A Computational Interpretation of the Axiom of Determinacy in Arithmetic

Takanori Hida

Research Institute for Mathematical Sciences, Kyoto University
Kitashirakawa Oiwakecho, Sakyo-ku, Kyoto 606-8502, Japan
hida@kurims.kyoto-u.ac.jp

Abstract

We investigate the computational content of the axiom of determinacy (AD) in the setting of classical arithmetic in all finite types with the principle of dependent choices (DC). By employing the notion of realizability interpretation for arithmetic given by Berardi, Bezem and Coquand (1998), we interpret the negative translation of AD. Consequently, the combination of the negative translation with this realizability semantics can be seen as a model of DC, AD and the negation of the axiom of choice at higher types. In order to understand the computational content of AD, we explain, employing Coquand's game theoretical semantics, how our realizer behaves.

1998 ACM Subject Classification F.4.1 Mathematical Logic

Keywords and phrases The axiom of determinacy, Gale-Stewart's theorem, Syntactic continuity, Realizability interpretation, Coquand's game semantics

Digital Object Identifier 10.4230/LIPIcs.CSL.2012.335

1 Introduction

The theory of infinite games has proven to be very effective in the study of various fields of logic and mathematics. There are a number of related works, and lots of game concepts have been proposed. Prominent among infinite game theory is the *two person infinite game with perfect information*, in which two players collaborate to define an infinite sequence of natural numbers by choosing a natural number alternately. There are many intriguing questions over this game, e.g., which games can be shown to be *determined*, in the sense that one of the two players has a winning strategy? D. Gale and F.M. Stewart [6] proved that all games for open or closed pay-off sets are determined. As the study of determinacy has revealed several remarkable consequences to mathematics, the axiom of determinacy (AD, for short) was introduced out of theoretical interest [17]: For every subset A of the Baire space ω^ω, the game $G(A)$ is determined. A substantial amount of research has been conducted over this topic and a number of deep results have been obtained (see [8]).

The focus of this paper, however, is on a somewhat different aspect from prior set-theoretical ones: what is the computational content of AD? For this purpose, we employ the notion of realizability interpretation for arithmetic given in [2]. Realizability interpretation, which is one formalization of BHK-interpretation, assigns a term to a valid formula. A realizer of a formula provides computational evidence for that formula, and thus endows it with computational content. Although there are other techniques for program extraction from formal proofs such as Curry-Howard correspondence [19], the realizability interpretation is better suited for our purpose: This methodology enables us to give interpretations even for some *non-trivial axioms* and *proofs using axioms*, because the definition of the realizability relation proceeds by induction not on *proofs* but on *formulas*.

© Takanori Hida;
licensed under Creative Commons License NC-ND

Computer Science Logic 2012 (CSL'12).
Editors: Patrick Cégielski, Arnaud Durand; pp. 335–349
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Several notions of realizability interpretation have been presented [21]. Nevertheless, in terms of the usage of realizability for the exploration of the computational content of classical proofs, existing research can be classified into two categories: direct and indirect approaches. The former studies computational meanings immediately in the setting of classical logic. The fact that such an approach is possible comes as rather a surprise; from the viewpoint of computation, classical logic is much more difficult to deal with than intuitioninstic logic. Among this line of research, Krivine's classical realizability is of great importance [12, 13, 14]. This technique is developed as a generalization of forcing and, using the orthogonal structure between terms and stacks, provides a way of examining computational content of classical logic. On the other hand, the indirect approach, which we follow in this paper, consists of two steps. The first step is to embed classical proofs into an intuitionistic system by a negative translation; then by using some notion of intuitionistic realizability, we interpret the translated proofs.

The rest of this paper is organized as follows. In the next section, we define the basic terminology and provide background information briefly. In order to realize a negative translation of AD, we formalize the statement of Gale-Stewart's theorem in arithmetic and prove it in section 3. After presenting the notion of realizability in section 4, we interpret the negative translation of the statement "every subset of ω^ω is open" and, as a consequence, obtain a required realizer in section 5. Corollaries of this result include the (relative) consistency of the principle of dependent choices and AD in arithmetic. Since our purpose is not just to give a realizer of AD but also to know its computational meaning, we explain the behavior of the realizer using Coquand's game theoretical semantics in section 6. In the final section, we discuss future work.

2 Notations and Definitions

2.1 Infinite Games

▶ **Definition 1.** (Two person, infinite game with perfect information)

For an arbitrary subset A of the set ω^ω of all infinite sequences of natural numbers, G(A) denotes the following game:

- There are two players, Player I and Player II,
- At each round i, Player I chooses an $x_i \in \omega$, then Player II chooses a $y_i \in \omega$,

I	x_0		x_1		$\cdots$		x_i		$\cdots$	
II		y_0		y_1		$\cdots$		y_i		$\cdots$

- Player I wins the game $G(A)$ if the infinite sequence $\langle x_0, y_0, x_1, y_1, \ldots \rangle$ is in A.

Each choice is called a *move* of the game, and the infinite sequence $\langle x_0, y_0, x_1, y_1, \ldots \rangle$ is called a *play* of the game. We refer to A as the *pay-off set* for the game $G(A)$. *Perfect information* means that both players have complete access to the way the game has been played so far.

▶ **Definition 2.** (Strategies)

A strategy for Player I is a function $\sigma : \{s \in \omega^{<\omega} \mid s \text{ is of even length}\} \to \omega$,

A strategy for Player II is a function $\tau : \{s \in \omega^{<\omega} \mid s \text{ is of odd length}\} \to \omega$,

where $\omega^{<\omega}$ is the set of all finite sequences of natural numbers.

Each player decides his or her move according to a strategy as follows:

▶ **Definition 3.** (The plays $\sigma * y$ and $x * \tau$)

Let σ be a strategy for Player I. For each $y = \langle y_0, y_1, \dots \rangle$, $\sigma * y$ denotes the play $\langle a_0, y_0, a_1, y_1, \dots \rangle$, where $a_0 = \sigma(\langle\rangle)$ and $a_{n+1} = \sigma(\langle a_0, y_0, \dots, a_n, y_n \rangle)$.

Let τ be a strategy for Player II. For each $x = \langle x_0, x_1, \dots \rangle$, $x * \tau$ denotes the play $\langle x_0, b_0, x_1, b_1, \dots \rangle$, where $b_n = \tau(\langle x_0, b_0, \dots, x_n \rangle)$.

▶ **Definition 4.** (Winning strategies)

A strategy σ is a winning strategy for Player I in $G(A)$ if $\sigma * y \in A$ for all $y \in \omega^\omega$.

A strategy τ is a winning strategy for Player II in $G(A)$ if $x * \tau \notin A$ for all $x \in \omega^\omega$.

▶ **Definition 5.** (Determined)

A game $G(A)$ is determined if either Player I or Player II has a winning strategy in this game.

A set $A \subset \omega^\omega$ is determined if the game $G(A)$ is determined.

One natural question over this property would be: How much determinacy is derivable? It is easy to see that all finite and cofinite subsets are determined. More interestingly, it has been proven that all open and closed subsets [6], and all Borel subsets [15] are determined. (Recall that the standard topology on ω^ω is induced by an open base $\{O(s) \mid s \in \omega^{<\omega}\}$, where $O(s) := \{f \in \omega^\omega \mid s \text{ is an initial segment of } f\}$. This space is called the *Baire space*).

▶ **Definition 6.** (The axiom of determinacy (AD))

The axiom of determinacy (AD) is the statement that every $A \subset \omega^\omega$ is determined.

The relationship between AD and choice principles is worth pointing out: AD contradicts the (full) axiom of choice (AC) in ZF set theory[6], but implies a restricted version of the axiom of countable choice [18]. As regards the principle of dependent choices (DC), which is an essential tool in exploring the consequences of AD, it is known that DC is independent from ZF+AD [9].

There are a number of striking results around AD, such as its role in the study of consistency strength and applications to infinite combinatorics. The investigation of determinacy extends even to the area of second order arithmetic, e.g., [16]. The reader can find more information in, e.g., [8].

2.2 Systems of Arithmetic

In order to investigate AD in arithmetic, let us fix the basic terminology of arithmetic and present fundamental results. Firstly, we describe minimal (HA^ω_-), intuitionistic (HA^ω) and classical (HA^ω_c) arithmetic in all finite types. We borrow most of our notation from [2].

▶ **Definition 7.** (Formal systems HA^ω_-, HA^ω and HA^ω_c)
Types, terms and formulas of the three systems are the same and given by the following grammars:

Types $\tau, \tau' ::= \mathtt{N} \mid \tau \to \tau'$

Terms $t, u ::= x^\tau \mid \lambda x^\tau . t^{\tau'} \mid t^{\tau \to \tau'} u^\tau \mid \mathtt{0}^{\mathtt{N}} \mid \mathtt{s}^{\mathtt{N} \to \mathtt{N}} \mid \mathtt{Rec}_\tau^{\tau \to ((\mathtt{N} \to \tau \to \tau) \to (\mathtt{N} \to \tau))}$

where t^τ or $t : \tau$ indicate that a term t is of type τ.

Formulas $\phi, \psi ::= \bot \mid t^{\mathtt{N}} = t'^{\mathtt{N}}$ (prime formula) $\mid \phi \wedge \psi \mid \phi \Rightarrow \psi \mid \forall x : \tau\, \phi \mid \exists x : \tau\, \phi$

For every formula ϕ, we write $\neg\,\phi$ in place of $\phi \Rightarrow \bot$ for brevity.

Higher type equations are abbreviations, e.g., $f^{\mathtt{N} \to \mathtt{N}} = g^{\mathtt{N} \to \mathtt{N}}$ stands for $\forall n : \mathtt{N}\, (fn = gn)$.

Theory of HA^ω_-

- Axioms and rules for first order many sorted minimal logic (with each sort corresponding to a type).
- Equality axioms and the induction schema:

$$t = t \qquad (eq_1)$$
$$t_1 = s_1 \Rightarrow \cdots \Rightarrow t_k = s_k \Rightarrow ft_1 \cdots t_k = fs_1 \cdots s_k \qquad (eq_2)$$
$$t_1 = s_1 \Rightarrow \cdots \Rightarrow t_k = s_k \Rightarrow \{P(t_1, \ldots, t_k) \Leftrightarrow P(s_1, \ldots, s_k)\} \qquad (eq_3)$$
$$\phi(\mathtt{0}) \wedge \forall n \{\phi(n) \Rightarrow \phi(\mathtt{s}n)\} \Rightarrow \forall n\, \phi(n) \qquad (Ind)$$

- Successor axioms:

$$\neg\, \mathtt{s}n = \mathtt{0} \quad (Suc_1) \qquad \mathtt{s}n = \mathtt{s}m \Rightarrow n = m \quad (Suc_2)$$

- The defining equations of the constant $\mathtt{Rec}_\tau$ for each type τ:

$$\mathtt{Rec}\, tu\mathtt{0} = t \quad (Rec_0) \qquad \mathtt{Rec}\, tu(\mathtt{s}v) = uv(\mathtt{Rec}\, tuv) \quad (Rec_s)$$

- λ-calculus axiom and rules:

$$(\lambda x.t)u = t[u/x] \quad (\beta)$$
$$\frac{t = t'}{tu = t'u}\,(Ap_1) \qquad \frac{u = u'}{tu = tu'}\,(Ap_2) \qquad \frac{t = t'}{\lambda x.t = \lambda x.t'}\,(ir)$$

The theory of HA^ω (resp. HA^ω_c) is obtained from that of HA^ω_- by changing the base logic from minimal to intuitionistic (resp. classical).

From now on, we assume that the variables i, j, k, l, m, n are of type $\mathtt{N}$, f, g are of $\mathtt{N} \to \mathtt{N}$ and χ is of $(\mathtt{N} \to \mathtt{N}) \to \mathtt{N}$, and omit types whenever there is no fear of confusion.

We then consider the following schemata (parametric in ϕ):

▶ **Definition 8.** (Schemata $Comp(\tau)$, $AC(\tau, \tau')$ and $DC(\tau)$)

$Comp(\tau)$ $\quad \exists \chi : \tau \to \mathtt{N}\ \forall x : \tau\ \{\chi(x) = \mathtt{1} \Leftrightarrow \phi(x)\}$.

$AC(\tau, \tau')$ $\quad \forall x : \tau\, \exists y : \tau'\, \phi(x, y) \Rightarrow \exists f : \tau \to \tau'\ \forall x : \tau\ \phi(x, f(x))$.

$DC(\tau)$ $\quad \forall x : \tau\, \exists y : \tau\, \phi(x, y) \Rightarrow \forall a : \tau\ \exists f : \mathtt{N} \to \tau\, \{f(\mathtt{0}) = a \wedge \forall n\, \phi(f(n), f(n + \mathtt{1}))\}$.

Using this notation, CAC (*the axiom of countable choice*) and DC (*the principle of dependent choices*) are expressed as $AC(\mathtt{N}, \tau)$ for all types τ and $DC(\tau)$ for all types τ, respectively.

▶ Remark. We refer to the schema $Comp(\tau)$ as comprehension under the identification of a set $\{x : \tau \mid \phi(x)\}$ with a function $\chi : \tau \to \mathtt{N}$ satisfying $\forall x : \tau\, \{\chi(x) = \mathtt{1} \Leftrightarrow \phi(x)\}$, namely a (generalized) characteristic function for $\{x : \tau \mid \phi(x)\}$. This schema is not counted as an axiom of HA^ω_c, and this may be the reason why [2] avoids the standard notation "PA^ω", in which comprehension is usually assumed. The absence of comprehension in our systems of arithmetic will be crucial in section 6.

▶ **Proposition 9.** For any type τ and τ', we have

1. $HA^\omega_- \vdash AC(\tau, \tau) \Rightarrow DC(\tau)$.
2. $HA^\omega_c \vdash DC(\mathtt{N} \to \tau) \Rightarrow AC(\mathtt{N}, \tau)$, and hence DC implies CAC.
3. $HA^\omega_- \vdash AC(\tau, \tau') \Rightarrow AC(\tau, \mathtt{N})$.
4. $HA^\omega_c \vdash AC(\tau, \mathtt{N}) \Rightarrow Comp(\tau)$. In particular, CAC implies $Comp(\mathtt{N})$.

For each formula ϕ of HA^ω_c, let ϕ^K denote the negative translation of ϕ obtained by prefixing all prime formulas and existentially quantified formulas by double negations. For instance, $\{\forall n\, \exists m\, (n + \mathtt{1} = m)\}^K$ is $\forall n \neg\neg \exists m \neg\neg (n + \mathtt{1} = m)$.

Let us point out a fact, which will be crucial in section 5. This translation enables us to embed classical arithmetic further into minimal arithmetic:

▶ **Proposition 10.** [2] $HA^\omega_c + DC \vdash \phi$ implies $HA^\omega_- + DC^K \vdash \phi^K$.

3 Gale-Stewart's Theorem in Classical Arithmetic

D. Gale and F.M. Stewart [6] proved in ZF set theory that all open subsets of the Baire space are determined (Open Determinacy). In this section, we formalize that statement in HA^ω_c and show informally that it is also provable in classical arithmetic.

Before proceeding any further, it would be better to introduce several abbreviations in order to enhance the readability of the following discussion:

- "n is odd" is the prime formula $\texttt{odd?}(n) = \texttt{1}$, where the term $\texttt{odd?}(n)$ of HA^ω_c is equal to $\texttt{1}$ when n is odd, and $\texttt{0}$ otherwise.
- "$k \leq m$" is the prime formula $k \dot{-} m = \texttt{0}$, where $\dot{-}$ is the term for the *truncated subtraction*: $k \dot{-} m$ is $k - m$ when $k > m$, and $\texttt{0}$ otherwise.
- "$OP(\chi)$" is the formula $\forall f\ \{\chi(f) = \texttt{1} \Rightarrow \exists m\ \forall g\ (\texttt{eq}_{\leq m}(f,g) = \texttt{1} \Rightarrow \chi(g) = \texttt{1})\}$, where the term "$\texttt{eq}_{\leq m}(f,g)$" of HA^ω_c is equal to $\texttt{1}$ when $f(k) = g(k)$ for all $k \in \{0,\ldots,m\}$, and $\texttt{0}$ otherwise.
- "OP" is the formula $\forall\chi\ OP(\chi)$.

▶ Remark. It will be easy to confirm that functions like $\texttt{odd?}, \dot{-}$ and $\texttt{eq}_{\leq m}(f,g)$ can be implemented as terms of HA^ω. Notice also that these defined symbols do not add any power, for HA^ω_- proves the equivalence between the prime formula $\texttt{odd?}(n) = \texttt{1}$ (resp. $\texttt{eq}_{\leq m}(f,g) = \texttt{1}$) and the formula $\exists k\ (k \leq n \wedge n = \texttt{2}k + \texttt{1})$ (resp. $\forall k\ \{k \leq m \Rightarrow f(k) = g(k)\}$). Henceforth, we introduce defined symbols in this way, i.e., without presenting the implementation as terms of HA^ω.

▶ Remark. $\chi : (\texttt{N} \to \texttt{N}) \to \texttt{N}$ can be seen as a (generalized) characteristic function for some $A \subset \omega^\omega$: $\forall f\, \{\chi(f) = \texttt{1} \Leftrightarrow f \in A\}$. With this in mind, the formula $OP(\chi)$ is read as "χ represents an open subset of the Baire space ω^ω".

In the sequel, we need an encoding of $\omega^{<\omega}$ into ω in order to formalize the theory of infinite games within arithmetic; fix a primitive recursive bijection $\langle\langle\cdot,\ldots,\cdot\rangle\rangle : \omega^{<\omega} \to \omega$. We also write $(n)_j := a_j$ $(0 \leq j < k)$ and $lh(n) := k$ if $n = \langle\langle a_0,\ldots,a_{k-1}\rangle\rangle$.

By employing this encoding, the plays in Definition 3 can be expressed by the following terms, where σ, y, τ and x are of type $\texttt{N} \to \texttt{N}$:

$$\sigma * y\,(i) := \begin{cases} y((i \dot{-} \texttt{1})/\texttt{2}) & (i : \text{odd}) \\ \sigma(\langle\langle\sigma * y\,(\texttt{0}),\ldots,\sigma * y\,(i \dot{-} \texttt{1})\rangle\rangle) & (i : \text{even}) \end{cases},$$

$$x * \tau\,(i) := \begin{cases} \tau(\langle\langle x * \tau\,(\texttt{0}),\ldots,x * \tau\,(i \dot{-} \texttt{1})\rangle\rangle) & (i : \text{odd}) \\ x(i/\texttt{2}) & (i : \text{even}) \end{cases}.$$

Strictly speaking, we should use different symbols for $*$ in $\sigma * y : \texttt{N} \to \texttt{N}$ and $x * \tau : \texttt{N} \to \texttt{N}$, since now all of σ, y, τ and x are of the same type. However, no confusion may be caused by this, as it is clear from the context.

For convenience, we also adopt the following abbreviations:

- "*I has a w.s. in* $G(\chi)$" is the formula $\exists\sigma : \texttt{N} \to \texttt{N}\ \forall y : \texttt{N} \to \texttt{N}\ \chi(\sigma * y) = \texttt{1}$.
- "*II has a w.s. in* $G(\chi)$" is the formula $\exists\tau : \texttt{N} \to \texttt{N}\ \forall x : \texttt{N} \to \texttt{N}\ \neg\,\chi(x * \tau) = \texttt{1}$.
- "$Det(\chi)$" is the formula $\neg\,(I$ *has a w.s. in* $G(\chi)) \Rightarrow (II$ *has a w.s. in* $G(\chi))$.
- "AD" is the formula $\forall\chi\ Det(\chi)$.

Now, let us formalize open determinacy in the language of HA^ω_c and prove it within arithmetic. Although the proof is presented informally, it can easily be formalized in $HA^\omega_c + CAC$.

▶ **Theorem 11.** (Gale-Stewart [6]) $HA^\omega_c + CAC \vdash \forall\chi\ \{OP(\chi) \Rightarrow Det(\chi)\}$.

Proof. For each $x : \mathtt{N}$ with $x = \langle\langle n_0, \ldots, n_{k-1}\rangle\rangle$ and $f : \mathtt{N} \to \mathtt{N}$, let us define $x \mathbin{@} f : \mathtt{N} \to \mathtt{N}$ by

$$x \mathbin{@} f := \langle n_0, \ldots, n_{k-1}, f(\mathtt{0}), f(\mathtt{1}), f(\mathtt{2}), \ldots\rangle.$$

Using this notation, for each $\chi : (\mathtt{N} \to \mathtt{N}) \to \mathtt{N}$ and $x : \mathtt{N}$, we introduce $\chi/_x : (\mathtt{N} \to \mathtt{N}) \to \mathtt{N}$ by

$$\chi/_x\,(f) = \mathtt{1} \Leftrightarrow \chi(x \mathbin{@} f) = \mathtt{1}.$$

▶ **Lemma 12.** For every $x : \mathtt{N}$ with $lh(x)$ odd, if $\neg\,(I\ has\ a\ w.s.\ in\ G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 2}\rangle\rangle}))$, then there exists a y such that $\neg\,(I\ has\ a\ w.s.\ in\ G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 2}, (x)_{lh(x) \dot{-} 1}, y\rangle\rangle}))$.

Proof. We show the contraposition of the above statement. If there exists an x such that $lh(x)$ is odd, and $(I\ has\ a\ w.s.\ in\ G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 1}, y\rangle\rangle}))$ holds for all y, then CAC yields a $\varphi : \mathtt{N} \to (\mathtt{N} \to \mathtt{N})$ such that $\varphi(y)$ is a winning strategy for Player I in the game $G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 1}, y\rangle\rangle})$. Define a strategy $\rho : \mathtt{N} \to \mathtt{N}$ for Player I by:

$$\rho(n) = \begin{cases} (x)_{lh(x) \dot{-} 1} & (n = \langle\langle\rangle\rangle) \\ \varphi(y)\langle\langle p_0, \ldots, p_{2l \dot{-} 1}\rangle\rangle & (n = \langle\langle (x)_{lh(x) \dot{-} 1}, y, p_0, \ldots, p_{2l \dot{-} 1}\rangle\rangle) \\ \mathtt{0} & (\text{else}) \end{cases}.$$

Then, $\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 2}\rangle\rangle}(\rho * z) = \chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 1}, z(\mathtt{0})\rangle\rangle}(\varphi\,(z(\mathtt{0})) * \mathtt{shift}(z)) = \mathtt{1}$ holds for all $z : \mathtt{N} \to \mathtt{N}$, where $\mathtt{shift}(z)$ is $\lambda n.\, z(n + \mathtt{1})$. This means that ρ is a winning strategy for Player I in the game $G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 2}\rangle\rangle})$. ◀

▶ **Lemma 13.** There exists a $\tau : \mathtt{N} \to \mathtt{N}$ such that for all $x : \mathtt{N}$, we have

$$\{lh(x)\ is\ odd \wedge \neg\,(I\ has\ a\ w.s.\ in\ G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 2}\rangle\rangle})) \Rightarrow$$
$$\neg\,(I\ has\ a\ w.s.\ in\ G(\chi/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 1}, \tau(x)\rangle\rangle}))\}.$$

Proof. Apply CAC to the statement of the previous lemma. ◀

Assume $OP(\chi)$ and $\neg\,(I\ has\ a\ w.s.\ in\ G(\chi))$. We show that the above τ is indeed a winning strategy for Player II. Assume for contradiction that there were an $x : \mathtt{N} \to \mathtt{N}$ such that $\chi(x * \tau) = \mathtt{1}$; then there exists an m such that, for all g, $\mathtt{eq}_{\leq m}(x * \tau, g) = \mathtt{1}$ implies $\chi(x * \tau) = \chi(g)$. In particular, if $\mathtt{eq}_{\leq 2m+1}(x * \tau, g) = \mathtt{1}$ holds, then $\chi(x * \tau)$ is equal to $\chi(g)$. Therefore, for all $y : \mathtt{N} \to \mathtt{N}$, it follows that $\chi/_{\langle\langle x * \tau(\mathtt{0}), \ldots, x * \tau(\mathtt{2}m + \mathtt{1})\rangle\rangle}((\lambda n.\mathtt{0}) * y) = \chi(\langle\langle x * \tau\,(\mathtt{0}), \ldots, x * \tau\,(\mathtt{2}m + \mathtt{1})\rangle\rangle \mathbin{@} ((\lambda n.\mathtt{0}) * y)) = \mathtt{1}$.

On the other hand, $(m + 1)$-times applications of Lemma 13 to the hypothesis $\neg(I\ has\ a\ w.s.\ in\ G(\chi))$ yields $\neg\,(I\ has\ a\ w.s.\ in\ G(\chi/_{\langle\langle x * \tau(0), \ldots, x * \tau(2m + 1)\rangle\rangle}))$. This means that for the strategy $\lambda n.\mathtt{0}$, there exists a $y : \mathtt{N} \to \mathtt{N}$ with $\chi/_{\langle\langle x * \tau(0), \ldots, x * \tau(2m + 1)\rangle\rangle}((\lambda n.\mathtt{0}) * y) \neq \mathtt{1}$. A contradiction. ◀

▶ Remark. By Proposition 9 and Theorem 11, we immediately see that $HA^{\omega}_{c} + DC$ proves $\forall\chi\ \{OP(\chi) \Rightarrow Det(\chi)\}$.

4 Realizability Interpretation

This section is a recapitulation of the notion of the realizability interpretation given in [2]. Since we would like to interpret HA^{ω} in a programming language with this methodology, we need first to present the (infinitary) programming language $\mathscr{P}$. Roughly speaking, $\mathscr{P}$ is an extension of Gödel's system T with list operators, the fixed-point combinator and some auxiliary constructs (needed for realizing DC^{K}). The types and terms of $\mathscr{P}$ are extensions of that of HA^{ω}.

▶ **Definition 14.** (The programming language $\mathscr{P}$)

Types Given by the following grammar: $\tau, \tau' ::= \mathtt{N} \mid \mathtt{Unit} \mid \mathtt{Abs} \mid \tau \to \tau' \mid \tau \times \tau' \mid [\tau]$
Here, $[\tau]$ is the type for lists of objects of type τ.
Seen as a type of $\mathscr{P}$, a type of HA^{ω} is called an $\mathtt{N}$-*type*.

Terms Given by the following grammar:

$$\begin{array}{lr}
t, u ::= x^{\tau} \mid \lambda x^{\tau}.t^{\tau'} \mid t^{\tau \to \tau'} u^{\tau} & \text{(lambda terms)} \\
\quad \mid \mathtt{0}^{\mathtt{N}} \mid \mathtt{s}^{\mathtt{N} \to \mathtt{N}} \mid \mathtt{Rec}_{\tau}^{\tau \to ((\mathtt{N} \to \tau \to \tau) \to (\mathtt{N} \to \tau))} & \text{(system T constants)} \\
\quad \mid Y_{\tau}^{(\tau \to \tau) \to \tau} & \text{(the fixed-point combinator)} \\
\quad \mid \langle , \rangle^{\tau_1 \to \tau_2 \to (\tau_1 \times \tau_2)} \mid \pi_i^{(\tau_1 \times \tau_2) \to \tau_i} \ (i = 1, 2) & \text{(pairing and projection)} \\
\quad \mid \mathtt{nil}^{[\tau]} \mid \mathtt{cons}^{\tau \to [\tau] \to [\tau]} \mid \mathtt{Lrec}^{(\tau \to [\tau] \to \sigma \to \sigma) \to \sigma \to [\tau] \to \sigma} & \text{(list operators)} \\
\quad \mid \mathtt{Dummy}^{\mathtt{Abs}} \mid \mathtt{Axiom}_{\mathtt{i}}^{\mathtt{N} \to \mathtt{Abs}} (i = 1, 2) \mid ()^{\mathtt{Unit}} & \text{(technical constants)} \\
\quad \mid (\lambda\!\!\lambda n.t_n^{\tau})^{\mathtt{N} \to \tau} & \text{(an infinite term)}
\end{array}$$

Infinite operator $\lambda\!\!\lambda$ allows us to built a single term $\lambda\!\!\lambda n.t_n^{\tau}$ out of an arbitrary sequence $t_0^{\tau}, t_1^{\tau}, \ldots$ of terms of type τ.
We abbreviate $\mathtt{s0}$ as $\mathtt{1}$, $\mathtt{ss0}$ as $\mathtt{2}$ and so on, and refer to them as *numerals* hereafter.

Formulas For each type τ, $t_1^{\tau} = t_2^{\tau}$ is a formula.

Theory

- The defining equations of the constant $\mathtt{Rec}_{\tau}$ for each type τ (see Definition 7).
- λ-calculus axiom and rules (see Definition 7).
- The axiom for the fixed-point combinator $\mathtt{Y}_{\tau}$: $\mathtt{Y}t = t(\mathtt{Y}t)$ $\quad (Y)$
- Pairing axioms and list axioms:

$$\begin{array}{lr}
\pi_i \langle t_1, t_2 \rangle = t_i \ (i = 1, 2) & (pr_i) \\
\mathtt{Lrec}(f, u, \mathtt{nil}) = u & (Lrec_0) \\
\mathtt{Lrec}(f, u, \mathtt{cons}(t, L)) = f(t, L, \mathtt{Lrec}(f, u, L)) & (Lrec_1)
\end{array}$$

- The axiom for infinite terms: $(\lambda\!\!\lambda n.t_n)\mathtt{k} = t_k$ $\quad (\beta\!\!\!/)$

▶ Remark. Although infinite terms and unfamiliar constants appear to be ad hoc, such terms are *not* included for computational purposes. In fact, every theorem of $HA^{\underline{\omega}} + DC^K$ can be realized without them [2]; moreover, these technical terms are not so important for the rest of this paper. However, the infinite operator $\lambda\!\!\lambda$ and *two* constants $\mathtt{Axiom}_1$ and $\mathtt{Axiom}_2$ are necessary for testing termination of realizers of CAC^K and DC^K [2].

Let us list several known facts about $\mathscr{P}$ [2, Section 3.4]:

- There exists a reduction relation $\rightsquigarrow$ such that its reflexive, symmetric and transitive closure coincides with $=$.
  Moreover, this reduction relation enjoys the following:
- The Church-Rosser property,
- Every closed normal form of type $\mathtt{N}$ (resp. $\mathtt{Unit}$) is a numeral (resp. $()$),
- Every closed normal form of type $\mathtt{Abs}$ is either $\mathtt{Dummy}$ or of the form $\mathtt{Axiom}_{\mathtt{i}}\mathtt{k}$.

We present two preparatory definitions in advance of the main definition of the realizability relation. With each formula ϕ of HA^{ω}, we associate a type $|\phi|$ of $\mathscr{P}$ as follows:

▶ **Definition 15.** (Associated type $|\phi|$ of $\mathscr{P}$)

- $|t = t'|$ is $\mathtt{Unit}$,
- $|\bot|$ is $\mathtt{Abs}$,

- $|\phi \Rightarrow \psi|$ is $|\phi| \to |\psi|$,
- $|\phi \wedge \psi|$ is $|\phi| \times |\psi|$,
- $|\forall x : \tau\, \phi|$ is $\tau \to |\phi|$,
- $|\exists x : \tau\, \phi|$ is $\tau \times |\phi|$.

For every closed term t of $\mathscr{P}$ of the $\mathtt{N}$-type, the technical notion of *reducibility* is given by induction on the $\mathtt{N}$-type:

▶ **Definition 16.** (Reducible terms of $\mathtt{N}$-type)

- $t : \mathtt{N}$ is reducible if t reduces to $\mathtt{k}$ for some $k \in \omega$,
- $t : \tau \to \tau'$ is reducible if $tu : \tau'$ is reducible for all reducible $u : \tau$.

All terms of HA^ω are clearly reducible.

We cite the following property of $\mathscr{P}$ from [2, Section 3.4] without proof. Note that this so-called syntactic continuity[1] can also be taken as a topological continuity: χ is a continuous function from the Baire space ω^ω to the discrete space ω.

▶ **Proposition 17.** For every reducible terms $\chi : (\mathtt{N} \to \mathtt{N}) \to \mathtt{N}$ and $f : \mathtt{N} \to \mathtt{N}$, there exists an $m \in \omega$ such that for all reducible $g : \mathtt{N} \to \mathtt{N}$ with $f(\mathtt{i}) = g(\mathtt{i})$ $(\forall\, i \leq m)$, we have $\chi f = \chi g$.

This proposition says that the closed normal form of χf is determined only from finitely many outputs of f. This property does not come as a surprise, for f is used (essentially) finitely many times during the reduction of χf to a numeral.

We are now in a position to define the realizability relation $t : |\phi| \circledR \phi$ for a closed formula ϕ of HA^ω possibly including closed reducible terms of $\mathscr{P}$, and for a closed term t of $\mathscr{P}$ of type $|\phi|$. This notion of realizability is essentially the so-called modified realizability [11], except the restriction to the reducible terms when interpreting quantifiers, and the existence of a term t satisfying that $t \circledR \bot$.

▶ **Definition 18.** (Realizability relation)

- $t : \mathtt{Abs} \circledR \bot$ if $t = \mathtt{Axiom_i k}$ for some $k \in \omega$ and $i = 1, 2$,
- $t : \mathtt{Unit} \circledR t_1 = t_2$ if $t = ()$ and both t_1 and t_2 reduce to the same numeral in $\mathscr{P}$,
- $t : |\phi_1| \to |\phi_2| \circledR \phi_1 \Rightarrow \phi_2$ if $tu \circledR \phi_2$ whenever $u \circledR \phi_1$,
- $t : |\phi_1| \times |\phi_2| \circledR \phi_1 \wedge \phi_2$ if $t = \langle t_1, t_2 \rangle$ and $t_i \circledR \phi_i$ $(i = 1, 2)$,
- $t : \tau \to |\phi| \circledR \forall x : \tau\ \phi$ if $tu \circledR \phi\ [u/x]$ for all reducible $u : \tau$,
- $t : \tau \times |\phi| \circledR \exists x : \tau\ \phi$ if $t = \langle p, u \rangle$ with $p : \tau$ reducible and $u \circledR \phi\ [p/x]$.

▶ Remark. Since $\mathscr{P}$ contains the fixed-point combinator $\mathtt{Y}$, the non-termination problem arises. For exactly this reason, quantification should be restricted to reducible terms, in other words, to hereditarily normalizing terms. Otherwise, there could be problems such as, (i) a realizer of an existential formula may fail to give a witness, and (ii) the identity axiom "$\forall x\ (x = x)$" cannot be realized.

[1] It has been proved by Čeitin (independently by Kreisel, Lacombe and Shoenfield) that for every effective operation e and total recursive index y, a modulus of continuity for e at y can be computed by a partial recursive function under the assumption of Markov principle (see [1, Chapter IV, Theorem 3.1]). However, it would be impossible here to adopt this theorem to show the existence of a modulus-of-continuity functional, i.e., a partial recursive function which compute a modulus of continuity. This is because our setting is far from intensional. Furthermore, f and g can be non-recursive here due to the existence of the infinite operator. In fact, it is known that there is *no* extensional modulus-of-continuity functional [1, Chapter IV, Section 3.3].

▶ Remark. If, for every term t and formula ϕ, t ® ϕ holds if and only if t ® ϕ^K holds, then the negative translation K plays no effective role in realizing a formula. The first clause of the above definition is demanded to break this equivalence, at the price of the non-existence of a realizer for the ex falso axiom: $\bot \Rightarrow \phi$. Moreover, this definition allows us to use $\mathtt{Axiom_i}$ for computing witnesses [2, Section 4.4]: These terms catch a witness $\mathtt{n}$ of an existential formula $\exists n\, \phi(n)$ during reduction and freeze that datum as in the form of $\mathtt{Axiom_i n}$. When the execution of a program stops, one can pick up that $\mathtt{n}$ out of the residue of the calculation.

We say a formula ϕ is *realizable* when there exists a $\{\lambda\!\!\lambda, \mathtt{Axiom_i}(i = 1, 2)\}$-free term t satisfying that $t : |\phi|$ ® ϕ. The main theorem of [2] reads in this notation as follows:

▶ **Theorem 19.** [2] Every theorem of $HA^{\omega}_{=} + DC^K$ is realizable.

▶ Remark. In [2], the most difficult cases, the realizations of $CAC^K$and DC^K, are managed with bar recursion and continuity. The problem of the termination of realizers, which boils down to the problem of the termination of bar recursion used in them, is proved non-constructively. The difficulty is to be attributed to the fact that the *negatively translated* choice principles are much more powerful than the choice principles themselves in HA^{ω}. In fact, [7] shows that $HA^{\omega} + AC(\mathtt{N}, \mathtt{N}) + AC(\mathtt{N}, \mathtt{N} \to \mathtt{N})$ is conservative over Heyting arithmetic.

▶ **Corollary 20.** $\forall\chi\ \{OP(\chi) \Rightarrow Det(\chi)\}^K$ is realizable.

By closely following the proof of Theorem 11, we obtain the following realizer of the formula $\forall\chi\ \{OP(\chi) \Rightarrow Det(\chi)\}^K$ (see Appendix for detail): $\lambda\chi\theta\zeta\eta.\Phi P_2(\zeta, \eta, \theta) H_2(\chi)[\,]$.

5 A Realizer of the Negative Translation of AD

In view of Corollary 20, it suffices to realize OP^K for realizing AD^K.

Let us consider again the formula OP:

$$\forall\chi\ \forall f\ \{\chi(f) = \mathtt{1} \Rightarrow \exists m\ \forall g\ (\mathtt{eq}_{\leq m}(f, g) = \mathtt{1} \Rightarrow \chi(g) = \mathtt{1})\},$$

which expresses that "every χ represents an open subset of ω^ω". More precisely, OP states that for every element f of (the set represented by) χ, there exists an m such that a basic open neighborhood $\{g \mid \mathtt{eq}_{\leq m}(f, g) = \mathtt{1}\}$ at f is contained in χ. Following [1], we call this m a *modulus* for χ at f.

Let us recall Proposition 17 here: for every reducible χ and f, the existence of a modulus m for χ at f is assured there from the *external* viewpoint. This, however, does not imply the existence of an *internally definable* term t such that $t(\chi, f)$ is a modulus for χ at f. It would be impossible to build such a term t in our setting (see footnote 1).

To realize OP itself will also be impossible, for if there were a realizer s of OP, a modulus could be computed internally as follows: Take any reducible χ and f with $\chi(f) = \mathtt{1}$. Then, from s ® OP and $()$ ® $\chi(f) = \mathtt{1}$, we see that $s\chi f()$ witnesses $\exists m\ \forall g\ (\mathtt{eq}_{\leq m}(f, g) = \mathtt{1} \Rightarrow \chi(g) = \mathtt{1})$, and hence, $\pi_1(s\chi f())$ reduces to a modulus.

The thing is quite different when it comes to realizing the negative translation OP^K of OP. In contrast to the realization of OP, where we do have to calculate a modulus only from χ and f internally, it suffices to indicate the existence of a modulus m externally when realizing OP^K. This point—internal or external—is to be noted as an essential difference between intuitionistic and classical logic.

▶ Remark. The reader may still have some doubt if it is really possible to realize even the negative translation of such a strange statement. This happens by virtue of the absence of the comprehension schema $Comp(\mathtt{N} \to \mathtt{N})$ at type $\mathtt{N} \to \mathtt{N}$. In fact, $Comp(\mathtt{N} \to \mathtt{N})$ implies the existence of a "set" χ satisfying $\forall f\ \{\chi(f) = \mathtt{1} \Leftrightarrow \forall n\,(f(n) = \mathtt{0})\}$; but such χ is *not* open.

Now we prove:

▶ **Lemma 21.** OP^K is realizable.

Proof. First of all, let us recall the formula OP^K:

$$\forall\chi\,\forall f\,\{\neg\neg\,\chi(f)=\mathtt{1}\Rightarrow\neg\neg\,\exists m\,\forall g\,(\neg\neg\,\mathtt{eq}_{\leq m}(f,g)=\mathtt{1}\Rightarrow\neg\neg\,\chi(g)=\mathtt{1})\}.$$

To realize this, we introduce a term Θ by

$$\begin{aligned}&\Theta\,\chi fuvn := v\langle n,\lambda gpq.\,t(n,g,p,q)\rangle,\ \text{with}\\&t(n,g,p,q) := \mathtt{if}\ \mathtt{eq}_{\leq n}(f,g)=\mathtt{0}\ \mathtt{then}\ p(\lambda r.\mathtt{Dummy})\ \mathtt{else}\\&\qquad\qquad\qquad\qquad\qquad\qquad\mathtt{if}\ \chi(f)=\chi(g)\ \mathtt{then}\ uq\ \mathtt{else}\ \Theta\,\chi fuv(\mathtt{s}n),\end{aligned}$$

where $\mathtt{if\ n=m\ then}\cdots\mathtt{else}\cdots$ is a syntactic sugar.

In the following, we show that $\lambda\chi fuv.\Theta\,\chi fuv\mathtt{0}\circledR OP^K$. Take arbitrary reducible terms χ and f. We need to prove $\Theta\,\chi fUV\mathtt{0}\circledR\bot$ for every term U and V with $U\circledR\neg\neg\,\chi\,(f)=\mathtt{1}$ and $V\circledR\neg\,\exists m\ \forall g\ (\neg\neg\,\mathtt{eq}_{\leq m}(f,g)=\mathtt{1}\Rightarrow\neg\neg\,\chi(g)=\mathtt{1})$.

We first claim that, for every n, $\Theta\,\chi fUV\mathtt{sn}\circledR\bot$ implies $\Theta\,\chi fUV\mathtt{n}\circledR\bot$. Assume that $\Theta\chi fUV\mathtt{sn}\circledR\bot$. Since we have $V\circledR\{\exists m\,\forall g\,(\neg\neg\,\mathtt{eq}_{\leq m}(f,g)=\mathtt{1}\Rightarrow\neg\neg\,\chi(g)=\mathtt{1})\Rightarrow\bot\}$ and $\Theta\,\chi fUV\mathtt{n}=V\langle\mathtt{n},\lambda gpq.\,t(\mathtt{n},g,p,q)\rangle$, in order to show our first claim, it suffices to prove $\lambda gpq.\ t(\mathtt{n},g,p,q)\circledR\forall g\ (\neg\neg\,\mathtt{eq}_{\leq\mathtt{n}}(f,g)=\mathtt{1}\Rightarrow\neg\neg\,\chi(g)=\mathtt{1})$. Take an arbitrary reducible term $g:\mathtt{N}\to\mathtt{N}$ and terms P and Q satisfying $P\circledR\neg\neg\,\mathtt{eq}_{\leq\mathtt{n}}(f,g)=\mathtt{1}$ and $Q\circledR\neg\,\chi\,(g)=\mathtt{1}$. We have to examine the following three cases to verify $t(\mathtt{n},g,P,Q)\circledR\bot$:

Case 1: $\exists i\leq n\ f(i)\neq g(i)$ — $t(\mathtt{n},g,P,Q)$ reduces to $P(\lambda r.\mathtt{Dummy})$. Since we have $\mathtt{eq}_{\leq\mathtt{n}}(f,g)=\mathtt{0}$ and $P\circledR\neg\neg\,\mathtt{eq}_{\leq\mathtt{n}}(f,g)=\mathtt{1}$, we conclude that $P(\lambda r.\mathtt{Dummy})\circledR\bot$.

Case 2: $\forall i\leq n\ f(i)=g(i)$ and $\chi(f)=\chi(g)$ — $t(\mathtt{n},g,P,Q)$ reduces to UQ. Since we have $U\circledR\neg\neg\,\chi\,(f)=\mathtt{1}$ and $Q\circledR\neg\,\chi\,(g)=\mathtt{1}$, it follows that $UQ\circledR\bot$.

Case 3: $\forall i\leq n\ f(i)=g(i)$ and $\chi(f)\neq\chi(g)$ — In this case, $t(\mathtt{n},g,P,Q)$ reduces to $\Theta\chi fUV\mathtt{sn}$. Hence we have $t(\mathtt{n},g,P,Q)\circledR\bot$ by the hypothesis.

Next, we claim that if m is a modulus for χ at f, then $\Theta\,\chi fUV\mathtt{m}\circledR\bot$ holds. The proof proceeds along the same line as above except the last case, which no longer happen by the fact that m is a modulus.

Since Proposition 17 assures the existence of a modulus m for χ at f, though we know the existence only externally, $\Theta\,\chi fUV\mathtt{0}\circledR\bot$ follows from the foregoing arguments. ◀

▶ Remark. Inspired by [2], U. Berger and P. Oliva presented a similar result axiomatically in [3]. Instead of implementing a bar recursion as a term of $\mathscr{P}$ using the fixed-point combinator $\mathtt{Y}$, they extended the calculus by *directly adding* the so-called *modified bar recursion* (MBR), which allows us to approximate a choice function and to realize DC^K. We have an impression that OP^K is not realizable in their framework. If an unbounded search used as in Θ were primitive recursively definable (p.r.d.) in MBR, the functional $\hat{\mu}$ would also be p.r.d. in MBR, where $\hat{\mu}(\chi,f):=\min\{\,k\mid\chi(f\!\upharpoonright\!k\ @\ \lambda n.\mathtt{0})=\chi(f\!\upharpoonright\!k\ @\ \lambda n.\mathtt{1})\}$. If so, Kohlenbach's bar recursion (KBR) is p.r.d. in MBR, because KBR is p.r.d. in $\hat{\mu}$ plus Spector's bar recursion (SBR) [10], and SBR is p.r.d. in MBR [4]. However, KBR is *not* p.r.d. in MBR [4].

▶ **Theorem 22.** AD^K is realizable.

Proof. Follows easily from Corollary 20 and Lemma 21. ◀

▶ **Corollary 23.** $\neg\,AC(\mathtt{N}\to\mathtt{N},\mathtt{N})^K$ is realizable.

Proof. $HA^\omega_c + AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$ proves an instance $\exists \chi \forall f \{\chi(f) = \mathtt{1} \Leftrightarrow \forall n (f(n) = \mathtt{0})\}$ of $Comp(\mathtt{N} \to \mathtt{N})$ by Proposition 9 . In view of the remark above Lemma 21, we find that $HA^\omega_c + AC(\mathtt{N} \to \mathtt{N}, \mathtt{N}) + OP$ is inconsistent. Thus, HA^ω_c proves $OP \Rightarrow \neg AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$, and hence, $OP^K \Rightarrow \neg AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})^K$ is realizable. The assertion follows from Lemma 21. ◀

▶ Remark. Corollary 23 shows that it is impossible to realize the axiom of choice at higher order in the framework of [2]. But even further is indicated by the above discussion: To realize such an axiom would be hopeless in any reasonable setting—at least if one sticks to the usual indirect approach. If we assume continuity, we will fail to realize $AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})^K$. This is because we may interpret OP^K, which contradicts the axiom of choice at higher type as we saw. On the other hand, if we drop the assumption of continuity, to realize even CAC becomes difficult. It is only a novel idea, if any, that can open the possibility for the higher order.

From these results, we find that the combination of the negative translation K and the realizability semantics à la [2] can be seen as a model of $HA^\omega_c + DC + AD + \neg AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$. Therefore, we can reduce the consistency of this system to that of $\mathscr{P}$:

▶ **Corollary 24.** $HA^\omega_c + DC + AD + \neg AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$ is consistent.

The next is a straightforward consequence of the previous corollary and Proposition 9.

▶ **Corollary 25.** $HA^\omega_c + DC \vdash AC(\mathtt{N}, \tau)$, but $HA^\omega_c + DC \nvdash AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$.

As far as the author knows, Corollaries 24 and 25 do not follow trivially from known results in set theory.[2] One future work is to investigate whether or not Corollary 24 remains true in the presence of $Comp(\mathtt{N} \to \mathtt{N})$.

Note that the foregoing three corollaries are still valid even if we replace $AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$ by $AC(\mathtt{N} \to \mathtt{N}, \tau)$ for an arbitrary type τ (see Proposition 9).

6 How Does the Realizer Behave?

By combining the realizer of OP^K given in the proof of Lemma 21 and the realizer of $\forall \chi \{OP(\chi) \Rightarrow Det(\chi)\}^K$ given after Corollary 20 (see also Appendix), we obtain a realizer of AD^K. This section is devoted to the explanation of its behavior.

Our realizer takes the following two steps:

Step 1: construct a strategy τ for Player II

In order to facilitate understanding of this step, let us employ Coquand's game theoretical semantics [5]. Firstly we give a recapitulation of that semantics below. The semantics is defined for an infinitary propositional calculus. The formulas of this calculus are defined inductively as: (i) 0 and 1 are (atomic) formulas, and (ii) if ϕ_i $(i \in I)$ are formulas, where I is a countable set, then both $\bigwedge_{i \in I} \phi_i$ and $\bigvee_{i \in I} \phi_i$ are formulas. Note that each arithmetical formula can be represented as a formula of this infinitary propositional calculus in a natural way. For instance, AD is expressed as:

$$\bigvee_{\sigma} \bigwedge_{y} \chi(\sigma * y) = \mathtt{1} \vee \bigvee_{\tau} \bigwedge_{x} \overline{\chi(x * \tau) = \mathtt{1}}. \qquad (*)$$

[2] As regards Corollary 24, one may think that at least the consistency of $HA^\omega_c + DC + AD$ follow trivially from that of $ZF + DC + AD$. This is certainly so, but the consistency of $ZF + DC + AD$ itself is *much stronger* than that of ZFC [8].

Henceforth, some formulas of HA^ω_c will be considered as formulas of this calculus without further explanation.

We then introduce the notion of *classical validity* by specifying the set $\mathcal{V}$ of classically valid formulas. $\mathcal{V}$ is the smallest set of formulas satisfying: (i) $1 \in \mathcal{V}$, (ii) $\bigwedge_{i \in I} \phi_i \in \mathcal{V}$ if $\phi_i \in \mathcal{V}$ for all $i \in I$, and (iii) $\bigvee_{i \in I} \phi_i \in \mathcal{V}$ if there exists an $i_0 \in I$ such that either ϕ_{i_0} is 1, or ϕ_{i_0} is of the form $\bigwedge_{j \in J} \phi_{i_0 j}$ with $\phi_{i_0 j} \vee \bigvee_{i \in I} \phi_i \in \mathcal{V}$ for all $j \in J$.[3]

Game theoretical semantics for this calculus is given as a perfect information game over a formula between two players: ∃loise, who plays for existential formulas, and ∀belard, who plays for universal formulas. Here, we regard atomic formulas as both universal and existential. The game for a formula ϕ is played as follows: If ∃loise (resp. ∀belard) has to play and ϕ is atomic, then ∃loise (resp. ∀belard) wins if ϕ is 1 (resp. 0). If ϕ is universal of the form $\bigwedge_{i \in I} \phi_i$, then ∀belard has to choose an $i \in I$ and ∃loise starts the game for ϕ_i. If ϕ is existential of the form $\bigvee_{i \in I} \phi_i$, then ∃loise chooses an $i \in I$ and wins if ϕ_i is 1, looses if ϕ_i is 0. When ϕ_i is universal of the form $\bigwedge_{j \in J} \phi_{ij}$, ∃loise can start the game not for ϕ_{ij} but for $\phi_{ij} \vee \bigvee_{i \in I} \phi_i$ after ∀belard returns a $j \in J$. The rule of this game is rather unfair to ∀belard; it is only ∃loise who is allowed to change her mind and backtrack in her choice. It will be easy to verify that

► **Proposition 26.** [5] ∃loise has a winning strategy for $\phi \;\Leftrightarrow\; \phi$ is classically valid.

Then, in order to describe this step, let us consider the following instance of the axiom of countable choice used in the proof of Lemma 13:

$$CAC_\phi := \bigvee_x \bigwedge_y \overline{\phi_{xy}} \vee \bigvee_\tau \bigwedge_x \phi_{x\tau(x)}, \text{ with}$$

$$\phi_{xy} := \overline{(lh\,(x)\ \ is\ odd)}^K \vee (I\ has\ a\ w.s.\ in\ G(X/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 2} \rangle\rangle}))^K \vee \\ \overline{(I\ has\ a\ w.s.\ in\ G(X/_{\langle\langle (x)_0, \ldots, (x)_{lh(x) \dot{-} 1}, y \rangle\rangle}))^K},$$

where the formula $\overline{\phi}$ is the complement of a formula ϕ obtained by interchanging 0 and 1, $\bigvee$ and $\bigwedge$. Observe that ϕ_{xy} is the direct translation of the statement of Lemma 12.

How our realizer constructs a strategy τ is illustrated by the following dialog.[4] ∀belard's answers should be read as values provided by arguments of our realizer, in other words, the environment. ∃loise's way of answering should be compared to the way our realizer returns values to the environment:

∃loise: Let me kick off the game by choosing, say $\tau_0 = \lambda n.0$.

∀belard: Then, my choice is $x = x_0$. By this, I can win in the game for $\phi_{x_0\, \tau_0(x_0)} (= \phi_{x_0 0})$.
► Now the formula is $CAC_\phi \vee \phi_{x_0 0}$.

∃loise: What is your answer when I play $x = x_0$ in the game for $\bigvee_x \bigwedge_y \overline{\phi_{xy}}$?

∀belard: In that case, I choose $y = y_0$. This can make you lose in the game for $\overline{\phi_{x_0 y_0}}$.
► Now the formula is $CAC_\phi \vee \phi_{x_0 0} \vee \overline{\phi_{x_0 y_0}}$.

∃loise: Since it is you who said that $\overline{\phi_{x_0 y_0}}$ is false, $\phi_{x_0 y_0}$ should be true, right?
(∀belard: Oops!) Then I backtrack my previous choice τ_0 and select
$\tau_1 := \lambda x.\ \texttt{if}\ x = x_0\ \texttt{then}\ y_0\ \texttt{else}\ \tau_0(x)$.

∀belard: Well, $x = x_1$ is fine. This time, I can defeat you in the game for $\phi_{x_1 \tau_1(x_1)}$.

[3] Since only the original paper [5] employs $\phi_{i_0 j} \vee \bigvee_{i \in I - \{i_0\}} \phi_i$ instead of $\phi_{i_0 j} \vee \bigvee_{i \in I} \phi_i$ in the formulation of the classical validity, we shall adopt the formulation given in the subsequent papers.

[4] Strictly speaking, the following "game" is different from the concept defined so far; now that τ ranges over the (uncountable) set of all functions from $\mathtt{N}$ to $\mathtt{N}$. In fact, by introducing conjunctions and disjunctions over $\mathtt{N} \to \mathtt{N}$, cut-elimination theorem, which holds in the original version, is no longer true [2, Section 2.3]. It serves only as an explanation of the behavior.

$\blacktriangleright$ Now the formula is $CAC_\phi \vee \phi_{x_0 0} \vee \overline{\phi_{x_0 y_0}} \vee \phi_{x_1 0}$.

∃loise: If I choose $x = x_1$ in the game for $\bigvee_x \bigwedge_y \overline{\phi_{xy}}$, what is your choice?

∀belard: Again that question!? $\cdots$ (sigh). $y = y_1$ is the best option; I will win in the game for $\overline{\phi_{x_1 y_1}}$.

$\blacktriangleright$ Now the formula is $CAC_\phi \vee \phi_{x_0 0} \vee \overline{\phi_{x_0 y_0}} \vee \phi_{x_1 0} \vee \overline{\phi_{x_1 y_1}}$.

∃loise: Wait a minute. It means $\phi_{x_1 y_1}$ should be true, doesn't it. (∀belard: Oh no!)
I do not have to stick to my previus move any more; let me choose
$\tau_2 := \lambda x.\,\mathtt{if}\ x = x_1\ \mathtt{then}\ y_1\ \mathtt{else}\ \tau_1(x)$.

$\vdots$

Both players continue playing in this way and, at each round, ∃loise updates a strategy τ using ∀belard's previous answers x_i and y_i. If ∀belard decides his move on *finitely much information* from the move of ∃loise, then, for some n and m with $n < m$, his choice for x in the n-th and m-th round will be the same one: $x_n = x_m$. (This assumption on ∀belard comes true in $\mathscr{P}$ due to continuity). Observe that, at that point, the formula is $CAC_\phi \vee \cdots \vee \overline{\phi_{x_n y_n}} \vee \cdots \vee \phi_{x_m y_n}$.

Step 2: derive the determinacy

Our realizer at last witnesses the determinacy. Let us continue employing the terminology of Coquand's game.

∃loise plays the game for the formula $CAC_\phi \vee \cdots \vee \overline{\phi_{x_n y_n}} \vee \cdots \vee \phi_{x_m y_n}$. Since either $\phi_{x_m y_n}$ or $\overline{\phi_{x_n y_n}}$ is true, ∃loise can certainly win by playing the games for $\phi_{x_m y_n}$ and $\overline{\phi_{x_n y_n}}$ alternately. If it turns out that $\phi_{x_m y_n}$ is true, the realizer concludes that τ satisfies $\bigwedge_x \phi_{x\tau(x)}$. This is because τ has been constructed so that $\phi_{x\,\tau(x)}$ holds for all possible moves x of ∀belard. With the help of the property OP^K, which is verified by just a simple unbounded search for a modulus, τ is understood as a winning strategy for Player II, in other words, $\bigwedge_x \overline{\chi(x * \tau) = \mathtt{1}}$ is verified. (Recall the proof of Theorem 11 here: the construction of a winning strategy τ for Player II is conducted without appealing to OP. In other words, OP has *nothing* to do with the construction of τ—the role of OP is to confirm that if a $\tau : \mathtt{N} \to \mathtt{N}$ satisfies $\bigwedge_x \phi_{x\tau(x)}$, then it is indeed a required one). On the other hand, when $\overline{\phi_{x_n y_n}}$ is true, our realizer tries to disprove the non-existence of a winning strategy for Player I by actually constructing a winning strategy for Player I as in the proof of Lemma 12 and, as a consequence, $\bigvee_\sigma \bigwedge_y \chi(\sigma * y) = \mathtt{1}$ is verified. Neither ∀belard nor ∃loise knows which player—Player I or Player II—is shown to have a winning strategy. This is because we cannot calculate the modulus m internally (cf. the previous section).

In summary, our realizer behaves as follows: It first constructs a strategy τ for Player II *not* by choosing values herself *but* by making use of ∀belard's returns $x_0, y_0, x_1, y_1, \ldots$ against her attempt at exposing falsehood. When a good approximation is made, it either verifies that τ is indeed a winning strategy for Player II with the help of OP, or shifts the blame to the assumption that Player I has no winning strategy.

The behavior of our realizer reflects the proof of Theorem 11: Since we proved that theorem by constructing a winning strategy for Player II under the assumption that Player I does not have a winning strategy, the resulting realizer constructs a winning strategy for Player II. If we change the proof so that it constructs a winning strategy for Player I assuming that there is no winning strategy for Player II, the corresponding realizer will try to construct a winning strategy for Player I. One future work is, based on a game theoretical intuition, to build a realizer of AD^K that works *symmetrically*, in other words, a realizer which behaves in such a way that winning strategies for Player I and Player II are constructed alternately by backtracking. (see $(*)$—the formula itself is symmetric).

7 Future Work

As emphasized previously, our focus is on the computational content of the AD rather than on the set theoretical applications. Since, insofar as the author knows, there are not so much research on the computational aspect of AD, the author wishes more work would be conducted in this area. This paper will conclude with suggestions for future research:

Indirect approach to AD: Over the property determinacy, several axioms have been proposed and explored [8]. It would be interesting to see whether these variants are realizable. The following intriguing problem should also be addressed: To realize AD^K not in arithmetic but in stronger systems, e.g., in ZF set theory.

Direct approach to AD: Krivine's classical realizability is a machinery which enables us to extract the computational content directly from second order classical logic. All axioms of ZF set theory are realizable in that framework [12]. Moreover, by adding the quote (or clock) instruction to the calculus, both CAC and DC become realizable [13]. It would be interesting to ask, for realizing AD, what kind of instruction we should add to the calculus? What instruction is indispensable? If we can realize AD with some instructions, this technology will attract more attention of set theorists. This is because Krivine's classical realizability yields a new model of ZF+DC+AD, if AD is realizable.

Consistency in arithmetic: In ZF set theory, (full) AC contradicts AD [6]. This is because a well-ordering of the set of all strategies, the existence of which follows from an equivalent of AC, enables us to build a non-determined pay-off set by means of transfinite induction. It seems hard to adjust that proof to the setting of arithmetic. Does full AC and AD contradict in HA^ω_c? Or, does $AC(\mathtt{N} \to \mathtt{N}, \mathtt{N})$ refute AD in HA^ω_c?

Acknowledgements I thank my master thesis supervisors Kazushige Terui and Masahito Hasegawa for lots of helpful comments, Stefano Berardi for explaining the underlying ideas of his joint work [2], and Makoto Tatsuta for drawing my attention to [1]. Finally, I would like to thank the anonymous referee who made a lot of useful suggestions.

A The Realizer of the Negative Translation of Open Determinacy

$\lambda\chi\theta\zeta\eta.\Phi P_2(\zeta,\eta,\theta)H_2(\chi)[\,] \circledR \forall\chi\ \{OP(\chi) \Rightarrow Det(\chi)\}^K$, where

$$\Phi PHL := P\,(\texttt{fun}\ L,\ \lambda x.\ \texttt{rea}\ L\ x\ \ (\lambda x'x''.\ Hx\lambda\langle y,z\rangle.\Phi PH\,((x,y,z):L)))\,;$$

$$\texttt{fun}\ [(x_0,y_0,z_0),\ldots,(x_n,y_n,z_n)]\ x\ \ := y_i\ (\text{when}\ \exists\, i \le n\ (x = x_i)),\ \ := \texttt{0}\ (\text{otherwise});$$

$$\texttt{rea}\ [(x_0,y_0,z_0),\ldots,(x_n,y_n,z_n)]\ x\ a := z_i\ (\text{when}\ \exists\, i \le n\ (x = x_i)), := a\ (\text{otherwise});$$

$$H_2(\chi) := \lambda sh.h\langle \texttt{0},\ \lambda\langle w_1,w_2\rangle v.\ \texttt{if}\ (lh(s)\ \text{is odd})\ \texttt{then}\ c_1\chi s(\lambda q.q())H_1(h)\ P_1(s,w_2)\ \texttt{else}\ w_1(\lambda e.\ \texttt{Dummy})\rangle;$$

$$P_2(\zeta,\eta,\theta) := \lambda\langle\tau,q\rangle.\eta\langle\tau,\lambda x\xi.\theta(x*\tau)\xi(\lambda\langle m,u\rangle.Q_1(\zeta,q,x*\tau,m)Q_2(u,x*\tau))\,\rangle;$$

$$H_1(h) := \lambda m'z'.h\langle m',\ \lambda zx.xz'\rangle;$$

$$P_1(s,w_2) := \lambda\langle\sigma,p\rangle.w_2\,(\lambda r.\ r\langle F'(\sigma,s),\ \lambda y'.\ p\ y'(\texttt{0})\ \texttt{shift}(y')\rangle)\,;$$

$$c_1 := \lambda\chi suh'p'.\texttt{if}\ (lh(s)\ \text{is odd})\ \texttt{then}\ \Phi p'h'[\,]\ \texttt{else}\ u(\lambda r.\texttt{Dummy});$$

$$Q_1(\zeta,q,x*\tau,m) := \texttt{Rec}\ \zeta\,(\lambda nz.\ q\ \langle\langle x*\tau\,(\texttt{0}),\ldots,x*\tau\,(\texttt{2}n)\rangle\rangle\langle\lambda d.\ d(),\ z\rangle)\,m+\texttt{1};$$

$$Q_2(u,x*\tau,m) := \lambda l.\ l\langle\lambda j.\ \texttt{0},\lambda y.\ u\ F(x*\tau,m,y)\ (\lambda k.k())\rangle;$$

$$F(x*\tau,m,y)(n) := \lambda n.\texttt{if}\ n \le \texttt{2}m+\texttt{1}\ \texttt{then}\ x*\tau\ (n)\ \texttt{else}\ ((\lambda n.\texttt{0})*y)\ (n \mathbin{\dot{-}} \texttt{2}m \mathbin{\dot{-}} \texttt{2});$$

$$F'(\sigma,s)(n) := \begin{cases} (s)_{lh(s)\dot{-}1} & (n = \langle\langle\rangle\rangle) \\ \sigma(m)\langle\langle k_0,\ldots,k_{2j+1}\rangle\rangle & (n = \langle\langle (s)_{lh(s)\dot{-}1}, m, k_0,\ldots,k_{2j+1}\rangle\rangle) \\ \texttt{0} & (\text{else}) \end{cases} .$$

References

1 M.J. Beeson: Foundations of constructive mathematics. A series of modern surveys in mathematics, Springer Verlag, pp. xxiii+466, 1985

2 S. Berardi, M. Bezem and Th. Coquand: On the computational content of the axiom of choice. J. Symbolic Logic vol. 63, no. 2, pp. 600–622, 1998

3 U. Berger and P. Oliva: Modified bar recursion and classical dependent choice. Lecture Notes in Logic vol. 20, pp. 89–107, 2005

4 U. Berger and P. Oliva: Modified bar recursion. Math. Structures Comput. Sci. vol. 16, issue 2, pp. 163–183, 2006

5 Th. Coquand: A semantics of evidence for classical arithmetic. J. Symbolic Logic vol. 60, no. 1, pp. 325–337, 1995

6 D. Gale and F.M. Stewart: Infinite games with perfect information. In: H.W. Kuhn and A.W. Tucker (eds.) Contributions to the theory of games, vol 2. Ann. Math. Studies 28. Princeton Univ. Press, pp. 245–266, 1953

7 N. Goodman: Intuitionistic arithmetic as a theory of constructions. PhD thesis, Stanford University, pp. 222, 1968

8 A. Kanamori: The higher infinite, second edition. Springer monographs in mathematics, pp. xxii+536, 2003

9 A.S. Kechris: The axiom of determinacy implies dependent choices in $\mathbf{L}(\mathbb{R})$. J. Symbolic Logic vol. 49, no. 1, pp. 161–173, 1984

10 U. Kohlenbach: Theory of majorizable and continuous functionals and their use for the extraction of bounds from non-constructive proofs: effective moduli of uniqueness for best approximations from ineffective proofs of uniqueness (German). PhD thesis, Goethe-Universität Frankfurt, pp. xxii+278, 1990

11 U. Kohlenbach: Applied proof theory: Proof interpretations and their use in mathematics. Springer monographs in mathematics, pp. xix+532, 2008

12 J.-L. Krivine: Typed lambda-calculus in classical Zermelo-Frænkel set theory. Arch. Math. Logic 40, no. 3, pp. 189–205, 2001

13 J.-L. Krivine: Dependent choice, 'quote' and the clock. Theoret. Comput. Sci. vol. 308, pp. 259–276, 2003

14 J.-L. Krivine: Realizability algebras II: new models of ZF+DC. `http://www.pps.jussieu.fr/~krivine/articles/R_ZF.pdf`

15 D.A. Martin: Borel determinacy. Ann. Math. vol. 102, no. 2, pp. 363–371, 1975

16 A. Montalbán and R.A. Shore: The limits of determinacy in second order arithmetic. Proc. London Math. Soc. vol. 104, part 2, pp.223–252, 2012

17 J. Mycielski and H. Steinhaus: A mathematical axiom contradicting the axiom of choice. Bulletin de l'Académie Polonaise des Sciences, Série des Sciences Mathématiques, Astronomiques et Physiques 10, pp.1–3, 1962

18 J. Mycielski and S. Swierczkowski: On the Lebesgue measurability and the axiom of determinateness. Fundamenta Mathematicae vol. 54, pp. 67–71, 1964

19 M.H. Sørensen and P. Urzyczyn: Lectures on the Curry-Howard isomorphism. Stud. Logic Found. Math. 149, Elsevier, pp. xiv+442, 2006

20 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 (ed.) Recursive function theory: Proc. Symposia in Pure Mathematics, vol. 5, American Mathematical Society, Providence, Rhode Island, pp. 1–27, 1962

21 A.S. Troelstra: Realizability. In: S.R. Buss (ed.) Handbook of proof theory. Stud. Logic Found. Math. 137, Elsevier, pp. 407–473, 1998