The Peirce Translation and the Double Negation Shift

Martín Escardó[1] and Paulo Oliva[2]

[1] University of Birmingham, UK
[2] Queen Mary University of London, UK

Abstract. We develop applications of selection functions to proof theory and computational extraction of witnesses from proofs in classical analysis. The main novelty is a translation of classical minimal logic into minimal logic, which we refer to as the *Peirce translation*, and which we apply to interpret both a strengthening of the double-negation shift and the axioms of countable and dependent choice, via infinite products of selection functions.

1 Introduction

In previous work [5, 7], we investigated *selection functions*

$$\varepsilon \in JA \;\equiv\; ((A \to R) \to A)$$

for *generalised quantifiers*

$$\phi \in KA \;\equiv\; ((A \to R) \to R),$$

where R is a fixed object of generalised truth values. Moreover, we developed various applications to higher-type computability, algorithms, game theory, and proof theory, among others. In this paper, we develop further applications to proof theory. Before discussing the new applications, we introduce background from the above work.

Selection functions in higher-type computability. In [5], the first author considered the particular case where the object A is a domain, and the object R is the domain of boolean values. The particular quantifier ϕ studied was the bounded existential quantifier $\exists_S$ for a subset S of A, with the requirement that $\varepsilon(p)$ be an element of S such that if $p(s)$ holds for some $s \in S$, then $p(\varepsilon(p))$ holds, that is:

$$\phi(p) = p(\varepsilon(p)), \tag{1}$$

for all $p \in A \to R$. The set $S \subseteq A$ is called *exhaustible* if the quantifier $\phi = \exists_S$ is computable, and *searchable* if additionally there is a computable functional $\varepsilon \in JA$ satisfying (1).

It turns out that any searchable set (of total elements) is topologically compact, and, mimicking the Tychonoff theorem from topology, it was shown that searchable sets are closed under countable products. This relies on countable-product functionals of type

$$(JA)^n \to JA^n \qquad\qquad (n \leq \omega).$$

In [7], we considered much more general choices for A and R (objects of a cartesian closed category), and for ϕ (e.g. supremum functional when R are the reals in the category of sets, or in suitable categories of spaces). Again, we required that the selection function ε be related to the quantifier ϕ as in Equation (1). Moreover, we considered the above product in more generality, allowing the object A to vary:

$$\bigotimes : \ \prod_{i<n} JA_i \to J\left(\prod_{i<n} A_i\right) \qquad\qquad (n \leq \omega).$$

The case $n = \omega$ is restricted to a category of continuous maps of certain spaces, which include Kleene–Kreisel spaces of continuous functionals, and requires that R be discrete (e.g. the natural numbers or more generally the types defined in [5, Definition 4.12]) to be well defined.

Selection functions in game theory. Let A_i be the set of possible moves of a sequential game at round i, and let R be the set of possible outcomes of the game. Moreover, let $p\colon \prod_i A_i \to R$ be a function that gives the outcome of a play (or payoff of a profile), and consider quantifiers $\phi_i \in KA_i$ for each round i defining the "goal" for that round (see [7] for details). Finally, assume the quantifiers ϕ_i have associated selection functions $\varepsilon_i \in JA_i$ that choose moves to locally optimise the play, which may be regarded as policy functions. It turns out that product of selection functions calculate optimal plays, profiles in Nash equilibrium, and optimal strategies. As a simple example, for Abelard and Eloise playing in alternating rounds, we take R to be the booleans, and we use universal quantifiers for Abelard, and existential quantifiers for Eloise. If a draw is possible, we instead consider $R = \{-1, 0, 1\}$, and we replace these quantifiers by infimum and supremum functionals respectively. For Nash equilibria, consider $R = \mathbb{R}^n$, with supremum functionals in all rounds (cf. [7]).

Selection functions in bar recursion. Moreover, we showed in [7] that:

1. The infinite case $n = \omega$ of the product of selections functions is the iteration of the finite case $n = 2$,

   $$\otimes\colon JA \times JB \to J(A \times B),$$

   in the sense that

   $$\bigotimes_i \varepsilon_i = \varepsilon_0 \otimes \bigotimes_i \varepsilon_{i+1}.$$

2. This iteration is an instance of the bar recursion scheme.

In the companion paper [6], we establish relations to traditional instances of bar recursion, such as Spector's bar recursion [11] and modified bar recursion [2, 3].

Selection functions in category theory. We also showed in [7] that the construction J over any cartesian closed category gives rise to a strong monad, with a monad morphism into the well-known continuation monad K [9]. The morphism $J \to K$ assigns the quantifier $\phi \in KA$ defined by Equation (1) to any given selection function $\varepsilon \in JA$. Moreover, the case $n = 2$ of the product of selection functions turns out to be simply the canonical map that makes any strong monad into a monoidal monad.

Selection functions in proof theory. We now move to the results developed in this paper. We interpret the objects A and R as logical formulae, and the morphisms as proofs in intuitionistic or minimal logic, or as computable realisers of entailments. For $T = J$ or $T = K$, or more generally any strong monad T, one has the intuitionistic laws

$A \to TA$	(unit)	$T(A \to B) \to TA \to TB$	(functor)
$TTA \to TA$	(multiplication)	$A \wedge TB \to T(A \wedge B)$	(strength).

In the terminology of [1], the construction T is a lax modal operator.

Application to the double negation shift. It turns out that the infinite product of selection functions realises, in the sense of formalised modified realisability, the following shift principle for $T = J$, assuming that R has a discrete type of realisers:

$$T\text{-shift} \quad : \quad \forall n T A(n) \to T \forall n A(n).$$

The well-known double negation shift is the case $T = K$ with $R = \bot$, but it is realised only for special types of formulae A, including those in the image of the negative translation, whereas the J-shift is realised for *all* formulae A. We also show that the double negation shift for formulas A in the image of a negative translation follows from the J-shift. With this, we will get an alternative way of interpreting classical analysis and extracting computational witnesses via infinite products of selection functions.

Application to the elimination of Peirce's law. It is well known that several forms of the negative translation can be understood in terms of the continuation monad K. It is also well known that any monad T gives rise to a translation (see e.g. [1]). Here we consider the T-*translation* inductively defined as

$$\begin{array}{llllll} P^T & = TP & (A \wedge B)^T & = A^T \wedge B^T & (A \vee B)^T & = T(A^T \vee B^T) \\ (\exists x A)^T & = T(\exists x A^T) & (\forall x A)^T & = \forall x A^T & (A \to B)^T & = A^T \to B^T. \end{array}$$

That is, we prefix T in front of atomic formulae, disjunctions and existential quantifications. For $T = K$ and $R = \bot$, this amounts to the standard Gödel-Gentzen negative translation [13], and for $R = A$, with A a Σ^0_1-formula, this corresponds to Friedman's A-translation [8] of the negative translation. From well-known properties of monads on cartesian closed categories, one sees by induction that any C in the image of the T-translation is a T-algebra and in particular $TC \to C$ is provable. Putting this together:

1. $TC \to C$ is provable in minimal logic ML for formulae C in the image of the T-translation.
2. For $T = K$ and $R = \bot$ this principle amounts to double negation elimination.
3. For $T = J$ this is the instance $((C \to R) \to C) \to C$ of Peirce's law, and hence we also refer to the J-translation as the *Peirce translation*.

Because there is a monad morphism $J \to K$, any K-algebra is a J-algebra, which gives the standard fact that the usual negative translations also eliminate Peirce's law. Notice that the implication $JA \to KA$ can be reversed if and only if $R \to A$. In fact, a main difference between the K-translation and the J-translation is that the former also eliminates ex-falso-quodlibet EFQ ($\bot \to A$), whereas the latter is sound with respect to EFQ but does not eliminate it.

Notation. We use X, Y, Z for variables ranging over types. Although in HA^ω one does not have dependent types, we will develop the rest of the paper working with types such as $\Pi_{i \in \mathbb{N}} X_i$ rather than the special case X^ω, when all X_i are the same. The reason for this generalisation is that all results below go through for the more general setting of dependent types. Nevertheless, we hesitate to define a formal extension of HA^ω with dependent types, leaving this to future work. We often write $\Pi_i X_i$ for $\Pi_{i \in \mathbb{N}} X_i$. If x has type X_n and s has type $\Pi_{i=0}^{n-1} X_i$ then $s * x$ is the concatenation of s with x, which has type $\Pi_{i=0}^{n} X_i$.

Acknowledgements. The second author gratefully acknowledges support of the Royal Society (grant 516002.K501/RH/kk).

2 Products of Selection Functions

In this background section we briefly recall some functionals defined and studied in more detail in [6, 7]. Given selection functions $\varepsilon \in JX$ and $\delta \in JY$, define their product $\varepsilon \otimes \delta \in J(X \times Y)$ by

$$(\varepsilon \otimes \delta)(p) = (a, b(a)) \text{ where } b(x) = \delta(\lambda y.p(x, y)) \text{ and } a = \varepsilon(\lambda x.p(x, b(x))).$$

Then, the infinite product functional is defined in [7] by the equation

$$\bigotimes_i \varepsilon_i = \varepsilon_0 \otimes \bigotimes_i \varepsilon_{i+1}.$$

Also, given a selection function $\varepsilon \in JX$ and a family of selection functions $\delta \in X \to JY$, define their *dependent product* $\varepsilon \otimes_d \delta \in J(X \times Y)$ as

$$(\varepsilon \otimes_d \delta)(p) = (a, b(a)) \text{ where } b(x) = \delta(x)(\lambda y.p(x, y)) \text{ and } a = \varepsilon(\lambda x.p(x, b(x))).$$

For $\varepsilon \colon \Pi_{k\in\mathbb{N}}((\Pi_{j<k}X_j) \to (JX_k))$ and $s \colon \Sigma_{k\in\mathbb{N}}(\Pi_{j<k}X_j)$, define the *iterated dependent product of selection functions* IPS as

$$\mathsf{IPS}_s(\varepsilon) \overset{J(\Pi_{i=k}^{\infty} X_i)}{=} \varepsilon_s \otimes_d \lambda x^{X_k}.\mathsf{IPS}_{s*x}(\varepsilon).$$

The recursive definitions for $\bigotimes$ and IPS uniquely define functionals in the models of partial and total continous functionals (cf. [7]). Finally, we remark that $\bigotimes$ and IPS are actually inter-definable over system T, as stated in [6].

3 Shift Principles, Countable Choice, and Dependent Choice

In this section we investigate Heyting arithmetic (HA) and classical extensions of HA induced by the monads $T = J$ and $T = K$, as discussed in the introduction. Given a formal system S we write S^ω for the finite type generalisation of S with a neutral treatment of equality (cf. [12]). Before specialising to the cases of interest, we consider the general T-translation for an arbitrary strong monad T.

We refer as *T-logic* to the extension of intuitionistic logic with the *T-elimination axiom* $TA \to A$. As such, classical logic amounts to K-logic if we choose $R = \perp$ in the definition of K. Using this language, the discussion of the introduction shows that the T-translation eliminates the T-elimination axiom, thereby mapping T-logic into intuitionistic logic (see Section 4 below for a sharper analysis of this fact). As is well known, in the case $T = K$ with $R = \perp$, this mapping actually lands in minimal logic ML, that is, intuitionistic logic without EFQ.

We refer as *T-arithmetic* (TA) to the extension of HA with T-logic. Then Peano arithmetic (PA) is K-arithmetic for $R = \perp$. If a formula does not have occurrences of disjunction or existential quantification, its T-translation only prefixes T to atomic formulae, and hence the T-translations of the Peano axioms follow from the Peano axioms. This shows that the T-translation maps TA into HA.

However, the T-translation does not map $\mathsf{TA}^\omega + \mathsf{AC}_\mathbb{N}$ into $\mathsf{HA}^\omega + \mathsf{AC}_\mathbb{N}$, where $\mathsf{AC}_\mathbb{N}$ is the axiom of countable choice

$$\mathsf{AC}_\mathbb{N} \quad : \quad \forall n^\mathbb{N} \exists x^X A(n, x) \to \exists f \forall n\, A(n, fn),$$

and this failure applies to the particular cases $T = J$ and $T = K$ too. In fact, the T-translation of $\mathsf{AC}_{\mathbb{N}}$ is

$$\mathsf{AC}^T_{\mathbb{N}} \quad : \quad \forall n T \exists x A^T(n,x) \to T \exists f \forall n A^T(n, fn),$$

which is not an instance of $\mathsf{AC}_{\mathbb{N}}$. In order to overcome this, the following was first observed by Spector [11] for the special case $T = K$ and $R = \bot$, where

$$T\text{-}\mathsf{shift}(A) \quad : \quad \forall n^{\mathbb{N}} T A(n) \to T \forall n A(n).$$

Proposition 1. $\mathsf{AC}_{\mathbb{N}}$ *and* T-shift *together imply* $\mathsf{AC}^T_{\mathbb{N}}$*. Hence, the T-translation maps* $\mathsf{TA}^\omega + \mathsf{AC}_{\mathbb{N}}$ *into* $\mathsf{HA}^\omega + \mathsf{AC}_{\mathbb{N}} + T\text{-}\mathsf{shift}$.

Proof Applying T-shift to the premise $\forall n T \exists x A^T(n,x)$ of $\mathsf{AC}^T_{\mathbb{N}}$, we deduce that $T \forall n \exists x A^T(n,x)$. Functoriality of T applied to $\mathsf{AC}_{\mathbb{N}}$ with A instantiated to A^T gives that $T \forall n \exists x A^T(n,x)$ implies $T \exists f \forall n\, A^T(n, fn)$, and hence we get $T \exists f \forall n A^T(n, fn)$ by modus ponens, which is the conclusion of $\mathsf{AC}^T_{\mathbb{N}}$. $\square$

Spector, in the context of the dialectica interpretation, showed that a form of bar recursion, now known as *Spector bar recursion*, realises the *double negation shift* (DNS), which amounts to the T-shift for $T = K$ and $R = \bot$. Moreover, via different forms of bar recursion with R a Σ^0_1 formula, it is shown in [2, 3] how computational information can also be extracted via (modified) realisability from proofs in classical analysis in the presence of countable choice. But the K-shift is established only for formulae $\exists x A^K$ where A^K is in the image of the K-translation. Now notice that any formula A^K we have $\bot \to \exists x A^K$.

Proposition 2. *Over minimal logic, if* $R \to A$ *then* $J\text{-}\mathsf{shift}(A) \to K\text{-}\mathsf{shift}(A)$.

Proof We know that $JA \to KA$ for any A, and the assumption $R \to A$ is easily seen to give the converse, and hence $JA \leftrightarrow KA$. (Moreover, notice that if $KA \to JA$ holds then $R \to A$, and hence the assumption $R \to A$ is optimal.) $\square$

Hence the following gives an alternative way of realising the K-shift for the purposes of extracting witnesses from classical proofs with countable choice. The notions in the assumptions of the following theorem are defined in [3, 12].

Theorem 1. *Assuming continuity and relativised bar induction, the infinite product of selection functions* $\bigotimes$ *modified-realises* $J\text{-}\mathsf{shift}(A)$ *for any A, provided the parameter R in the definition of J is a formula with a discrete type of realisers.*

We omit the proof for lack of space, but we fully prove a stronger result in Section 5. The restriction on R is needed for the infinite product to be well-defined [7], and notice that it is fulfilled if R is Σ^0_1 or a Harrop formula.

We emphasise that the above theorem states that the infinite product functional *itself* realises the shift principle. This is in contrast with the work discussed above, where the bar recursive functionals in question are *used in order to define* functionals that realise shift principles, but do not realise the shift principles themselves as they do not have the required types. We regard as rather striking the fact that a functional that was originally introduced to mimic a theorem from topology in a computational setting, as discussed in the introduction, turns out to have a natural logical reading related to traditional work in proof theory, and we think that this deserves further investigation. In summary, the J-shift turns out to be a logical analogue of the Tychonoff theorem from topology.

We now compare TA^ω and HA^ω with respect to the axiom of *dependent choice*

$$\mathsf{DC}_X \quad : \quad \forall n^{\mathbb{N}}, x^X \exists y^X A_n(x,y) \to \forall x_0 \exists \alpha(\alpha_0 = x_0 \wedge \forall n A_n(\alpha_n, \alpha_{n+1})).$$

Proposition 3. $\mathsf{DC}_{\mathbb{N}}$ *and T-shift together imply* $\mathsf{DC}^T_{\mathbb{N}}$*. Hence, the T-translation maps* $\mathsf{TA} + \mathsf{DC}_{\mathbb{N}}$ *into* $\mathsf{HA} + \mathsf{DC}_{\mathbb{N}} + T\text{-}\mathsf{shift}$.

Proof The argument is essentially the same as that of Proposition 1, but one applies the T-shift twice, to move T outside two numerical universal quantifiers. □

In general, however, when X is a higher-type, the situation is subtler, because the T-shift will not be available for $T = J$ (let alone $T = K$). The case $T = K$ has been addressed in [2, 3], and in Section 5 below we address the case $T = J$ (which has the case $T = K$ as a corollary).

Proposition 4. *The* T-shift *principle is equivalent to*

$$\forall n(\forall k < n\, A(k) \to TA(n)) \to T\forall n A(n),$$

which we will refer to as the course-of-values T-shift.

Proof It is straightforward that this condition implies the T-shift. Conversely, assume $\forall n(\forall k < n\, A(k) \to TA(n))$. By the extension law $(B \to TC) \to (TB \to TC)$ of strong monads in a cartesian closed category and induction on n, we deduce that $\forall n(\forall k < n\, TA(k) \to TA(n))$. Hence $\forall n TA(n)$ by course-of-values induction, and the T-shift gives the desired result. □

Theorem 2. $\mathsf{IPS}_{\langle\rangle}$ *modified-realises the course-of-values* J-shift(A) *for any A, provided the parameter R in the definition of J has a discrete type of realisers.*

In Section 5 we show that IPS also realises a more general logical principle that implies DC^J_X for any type X, not just $X = \mathbb{N}$ as above.

4 Extraction of Witnesses via the J-translation

Let ML stand for *minimal logic*[3] and consider the T-elimination scheme

$$T\text{-}\mathsf{elim} \quad : \quad TA \to A.$$

As discussed above, for $T = J$, this is the instance $((A \to R) \to A) \to A$ of Peirce's law. If $R = \perp$, the proof system $\mathsf{ML} + J\text{-}\mathsf{elim} + \mathsf{EFQ}$ amounts to full first-order classical logic CL. For R arbitrary, the J-translation is such that the translated instance of Peirce's law $JA \to A$ becomes provable in minimal logic. More generally:

Lemma 1. *For any strong monad T:*

1. $\mathsf{ML} \vdash TA^T \to A^T$.
2. $\mathsf{ML} + T\text{-}\mathsf{elim} \vdash A^T \to A$.
3. $\mathsf{ML} + T\text{-}\mathsf{elim} \vdash A$ *if and only if* $\mathsf{ML} \vdash A^T$.

These facts are well known (see e.g. [1]) and are easily proved by induction on formulae, although they are usually stated for intuitionistic logic rather than minimal logic.

[3] Intuitionistic logic without the ex-falso-quodlibet axiom scheme $\mathsf{EFQ} : \perp \to A$ (see e.g. [13]).

Theorem 3. *Assume that $P(x,y) \to R$ and that the variable y is not free in R. If*

$$\mathsf{ML} + J\text{-}\mathsf{elim} \vdash \forall x \exists y P(x,y)$$

then also $\mathsf{ML} \vdash \forall x \exists y P(x,y)$.

Proof First notice that under the assumption $P(x,y) \to R$ we have

(i) $\mathsf{ML} \vdash JP(x,y) \to P(x,y)$,

(ii) $\mathsf{ML} \vdash J\exists y P(x,y) \to \exists y P(x,y)$.

If $\mathsf{ML} + J\text{-}\mathsf{elim} \vdash \forall x \exists y P(x,y)$ then $\mathsf{ML} + J\text{-}\mathsf{elim} \vdash \exists y P(x,y)$, and hence Lemma 1 gives $\mathsf{ML} \vdash J\exists y JP(x,y)$, which by (i) and (ii), implies that $\mathsf{ML} \vdash \exists y P(x,y)$. □

The first part of the next proposition shows that if multiple instances of J-elimination are used in a proof, for different parameters R, one can reduce to a single instance with the conjunction of all the parameters. For example, this can be applied to the above theorem if one needs to use several instances of Peirce's law. The second part shows that the J- and K-translations coincide over intuitionistic logic.

Proposition 5.

1. $\mathsf{ML} + J_{R_0 \wedge R_1}\text{-}\mathsf{elim} \vdash J_{R_0}\text{-}\mathsf{elim} \wedge J_{R_1}\text{-}\mathsf{elim}$.
2. *For $R \equiv \perp$ we have that* $\mathsf{ML} + \mathsf{EFQ} \vdash A^K \leftrightarrow A^J$.

Proof The first part is routine verification. The second part follows from Proposition 2. □

The following theorem (cf. Proposition 1 of [3]) shows how one can extract witnesses from proofs of Π^0_2-statements in classical analysis via the J-translation and the J-shift (as opposed to via the negative translation and the double negation shift).

Theorem 4. *If* $\mathsf{PA}^\omega + \mathsf{AC}_\mathbb{N} + \mathsf{DC}_\mathbb{N} \vdash \forall x^X \exists n^\mathbb{N} P(x,n)$ *then one can extract a term t in system T extended with the product functional $\bigotimes$ such that $P(x,tx)$.*

Proof Write $C = \mathsf{AC}_\mathbb{N} \wedge \mathsf{DC}_\mathbb{N}$. By prefixing each atomic formula with a double negation, EFQ is eliminated. Hence the assumption of the theorem implies that $\mathsf{MA}^\omega + J_\perp\text{-}\mathsf{elim} + C \vdash \forall x \exists n \neg\neg P(x,n)$. Because the proof is in ML, we can replace $\perp$ by any formula, which we take to be R:

$$\mathsf{MA}^\omega + J_R\text{-}\mathsf{elim} + C \vdash \forall x \exists n (P(x,n) \to R) \to R.$$

If we now take $R \equiv \exists n P(x,n)$, we conclude that $\mathsf{MA}^\omega + J_R\text{-}\mathsf{elim} + C \vdash \forall x \exists n P(x,n)$. By the J-translation we have $\mathsf{MA}^\omega + C^J \vdash \forall x J \exists n JP(x,n)$, and, by the choice of R, $\mathsf{MA}^\omega + C^J \vdash \forall x \exists n P(x,n)$. We are now done because C^J follows, in $\mathsf{MA}^\omega + C$, from J-shift, which, by Theorem 1, is realised by $\bigotimes$, and because $\mathsf{AC}_\mathbb{N}$ and $\mathsf{DC}_\mathbb{N}$, and hence C, are also realised. □

5 Full Dependent Choice

Recall that it has been standard to interpret the axiom of *countable* choice computationally by reducing it to the computational interpretation of the double negation shift (cf. [2, 3, 11] and Theorem 4 above). When it comes to the computational interpretation of the *dependent* choice, one normally does it directly, as it is seems not possible to reduce the negative translation of DC using the simple double negation shift. In this section, continuing the discussion started in Section 3, we show that what is needed in order to approach this from a logical point of view is a *dependent* variant of the shift principle. The binary version of the course-of-values J-shift (cf. Proposition 4) is

$$JA \wedge (A \to JB) \to J(A \wedge B).$$

Now, let us consider a dependent version of this, where JB depends on the witness for A:

$$J\exists x A(x) \wedge \forall x \in A\, J\exists y B(x,y) \to J\exists x, y(A(x) \wedge B(x,y)).$$

Using finite sequences, this can be generalised to an arbitrary finite number of predicates:

$$\bigwedge_{i=0}^{n} \forall s \in (\bigwedge_{j=0}^{i-1} A_j)\, J\exists x_i A_i(s * x_i) \to J\exists t \bigwedge_{i=0}^{n} A_i(t_0, \ldots, t_i),$$

where $\bigwedge_{i=0}^{n}$ and $\bigwedge_{j=0}^{i-1}$ stand for bounded universal quantifications. Generalising this further to the case of infinitely many predicates, we have

$$J^d\text{-}\mathsf{shift} \quad : \quad \forall s \in (\bigwedge_{j=0}^{|s|-1} A_j)\, J\exists x A_{|s|}(s * x) \to J\exists \alpha \forall n A_n([\alpha](n+1)),$$

which we call the *dependent* J-shift. Note that $[\alpha](n)$ stands for the initial segment of the infinite sequence α of length n, i.e. $[\alpha](n) = \langle \alpha(0), \alpha(1), \ldots, \alpha(n-1) \rangle$.

We show now how the computational content of (the classical) dependent choice can be reduced to that of the dependent J-shift. More precisely, the dependent J-shift together with dependent choice proves the J-translation of dependent choice (as with countable choice). We use the following variant of *dependent choice* based on finite sequences [10, Section 2.3]:

$$\mathsf{SDC} \ : \ \forall s(\forall j < |s|\, A_j([s](j)) \to \exists x A_{|s|}(s * x)) \to \exists \alpha \forall n A_n([\alpha](n+1)).$$

Lemma 2. *The following are provable in* MA^ω*:*

1. $\mathsf{SDC} \vdash \mathsf{DC}$.
2. $\mathsf{SDC} + J^d\text{-}\mathsf{shift} \vdash \mathsf{SDC}^J$.

Theorem 5. *Let R be a Σ^0_1-formula. Assuming continuity and relativised bar induction,* $\mathsf{IPS}_{\langle\rangle}$ *modified-realises* J^d-shift.

Proof Assume the realiser for $\exists y^{Y_n} A_n(s * y)$ has type $X_n(s) \equiv Y_n \times Z_n(s)$. Also, assume

ε_s mr $\forall j < |s|\, A_j([s](j)) \to J\exists y A_{|s|}(s * y)$
q mr $\exists \alpha \forall n A_n([\alpha](n+1)) \to R$.

Then ε_s and q have types

$$\Pi_{j<|s|} \Pi_{z_j \in Z_j(s_0,\ldots,s_j)} JX_{|s|}(s) \quad \text{and} \quad \Sigma_{\alpha \in \Pi_k Y_k} \Pi_n Z_n([\alpha](n+1)) \to R,$$

respectively. We prove $\mathsf{IPS}_{\langle\rangle}(\varepsilon)(q)$ mr $\exists \alpha \forall n A_n([\alpha](n+1))$. We proceed by relativised bar induction (cf. [3]) to prove $\forall s P(s)$, where

$P(s) \equiv \mathsf{IPS}_s(\varepsilon)(q_s) \,\mathsf{mr}\, \exists\alpha\forall n\, A_{|s|+n}(s^0 * [\alpha](n+1))$.

Note that s is a finite sequence of pairs. We write s^0 (respectively, s^1) for the finite sequence consisting of the first (respectively, second) component of each pair. The bar induction will be relativised to the predicate

$R(s) \equiv \forall j < |s|\, (s^1_j \,\mathsf{mr}\, A_j(\langle s^0_0, \ldots, s^0_{j-1}\rangle))$.

We now prove the two hypothesis (i) and (ii) of the bar induction.

(i) $\forall \alpha^R \exists k P([\alpha](k))$. Given α, pick k to be a point of continuity of q on α. We must show $P([\alpha](k))$, i.e.

$$\mathsf{IPS}_{[\alpha](k)}(\varepsilon)(q_{[\alpha](k)}) \,\mathsf{mr}\, \exists\beta\forall n\, A_{k+n}(([\alpha](k))^0 * [\beta](n+1)).$$

Abbreviate $\gamma, \delta \equiv \mathsf{IPS}_{[\alpha](k)}(\varepsilon)(q_{[\alpha](k)})$. The above follows from, for all n,

$$\delta(n) \,\mathsf{mr}\, A_{k+n}(([\alpha](k))^0 * [\gamma](n+1)).$$

Unfolding the definition of IPS, this is equivalent to, for all n,

$$(\varepsilon_{[\alpha](k)*r}(\lambda x.q_{[\alpha](k)*r*x}(\mathsf{IPS}_{[\alpha](k)*r*x}(\varepsilon)(q_{[\alpha](k)*r*x}))))_1 \,\mathsf{mr}\, A_{k+n}(([\alpha](k))^0 * [\gamma](n+1)),$$

where $r = \mathsf{IPS}_{[\alpha](k)}(\varepsilon)(q_{[\alpha](k)})[k, k+n-1]$ (computed by course-of-values). By the fact that k is a point of continuity of q on α, this is equivalent to

$$(\varepsilon_{[\alpha](k)*r}(\lambda x.q_{[\alpha](k)*r*x}(\mathbf{0})))_1 \,\mathsf{mr}\, A_{k+n}(([\alpha](k))^0 * [\gamma](n+1)).$$

By course-of-values induction $[\alpha](k) * r \in R$, Hence, by the assumption on ε it remains to show that

$$\lambda x.q_{[\alpha](k)*r*x}(\mathbf{0}) \,\mathsf{mr}\, \exists y A_{k+n}(([\alpha](k))^0 * [\gamma](n) * y) \to R$$

which follows from the assumptions on q, and that $\alpha \in R$.

(ii) $\forall s^R(\forall t, x(R(s * t * x) \to P(s * t * x)) \to P(s))$. Let $s \in R$ be given, and assume (1) $\forall t, x(R(s * t * x) \to P(s * t * x))$. We must show $P(s)$, i.e.

$$\mathsf{IPS}_s(\varepsilon)(q_s) \,\mathsf{mr}\, \exists\alpha\forall n A_{|s|+n}(s^0 * [\alpha](n+1)).$$

Again let $\gamma, \delta \equiv \mathsf{IPS}_s(\varepsilon)(q_s)$. Then, $P(s)$ follows from

$$(\mathsf{IPS}_s(\varepsilon)(q_s)(n))_1 \,\mathsf{mr}\, A_{|s|+n}(s^0 * [\gamma](n+1)),$$

which, by the definition of IPS is

$$(\varepsilon_{s*r}(\lambda x.q_{s*r*x}(\mathsf{IPS}_{s*r*x}(\varepsilon)(q_{s*r*x}))))_1 \,\mathsf{mr}\, A_{|s|+n}(s^0 * [\gamma](n+1)),$$

where $r = \mathsf{IPS}_s(\varepsilon)(q_s)[|s|, |s|+n-1]$. This follows from

$$(2)\ \lambda x.q_{s*r*x}(\mathsf{IPS}_{s*r*x}(\varepsilon)(q_{s*r*x})) \,\mathsf{mr}\, \exists x_n A_{|s|+n}(s^0 * [\gamma](n) * x_n) \to R.$$

Now, assume x is such that $R(s * r * x)$. Then, by (1) we have, $P(s * r * x)$, i.e.

$$(3)\ \mathsf{IPS}_{s*r*x}(\varepsilon)(q_{s*r*x}) \,\mathsf{mr}\, \exists\alpha\forall n A_{|s*r*x|+n}((s * r * x)^0 * [\alpha](n+1)).$$

By the assumption on q we have that (3) implies (2), which concludes the proof. $\square$

Corollary 1. *If* $\mathsf{PA}^\omega + \mathsf{AC}_\mathbb{N} + \mathsf{SDC} \vdash \forall x^X \exists n^\mathbb{N} P(x,n)$ *then one can extract a term* t *in system* T *extended with* $\bigotimes$ *such that* $P(x,tx)$.

Proof $\mathsf{AC}_\mathbb{N}$ and SDC are modified-realizable in system T. The result follows because $\bigotimes$ is inter-definable with IPS (cf. [6]), and hence the J^d-shift is modified-realizable in $T + \bigotimes$. $\square$

6 Concluding remarks

We have developed a proof translation based on the selection monad J, and shown how to realise a corresponding J-shift principle, which is more general than the double negation shift. We plan to investigate the use of the product of selection functions $\bigotimes$ for extraction of computational content from proofs involving countable/dependent choice, as done by Seisenber [10] with modified bar recursion. Based on the experimental results and theoretical conjectures of [4] and [5, Section 8.10], we wish to investigate whether $\bigotimes$ would give rise to more efficient computational extraction of witnesses.

References

1. P. Aczel. The Russell-Prawitz modality. *Math. Structures Comput. Sci.*, 11(4):541–554, 2001. Modalities in type theory (Trento, 1999).
2. S. Berardi, M. Bezem, and T. Coquand. On the computational content of the axiom of choice. *The Journal of Symbolic Logic*, 63(2):600–622, 1998.
3. U. Berger and P. Oliva. Modified bar recursion and classical dependent choice. *Lecture Notes in Logic*, 20:89–107, 2005.
4. M. H. Escardó. Infinite sets that admit fast exhaustive search. In *Proceedings of LICS*, pages 443–452, 2007.
5. M. H. Escardó. Exhaustible sets in higher-type computation. *Logical Methods in Computer Science*, 4(3), 2008.
6. M. H. Escardó and P. Oliva. Computational interpretations of analysis via products of selection functions. In F. Ferreira, B. Lowe, E. Mayordomo, and L. M. Gomes, editors, *Computability in Europe 2010*. LNCS, Springer, 2010.
7. M. H. Escardó and P. Oliva. Selection functions, bar recursion, and backward induction. *Mathematical Structures in Computer Science*, 20(2):127–168, 2010.
8. H. Friedman. Classically and intuitionistically provably recursive functions. In D. Scott and G. Müller, editors, *Higher Set Theory*, volume 669 of *Lecture Notes in Mathematics*, pages 21–28. Springer, Berlin, 1978.
9. T. G. Griffin. A formulae-as-types notion of control. In *17th Annual ACM Symp. on Principles of Programming Languages, POPL'90, San Francisco, CA, USA*, pages 17–19, 1990.
10. M. Seisenberger. Programs from proofs using classical dependent choice. *Annals of Pure and Applied Logic*, 153(1–3):97–110, 2008.
11. C. Spector. Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles in current intuitionistic mathematics. In F. D. E. Dekker, editor, *Recursive Function Theory: Proc. Symposia in Pure Mathematics*, volume 5, pages 1–27. American Mathematical Society, Providence, Rhode Island, 1962.
12. A. S. Troelstra. *Metamathematical Investigation of Intuitionistic Arithmetic and Analysis*, volume 344 of *Lecture Notes in Mathematics*. Springer, Berlin, 1973.
13. A. S. Troelstra and H. Schwichtenberg. *Basic Proof Theory*. Cambridge University Press, Cambridge (2nd edition), 2000.