Q_1)$$

is a derivation of F.

4 Extensions

In the case of prenex formulas the skolemization expresses the relative order of the quantifiers, so that unification respects the eigenvariable condition. The basic idea is the following: if F is the formula

$$\exists x . \forall y . P(x, y)$$

its Herbrand form is

$$P(x, f_y(x))$$

Then unification cannot lead to a term containing f_y to substitute to x (we would have an occur-check). As a consequence, the term substituted to x does not contain the variable y, which is the correct condition.

But, unfortunately, in the case of non-prenex formulas, the skolemization is not powerful enough. For instance, we saw that the formula

$$\exists y . A(y) \vee B \supset \exists x . (A(x) \vee B)$$

has the Herbrand form

$$F_H(x) = A(f_y) \vee B \supset A(x) \vee B$$

which is provable for the substitution $\sigma = \left[\begin{smallmatrix} x \\ f_y \end{smallmatrix}\right]$. But the dependency of $\sigma(x)$ over y (f_y) implies that $\mathcal{L}-\exists$ must be applied before $\mathcal{R}-\exists$ in the proof, which is not possible without contraction, as we have already seen.

So skolemization gives a necessary condition on σ, but not a sufficient one. The idea for extending the decision procedure in the case of non-prenex formulas is the following: we keep skolemization and the search for paths for the Herbrand form, satisfying the condition (1), and we try to reconstruct derivations from paths, like we did in the prenex case. But now, this step can lead to a failure: the path does *not* necessarily correspond to a derivation. But we keep the completeness of the method: if a formula is provable in DPC then there is at least one path for it on which the algorithm is successful.

Of course, such reconstructed proofs from paths have a certain shape, depending on the choices we made when applying the rules. But all the proofs we can construct from a given path are "all the same", in a sense we are going to define below.

Moreover, we can direct the construction to bring out intuitionnistic proofs when they exist. We prove the existence and the completeness of such a construction, that is: if a formula is provable in intuitionnistic DPC then there is at least one path for it on which the algorithm successfully returns an intuitionnistic proof of it.

4.1 Canonical proofs

Definition 11 (potential quantifier) *Let S be a sequent, and $Q_1x_1.F_1, \ldots, Q_mx_m.F_m$ its quantified subformulas, $x_1, \ldots, x_n$ being the essentially existential variables. Let $\mathcal{D}$ be a derivation of S, and σ the substitution of existential variables in $\mathcal{D}$. We define the relation $\sqsubset$ on $\{Q_i\}_{i=1,\ldots,m}$ with*

$$Q_i \sqsubset Q_j \quad \stackrel{def}{\Longleftrightarrow} \quad (\ (Q_jx_j.F_j \preceq F_i) \vee (j \in \{1,\ldots,n\} \wedge x_i \preceq \sigma(x_j))\ )$$

A quantified formula $Qx.F$ of S is said to be potential *if Q is minimal for $\sqsubset$.*

It just means that the corresponding rule can be applied on $Qx.F$, and we have the fact:

Proposition 6 *Let $\mathcal{D}$ be a derivation of S, and Q, Q' two quantifiers appearing in $\mathcal{D}$. If $Q \sqsubset Q'$ then Q is introduced* before *Q' in $\mathcal{D}$ (that means* below *in the bottom-up representation we chose in this paper).*

Definition 12 (potential conjunction) *Let S be a sequent, and $\mathcal{D}$ a derivation of S. A conjunctive formula $A \circ B$ of S is said to be* potential *if the corresponding rule can be applied on $A \circ B$, keeping the same path $\mathcal{P}(\mathcal{D})$ for the resulting proof. (That is, if S can be split into $S_1 \cup S_2 \cup A \circ B$, and $\mathcal{P}(\mathcal{D})$ into $\mathcal{P}_1$ and $\mathcal{P}_2$, such that $\mathcal{P}_1$ is a path for $S_1 \cup A$ and $\mathcal{P}_2$ for $S_2 \cup B$)*

We define the notion of *canonical derivation* by induction on a derivation.

Definition 13 (canonical proof) *A derivation $\mathcal{D}$ of a sequent $S = \Gamma \vdash \Delta$ is said to be* canonical *if*

- *either $\mathcal{D}$ is an axiom;*
- *or $\mathcal{D}$ is of the form*

$$\frac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_2 \\ S_1 & S_2\end{array}}{\Gamma \vdash \Delta}(R) \qquad \textit{or} \qquad \frac{\begin{array}{c}\mathcal{D}_1 \\ S_1\end{array}}{\Gamma \vdash \Delta}(R)$$

  with $\mathcal{D}_1$ and $\mathcal{D}_2$ canonical, and

  - *If a formula of S is not satisfied in $\mathcal{D}$ then R is a weakening rule;*
  - *or else, if S contains a potential quantification then R is a quantification rule;*
  - *or else, if Δ contains a negation then $R = \mathcal{R}-\neg$;*

– *or else, if S contains a disjunction then R is a disjunctive rule;*

– *or else, if Δ contains a potential conjunction then $R = \mathcal{R} - \wedge$;*

– *or else, if Γ contains a potential conjunction then R is a conjunctive rule;*

– *or else, if Γ contains a negation then $R = \mathcal{L} - \neg$.*

This choice may seem arbitrary, and we could have chosen another, but it will be justified by proposition 10.

Definition 14 (equivalent derivations) *Two derivations $\mathcal{D}_1$ and $\mathcal{D}_2$ are said* equivalent *if*

$$\mathcal{P}(\mathcal{D}_1) = \mathcal{P}(\mathcal{D}_2)$$

(i.e.) *if they have the same axioms.*

The main result is the following:

Proposition 7 *Every derivation $\mathcal{D}$ is equivalent to a canonical derivation $\bar{\mathcal{D}}$.*

Proof: The proof is by induction on the derivation $\mathcal{D}$ (of $S = \Gamma \vdash \Delta$). If $\mathcal{D}$ is an axiom, then the result is clear with $\bar{\mathcal{D}} = \mathcal{D}$. Otherwise, $\mathcal{D}$ is

$$\text{either} \quad \frac{\begin{array}{cc}\mathcal{D}_1 & \mathcal{D}_2 \\ S_1 & S_2\end{array}}{\Gamma \vdash \Delta}(R) \qquad \text{or} \qquad \frac{\begin{array}{c}\mathcal{D}_1 \\ S_1\end{array}}{\Gamma \vdash \Delta}(R)$$

Then we reason by case, following the definition of a canonical derivation:

(a) If a formula F of S is not satisfied in $\mathcal{D}$:

Lemma 5 *If F is not satisfied in a derivation $\mathcal{D}$ of S, F, then there is an equivalent derivation $\mathcal{D}'$ of S, smaller than $\mathcal{D}$.*

Proof: The proof is by induction on the length of $\mathcal{D}$. $\mathcal{D}$ cannot be an axiom. If the last rule of $\mathcal{D}$ is a weakening on F, then the result is clear. Otherwise, $\mathcal{D}$ has the form

$$\frac{\begin{array}{c}\mathcal{D}_1 \\ S', F\end{array}}{S, F}(R)$$

and by induction hypothesis, there is an equivalent derivation $\mathcal{D}'_1$ of S', smaller than $\mathcal{D}_1$, so there is an equivalent derivation of S by R, smaller than $\mathcal{D}$. □

From the lemma above, there is a derivation of $S \backslash F$ smaller than $\mathcal{D}$ and by induction hypothesis there is an equivalent canonical derivation of $S \backslash F$, and so, by a weakening on F, there is a canonical derivation of S equivalent to $\mathcal{D}$.

(b) or else, if S contains a potential quantification $Qx.F(x)$:

Lemma 6 *If $\mathcal{D}$ is a derivation of $\Gamma \vdash \Delta, \forall x.P(x)$ then the derivation $\mathcal{D}$ in which we have removed the rule $\mathcal{R} - \forall$ corresponding to $\forall x.P(x)$, and replaced every occurrence of $\forall x.P(x)$ by $P(x)$ is a derivation of $\Gamma \vdash \Delta, P(x)$. (Likewise for $\Gamma, \exists x.P(x) \vdash \Delta$).*

Proof: The proof is an easy induction on the derivation $\mathcal{D}$, as above. □

Lemma 7 *If $\mathcal{D}$ is a derivation of $\Gamma \vdash \Delta, \exists x.P(x)$, where $\exists x.P(x)$ is a potential quantifier, then the derivation $\mathcal{D}$ in which we have removed the rule $\mathcal{R}-\exists$ corresponding to $\exists x.P(x)$, and replaced every occurrence of $\exists x.P(x)$ by $P(t)$, where t is the substituted term for x, is a derivation of $\Gamma \vdash \Delta, P(t)$. (Likewise for $\Gamma, \forall x.P(x) \vdash \Delta$).*

Proof: The proof is an easy induction on the derivation $\mathcal{D}$, as above. □

Assume that S is of the form $\Gamma \vdash \Delta_1, Qx.F(x)$. Then, from one of the two previous lemmas, we have an equivalent derivation of $\Gamma \vdash \Delta, F(x)$ (or $\Gamma \vdash \Delta, F(t)$), smaller than $\mathcal{D}$; so, by induction hypothesis, we have an equivalent canonical derivation of $\Gamma \vdash \Delta, F(x)$ (or $\Gamma \vdash \Delta, F(t)$), from which we get an equivalent canonical derivation of S by the corresponding rule ($\mathcal{R}-\forall$ or $\mathcal{R}-\exists$ here).

(c) or else, if Δ contains a negation:

Lemma 8 *If $\mathcal{D}$ is a derivation of $\Gamma \vdash \Delta, \neg F$, in which $\neg F$ is satisfied, then there is an equivalent derivation of $\Gamma, F \vdash \Delta$, smaller than $\mathcal{D}$.*

Proof: The proof is an easy induction on the derivation $\mathcal{D}$, as above. □

Then, if S has the form $\Gamma \vdash \Delta_1, \neg F$, there is, from the previous lemma, an equivalent derivation of $\Gamma, F \vdash \Delta_1$, smaller than $\mathcal{D}$, so the induction hypothesis gives an equivalent canonical derivation of $\Gamma, F \vdash \Delta_1$. So we get an equivalent canonical derivation of S by $\mathcal{R}-\neg$.

(d) or else, if S contains a disjunction:

Lemma 9 *If $\mathcal{D}$ is a derivation of $\Gamma \vdash \Delta, A \vee B$, in which $A \vee B$ is satisfied, then there is an equivalent derivation of $\Gamma \vdash \Delta, A, B$, smaller than $\mathcal{D}$. (Likewise for $\Gamma \vdash \Delta, A \supset B$ and $\Gamma, A \wedge B \vdash \Delta$).*

Proof: The proof is an easy induction on the derivation $\mathcal{D}$, as above. □

Then, if S has the form $\Gamma \vdash \Delta_1, A \vee B$, there is, from the previous lemma, an equivalent derivation of $\Gamma \vdash \Delta_1, A, B$, smaller than $\mathcal{D}$, so the induction hypothesis gives an equivalent canonical derivation of $\Gamma \vdash \Delta_1, A, B$. So we get an equivalent canonical derivation of S by $\mathcal{R}-\vee$. (Likewise for $\Gamma \vdash \Delta, A \supset B$ and $\Gamma, A \wedge B \vdash \Delta$).

(e) or else, if Δ contains a potential conjunction:

Lemma 10 *If $\mathcal{D}$ is a derivation of $\Gamma \vdash \Delta, A \wedge B$, in which $A \wedge B$ is satisfied and potential, then there is a derivation $\mathcal{D}_1$ of $\Gamma_1 \vdash \Delta_1, A$ and a derivation $\mathcal{D}_2$ of $\Gamma_2 \vdash \Delta_2, B$, both smaller than $\mathcal{D}$, with $\Gamma = \Gamma_1 \cup \Gamma_2$, $\Delta = \Delta_1 \cup \Delta_2$ and $\mathcal{P}(\mathcal{D}) = \mathcal{P}(\mathcal{D}_1) \cup \mathcal{P}(\mathcal{D}_2)$.*

Proof: The proof is by induction on the size of $\mathcal{D}$. $\mathcal{D}$ cannot be an axiom. If the last rule of $\mathcal{D}$ is applied to $A \wedge B$, then the result is obvious. Two cases remain:

- The last rule of $\mathcal{D}$ has one premise, for instance

$$\dfrac{\begin{array}{c}\mathcal{D}'\\ \Gamma \vdash \Delta, F, A \wedge B\end{array}}{\neg F, \Gamma \vdash \Delta, A \wedge B}(\mathcal{L}-\neg)$$

  By induction hypothesis, there exist a derivation $\mathcal{D}_1$ of $\Gamma_1 \vdash \Delta_1, A$ and a derivation $\mathcal{D}_2$ of $\Gamma_2 \vdash \Delta_2, F, B$, both smaller than $\mathcal{D}'$, with $\Gamma = \Gamma_1 \cup \Gamma_2$, $\Delta = \Delta_1 \cup \Delta_2$ and $\mathcal{P}(\mathcal{D}_1) \cup \mathcal{P}(\mathcal{D}_2) = \mathcal{P}(\mathcal{D}') = \mathcal{P}(\mathcal{D})$. So we have the derivation

$$\dfrac{\begin{array}{cc}\mathcal{D}_1 & \begin{array}{c}\mathcal{D}_2\\ \Gamma_2 \vdash \Delta_2, F, B\end{array}\\ \Gamma_1 \vdash \Delta_1, A & \dfrac{}{\neg F, \Gamma_2 \vdash \Delta_2, B}(\mathcal{L}-\neg)\end{array}}{\neg F, \Gamma \vdash \Delta, A \wedge B}(\mathcal{R}-\wedge)$$

  where the derivation of $\neg F, \Gamma_2 \vdash \Delta_2, B$ is smaller than $\mathcal{D}$.

- The last rule of $\mathcal{D}$ has two premises, for instance

$$\dfrac{\begin{array}{cc}\mathcal{D}_3 & \mathcal{D}_4 \\ \Gamma_3 \vdash \Delta_3, A', A \wedge B & B', \Gamma_4 \vdash \Delta_4\end{array}}{A' \supset B', \Gamma_3, \Gamma_4 \vdash \Delta_3, \Delta_4, A \wedge B}(\mathcal{L}-\supset)$$

By induction hypothesis, there exist a derivation $\mathcal{D}_1$ of $\Gamma_1 \vdash \Delta_1, A$ and a derivation $\mathcal{D}_2$ of $\Gamma_2 \vdash \Delta_2, B, A'$, both smaller than $\mathcal{D}_3$, with $\Gamma_3 = \Gamma_1 \cup \Gamma_2$, $\Delta_3 = \Delta_1 \cup \Delta_2$ and $\mathcal{P}(\mathcal{D}_1) \cup \mathcal{P}(\mathcal{D}_2) = \mathcal{P}(\mathcal{D}_3)$. So we have the derivation

$$\dfrac{\begin{array}{cc}\mathcal{D}_1 & \dfrac{\begin{array}{cc}\mathcal{D}_2 & \mathcal{D}_4 \\ \Gamma_2 \vdash \Delta_2, B, A' & B', \Gamma_4 \vdash \Delta_4\end{array}}{A' \supset B', \Gamma_2, \Gamma_4 \vdash \Delta_2, \Delta_4, B}(\mathcal{L}-\supset) \\ \Gamma_1 \vdash \Delta_1, A & \end{array}}{A' \supset B', \Gamma_3, \Gamma_4 \vdash \Delta_3, \Delta_4, A \wedge B}(\mathcal{R}-\wedge)$$

where the derivation of $A' \supset B', \Gamma_2, \Gamma_4 \vdash \Delta_2, \Delta_4, B$ is smaller than $\mathcal{D}$ because $\mathcal{D}_2$ is smaller than $\mathcal{D}_3$.

□

Then, if S has the form $\Gamma \vdash \Delta, A \wedge B$, then, from the previous lemma, there exist derivations $\mathcal{D}_1$ and $\mathcal{D}_2$ of $\Gamma_1 \vdash \Delta_1, A$ and $\Gamma_2 \vdash \Delta_2, B$, both smaller than $\mathcal{D}$, with $\Gamma = \Gamma_1 \cup \Gamma_2$, $\Delta = \Delta_1 \cup \Delta_2$ and $\mathcal{P}(\mathcal{D}) = \mathcal{P}(\mathcal{D}_1) \cup \mathcal{P}(\mathcal{D}_2)$. By induction hypothesis there exist equivalent canonical derivations $\bar{\mathcal{D}_1}$ and $\bar{\mathcal{D}_2}$, so we have an equivalent canonical derivation of S by $\mathcal{R} - \wedge$.

(f) or else, if Γ contains a potential conjunction:

The proof is similar to the previous case.

(g) or else, if Γ contains a negation:

The proof is similar to the case (c).

□

4.2 Application

The idea is to build canonical proofs from paths. For this purpose, we refine the algorithm given in 3.2.2: let $\mathcal{P}$ be a path for S_H, and σ a substitution, such that the condition (1) is satisfied. We construct a proof of S by induction on the size of S:

- If there is an axiom in S, then we apply the rule *Axiom*, possibly preceded by a sequence of weakening rules;
- or else, if a formula of S is not satisfied, then we apply the corresponding weakening rule;
- or else, if there is a potential quantifier in S we apply the corresponding rule;
- or else, if there is a negation in Δ we apply the corresponding rule;
- or else, if there is a disjunction in S we apply the corresponding rule;
- or else, if there is potential conjunction in Δ we apply the corresponding rule;
- or else, if there is potential conjunction in Γ we apply the corresponding rule;
- or else, if there is negation in Γ we apply the corresponding rule;
- or else, we return a FAILURE.

We denote $\mathcal{A}(\mathcal{P})$ the resulting proof, if it exists. Clearly, we have the following results:

Proposition 8

(1) *If $\mathcal{A}(\mathcal{P})$ exists, then it is a canonical proof.*

(2) *$\mathcal{A}(\mathcal{P}(\mathcal{D}))$ exists, and is equivalent to $\mathcal{D}$.*

As a consequence of the proposition 7, we obtain the completeness of our algorithm:

Theorem 3 (Completeness) *If S is provable in DPC then there exist a path $\mathcal{P}$ for S (and a substitution σ) such that $\mathcal{A}(\mathcal{P})$ is defined and is a proof of S.*

4.3 Intuitionnistic proofs

The corresponding system in intuitionnistic logic, that we can call Intuitionnistic Direct Predicate Calculus, is obtained as usual by considering only intuitionnistic sequents that is of the form $\Gamma \vdash \Delta$ where Δ contains *at most one formula*. The rules are slightly modified:

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}(\mathcal{L}-W) \qquad \frac{\Gamma \vdash}{\Gamma \vdash A}(\mathcal{R}-W)$$

$$\frac{\Gamma \vdash A}{\neg A, \Gamma \vdash}(\mathcal{L}-\neg) \qquad \frac{A, \Gamma \vdash}{\Gamma \vdash \neg A}(\mathcal{R}-\neg)$$

$$\frac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}(\mathcal{L}-\wedge) \qquad \frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{\Gamma_1, \Gamma_2 \vdash A \wedge B}(\mathcal{R}-\wedge)$$

$$\frac{A, \Gamma_1 \vdash \Delta_1 \quad B, \Gamma_2 \vdash \Delta_2}{A \vee B, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}(\mathcal{L}-\vee) \qquad \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee_1) \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee_2)$$

$$\frac{\Gamma_1 \vdash A \quad B, \Gamma_2 \vdash \Delta}{A \supset B, \Gamma_1, \Gamma_2 \vdash \Delta}(\mathcal{L}-\supset) \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \supset B}(\mathcal{R}-\supset)$$

$$\frac{A(t), \Gamma \vdash \Delta}{\forall x.A(x), \Gamma \vdash \Delta}(\mathcal{L}-\forall) \qquad \frac{\Gamma \vdash A(a)}{\Gamma \vdash \forall x.A(x)}(\mathcal{R}-\forall)$$

$$\frac{A(a), \Gamma \vdash \Delta}{\exists x.A(x), \Gamma \vdash \Delta}(\mathcal{L}-\exists) \qquad \frac{\Gamma \vdash A(t)}{\Gamma \vdash \exists x.A(x)}(\mathcal{R}-\exists)$$

where $\Gamma, \Delta, \Gamma_1, \Gamma_2, \Delta_1, \Delta_2$ are sequences of formulas, with $|\Delta| \leq 1$ and $|\Delta_1, \Delta_2| \leq 1$, A, B formulas, a a variable not appearing in $\Gamma \cup \Delta$, and t a term.

Notice that it is equivalent to replace the rule $\mathcal{L}-\vee$ by the two rules

$$\frac{A, \Gamma_1 \vdash \Delta \quad B, \Gamma_2 \vdash}{A \vee B, \Gamma_1, \Gamma_2 \vdash \Delta}(\mathcal{L}-\vee_1) \qquad \frac{A, \Gamma_1 \vdash \quad B, \Gamma_2 \vdash \Delta}{A \vee B, \Gamma_1, \Gamma_2 \vdash \Delta}(\mathcal{L}-\vee_2)$$

since in $\mathcal{L}-\vee$ we have $|\Delta_1, \Delta_2| \leq 1$. The multiplicative form of the rules gives to the rule $\mathcal{L}-\vee$ a very different behaviour than the additive form

$$\frac{A, \Gamma \vdash \Delta \quad B, \Gamma \vdash \Delta}{A \vee B, \Gamma \vdash \Delta}(\mathcal{L}-\vee)$$

in which the goal is the same on each side.

If a formula is provable in Intuitionnistic Direct Predicate Calculus then we will say that its is *provable in DPC_i*.

Definition 15 (intuitionnistic derivation) *A derivation $\mathcal{D}$ of DPC is said to be* intuitionnistic *if every application of the rule $\mathcal{R}-\vee$ on $A \vee B$ is immediately followed by a weakening on A or B, and if every rule has at most one formula in conclusion.*

Clearly we have:

Proposition 9 *F is provable in DPC$_i$ if and only if there is an intuitionnistic derivation of F in DPC.*

Proof: Let $\mathcal{D}$ be a derivation of F in DPC$_i$. Excepted $\mathcal{R}-\vee_1$ and $\mathcal{R}-\vee_2$, every rule of DPC$_i$ is a rule of DPC, and remains unchanged. The rule

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee_1)$$

is translated into

$$\frac{\dfrac{\Gamma \vdash A}{\Gamma \vdash A, B}(\mathcal{R}-W)}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee)$$

Likewise for $\mathcal{R}-\vee_2$. The resulting derivation is clearly an intuitionnistic derivation.

Conversely, if there exists an intuitionnistic derivation of F in DPC, then the same translation clearly gives a derivation of F in DPC$_i$. □

The choice we made for the order of application of the rules in the previous algorithm is not innocent. The idea is to get intuitionnistic proofs when they exist. Indeed, two derivations can be equivalent, the one being intuitionnistic, the other not. For instance

$$\frac{\dfrac{\dfrac{}{A \vdash A}(Ax)}{\neg A, A \vdash}(\mathcal{L}-\neg)}{\neg A \vdash \neg A}(\mathcal{R}-\neg) \quad \text{and} \quad \frac{\dfrac{\dfrac{}{A \vdash A}(Ax)}{\vdash \neg A, A}(\mathcal{R}-\neg)}{\neg A \vdash \neg A}(\mathcal{L}-\neg)$$

are equivalent, but only the first one is intuitionnistic. That's the reason why we chose to apply $\mathcal{R}-\neg$ before $\mathcal{L}-\neg$ in our algorithm. Likewise, the derivations

$$\frac{\dfrac{\dfrac{\dfrac{}{A \vdash A}(Ax) \quad \dfrac{\dfrac{}{B \vdash B}(Ax)}{B, \neg B \vdash}(\mathcal{L}-\neg)}{A \supset B, \neg B, A \vdash}(\mathcal{L}-\supset)}{A \supset B, \neg B \vdash \neg A}(\mathcal{R}-\neg) \quad \dfrac{}{C \vdash C}(Ax)}{A \supset B, C, \neg B \vdash \neg A \wedge C}(\mathcal{R}-\wedge) \quad \text{and}$$

$$\frac{\dfrac{\dfrac{\dfrac{}{A \vdash A}(Ax)}{\vdash A, \neg A}(\mathcal{R}-\neg) \quad \dfrac{}{C \vdash C}(Ax)}{C \vdash A, \neg A \wedge C}(\mathcal{R}-\wedge) \quad \dfrac{\dfrac{}{B \vdash B}(Ax)}{B, \neg B \vdash}(\mathcal{L}-\neg)}{A \supset B, C, \neg B \vdash \neg A \wedge C}(\mathcal{L}-\supset)$$

are equivalent, but only the first one is intuitionnistic. That's the reason why we chose to apply $\mathcal{R}-\wedge$ before $\mathcal{L}-\supset$ and $\mathcal{L}-\vee$.

These choices are justified by the following result:

Proposition 10 *If a derivation $\mathcal{D}$ is intuitionnistic then the derivation $\bar{\mathcal{D}}$ is intuitionnistic too.*

Proof: The proof is by absurdum: assume that $\bar{\mathcal{D}}$ is not intuitionnistic. There are two possible reasons:

- either we have an application of $\mathcal{R}-\vee$ not followed by a weakening on A or B:

$$\frac{\dfrac{\dfrac{\cdots}{\Gamma \vdash A, B}(\mathcal{X})}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee)}{}$$

  But $\bar{\mathcal{D}}$ is canonical, so $A \vee B$ is satisfied in $\bar{\mathcal{D}}$ so in $\mathcal{D}$ too. But $\mathcal{D}$ is intuitionnistic so the application of $\mathcal{R}-\vee$ on $A \vee B$ in $\mathcal{D}$ is followed by a weakening on A or B, so A or B must be weakened in $\bar{\mathcal{D}}$, which is a contradiction.

- or there is a non-intuitionnistic sequent in $\bar{\mathcal{D}}$. Let us consider the "first" non-intuitionnistic sequent in $\bar{\mathcal{D}}$. There are two possibilities:

– the corresponding rule is

$$\frac{\Gamma \vdash A, F}{\Gamma, \neg A \vdash F}(\mathcal{L}-\neg)$$

$\bar{\mathcal{D}}$ is canonical so A is satisfied in $\bar{\mathcal{D}}$, so in $\mathcal{D}$ too, and F is a quantifier or a conjunction not potential. Let us consider the rule in $\mathcal{D}$ where A is principal, that is

$$\frac{\Gamma' \vdash A}{\Gamma', \neg A \vdash}(\mathcal{L}-\neg)$$

since $\mathcal{D}$ is intuitionnistic. But this contradicts the fact that F were not potential in $\bar{\mathcal{D}}$.

– the corresponding rule is

$$\frac{\Gamma_1 \vdash F, A \quad B, \Gamma_2 \vdash}{A \supset B, \Gamma_1, \Gamma_2 \vdash F}(\mathcal{L}-\supset)$$

$\bar{\mathcal{D}}$ is canonical so F is a non-potential conjunction, or an atom. Then let us consider the rule in $\mathcal{D}$ where $A \supset B$ is principal, that

$$\frac{\Gamma_1' \vdash A \quad B, \Gamma_2' \vdash F'}{A \supset B, \Gamma_1', \Gamma_2' \vdash F'}(\mathcal{L}-\supset)$$

Necessarily, F' must be F, otherwise it would contradict the non potentiality of F in the above rule. But, since F is satisfied, and connected to A seen the derivation $\mathcal{D}$, this application is not possible.

□

As a consequence we obtain the completeness of our algorithm with respect to intuitionnistic provability:

Theorem 4 (Completeness) *If a sequent S is provable in DPC_i then there exist a path $\mathcal{P}$ (and a substitution σ) such that $\mathcal{A}(\mathcal{P})$ is defined and is an intuitionnistic proof of S.*

5 Implementation in the system Coq

An implementation of the decision procedure we just presented has been realized in the system Coq, version 5.10 [2], a proof assistant developed at INRIA-Rocquencourt and ENS Lyon.

The implementation of the decision procedure itself has been realized in *Caml Light* [11, 8], independently of the system Coq, with its own representation of terms and proofs. The interface with the system Coq. which is written in *Caml Light*, is then just a translation of the goal to prove into our representation of terms, and of the resulted proof (if it exists) in a natural deduction proof. We just give here the main lines of this implementation, and we won't focus on the possible algorithmic optimizations (see [1], pages 137–141).

5.1 Implementation of the decision procedure in *Caml Light*

We assume that we have types `term`, `formula` and `proof` to represent terms, formulas and proofs of Direct Predicate Calculus. A possible representation of terms and formulas may be the following:

```
type term = Var of string
          | Fun of string * (term list);;

type formula = Atom   of string * int * (term list)
             | Neg    of formula
             | Imp    of formula * formula
             | Conj   of formula * formula
             | Disj   of formula * formula
             | ForAll of string * formula
             | Exists of string * formula;;
```

even if it is not exactly the one we chose.

Notice that atomic formulas are of the form `Atom(name,n,tl)`, where `name` is the name of the predicate, `tl` the list of terms on which it is applied and `n` an integer. This integer `n` allows to separate formulas: formulas of $\mathcal{L}_0$ are represented with the integer 0 and separated formulas of $\mathcal{L}$ with different integers for each occurrence of an atomic formula.

We define the following functions:

```
value separate   : formula -> formula;;
value pi_formula : formula -> formula;;
```

where `pi_formula` is the canonical projection from $\mathcal{L}$ into $\mathcal{L}_0$ (denoted π) and `separate` a function associating to a formula F_0 of $\mathcal{L}_0$ a separated formula F of $\mathcal{L}$ such that $\pi(F) = F_0$ (by doing a prefix traversal of F_0 and giving increasing integers for each atomic formula).

5.1.1 Paths search

We consider a formula F of $\mathcal{L}$ for which we want to know if it is provable in Direct Predicate Calculus and, in that case, what are the possible derivations.

The first step of the decision procedure is to determine the Herbrand form of F. The function

```
value Herbrand : formula -> (string list * formula);;
```

takes a formula F and returns the list of its essentially existential variables together with its Herbrand form F_H.

The next step is to look for pairs $(\sigma, \mathcal{P})$ where σ is a substitution and $\mathcal{P}$ a path for $\vdash F_H$ satisfying the condition (1) of the theorem 1. For this purpose, we proceed using the method of [1], that we briefly presented in 3.

We first determine the set

$$\mathcal{M} = \{ (P, P', u) \quad | \quad P \in \mathcal{A}^+ \wedge P' \in \mathcal{A}^- \wedge P[u] \approx P'[u] \}$$

where $\mathcal{A}^+$ is the set of positive atomic subformulas of F_H and $\mathcal{A}^-$ the set of negative ones. So, we have to solve the unification problems $\vec{u} = \vec{v}$ where $A^i(\vec{u})$ and $A^j(\vec{v})$ are two atomic subformulas of F_H, respectively appearing positively and negatively. The first-order unification is decidable. A lot of algorithms are known, and we used Martelli and Montanari's one (see [10], and also [7], pages 224–226).

The function `all_matches` compute the set $\mathcal{M}$

```
value all_matches : formula -> (formula * formula * unifier) list;;
```

Then we look for the pairs $(\mathcal{U}, \sigma)$, where $\mathcal{U}$ is a subset of $\mathcal{M}$ and σ a substitution, such that

- $\mathcal{U}$ is not empty;
- All atomic formulas of $\mathcal{U}$ are distinct;
- $\bigvee_{(P,P',u)\in\mathcal{U}} u$ is defined and $\sigma = \bigvee_{(P,P',u)\in\mathcal{U}} u$;
- If $A \circ B$ is a conjunctive subformula of F_H then A and B are not connected by $\mathcal{U}$;
- If $\mathcal{U}$ satisfies a conjunctive formulas $A \circ B$ then $\mathcal{U}$ satisfies A and B.

Such a pair $(\mathcal{U}, \sigma)$ clearly satisfies the conditions (a–e) and (1). They are represented by the type

```
type path == (formula * formula * unifier) list * unifier;;
```

and the function

```
value paths : (formula * formula * unifier) list -> formula
              -> path list;;
```

compute all such pairs from $\mathcal{M}$ and F_H. We do not compute all the subsets of $\mathcal{M}$, which would be costly, but we use the method described in [1], page 139.

It remains to satisfy the condition (f), that is the absence of conjunctive cycle. For this purpose we build the graph G introduced in 3.2.2 and we check if it has a cycle or not (this can be done in linear time). The function

```
value valid_paths : formula -> path list -> path list;;
```

takes the formula F_H, a list of potential paths for F_H, and returns the set of paths for F_H (those without conjunctive cycle).

So we can now build proofs from paths.

5.1.2 From paths to proofs

Let $\mathcal{P}$ be a path for F_H and σ a substitution such that the condition (1) is satisfied. First we replace in $\mathcal{P}$ and σ the Herbrand terms by the corresponding variables (they are no longer useful), that is we replace every occurrence of $f_x(\vec{u})$ by x. We keep the notation $\mathcal{P}$ and σ for the resulting path and substitution.

The making of a derivation from $\mathcal{P}$ follows the algorithm given in 4.2. The function

```
value proof_of_path : formula -> path -> proof;;
```

applies the algorithm and returns a proof of F, or raises the exception `FAILURE`.

Then we can give the main body of the decision procedure:

```
exception Not_provable_in_DPC of string;;

(*** prove : formula -> proof ***)

let prove f =
  let f0 = separate f in
  let (ex,f1) = herbrand f0 in
  let M = all_matches f1 in
  if M=[] then
    raise (Not_provable_in_DPC "M is empty.")
  else
    let ps = paths M f1 in
    if ps=[] then
    raise (Not_provable_in_DPC "No path.")
  else
    let vp = valid_paths f1 all_paths in
    if vp=[] then
    raise (Not_provable_in_DPC "Every path has a conjunctive cycle.")
  else
    let rec quant = function
        p::rest -> try proof_of_path f1 p
                   with FAILURE -> quant rest
      | [] -> raise (Not_provable_in_DPC
                              "Path(s) do not correspond to proofs.")
    in quant vp
;;
```

5.1.3 Complexity

Our purpose is to evaluate the complexity of the decision procedure.

Let F be a formula and n the number of atomic subformulas of F. This number n is of the size of F (even if it can be very less, like in $\neg\neg\ldots\neg A$) and we will consider them as equal. Let a^+ be the number of positive atomic subformulas and a^- the number of negative ones.

The first step is to determine the set $\mathcal{M}$, (*i.e.*) the triplets (P, P', u) satisfying (c) and (1). There are at most $\frac{n^2}{4}$ pairs (P, P') with $P \in \mathcal{A}^+$ and $P' \in \mathcal{A}^-$, and so the computing of $\mathcal{M}$ is in time $O(n^2)$.

Then we look for paths among the subsets of $\mathcal{M}$. For a subset of $\mathcal{M}$ of cardinal k, we must check the conditions on paths. We admit that this check can be done in time kn.

Let p_k be the number of paths of cardinal k, with $1 \leq k \leq a = min(a^+, a^-)$. Seen the conditions (b) and (1), we have

$$p_k \leq k! C_{a^+}^k C_{a^-}^k$$

The search for all paths takes a total time

$$\begin{aligned} T_n &\leq \sum_{k=1}^{a} k\, n\, p_k \\ &\leq n \sum_{k=1}^{a} k\, k!\, C_{a^+}^k C_{a^-}^k \\ &= n \sum_{k=1}^{a} k\, k!\, C_{n-a}^k C_a^k \end{aligned}$$

But we have $a \leq n/2$ so C_{n-a}^k and C_a^k are always less than $C_{n/2}^{n/4}$, and so

$$\begin{aligned} T_n &\leq n\, a^2\, a! \left(C_{n/2}^{n/4}\right)^2 \\ &\leq n \left(\frac{n}{2}\right)^2 \left(\frac{n}{2}\right)! \left(C_{n/2}^{n/4}\right)^2 \\ &\sim \frac{n^3}{2} \left(\frac{n}{e}\right)^{\frac{n}{2}} \end{aligned}$$

using the Stirling formula.

The complexity is "reasonably exponential", when we remember that propositional calculus decidability is already exponential.

5.2 Overview of the system Coq

Coq [2] is a proof assistant for higher-order logic. It allows to write specifications and propositions, to check mathematical proofs, and to automatically synthesize computer programs from the proof of their specifications.

The language of Coq is the *Calculus of Inductive Constructions*, and its proof system is an intuitionnistic natural deduction. We are going to briefly present the syntax and the principles of Coq, in order to explain how we implemented the decision procedure for Direct Predicate Calculus in Coq.

5.2.1 Terms and propositions

In Coq, all types are terms. Given basic types, it is possible to build other types from three elementary constructions:

- *application* of a term `f` to a term `x`, denoted `(f x)`;
- *abstraction* of a variable `x` of type `T` in a term F, denoted `[x:T]`F;
- *product* of a type T_1 and a type T_2, denoted `(`x`:`T_1`)`T_2. When the product is not dependent (that is x does not appear in T_2) it is also denoted T_1 `->` T_2, and then represents the type of functional objects.

Coq allows also to define *inductive types*, as the type of naturals for instance. In a language of the ML family, like *Caml Light*, it is possible to define such a type as:

```
type nat = O
         | S of nat;;
```

In Coq, we define the same inductive type in a similar way:

```
Inductive Set nat = O : nat
                  | S : nat -> nat.
```

The function associating x to $x+2$ will be denoted `[x:nat](S (S x))`, and it has the type `nat->nat`.

In Coq, logical propositions have type `Prop`. Atomic propositions are obtained as the application of predicate to terms. For instance, if `P` is a predicate of type `nat->Prop`, then `(P (S O))` is an atomic proposition. Propositions are built from the usual connectives and quantifiers, with the following syntax:

- If `P` and `Q` are two propositions then `P->Q` is the proposition $P \supset Q$,
- If `P` and `Q` are two propositions then `P/\Q` is the proposition $P \wedge Q$,
- If `P` and `Q` are two propositions then `P\/Q` is the proposition $P \vee Q$,
- `True` is the tautological proposition, `False` the absurd proposition,
- If `P` is a proposition then `~P` is the proposition $\neg P$, defined as $P \supset False$,
- If `P` is a proposition in which x is a free variable of type T then `(x:T)P` is the proposition $\forall x : T.P$,
- If `P` is a proposition in which x is a free variable of type T then `(Ex [x:T]P)` is the proposition $\exists x : T.P$.

We can also write higher-order propositions, like `(A:Prop)A->A` or `(P:nat->Prop)(Ex [x:nat](P x))`. If `P` is a predicate of type `nat->Prop`, the "drinkers' theorem" is written `(Ex [x:nat](y:nat)(P x)->(P y))`.

Let us show now how to prove propositions in the system Coq.

5.2.2 Tactics and proofs

If we want to prove the proposition $A \supset (A \supset B) \supset B$ in Coq, we write for instance

```
Coq < Lemma example1 : (A,B:Prop) A->(A->B)->B.
```

We obtain the following goal:

```
1 subgoal

  ============================
   (A,B:Prop)(A->(A->B)->B)
```

In a goal, the proposition to prove is below the double line and the context in which we do the proof — here an empty context for the moment — is on top of it. We can see the goal as the sequent $\Gamma \vdash P$, where Γ is the context (also called environment) and P the proposition to prove.

We call *tactic* every Coq command which applies one or more inference rules to the current goal. To each tactic is associated a *validation*, that is a structure which allows to verify, once the proof is done, that the inference rules have been correctly used. Then, we are always ensured to stay in a coherent state, and to have a sequent which can be derived from the initial one.

The tactic `Intro`, for instance, allows to introduce universally quantified variables and hypothesis in the context. Let us apply the tactic `Intro` to our goal:

```
example1 < Intro.
1 subgoal

  A : Prop
  ============================
   (B:Prop)(A->(A->B)->B)
```

By applying it as many times as possible (`Intros` does it) we get the goal:

```
  A : Prop
  B : Prop
  H : A
  H0 : A->B
  ============================
   B
```

We apply an hypothesis (or a lemma, an axiom,...) with the tactic `Apply`. Here we apply the hypothesis H_0 with the command:

```
example1 < Apply H0.
1 subgoal

  A : Prop
  B : Prop
  H : A
  H0 : A->B
  ============================
   A
```

A is an hypothesis. So we can finish the proof with the tactic `Assumption` :

```
example1 < Assumption.
Subtree proved!
```

In a more general way, there exist for each connective an introduction and an elimination rule. The elimination of a conective in realized through the tactic `Elim`. For instance, on the goal

```
  A : Prop
  B : Prop
  C : Prop
  H : A \/ B
  ============================
   C
```

the command `Elim H` produces the two subgoals:

```
2 subgoals

  A : Prop
  B : Prop
  C : Prop
  H : A \/ B
  ============================
   A->C

subgoal 2 is:
 B->C
```

5.3 Interface with the system Coq

We are now in position to interface our decision procedure with the system Coq. For this purpose, we must first translate the goal in the language $\mathcal{L}$, and then translate one of the resulting proofs, if there exist, into a sequence of Coq tactics.

The translation of the goal to prove into $\mathcal{L}$ is not a problem. It is a purely syntactic translation, in which we forget the types of terms. We have just to check if the goal is a first-order proposition. For this, it is sufficient to check that none of the quantified variables is a predicate or a function applied to terms.

Remark. The proposition $\forall A.(A \supset A)$ does not belong to first-order logic, but $A \supset A$ is trivially provable in Direct Predicate Calculus. More generally, if we can prove $P(x_1, \ldots, x_n)$ in DPC, then we can prove $\forall x_1 \ldots \forall x_n.P(x_1, \ldots, x_n)$ in Coq, even if the quantifiers $\forall x_1, \ldots, \forall x_n$ are not first-order ones. So we can first introduce all the prenex universal quantifications, then apply the decision procedure on the resulting goal (in which $x_1, \ldots, x_n$ have become free variables, that is constants). This way, the method is extended to a certain class of higher-order formulas.

Once the goal is translated into a formula F of $\mathcal{L}$, we can apply the decision procedure. If it fails, we leave the goal unchanged and we just warm the user that the tactic failed — which does not mean, of course, that the goal is not provable in Coq — and why.

If it succeeds, we get a set of derivations of F in DPC, from which we keep intuitionnistic ones. If there is almost one intuitionnistic derivation $\mathcal{D}$ of F then we translate it into Coq tactics. In the practice, we used internal Coq functions to do this translation, but, to clarify this step, we will give here the corresponding Coq top-level tactics.

Axioms. Every axiom

$$\frac{}{A \vdash A}(Ax)$$

in the derivation $\mathcal{D}$ means that A is in the local context. So we translate the rule *Axiom* by the tactic `Assumption`.

Structural rules. The weakening rule

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}(\mathcal{L}-W)$$

means that we forget the hypothesis A. We just do nothing and continue the translation.

On the other hand, the weakening rule

$$\frac{\Gamma \vdash}{\Gamma \vdash A}(\mathcal{R}-W)$$

means that we replace the goal to prove by $False$. The command `ElimType False` has this effect.

Logical rules. Here is the translation of logical rules of Intuitionnistic Direct Predicate Calculus into Coq tactics. We assume that this translation is recursively called on the generated subgoals. We denote $\mathcal{C}$ the local context of Coq.

$$\frac{\Gamma, A \vdash}{\Gamma \vdash \neg A}(\mathcal{R}-\neg) \qquad \texttt{Red, Intro},$$
because $\neg A$ is defined as $A \supset False$

$$\frac{\Gamma \vdash A}{\Gamma, \neg A \vdash}(\mathcal{L}-\neg) \qquad \texttt{Apply H},$$
where $H : \neg A \in \mathcal{C}$

$$\frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{\Gamma \vdash A \wedge B}(\mathcal{R}-\wedge) \qquad \texttt{Split}$$

$$\frac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}(\mathcal{L}-\wedge) \qquad \texttt{Elim H, Intro, Intro},$$
where $H : A \wedge B \in \mathcal{C}$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee_1) \qquad \texttt{Left}$$

$\dfrac{\Gamma \vdash B}{\Gamma \vdash A \vee B}(\mathcal{R}-\vee_2)$	`Right`
$\dfrac{A, \Gamma_1 \vdash \Delta_1 \quad B, \Gamma_2 \vdash \Delta_2}{A \vee B, \Gamma \vdash \Delta}(\mathcal{L}-\vee)$	`Elim H`, then `ElimType False` on the goal i such that $\Delta_i = \emptyset$, and `Intro` on the other, where $H : A \vee B \in \mathcal{C}$
$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}(\mathcal{R}-\supset)$	`Intro`
$\dfrac{\Gamma_1 \vdash A \quad B, \Gamma_2 \vdash \Delta}{A \supset B, \Gamma \vdash \Delta}(\mathcal{L}-\supset)$	`Cut B`, then `Apply H` on the second subgoal (where $H : A \supset B \in \mathcal{C}$) and `Intro` on the first one
$\dfrac{\Gamma \vdash P}{\Gamma \vdash \forall x.P}(\mathcal{R}-\forall)$	`Intro`
$\dfrac{P(t), \Gamma \vdash \Delta}{\forall x.P(x), \Gamma \vdash \Delta}(\mathcal{L}-\forall)$	`Cut` $P[x \leftarrow t]$, then `Apply (H t)` on the second subgoal (where $H : (x : T)P \in \mathcal{C}$) and `Intro` on the first subgoal
$\dfrac{\Gamma \vdash P(t)}{\Gamma \vdash \exists x.P(x)}(\mathcal{R}-\exists)$	`Exists` t
$\dfrac{P, \Gamma \vdash \Delta}{\exists x.P, \Gamma \vdash \Delta}(\mathcal{L}-\exists)$	`Elim H, Intro, Intro`, where $H : (Ex\ [x : T]P) \in \mathcal{C}$

The resulting tactic is called `Linear`, and thus called with:

```
Coq < Linear.
```

However, this tactic does not allow to use previous results or hypothesis of the context. It is useful to use lemmas in a proof, when some subgoals arise frequently, or when we need to do a part of the proof (a higher-order reasoning) "by hand", and to finish the proof with the tactic `Linear`.

For this purpose, we extended the syntax of our tactic in the following way. Assume that we have in the context the hypotheses

$$H_1 : c_1 \quad , \quad H_2 : c_2 \quad , \quad \ldots \quad , \quad H_n : c_n$$

and the goal to prove

$$(x_1 : T_1)(x_2 : T_2) \ldots (x_k : T_k)\, c$$

for which we want to (possibly) use the previous hypotheses.

The idea is to apply the decision procedure on the goal $c_1 \supset c_2 \supset \ldots \supset c_n \supset c$, seen we have $c_1, \ldots, c_n$ as hypotheses. The syntax is

```
Coq < Linear with H1 H2 ... Hn.
```

Examples. Let us illustrate the behaviour of the tactic `Linear` on some examples. Assume we have a natural a, predicate $P, Q, odd, even$ of type $nat \rightarrow Prop$ and a function f of type $nat \rightarrow nat$.

```
Variable a : nat.
Variable P,Q,odd,even : nat->Prop.
Variable f : nat -> nat.
```

We can show the following facts:

```
Theorem E1 : (x:nat)(Ex [y:nat]((P x)->(P y))).
Linear.
Save.

Theorem E2 : (Ex [x:nat](P x))
             -> ((y:nat)(P y) -> (Q y))
             -> (Ex [z:nat](Q z)).
Linear.
Save.

Theorem E3 : (even a)
              -> ((x:nat)((even x)->(odd (S x))))
              -> (Ex [y:nat](odd y)).
Linear.
Save.

Theorem E4 : ((x:nat)((and (P x) (odd x)) -> (even (f x))))
     -> ((x:nat)((even x)->(odd (S x))))
     -> (even a)
     -> (P (S a))
     -> (Ex [z:nat](even (f z))).
Linear.
Save.
```

The drinkers' theorem is more delicate to prove. The proof by case (we have either $\forall x.Q(x)$, or $\exists x.\neg Q(x)$) is done "by hand", but the corresponding subgoals are automatically proved by the tactics `Linear` (with a lemma for the second one).

```
Variable U : Set.
Variable Q : U -> Prop.

Axiom excluded_middle : (P:Prop) (P \/ ~P).

Lemma em_Q : (x:U) (Q x) \/ ~(Q x).
Exact [x:U](excluded_middle (Q x)).
Save.

Theorem  Drinker's_theorem : (x:U)(Ex [x: U]((Q x) -> (x: U) (Q x))).
Intro t0.
Generalize (excluded_middle (Ex [x:U]~(Q x))); Intro H; Elim H.
Linear.
Intro H0.
Exists t0.
Linear with H0 em_Q.
Save.
```

To illustrate the use of lemmas in the decision procedure, let us consider a specification of the predicate $\leq$ on naturals, a monotonic function f of type $nat \to nat$, and let us show

$$\forall a\ \exists b\ f(a) \leq f(b+1)$$

```
Variable le : nat -> nat -> Prop.
Variable f  : nat -> nat.

Axiom le_n : (n:nat)(le n n).
```

```
Axiom le_S : (n,m:nat)(le n m) -> (le n (S m)).
Axiom monoticity : (n,m:nat)(le n m) -> (le (f n) (f m)).

Lemma le_x_Sx : (x:nat)(le x (S x)).
Linear with le_n le_S.
Save.

Theorem L1 : (a:nat)(Ex [b:nat](le (f a) (f (S b)))).
Linear with le_x_Sx monoticity.
Save.
```

At last, let us show some examples on which the tactic fails:

```
Parameter A,B,C,D : Prop.

Theorem T1 : A -> (A /\ A).
  (* Error: Not provable in DPC (No path)
   * during command Linear.
   *)
Auto.
Save.

Theorem T2 : ((or A B) -> C)
        -> (or (D -> A) (D -> B))
        -> D -> C.
  (* Error: Not provable in DPC (No path)
   * during command Linear.
   *)
Intros H H0 H1.
Elim H0; Auto.
Save.
```

Conclusion

We have presented a decision procedure for Direct Predicate Calculus. Starting from [6, 1], we have extended the decision procedure to non-prenex formulas and to intuitionnistic proofs. This work can be related with [9] but it differs on two points: first, we do not consider explicitly rules permutabilities but we construct a particular proof (discarding the others, which are equivalent); secondly, we do not wait to reach the axioms rules to fail in proof search but we first look for potential axioms and then we construct a proof from axioms.

References

[1] G. Bellin and J. Ketonen. A decision procedure revisited : Notes on direct logic, linear logic and its implementation. *Theoretical Computer Science*, 95:115–142, 1992.

[2] C. Cornes, J. Courant, J.-C. Filliâtre, G. Huet, P. Manoury, C. Paulin-Mohring, C. Muñoz, C. Murthy, C. Parent, A. Saïbi, and B. Werner. The Coq Proof Assistant Reference Manual Version 5.10. Technical Report 0177, INRIA, July 1995.

[3] J. Gallier. Constructive Logics. Part I: A Tutorial on Proof Systems and Typed λ–Calculi. Research Report 8, Digital Equipment Corporation, May 1991.

[4] J. Gallier. Constructive Logics. Part II: Linear Logic and Proof Nets. Research Report 8, Digital Equipment Corporation, May 1991.

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

[6] J. Ketonen and R. Weyhrauch. A decidable fragment of Predicate Calculus. *Theoretical Computer Science*, 32:297–307, 1984.

[7] R. Lalement. *Logique, réduction, résolution*. Etudes et recherches en Informatique. MASSON, 1990.

[8] X. Leroy and P. Weis. *Manuel de référence du langage Caml*. InterEditions, 1993.

[9] P. D. Lincoln and N. Shankar. Proof search in first-order linear logic and other cut-free sequent calculi. In *Symposium on Logic in Computer Science*. IEEE Computer Society Press, 1994.

[10] A. Martelli and U. Montanari. An efficient unification algorithm. *ACM Trans. Prog. Lang. Syst.*, 4(2):258–282, April 1982.

[11] P. Weis and X. Leroy. *Le langage Caml*. InterEditions, 1993.