An Intuitionistic Logic That Proves Markov's Principle

(revision fixing typos – April 2021)

Hugo Herbelin
INRIA - PPS
Paris, France
Hugo.Herbelin@inria.fr

Abstract—We design an intuitionistic predicate logic that supports a limited amount of classical reasoning, just enough to prove a variant of Markov's principle suited for predicate logic.

At the computational level, the extraction of an existential witness out of a proof of its double negation is done by using a form of statically-bound exception mechanism, what can be seen as a direct-style variant of Friedman's A-translation.

Keywords-Markov's principle; intuitionistic logic; proof-as-program correspondence; exceptions

I. INTRODUCTION

In arithmetic, Markov's principle is a weak classical scheme stating $\neg\neg\exists x\, A(x) \to \exists x\, A(x)$ for every decidable formula $A(x)$. Though not derivable [1] in Heyting Arithmetic (i.e. intuitionistic arithmetic), its formulation as a rule (Markov's rule) is admissible (see e.g. [2] for a classical proof). One of the important properties of Markov's principle is that it implies, via the double-negation translation, the conservativity of classical logic over intuitionistic logic for Π_2^0-formulas.

Markov's principle is realizable (in the sense of Kleene [3]) by unbounded search (ensured terminating by use of classical reasoning). It is also realizable along Gödel "Dialectica" functional interpretation [4] with witness obtained by selecting the first suitable instance among those occurring in the proof of $\neg\neg\exists x\, A(x)$. This is generally by one of these two ways that programs are extracted from proofs that use Markov's principle, especially in constructive analysis where Markov's principle amounts to the property $\neg\neg x \neq y \to x \neq y$ ($\neq$ asserts the existence of a rational number that separates x and y, see e.g. [5]).

Howard's correspondence [6] between natural deduction and λ-calculus emphasized that Prawitz's normalization of intuitionistic natural deduction [7] can be used as a method to compute with proofs. By later showing a connection between classical axioms and control operators, Griffin [8] revealed that normalization is a computation device also for classical logic. While intuitionistic logic satisfies the disjunction and existence properties, it is however not the case of classical logic and Markov's principle has this peculiarity to lie in between, being classical in its form and still retaining the disjunction and existence properties, as

was shown e.g. by Smorynski in [2]. One of our objective is then to study intuitionistic logic extended with Markov's principle as a proper constructive logic.

An elementary method to prove the intuitionistic admissibility of Markov's rule has been given by Friedman who introduced for this purpose a notion of A-translation [9] (see also Dragalin [10]). In fact, Friedman's proof does not prove $\neg\neg\exists x\, A(x) \to \exists x\, A(x)$ for any propositional formula $A(x)$ but only for those $A(x)$ that do not contain the implication connective (this was noticed for instance in Berger [11]). More generally, Friedman's method can be interpreted from the proof-theoretical point of view as a proof that $\neg\neg A \to A$ is admissible in intuitionistic predicate logic whenever A is a $\forall$-$\to$-free formula, the case of $\exists x\, A(x)$ for $A(x)$ $\to$-free being just a prototypical instance of the scheme.

We concentrate here on predicate logic and refer to in this context as Markov's principle the scheme that asserts $\neg\neg T \to T$ whenever T is $\forall$-$\to$-free. Extending intuitionistic logic with classical reasoning on $\forall$-$\to$-free formulas is then the same as adding Markov's principle and we show that the resulting logic is constructive in the sense that it satisfies normalization and the disjunction and existence properties.

At the computational level, the added classical rules can be seen as a mechanism of (statically-bound) exception throwing similar to the one introduced by Nakano in his (intuitionistic) `catch` and `throw` calculus [12][1].

II. IQC_{MP}: AN INTUITIONISTIC PREDICATE LOGIC THAT PROVES MARKOV'S PRINCIPLE

Usual intuitionistic predicate logic (IQC) is defined from a set of function symbols f, g, $\ldots$, each of a given arity and a set of predicate symbols P, Q, R, ... each with an arity too. Functions symbols of arity 0 are called constants and predicate symbols of arity 0 are called atomic propositions.

Terms are built from a set of variables x, y, ... by

$$t, u \quad ::= \quad f(\vec{t}) \mid x$$

where f ranges over function symbols and $\vec{t}$ denotes in $f(\vec{t})$ a sequence of terms of length the arity of f. Formulas are

---

[1]Ordinary (dynamically bound) exceptions, as found in the ML, C++ or Java programming languages can actually be considered too, this is part of ongoing work.

$$\frac{A \in \Gamma}{\Gamma \vdash_\Delta A} \;\text{AXIOM}$$

$$\frac{\Gamma \vdash_\Delta A_1 \qquad \Gamma \vdash_\Delta A_2}{\Gamma \vdash_\Delta A_1 \wedge A_2} \wedge_I \qquad \frac{\Gamma \vdash_\Delta A_1 \wedge A_2}{\Gamma \vdash_\Delta A_i} \wedge_E^i$$

$$\frac{\Gamma \vdash_\Delta A_i}{\Gamma \vdash_\Delta A_1 \vee A_2} \vee_I^i \qquad \frac{\Gamma \vdash_\Delta A_1 \vee A_2 \qquad \Gamma, A_1 \vdash_\Delta B \qquad \Gamma, A_2 \vdash_\Delta B}{\Gamma \vdash_\Delta B} \vee_E$$

$$\frac{\Gamma, A \vdash_\Delta B}{\Gamma \vdash_\Delta A \rightarrow B} \rightarrow_I \qquad \frac{\Gamma \vdash_\Delta A \rightarrow B \qquad \Gamma \vdash_\Delta A}{\Gamma \vdash_\Delta B} \rightarrow_E$$

$$\frac{\Gamma \vdash_\Delta A(x) \qquad x \text{ fresh}}{\Gamma \vdash_\Delta \forall x\, A(x)} \forall_I \qquad \frac{\Gamma \vdash_\Delta \forall x\, A(x)}{\Gamma \vdash_\Delta A(t)} \forall_E$$

$$\frac{\Gamma \vdash_\Delta A(t)}{\Gamma \vdash_\Delta \exists x\, A(x)} \exists_I \qquad \frac{\Gamma \vdash_\Delta \exists x\, A(x) \qquad \Gamma, A(x) \vdash_\Delta B \qquad x \text{ fresh}}{\Gamma \vdash_\Delta B} \exists_E$$

$$\frac{}{\Gamma \vdash_\Delta \top} \top_I \qquad \frac{\Gamma \vdash_\Delta \bot}{\Gamma \vdash_\Delta C} \bot_E$$

$$\frac{\Gamma \vdash_{T,\Delta} T}{\Gamma \vdash_\Delta T} \;\text{CATCH} \qquad \frac{\Gamma \vdash_\Delta T \qquad T \in \Delta}{\Gamma \vdash_\Delta C} \;\text{THROW}$$

Figure 1. Inference rules of IQC_{MP}

built from the standard connectives and quantifiers by the grammar

$$\begin{aligned} A, B \quad &::= \quad P(\vec{t}) \mid \top \mid \bot \mid A \rightarrow B \mid A \wedge B \mid A \vee B \\ &\mid \quad \forall x\, A \mid \exists x\, A \end{aligned}$$

where P ranges over predicate symbols and $\vec{t}$ is a sequence of terms whose length is the arity of P. Negation $\neg A$ is defined as $A \rightarrow \bot$. In $\forall x\, A$ and $\exists x\, A$, x is bound and freely subject to renaming (so-called α-conversion).

IQC_{MP} is an extension of IQC. Its inference rules are given on Figure 1 (we use natural deduction). The subclass of $\forall$-$\rightarrow$-free formulas plays a special role and we use T, U, ... to denote such formulas:

$$T, U \quad ::= \quad P(\vec{t}) \mid \top \mid \bot \mid T \wedge U \mid T \vee U \mid \exists x\, T$$

Contexts of formulas, written Γ, are ordered sequences of formulas. Contexts of $\forall$-$\rightarrow$-free formulas, written Δ, are ordered sequences of $\forall$-$\rightarrow$-free formulas. By $\neg\Delta$ is meant the context obtained by distributing $\neg$ over the formulas of Δ. Note that IQC can be characterized as the subset of IQC_{MP} obtained by removing the rules CATCH and THROW and keeping Δ empty.

The main difference between IQC_{MP} and IQC is that the former supports classical reasoning on $\forall$-$\rightarrow$-free formulas. This is implemented by the rules CATCH and THROW which say that to prove a $\forall$-$\rightarrow$-free formula T, one is allowed to change its mind during the proof and to restart a new proof of T at any time.

The main properties of IQC_{MP} is that it proves Markov's principle while still retaining the disjunction and existence properties that are characteristic of intuitionistic logic.

Let us write MP for the scheme $\neg\neg T \rightarrow T$ where T is a $\forall$-$\rightarrow$-free formula (this is the "proof-theoretic" way to think about Markov's principle) and $MP_\exists$ for the scheme $\neg\neg\exists\vec{x}\, A(\vec{x}) \rightarrow \exists\vec{x}\, A(\vec{x})$ where $A(\vec{x})$ is an $\rightarrow$-free proposition (this is the "standard" way to think about Markov's principle). Both schemes are logically equivalent.

Theorem 1: In IQC_{MP}, MP and $MP_\exists$ hold.

Proof: One gets a proof of $T \vdash_T \bot$ by applying THROW. By $\rightarrow_I$ and $\rightarrow_E$ we obtain a proof of $\neg\neg T \vdash_T \bot$. By applying $\bot_E$ followed by CATCH, we get a proof of $\neg\neg T \vdash T$ from which $\neg\neg T \rightarrow T$ derives (note that we freely use the lemma that $\Gamma \vdash_\Delta A$ implies $\Gamma' \vdash_{\Delta'} A$ for $\Gamma \subset \Gamma'$ and $\Delta \subset \Delta'$; this weakening lemma, as expected,

$$\frac{(a : A) \in \Gamma}{\Gamma \vdash_\Delta a : A} \ \text{AXIOM}$$

$$\frac{\Gamma \vdash_\Delta p_1 : A_1 \qquad \Gamma \vdash_\Delta p_2 : A_2}{\Gamma \vdash_\Delta (p_1, p_2) : A_1 \wedge A_2} \ \wedge_I \qquad \frac{\Gamma \vdash_\Delta p : A_1 \wedge A_2}{\Gamma \vdash_\Delta \pi_i p : A_i} \ \wedge_E^i$$

$$\frac{\Gamma \vdash_\Delta p : A_i}{\Gamma \vdash_\Delta \iota_i(p) : A_1 \vee A_2} \ \vee_I^i \qquad \frac{\Gamma \vdash_\Delta p : A_1 \vee A_2 \qquad \Gamma, a_1 : A_1 \vdash_\Delta p_1 : B \qquad \Gamma, a_2 : A_2 \vdash_\Delta p_2 : B}{\Gamma \vdash_\Delta \texttt{case } p \texttt{ of } [a_1.p_1 \,|\, a_2.p_2] : B} \ \vee_E$$

$$\frac{\Gamma, a : A \vdash_\Delta p : B}{\Gamma \vdash_\Delta \lambda a.p : A \to B} \ \to_I \qquad \frac{\Gamma \vdash_\Delta p : A \to B \qquad \Gamma \vdash_\Delta q : A}{\Gamma \vdash_\Delta p\,q : B} \ \to_E$$

$$\frac{\Gamma \vdash_\Delta p : A(x) \qquad x \text{ fresh}}{\Gamma \vdash_\Delta \lambda x.p : \forall x\, A(x)} \ \forall_I \qquad \frac{\Gamma \vdash_\Delta p : \forall x\, A(x)}{\Gamma \vdash_\Delta p\,t : A(t)} \ \forall_E$$

$$\frac{\Gamma \vdash_\Delta p : A(t)}{\Gamma \vdash_\Delta (t,p) : \exists x\, A(x)} \ \exists_I \qquad \frac{\Gamma \vdash_\Delta p : \exists x\, A(x) \qquad \Gamma, a : A(x) \vdash_\Delta q : B \qquad x \text{ fresh}}{\Gamma \vdash_\Delta \texttt{dest } p \texttt{ as } (x,a) \texttt{ in } q : B} \ \exists_E$$

$$\frac{}{\Gamma \vdash_\Delta () : \top} \ \top_I \qquad \frac{\Gamma \vdash_\Delta p : \bot}{\Gamma \vdash_\Delta \texttt{efq } p : C} \ \bot_E$$

$$\frac{\Gamma \vdash_{\alpha:T,\Delta} p : T}{\Gamma \vdash_\Delta \texttt{catch}_\alpha\, p : T} \ \text{CATCH} \qquad \frac{\Gamma \vdash_\Delta p : T \qquad (\alpha : T) \in \Delta}{\Gamma \vdash_\Delta \texttt{throw}_\alpha\, p : C} \ \text{THROW}$$

Figure 2. Proof-term annotation of IQC_{MP}

indeed holds in IQC_{MP}). The case of $MP_\exists$ holds by logical equivalence with MP. ∎

Theorem 2: $\Gamma \vdash A$ in IQC_{MP} iff $MP, \Gamma \vdash A$ in IQC iff $MP_\exists, \Gamma \vdash A$ in IQC.

Proof: By Theorem 1, $\Gamma, MP \vdash A$ implies $\Gamma \vdash A$ which obviously is a proof of IQC_{MP}. Conversely, we prove by induction on a derivation of $\Gamma \vdash_\Delta A$ in IQC_{MP} that $MP, \Gamma, \neg\Delta \vdash A$ holds in IQC. All cases are direct and we use MP for interpreting the rule CATCH. ∎

Theorem 3 (Disjunction property): In IQC_{MP}, if $\vdash A_1 \vee A_2$ then $\vdash A_1$ or $\vdash A_2$.

Theorem 4 (Existence property): In IQC_{MP}, if $\vdash \exists x\, A(x)$ then there exists t such that $\vdash A(t)$.

The proof of these last two theorems is the subject of the next section.

III. THE PROOF THEORY OF IQC_{MP}

We show that the proofs of IQC_{MP} have a computational interpretation as programs in a λ-calculus extended with a mechanism of statically-bound exceptions implemented with operators named `catch` and `throw`.

The language of proofs is defined by the grammar

$$
\begin{aligned}
p, q \ ::= \ & a \mid \iota_i(p) \mid (p, q) \mid (t, p) \mid \lambda a.p \mid \lambda x.p \mid () \\
& | \quad \texttt{case } p \texttt{ of } [a_1.p_1 \,|\, a_2.p_2] \\
& | \quad \pi_i(p) \mid \texttt{dest } p \texttt{ as } (x, a) \texttt{ in } q \\
& | \quad p\,q \mid p\,t \mid \texttt{efq } p \\
& | \quad \texttt{catch}_\alpha p \mid \texttt{throw}_\alpha p
\end{aligned}
$$

where a, b, ... range over a first set of proof variables. and α, β, ... range over another set of proof variables. The constructions $\lambda a.p$, $\texttt{case } p \texttt{ of } [a_1.p_1 \,|\, a_2.p_2]$ and $\texttt{dest } p \texttt{ as } (x, a) \texttt{ in } q$ bind a, a_1 and a_2. The constructions $\lambda x.p$ and $\texttt{dest } p \texttt{ as } (x, a) \texttt{ in } q$ bind x. The construction $\texttt{catch}_\alpha p$ binds α. The binders are considered up to the actual name used to represent the binder (α-conversion).

The annotation of IQC_{MP} with proof-terms is given in Figure 2 where the contexts Γ and Δ are now maps from variable names to formulas. For instance, the proof of Markov's principle in Theorem 1 is

$$\lambda a.\texttt{catch}_\alpha \texttt{ efq } (a\, \lambda b.\texttt{throw}_\alpha\, b)$$

with the typing derivation given on Figure 3.

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{b:T \vdash_{\alpha:T} b:T}\;\text{AXIOM}}{b:T \vdash_{\alpha:T} \mathtt{throw}_\alpha\, b:\bot}\;\text{THROW}}{\vdash_{\alpha:T} \lambda b.\mathtt{throw}_\alpha\, b:\neg T}\;\to_I}{\dfrac{\dfrac{\overline{a:\neg\neg T \vdash_{\alpha:T} a:\neg\neg T}\;\text{AXIOM}\qquad}{a:\neg\neg T \vdash_{\alpha:T} a\,(\lambda b.\mathtt{throw}_\alpha\, b):\bot}\;\to_E}{\dfrac{a:\neg\neg T \vdash_{\alpha:T} \mathtt{efq}\, a\,(\lambda b.\mathtt{throw}_\alpha\, b):T}{\dfrac{a:\neg\neg T \vdash \mathtt{catch}_\alpha\, \mathtt{efq}\, a\,(\lambda b.\mathtt{throw}_\alpha\, b):T}{\vdash \lambda a.\mathtt{catch}_\alpha\, \mathtt{efq}\, a\,(\lambda b.\mathtt{throw}_\alpha\, b):\neg\neg T \to T}\;\to_I}\;\text{CATCH}}\;\bot_E}}$$

Figure 3. Proof of MP

A subclass of proofs will play a particular role in extracting the intuitionistic content of weakly classical proofs of IQC_{MP}. These are the values defined by

$$V \quad ::= \quad a \mid \iota_i(V) \mid (V,V) \mid (t,V) \mid \lambda a.p \mid \lambda x.p \mid ()$$

Another class of expressions will be useful to define the reduction, it is the class of elementary evaluation contexts defined by

$$\begin{aligned}
F[\,] \quad ::= \quad & \mathtt{case}\ [\,]\ \mathtt{of}\ [a_1.p_1 \mid a_2.p_2] \\
\mid\ & \pi_i([\,]) \mid \mathtt{dest}\ [\,]\ \mathtt{as}\ (x,a)\ \mathtt{in}\ p \\
\mid\ & [\,]\,q \mid (\lambda x.q)\,[\,] \\
\mid\ & [\,]\,t \mid \mathtt{efq}\,[\,] \mid \mathtt{throw}_\alpha\,[\,] \\
\mid\ & \iota_i([\,]) \mid ([\,],p) \mid (V,[\,]) \mid (t,[\,])
\end{aligned}$$

For $F[\,]$ an elementary evaluation context and p a proof, we write $F[p]$ for the proof obtained by plugging p into the hole of $F[\,]$.

We can now define evaluation in IQC_{MP} as the congruent closure of the following reductions:

$$\begin{aligned}
(\lambda a.p)\,V &\to p[a \leftarrow V] \\
(\lambda x.p)\,t &\to p[x \leftarrow t] \\
\mathtt{case}\ \iota_i(V)\ \mathtt{of}\ [a_1.p_1 \mid a_2.p_2] &\to p_i[a_i \leftarrow V] \\
\mathtt{dest}\ (t,V)\ \mathtt{as}\ (x,a)\ \mathtt{in}\ p &\to p[x \leftarrow t][a \leftarrow V] \\
\pi_i(V_1,V_2) &\to V_i \\
F[\mathtt{efq}\ p] &\to \mathtt{efq}\ p \\
F[\mathtt{throw}_\alpha p] &\to \mathtt{throw}_\alpha p \\
\mathtt{catch}_\alpha \mathtt{throw}_\alpha p &\to \mathtt{catch}_\alpha p \\
\mathtt{catch}_\alpha \mathtt{throw}_\beta V &\to \mathtt{throw}_\beta V \quad (\alpha \neq \beta) \\
\mathtt{catch}_\alpha V &\to V
\end{aligned}$$

where the substitutions $p[a \leftarrow V]$ and $p[x \leftarrow t]$ are capture-free with respect to the three kinds of variables (x, a and α).

Note that this is a call-by-value reduction semantics and that we do not consider commutative cuts just because we are only concerned with the normalization of closed proofs and commutative cuts are not needed for that purpose.

Note also that because $\mathtt{catch}_\alpha$ only applies to proofs of $\forall$-$\to$-free formulas, V cannot contain subterms of the form $\lambda a.p$ or $\lambda x.p$ in the last two rules and one is sure that α does not occur in V. This is a crucial point of the design of the system which ensures that when a $\forall$-$\to$-free formula has

been proved using Markov's principle in the empty context, its call-by-value evaluation reveals that the call to Markov's principle is in fact useless.

The operators $\mathtt{catch}$ and $\mathtt{throw}$ behave like the similarly named operators of Nakano [12] or Crolard [13]. Like in [12], but on the contrary of [13] (or of Parigot's $\lambda\mu$-calculus [14] to which the calculus of [13] is equivalent), $\mathtt{catch}$ does not capture its surrounding context (i.e. there is no rule of the form $F[\mathtt{catch}_\alpha p] \to \mathtt{catch}_\beta\, p[\mathtt{throw}_\alpha\,[\,] \leftarrow \mathtt{throw}_\beta\, F[\,]])$. As such, $\mathtt{throw}$ behaves as an exception raiser and $\mathtt{catch}$ as an exception handler but still not as in standard programming languages like Java or ML, since there exceptions are dynamically bound (i.e. the substitution is not capture-free) while in IQC_{MP} they are statically-bound (i.e. the substitution is capture-free)[2]. Alternatively, $\mathtt{catch}_\alpha p$ can be seen as a blocked control operator (i.e. as an expression of the form $\#\,\mathtt{callcc}_\alpha\, p$ where $\#$ is a delimiter [15] that blocks the interaction of $\mathtt{callcc}_\alpha\, p$ with its outside and expects it first to evaluate – to a value – before being observed by the evaluation context of the delimiter).

We now check that the reduction system is compatible with typing.

Theorem 5 (Strengthening): If $\Gamma \vdash_{\Delta,T} V : T$ then $\Gamma \vdash_\Delta V : T$.

Proof: Obvious since the syntax of V refers to no p (and hence to no $\mathtt{catch}$ or $\mathtt{throw}$) as soon as $\to$ and $\forall$ are excluded in T. ∎

Theorem 6 (Subject reduction): If $\Gamma \vdash_\Delta p : A$ and $p \to q$ then $\Gamma \vdash_\Delta q : A$.

Proof: By checking all cases, using Strengthening for the last two rules. Note that since $\mathtt{catch}$, on the contrary of standard classical operators like $\mathtt{callcc}$, does not capture its context, its type remains unchanged and the $\forall$-$\to$ constraint on the formulas of Δ is preserved. ∎

We then characterize the set of normal forms in IQC_{MP}.

Theorem 7 (Characterization of normal forms): The set of typed normal forms for $\to$ corresponds to the entry r

---

[2]Compare substitution with capture
$(\lambda a.\mathtt{catch}_\alpha(a, \mathtt{throw}_\alpha 1))(\mathtt{throw}_\alpha 2) \to \mathtt{catch}_\alpha(\mathtt{throw}_\alpha 2, \mathtt{throw}_\alpha 1)$
to capture-free substitution
$(\lambda a.\mathtt{catch}_\alpha(a, \mathtt{throw}_\alpha 1))(\mathtt{throw}_\alpha 2) \to \mathtt{catch}_\beta(\mathtt{throw}_\alpha 2, \mathtt{throw}_\beta 1)$.

of the following grammar:

$$
\begin{aligned}
r \quad &::= \quad W \mid \mathtt{throw}_\alpha\, W \mid s \mid \mathtt{efq}\, s^+ \mid \mathtt{throw}_\alpha\, s \\
W \quad &::= \quad a \mid \iota_i(W) \mid (W, W) \mid (t, W) \mid \lambda x.r \mid \lambda a.r \mid () \\
s \quad &::= \quad \pi_i(s^+) \mid s^+\, r \mid s^+\, t \mid (\lambda a.r)\, s \\
&\quad \mid \quad \iota_i(s) \mid (s, r) \mid (W, s) \mid (t, s) \\
&\quad \mid \quad \mathtt{case}\, s^+ \,\mathtt{of}\, [a_1.r_1 \mid a_2.r_2] \\
&\quad \mid \quad \mathtt{dest}\, s^+ \,\mathtt{as}\, (x, a) \,\mathtt{in}\, r \\
&\quad \mid \quad \mathtt{catch}_\alpha\, s \mid \mathtt{catch}_\alpha\, \mathtt{efq}\, s^+ \mid \mathtt{catch}_\alpha\, \mathtt{throw}_\beta\, s \\
s^+ \quad &::= \quad s \mid a
\end{aligned}
$$

(we have to add $\mathtt{efq}\, W$ if we want to characterize untyped normal forms.)

Proof: By inspection of the form of proofs that are not reducible. If the normal proof is not a normal value W nor a $\mathtt{throw}_\alpha W$ waiting to erase its context up to its binding $\mathtt{catch}_\alpha$, it is a proof whose computation is blocked by a free variable. Such a normal proof either remains normal when placed into a normal context $F[\ ]$, in which case it has the form s, or it starts with $\mathtt{throw}$ or $\mathtt{efq}$ and captures its surrounding context. A normal proof in s cannot be a variable (otherwise it would be a value) but as soon as an elimination rule is traversed, a variable can occur (entry s^+). ∎

We then check that the reduction system is not too rudimentary and that it at least produces head-normal form on closed proofs.

Theorem 8 (Progress): If $\vdash p : A$ and p is not a (closed) value then p is reducible.

Proof: According to Theorem 7, closed normal forms are necessarily in the set W. But this set is a subset of the set of values. ∎

Theorem 9 (Normalization): If $\Gamma \vdash_\Delta p : A$ then p is normalizable.

Proof: We only give a hint. IQC_{MP} is a subsystem of classical natural deduction equipped with a constrained call-by-value reduction system. Then, its normalization directly derives from the *strong* normalization of classical natural deduction, represented e.g. as call-by-value simply-typed Parigot's $\lambda\mu$-calculus [14] equipped with sums, products and the empty type, and with the appropriate reduction rules: $\mathtt{catch}_\alpha\, p$ is interpreted by $\mu\alpha.[\alpha]p$, $\mathtt{throw}_\alpha\, p$ by $\mu\delta.[\alpha]p$ for δ fresh and $\mathtt{efq}\, p$ by $\mu\delta.[\mathtt{tp}_\perp]p$ where $\mathtt{tp}_\perp$ is the continuation constant associated to the empty type (see [16]). It is unclear whether normalization of this particular system has already been considered in the literature. The closest work seems to be David and Nour [17] who have a normalization proof for a symmetric $\lambda\mu$-calculus. This subsumes the call-by-value case but only implication is considered. Alternatively, we can reduce normalization in natural deduction to normalization in sequent calculus and embed the sequent calculus version of IQC_{MP} into Danos-Joinet-Schellinx (call-by-value) LK_Q [18] which (up to details about the representation of the axiom and contraction rules) is a subsystem of both LK^{tq} [18] (which has cut-elimination as well for

call-by-name as for call-by-value by embedding either into linear logic or into Urban's non-deterministic LK [19]) or $LK_{\mu\tilde\mu}$ [20] (for which a non-deterministic cut-elimination able to capture call-by-value exists [21])[3]. ∎

We are now ready to prove Theorems 3 and 4. Given a proof of $\vdash p : A_1 \vee A_2$, we know by progress and normalization that p eventually reduces to a value V which, by subject reduction, satisfies $\vdash V : A_1 \vee A_2$. By inspection of the possible forms of V, we know that we have either a proof of $\vdash A_1$ or a proof of $\vdash A_2$. Similarly, from $\vdash p : \exists x\, A(x)$ we derive $\vdash V : \exists x\, A(x)$ for some V and hence $\vdash A(t)$ for some term t.

IV. DISCUSSION AND RELATION TO OTHER WORKS

The codereliction of differential proof nets: In terms of polarity in linear logic [23], the $\forall\text{-}\!\to$-free constraint characterizes the formulas of intuitionistic logic that can be built only from positive connectives ($\oplus$, $\otimes$, 0, 1, !) and the why-not connective ("?"). In this framework, Markov's principle expresses that from such a $\forall\text{-}\!\to$-free formula A (e.g. $? \oplus_x (?A(x)\otimes?B(x))$) where the presence of "?" indicates that the proof possibly used weakening ($\mathtt{efq}$ or $\mathtt{throw}$) or contraction ($\mathtt{catch}$), a linear proof of A purged from the occurrences of its "?" connective can be extracted (meaning for the example above a proof of $\oplus_x(A(x) \otimes B(x))$). Interestingly, the removal of the "?", i.e. the steps from $?P$ to P, correspond to applying the codereliction rule of differential proof nets [24].

Nakano and Crolard's intuitionistic $\mathtt{catch}$ *and* $\mathtt{throw}$ *calculi:* Our calculus is very similar to the intuitionistic one proposed by Nakano [12]. However, in [12], the rule of introduction of implication requires Δ to be empty what prevents from deriving Markov's principle. Actually, as expressed in Theorem 5 of [12], the logical expressiveness Nakano's calculus is the one of LJ. Another variant of intuitionistic logic with control operators that does not increase the logical expressiveness can also be found in Crolard [25].

Friedman's A-translation: Expressed in our calculus, Friedman's A-translation [9] maps a proof of $\Gamma \vdash_\Delta A$ in IQC_{MP} to a proof of $\Gamma_\Delta \vdash A_\Delta$ in IQC, where B_Δ is obtained by replacing each atom of B by the disjunction of the formulas in Δ. Through this translation, the THROW rule (assuming w.l.o.g that it is used with atomic conclusions) is interpretable as an injection. Also, for B $\forall\text{-}\!\to$-free, we have $B_\Delta \to B \vee \bigvee \Delta$ from what we see that the CATCH rule is interpretable (the proof of $B_\Delta \to B \vee \bigvee \Delta$ can be logically seen as the property that the proof of a $\forall\text{-}\!\to$ formula B can be purified from all its calls to $\mathtt{throw}$ by locally following a call-by-value reduction strategy; interestingly, when, B is a conjunction, there are two asymmetrical proofs

[3]Another way to prove the normalization of IQC_{MP} without transiting by full classical logic is by embedding into IQC using a variant of Coquand-Hofmann's own variant [22] of Friedman's translation, as done in a work in progress.

of $C_\Delta \wedge D_\Delta \rightarrow (C \wedge D) \vee \bigvee \Delta$ which match the two asymmetrical ways to evaluate a pair along call-by-value reduction). Henceforth, our calculus can be seen as a *direct-style* variant[4] of A-translation in the same way as `callcc` provides with a direct-style representation of continuation-passing-style translation [8], [26].

Independence of premise: The principle of independence of premise (IP) is the scheme $(\neg B \rightarrow \exists x\, A(x)) \rightarrow \exists x\, (\neg B \rightarrow A(x))$. As a rule, the independence of premise principle is admissible in intuitionistic logic [27] but it is not admissible in IQC_{MP} because if it were, taking $A(x)$ an arbitrary atomic formula, one would obtain from MP that $\exists x\, (\neg\neg \exists y\, A(y) \rightarrow A(x))$, then, by intuitionistic reasoning, $\exists x\, (\exists y\, A(y) \rightarrow A(x))$, then, by Theorem 4, $\exists y\, A(y) \rightarrow A(t)$ for some term t. But by Theorem 7, one sees that no normal proof can have this type (the same reasoning holds in Heyting Arithmetic taking for A a formula such that neither $\vdash \neg\neg A$ nor $\vdash \neg A$ holds).

Markov's principle in arithmetic: Since any decidable formula can be expressed in terms of bounded existential quantification, conjunction and disjunction over decidable atoms, and hence as a $\forall$-$\rightarrow$-free formula, we believe that by using an axiom-free presentation of Heyting Arithmetic, one could directly extend IQC_{MP} to the arithmetic case. We would then get a constructive content of $\neg\neg \exists x\, A(x) \rightarrow \exists x\, A(x)$ for $A(x)$ decidable which, in contrast to the realizer that successively checks the truth of each instance of $A(n)$, would not reconstruct a witness from scratch but extract it from the proof. Especially, our constructivization of Markov's principle not only contains its own proof of termination but it also directly evaluates to the witness of the existential quantification.

Modified realizability: Markov's principle is not realizable with respect to Kreisel's modified realizability [28] which therefore does not suit to IQC_{MP}. When interpreting an implication, we have to anticipate the possibility that the function throw an exception. This suggests to introduce a modification of modified realizability where realizability is parametrized by a set Δ of possible $\forall$-$\rightarrow$-free formulas a program is allowed to escape to and to have that t realizes $A \rightarrow B$ with respect to Δ whenever $t\, u$ realizes B with respect to Δ for u realizing A with respect to *any* extension of Δ, using a kind of polymorphism similar to the one Coquand and Hofmann [22] introduced to extend Friedman's A-translation.

Completeness proofs: Gödel and Kreisel proved that completeness for classical predicate logic implies Markov's principle. More precisely, Berardi and Valentini [29] showed that Markov's principle is necessary as soon as the connective $\perp$ of the logic is interpreted as absurdity in the model (a similar phenomenon happens for the completeness of intuitionistic logic for which the interpretation of $\perp$ has to be weakened so as to obtain intuitionistic proofs; see e.g. Veldman [30]). We believe that both completeness for classical logic and for intuitionistic logic could be carried out in an extension of IQC_{MP} to second-order arithmetic without having to weaken the interpretation of $\perp$.

V. CONCLUSION

We showed that adding classical reasoning on $\forall$-$\rightarrow$-free formulas to intuitionistic logic preserves the intuitionistic character of the logic, as witnessed by the preservation of the disjunction and existence properties, while providing with an effective intuitionistic proof of Markov's principle. To compute with Markov's principle, we used a form of statically-bound exception mechanism.

REFERENCES

[1] G. Kreisel, "The non-derivability of ¬ (x) A(x) → (Ex) ¬ A(x), A primitive recursive, in intuitionistic formal systems (abstract)," *The Journal of Symbolic Logic*, vol. 23, no. 4, pp. 456–461, dec 1958.

[2] A. S. Troelstra, *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*, ser. Lecture Notes in Mathematics. Berlin: Springer-Verlag, 1973, vol. 344.

[3] S. C. Kleene, "On the interpretation of intuitionistic number theory," *The Journal of Symbolic Logic*, vol. 10, no. 4, pp. 109–124, 1945.

[4] K. Gödel, "Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes," *Dialectica*, vol. 12, no. 3, pp. 280–287, Dec. 1958.

[5] U. Kohlenbach, *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*, ser. Monographs in Mathematics. Springer, 2008.

[6] W. A. Howard, "The formulae-as-types notion of constructions," in *to H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*. Academic Press, 1980, unpublished manuscript of 1969.

[7] D. Prawitz, *Natural Deduction, a Proof-Theoretical Study*. Almqvist and Wiksell, Stockholm, 1965.

[8] T. G. Griffin, "The formulae-as-types notion of control," in *Conf. Record 17th Annual ACM Symp. on Principles of Programming Languages, POPL '90, San Francisco, CA, USA, 17-19 Jan 1990*. ACM Press, New York, 1990, pp. 47–57.

---

[4]Ongoing work suggests that Friedman's A-translation is more precisely related to a reduction system where exceptions are *dynamically-bound* (instead of *statically-bound*) and where the underlying structure (beside the call-by-value part used to reduce `catch`) is call-by-name. For statically-bound exceptions, the relation has to be sought towards Coquand-Hofmann's variant [22] of Friedman's A-translation.

[9] H. Friedman, "Classically and intuitionistically provably recursive functions," in *Higher Set Theory*, ser. Lecture Notes in Mathematics, D. S. Scott and G. H. Muller, Eds. Berlin/Heidelberg: Springer, 1978, vol. 669, pp. 21–27.

[10] A. G. Dragalin, "New kinds of realizability and Markov's rule," *Soviet Mathematical Doklady*, vol. 251, pp. 534–537, 1980.

[11] U. Berger, "A computational interpretation of open induction," in *19th IEEE Symposium on Logic in Computer Science (LICS 2004), 14-17 July 2004, Turku, Finland, Proceedings*. IEEE Computer Society, 2004, p. 326.

[12] H. Nakano, "A constructive formalization of the catch and throw mechanism," in *Proceedings, Seventh Annual IEEE Symposium on Logic in Computer Science, 22-25 June 1992, Santa Cruz, California, USA*. IEEE Computer Society, 1992, pp. 82–89.

[13] T. Crolard, "A confluent lambda-calculus with a catch/throw mechanism," *J. Funct. Program.*, vol. 9, no. 6, pp. 625–647, 1999.

[14] M. Parigot, "Lambda-mu-calculus: An algorithmic interpretation of classical natural deduction," in *Logic Programming and Automated Reasoning: International Conference LPAR '92 Proceedings, St. Petersburg, Russia*. Springer-Verlag, 1992, pp. 190–201.

[15] M. Felleisen, "The theory and practice of first-class prompts," in *Proceedings of the 15th ACM Symposium on Principles of Programming Languages (POPL '88)*. ACM Press, New York, Jan. 1988, pp. 180–190.

[16] Z. M. Ariola and H. Herbelin, "Minimal classical logic and control operators," in *Thirtieth International Colloquium on Automata, Languages and Programming , ICALP'03, Eindhoven, The Netherlands, June 30 - July 4, 2003*, ser. Lecture Notes in Computer Science, vol. 2719. Springer-Verlag, 2003, pp. 871–885.

[17] R. David and K. Nour, "Arithmetical proofs of strong normalization results for symmetric λ-calculi," *Fundam. Inform.*, vol. 77, no. 4, pp. 489–510, 2007.

[18] V. Danos, J.-B. Joinet, and H. Schellinx, "A new deconstructive logic: Linear logic," *J. Symb. Log.*, vol. 62, no. 3, pp. 755–807, 1997.

[19] C. Urban, "Classical logic and computation," Ph.D. Thesis, University of Cambridge, Oct. 2000.

[20] P.-L. Curien and H. Herbelin, "The duality of computation," in *Proceedings of the Fifth ACM SIGPLAN International Conference on Functional Programming, ICFP 2000, Montreal, Canada, September 18-21, 2000*, ser. SIGPLAN Notices 35(9). ACM, 2000, pp. 233–243.

[21] E. Polonovski, "Subsitutions explicites, logique et normalisation," Ph.D. Thesis, University Paris 7, Jun. 2003.

[22] T. Coquand and M. Hofmann, "A new method for establishing conservativity of classical systems over their intuitionistic version," *Mathematical Structures in Computer Science*, vol. 9, no. 4, pp. 323–333, 1999.

[23] J.-Y. Girard, "Linear logic," *Theor. Comput. Sci.*, vol. 50, pp. 1–102, 1987.

[24] T. Ehrhard and L. Regnier, "Differential interaction nets," *Electr. Notes Theor. Comput. Sci.*, vol. 123, pp. 35–74, 2005.

[25] T. Crolard, "Extension de l'isomorphisme de Curry-Howard au traitement des exceptions," Ph.D. Thesis, University Paris 7, Dec. 1996.

[26] C. Murthy, "Extracting constructive content from classical proofs," Ph.D. Thesis, Cornell University, 1990.

[27] A. Visser, "Aspects of diagonalization and provability," Ph.D. Thesis, University of Utrecht, Department of Philosophy, The Nederland, Nov. 1981.

[28] G. Kreisel, "Interpretation of analysis by means of functionals of finite type," in *Constructivity in Mathematics*, A. Heyting, Ed. North-Holland, 1959, pp. 101–128.

[29] S. Berardi and S. Valentini, "Krivine's intuitionistic proof of classical completeness (for countable languages)," *Ann. Pure Appl. Logic*, vol. 129, no. 1-3, pp. 93–106, 2004.

[30] W. Veldman, "An intuitionistic completeness theorem for intuitionistic predicate logic," *J. Symb. Log.*, vol. 41, no. 1, pp. 159–166, 1976.