Forcing Bar Induction in **System** $\mathbb{T}$

Jonathan Sterling

Carnegie Mellon University

Abstract

Using Martín Escardó's *effectful forcing* technique, we give a new and elegant proof of a well-known result: the constructive validity of Brouwer's monotone Bar Theorem for any **System** $\mathbb{T}$-definable bar [22]. We have not assumed any non-constructive (Classical or Brouwerian) principles in this proof, and have carried out the entire development formally in the Agda proof assistant [17] for Martin-Löf's Constructive Type Theory.

In 2013, Martín Escardó pioneered a technique called "effectful forcing" for demonstrating non-constructive (Brouwerian) principles for the definable functionals of Gödel's **System** $\mathbb{T}$ [7], including the continuity of functionals on the Baire space and uniform continuity of functionals on the Cantor space. Effectful forcing is a remarkably simpler alternative to standard sheaf-theoretic forcing arguments, using ideas from programming languages, including computational effects, monads and logical relations.

Following a suggestion from Thierry Coquand [2, 3], the author learnt that Brouwer's controversial Bar Theorem can be validated in a Beth model (or, more generally, a sheaf topos) by instantiating the premise of barhood at a "generic point", which would yield an inductive mental construction of barhood. In this paper, we put an analogous version of this idea into practice using Escardó's method.

Notation I find it useful to reflect the *modes* of mathematical statements and judgments using colors, where blue indicates an *input* to a statement, and red indicates an *output* (something that is synthesized).

1 Brouwer's Bar Thesis

There are many versions of the Bar Thesis and its corollary, the bar induction principle (see Section 1.1), but we will describe here a particularly perspicuous one. First we will define a point-free notion of topological space called a "spread".

Finite sequences For any set X, let $X^\star$ be the set of finite sequences or lists $\vec{u}$ of elements of X. We will write $\vec{u}\frown x$ for the list got by appending $x \in X$ to the list $\vec{u} \in X^\star$, and $x :: \vec{u}$ for the list got by prepending $x \in X$ to $\vec{u} \in X^\star$. Let α, β range

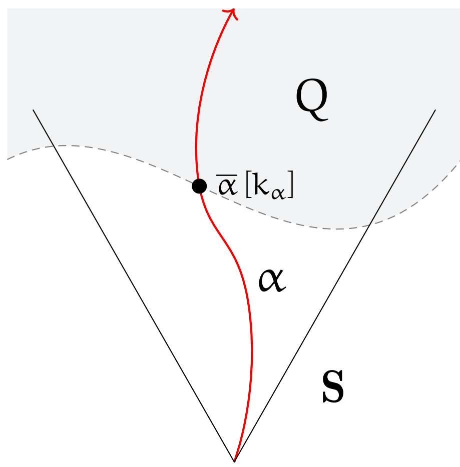

Figure 1: A visualization of when the monotone species $Q \subseteq \mathbf{S}$ is a bar, i.e. $\langle\rangle \lhd Q$. The dotted line represents where the bar begins, and α is an arbitrary choice sequence in the spread; $\overline{\alpha}\,[k_\alpha]$ is the first prefix of α which is in Q.

over *infinite* sequences $X^{\mathbb{N}}$; we write $\overline{\alpha}\,[k] \in X^\star$ for finite sequence which is the first k elements of α, $x :: \alpha \in X^{\mathbb{N}}$ for the infinite sequence got by prepending $x \in X$, and $\vec{u} \Subset \alpha \in X^{\mathbb{N}}$ for the infinite sequence got by prepending $\vec{u} \in X^\star$. Finally, write $|\vec{u}|$ for the length of $\vec{u} \in X^\star$.

Spreads A *spread* consists in a set $\mathbf{S} \subseteq \mathbb{N}^\star$ of finite sequences (nodes) $\vec{u} \in \mathbb{N}^\star$ such that the following hold:

$$\frac{}{\langle\rangle \in \mathbf{S}} \qquad \frac{\vec{u} \in \mathbf{S}}{\exists x \in \mathbb{N}.\ \vec{u}\frown x \in \mathbf{S}} \qquad \frac{\vec{u} \in \mathbf{S} \quad \vec{v} \preccurlyeq \vec{u}}{\vec{v} \in \mathbf{S}}$$

Viewed as a topological space, the admitted finite sequences are the spread's open sets (neighborhoods), and its points are the infinite sequences $\alpha \in \mathbb{N}^{\mathbb{N}}$ whose every prefix $\vec{u} \prec \alpha$ is admitted. The topology of a spread is given by the notion of a "bar" (analogous to a formal "cover").

In Intuitionistic mathematics, a subset or predicate is usually called a *species*; we will uphold that terminology.

Bars on a spread We say that a species of nodes $Q \subseteq \mathbf{S}$ bars a node $\vec{u}$ when every infinite sequence out of $\vec{u}$ has a prefix in Q. Formally, we define:

$$\vec{u} \lhd Q \triangleq \forall \alpha \succ \vec{u}.\ \exists k \in \mathbb{N}.\ \overline{\alpha}\,[k] \in Q \tag{1}$$

When Q bars the root node, i.e. $\langle\rangle \lhd Q$, we simply say that Q is a bar; this state of affairs is presented visually in Figure 1.

Inductive definition of bars A species Q is called *monotone* when, if $\vec{u} \in Q$, we also have $\vec{u}\frown x \in Q$ for any $x \in \mathbb{N}$. Separately, we define an *inductive* version of the barhood relation for a *monotone* species of nodes Q, defined as the least relation closed under the two rules of inference in Figure 2. This version of barhood is visualized as a tree in Figure 3.

$$\boxed{\vec{u} \blacktriangleleft Q \ \textit{presupposing} \ \vec{u} \in \mathbf{S}\,,\ Q \subseteq \mathbf{S}\,,\ Q\ \textit{monotone}}$$

$$\frac{\vec{u} \in Q}{\vec{u} \blacktriangleleft Q}\,\eta \qquad \frac{\forall x \in \{x \in \mathbb{N} \mid \vec{u}\frown x \in \mathbf{S}\}.\ \vec{u}\frown x \blacktriangleleft Q}{\vec{u} \blacktriangleleft Q}\,\digamma$$

Figure 2: Brouwer's definition of barhood.

Theorem 1.1. *Assuming Q is monotone, Brouwer's ζ inference is admissible:*

$$\frac{\vec{u} \blacktriangleleft Q}{\vec{u}\frown x \blacktriangleleft Q}\,\zeta$$

Proof. By case on the premise. If the premise was η, then by monotonicity of Q and η. Otherwise, if the premise was $\digamma$, then we have for any $y \in \mathbb{N}$, $\vec{u}\frown y \blacktriangleleft Q$. Choose $y \equiv x$. □

Theorem 1.2. *The inductive relation $\vec{u} \blacktriangleleft Q$ is a sound characterization of barhood: from $\vec{u} \blacktriangleleft Q$ we may conclude $\vec{u} \triangleleft Q$.*

Proof. We have to show that for any $\alpha \succ \vec{u}$, there exists a $k \in \mathbb{N}$ such that $\overline{\alpha}\,[k] \in Q$. In fact, it suffices to show that for any $\alpha \succ \langle\rangle$, there exists a $k \in \mathbb{N}$ such that $\overline{\vec{u} \Subset \alpha}\,[|\vec{u}| + k] \in Q$. We proceed by cases on the premise.

If the premise was η, then choose $k \equiv 0$. Otherwise, if the premise was $\digamma$, then we have for all $x \in \mathbb{N}$, $\vec{u}\frown x \triangleleft Q$; by our inductive hypothesis, for any $x \in \mathbb{N}$ and $\beta \succ \langle\rangle$, we have a $k' \in \mathbb{N}$ such that $\overline{\vec{u}\frown x \Subset \beta}\,[|\vec{u}| + 1 + k'] \in Q$. Let $x \equiv \mathsf{head}(\alpha)$ and $\beta \equiv \mathsf{tail}(\alpha)$; then choose $k \equiv k' + 1$. □

Proposition 1.3 (Brouwer's Bar Thesis)**.** *Brouwer's (monotone) Bar Thesis states that for any monotone species $Q \subseteq \mathbf{S}$, the inductive definition of barhood is also complete: from $\vec{u} \triangleleft Q$ we may conclude $\vec{u} \blacktriangleleft Q$.*

Instantiated at the Baire spread $\mathsf{B} \equiv \{\vec{u} \mid \vec{u} \in \mathbb{N}^\star\}$ (also called the *universal spread*), Proposition 1.3 becomes the standard statement of Brouwer's Bar Thesis; at the Cantor spread $\mathsf{C} \equiv \{\vec{u} \mid \vec{u} \in \mathbf{2}^\star\}$, it proves the Fan Theorem. In what follows, we will work only with the universal spread B without loss of generality.

Stated as above, Proposition 1.3 for the Baire and Cantor spreads is consistent with constructive foundations, but is not constructively valid. There is a computational procedure called "bar recursion" to witness the validity of the completeness rule above,

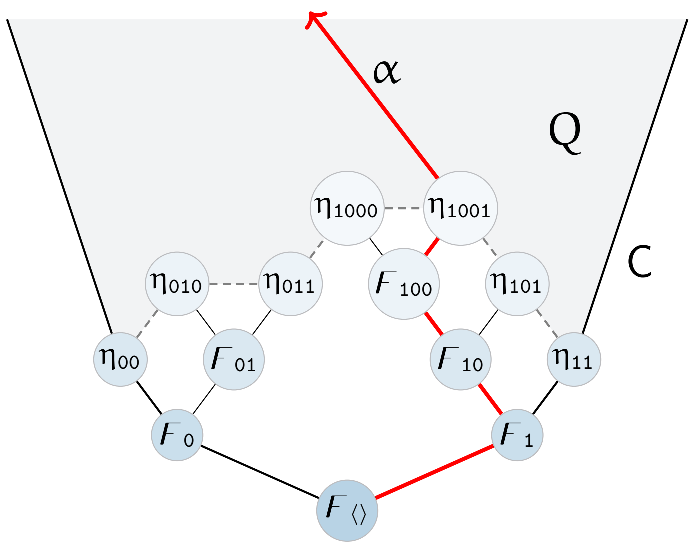

Figure 3: A visualization of the *inductive* characterization of barhood $\langle\rangle \blacktriangleleft \mathsf{Q}$ for a monotone species $\mathsf{Q} \subseteq \mathsf{C}$ in the Cantor spread $\mathsf{C} \triangleq \mathbf{2}^{\star}$, whose points are infinite sequences of bits. (Compare with Figure 1.) In the depicted tree, each node is labeled with the finite sequence of bits that it codes; to be precise, a node labeled $\eta_{\vec{u}}$ represents a proof of $\vec{u} \blacktriangleleft \mathsf{Q}$ which is a leaf node, and a node labeled $F_{\vec{u}}$ represents a proof of $\vec{u} \blacktriangleleft \mathsf{Q}$ which is a branching node.

but the recognition of its effectiveness depends crucially on the assumption of the Bar Thesis itself.

As far as *computational* realizations of constructive foundations are concerned, then, this places the Bar Thesis at the same level as other axioms such as Markov's Principle and Church's Thesis, whose effectiveness can be assumed without disturbing the computational character of the framework, but for which it is a trivial matter to exhibit a countermodel.

1.1 The Bar Induction Principle

In most of the relevant literature, Brouwer's Thesis is usually considered not in the form given in Proposition 1.3, but equivalently as the following *induction principle*.

Definition 1.4 (Monotone Bar Induction Principle)**.** *The monotone bar induction principle* $\mathbf{BI}_{mono}$ *says that for a species* $R \subseteq \mathbb{N}^\star$ *of nodes and a monotone bar* $\langle\rangle \lhd Q$*, if we can conclude* $\langle\rangle \in R$ *from the following conditions:*

1. *Base case:* R *includes the bar, i.e.* $Q \subseteq R$.
2. *Inductive step: We have* $\vec{u} \in R$ *if for all* $x \in \mathbb{N}$, $\vec{u}\frown x \in R$*. This condition is often called* upwards heredity.

Theorem 1.5. *The monotone bar induction principle* $\mathbf{BI}_{mono}$ *(Definition 1.4) is equivalent to Brouwer's Bar Thesis (Proposition 1.3).*

Proof. ($\Rightarrow$) Fix a monotone bar $\langle\rangle \lhd Q$ and define $R \triangleq \{\vec{u} \mid \vec{u} \blacktriangleleft Q\}$. By the $\mathbf{BI}_{mono}$ principle, it suffices to show the following:

1. Base case: We need to show that if $\vec{u} \in Q$ then $\vec{u} \in R$, which is to say, $\vec{u} \blacktriangleleft Q$; but this is just the η inference.
2. Inductive step: the upwards heredity of R amounts to exactly the $\digamma$ inference.

($\Leftarrow$) Fix a monotone bar $\vec{u} \lhd Q$ and a species $R \supseteq Q$ which is hereditary upwards. By Proposition 1.3, we have some $\vec{u} \blacktriangleleft Q$; proceed by induction on this tree:

Case η. Then $\vec{u} \in B$, whence $\vec{u} \in R$.

Case $\digamma$. Then for any $x \in \mathbb{N}$, we have $\vec{u}\frown x \blacktriangleleft Q$; by induction, we have $\vec{u}\frown x \in R$ for all $x \in \mathbb{N}$. By upwards heredity of R, we have $\vec{u} \in R$.

Then, instantiate $\vec{u} \equiv \langle\rangle$. □

There is another version of the Bar Induction principle called $\mathbf{BI}_{dec}$, which replaces monotonicity of the bar with decidability. In the literature, $\mathbf{BI}_{dec}$ seems to get the most attention, but we view monotonicity to be a more fundamental characteristic than decidability: monotonicity is what distinguishes *judgment* (i.e. acts of knowledge) from merely procedural activity of the Intuitionistic subject [15]. From our point of view, decidability is on the contrary a fairly arbitrary property, and depends less on the mathematical characteristics of an object than on its coding.

From an axiomatic point of view, however, $\mathbf{BI}_{dec}$ may be of interest, because it is weaker than $\mathbf{BI}_{mono}$, as we can see in Theorem 1.6, which is due to Dummett [6].

Lemma 1.1. *Any bar $\langle\rangle \triangleleft Q$ can be freely made into a monotone bar $\langle\rangle \triangleleft Q'$.*

Proof. For a concrete proof of this fact, the reader is referred to Dummett [6]; we prefer to give a *conceptual* proof, where the development of the monotone bar is guided using a *free* construction (i.e. one which is left adjoint to a forgetful functor).

A monotone species can viewed as a presheaf on the poset $\mathbb{N}^\star$. Letting $|\mathbb{N}^\star|$ be the poset of finite sequences of natural numbers under the discrete ordering, we have an obvious inclusion $i : |\mathbb{N}^\star| \to \mathbb{N}^\star$, and thence the reindexing functor $i^* : \widehat{\mathbb{N}^\star} \to \widehat{|\mathbb{N}^\star|}$ by precomposition.

By the left Kan extension, this reindexing functor has a left adjoint, $i_! \triangleq \mathit{Lan}_{i^*}(-) \dashv i^*$; calculating this extension pointwise, we have $i_!(Q) : \widehat{\mathbb{N}^\star} = \{\vec{u} \mid \exists \vec{v} \preccurlyeq \vec{u}.\ \vec{v} \in Q\}$, which is clearly also a bar when $Q : \widehat{|\mathbb{N}^\star|}$ is a bar. ☐

Theorem 1.6. *From* $\mathbf{BI}_{\mathit{mono}}$ *we can conclude* $\mathbf{BI}_{\mathit{dec}}$.

Proof. Fix a *decidable* bar $\langle\rangle \triangleleft Q$ and species $R \supseteq Q$ which is hereditary upwards; we need to show that $\langle\rangle \in R$. By Lemma 1.1, we have can exhibit a monotone bar $Q' \triangleq i_! Q$.

Next, define a new motive of induction, $R' \triangleq \{\vec{u} \mid \vec{u} \in R \vee \vec{u} \in Q'\}$. In order to apply the $\mathbf{BI}_{\mathit{mono}}$ principle, we need to show the following:

1. Base case: clearly $Q' \subseteq R'$.

2. Inductive step: we can show that R' is hereditary upwards. Suppose that for all $x \in \mathbb{N}$, we have $\vec{u}\frown x \in R'$; we need to show that $\vec{u} \in R'$. Because Q and Q' are decidable, we can proceed by case on whether $\vec{u} \in Q'$.

   Case $\vec{u} \in Q'$. Automatically, we have $\vec{u} \in R'$.

   Case $\vec{u} \notin Q'$. Unfolding definitions, this means that there does not exist any $\vec{v} \preccurlyeq \vec{u}$ such that $\vec{v} \in Q$. Therefore, by decidability of Q, we are justified in concluding $\vec{u}\frown x \in Q$ from $\vec{u}\frown x \in Q'$. From the base case of the $\mathbf{BI}_{\mathit{dec}}$ premise, we know that $\vec{u}\frown x \in R$, and then from the upwards heredity of R, we can conclude $\vec{u} \in R$. By definition, we have $\vec{u} \in R'$.

From the above, we can conclude $\langle\rangle \in R'$, which means that either $\langle\rangle \in R$ or $\langle\rangle \in Q'$. In the latter case, we clearly have $\langle\rangle \in Q$, whence by the base case for $\mathbf{BI}_{\mathit{dec}}$ we have $\langle\rangle \in R$. ☐

2 Gödel's System $\mathbb{T}$ as a theory of constructions

In his famous *Dialectica* interpretation [11], Kurt Gödel introduced **System** $\mathbb{T}$ to serve as a formal theory of constructions for Heyting arithmetic, which we briefly reproduce in modern form in Figure 4.

A constructive interpretation of first-order logic can be given by interpreting the logical connectives as predicates on **System** $\mathbb{T}$ terms. Then, it is possible to say what it means for a formula (such as $\vec{u} \triangleleft Q$) to be valid relative to **System** $\mathbb{T}$.

$\boxed{\iota\ \mathit{atype}}$ $\boxed{\sigma\ \mathit{type}}$ $\boxed{\Gamma\ \mathit{ctx}}$ $\boxed{\Gamma \vdash M : \sigma \ \textit{presupposing}\ \Gamma\ \mathit{ctx},\ \sigma\ \mathit{type}}$

$$\frac{}{\mathsf{nat}\ \mathit{atype}} \qquad \frac{\iota\ \mathit{atype}}{\iota\ \mathit{type}} \qquad \frac{\sigma\ \mathit{type} \quad \tau\ \mathit{type}}{\sigma \to \tau\ \mathit{type}} \qquad \mathsf{seq} \triangleq \mathsf{nat} \to \mathsf{nat} \tag{Types}$$

$$\frac{}{\cdot\ \mathit{ctx}} \qquad \frac{\Gamma\ \mathit{ctx} \quad \sigma\ \mathit{type}}{\Gamma, x : \sigma\ \mathit{ctx}}\ (x \notin \Gamma) \tag{Contexts}$$

$$\frac{}{\Gamma, x : \sigma, \Delta \vdash x : \sigma}\ \textbf{var} \qquad \frac{}{\Gamma \vdash \mathsf{z} : \mathsf{nat}}\ \textbf{zero} \qquad \frac{\Gamma \vdash m : \mathsf{nat}}{\Gamma \vdash \mathsf{s}(m) : \mathsf{nat}}\ \textbf{succ} \tag{Terms}$$

$$\frac{\Gamma, x : \mathsf{nat}, y : \sigma \vdash s[x, y] : \sigma \quad \Gamma \vdash z : \sigma \quad \Gamma \vdash n : \mathsf{nat}}{\Gamma \vdash \mathsf{rec}_\sigma([x, y].s[x, y]; z; n) : \sigma}\ \textbf{rec}$$

$$\frac{\Gamma, x : \sigma \vdash m[x] : \tau}{\Gamma \vdash \lambda x.m[x] : \sigma \to \tau}\ \textbf{lam} \qquad \frac{\Gamma \vdash m : \sigma \to \tau \quad \Gamma \vdash n : \sigma}{\Gamma \vdash m \bullet_\sigma n : \tau}\ \textbf{ap}$$

Figure 4: Syntax and typing rules of **System** $\mathbb{T}$, formulated in λ-calculus style.

We do not in this paper pursue the formal development of such a realizability interpretation; however, in Section 4 we will propose a version of barhood $\vec{u} \lhd_{\mathbb{T}} Q$ which captures precisely what it should mean for $\vec{u} \lhd Q$ to be true for a bar that is coded in **System** $\mathbb{T}$.

3 Denotational semantics of System $\mathbb{T}$

We will now proceed to develop two denotational semantics for **System** $\mathbb{T}$: a "standard" semantics and an interactive "dialogue" semantics; then we will prove that the two are coherent using a logical relations argument. This procedure is entirely due to Escardó [7]. Both semantics share the interpretation $\mathcal{V}[\iota]$ of the atomic types ι *atype*, as follows:

$$\mathcal{V}[\mathsf{nat}] \triangleq \mathbb{N} \tag{2}$$

3.1 Standard semantics of System $\mathbb{T}$

The standard semantics $\mathcal{V}[\![\sigma]\!]$ for the types σ *type* is as follows:

$$\mathcal{V}[\![\iota]\!] \triangleq \mathcal{V}[\iota]$$
$$\mathcal{V}[\![\sigma \to \tau]\!] \triangleq \mathcal{V}[\![\sigma]\!] \to \mathcal{V}[\![\tau]\!]$$

Contexts Γ *ctx* are interpreted as environments $\mathcal{G}[\![\Gamma]\!]$:

$$\mathcal{G}[\![\Gamma]\!] \triangleq \prod_{x \in |\Gamma|} \mathcal{V}[\![\Gamma(x)]\!]$$

$$
\begin{aligned}
[\![x : \sigma]\!]_\rho &\triangleq \rho(x)\\
[\![z : \mathsf{nat}]\!]_\rho &\triangleq 0\\
[\![s(m) : \mathsf{nat}]\!]_\rho &\triangleq 1 + [\![m : \mathsf{nat}]\!]_\rho\\
[\![\mathsf{rec}_\sigma([x,y].s[x,y]; z; n) : \sigma]\!]_\rho &\triangleq \mathit{Rec}\begin{pmatrix} (a,b) \mapsto [\![s[x,y] : \sigma]\!]_{\rho, x\mapsto a, y \mapsto b}\,, \\ [\![z : \sigma]\!]_\rho\,, \\ [\![n : \mathsf{nat}]\!]_\rho \end{pmatrix}\\
[\![\lambda x.m[x] : \sigma \to \tau]\!]_\rho &\triangleq a \mapsto [\![m[x] : \tau]\!]_{\rho, x \mapsto a}\\
[\![m \bullet_\sigma n : \tau]\!]_\rho &\triangleq [\![m : \sigma \to \tau]\!]_\rho\, [\![n : \sigma]\!]_\rho
\end{aligned}
$$

where

$$
\begin{aligned}
\mathit{Rec}(s, z, 0) &\triangleq z\\
\mathit{Rec}(s, z, n+1) &\triangleq s(n, \mathit{Rec}(s, z, n))
\end{aligned}
$$

Figure 5: The interpretation of **System** $\mathbb{T}$ terms in our standard semantics.

The interpretation of terms $[\![m : \sigma]\!]_\rho \in \mathcal{V}[\![\sigma]\!]$ for $\rho \in \mathcal{G}[\![\Gamma]\!]$, presupposing $\Gamma \vdash m : \sigma$, is entirely straightforward. We summarize it in Figure 5.

3.2 Escardó dialogues: ideal codes for functionals

First, let us define the set $\mathfrak{E}^X_Y(Z)$ of "Escardó dialogues", which code functionals of type $Y^X \to Z$, as the least set closed under the rules in Figure 6.

$$
\boxed{\mathfrak{E}^X_Y(-) : \mathbf{Set} \to \mathbf{Set}} \qquad \frac{z \in Z}{\eta(z) \in \mathfrak{E}^X_Y(Z)}\ \textbf{return} \qquad \frac{x \in X \quad e \in Y \to \mathfrak{E}^X_Y(Z)}{\beta\langle x\rangle(e) \in \mathfrak{E}^X_Y(Z)}\ \textbf{query}
$$

Figure 6: Rules for constructing Escardó dialogues.

An Escardó dialogue is an idealized procedure or algorithm for computing a functional; leaf nodes $\eta(z)$ return a result z, and branch nodes $\beta\langle x\rangle(e)$ query for the xth element y of the input choice sequence and proceed with $e(y)$. Such interactions are essentially a hyper-intensional, syntactic representation of a *neighborhood function*.

$\mathfrak{E}^X_Y(-)$ is a monad on **Set**, natural in $X, Y \in \mathbf{Set}$. We define the action $\mathfrak{E}(f) \in \mathfrak{E}^X_Y(A) \to \mathfrak{E}^X_Y(B)$ of the functor as follows for $f \in A \to B$:

$$
\begin{aligned}
\mathfrak{E}(f) &\in \mathfrak{E}^X_Y(A) \to \mathfrak{E}^X_Y(B)\\
\mathfrak{E}(f)(\eta(a)) &\triangleq \eta(f(a))\\
\mathfrak{E}(f)(\beta\langle x\rangle(e)) &\triangleq \beta\langle x\rangle(\mathfrak{E}(f) \circ e)
\end{aligned}
$$

η is the unit of the monad; we define the Kleisli extension $f^\star \in \mathfrak{E}^X_Y(A) \to \mathfrak{E}^X_Y(B)$ for $f \in A \to \mathfrak{E}^X_Y(B)$ as follows:

$$\begin{aligned} f^\star &\in \mathfrak{E}^X_Y(A) \to \mathfrak{E}^X_Y(B) \\ f^\star(\eta(a)) &\triangleq f(a) \\ f^\star(\beta\langle x\rangle(e)) &\triangleq \beta\langle x\rangle(f^\star \circ e) \end{aligned}$$

An Escardó dialogue $e \in \mathfrak{E}^X_Y(Z)$ may be executed on a choice sequence $\alpha \in Y^X$ to return a result $e \diamond \alpha \in Z$ as follows:

$$\begin{aligned} - \diamond - &\in \mathfrak{E}^X_Y(Z) \times Y^X \to Z \\ \eta(z) \diamond \alpha &\triangleq z \\ \beta\langle x\rangle(e) \diamond \alpha &\triangleq e(\alpha(x)) \diamond \alpha \end{aligned}$$

The following two lemmas are from Escardó [7].

Lemma 3.1. *For any $\alpha \in Y^X$, the execution map $- \diamond \alpha$ is a natural transformation $\mathfrak{E}^X_Y(-) \to \mathbf{1}_{\mathbf{Set}}$:*

$$\begin{tikzcd} \mathfrak{E}^X_Y(A) \arrow[r, "\mathfrak{E}(f)"] \arrow[d, "-\diamond\alpha"] & \mathfrak{E}^X_Y(B) \arrow[d, "-\diamond\alpha"] \\ A \arrow[r, "f"] & B \end{tikzcd}$$

Proof. Immediate by induction on the dialogue tree. □

Lemma 3.2. *Kleisli extension commutes with execution, in the following sense:*

$$\begin{tikzcd} \mathfrak{E}^X_Y(A) \arrow[rr, "f^\star"] \arrow[d, "-\diamond\alpha"] & & \mathfrak{E}^X_Y(B) \arrow[d, "-\diamond\alpha"] \\ A \arrow[r, "f"] & \mathfrak{E}^X_Y(B) \arrow[r, "-\diamond\alpha"] & B \end{tikzcd}$$

Proof. Immediate by induction on the dialogue tree. □

3.3 Interactive semantics of System $\mathbb{T}$

Now we are prepared to begin interpreting well-typed **System** $\mathbb{T}$ terms into Escardó dialogues. The semantic domains are as follows:

$$\begin{aligned} \mathcal{V}\langle\!\langle\iota\rangle\!\rangle &\triangleq \mathfrak{E}^{\mathbb{N}}_{\mathbb{N}}(\mathcal{V}[\iota]) \\ \mathcal{V}\langle\!\langle\sigma \to \tau\rangle\!\rangle &\triangleq \mathcal{V}\langle\!\langle\sigma\rangle\!\rangle \to \mathcal{V}\langle\!\langle\tau\rangle\!\rangle \\ \mathcal{G}\langle\!\langle\Gamma\rangle\!\rangle &\triangleq \prod_{x\in|\Gamma|} \mathcal{V}\langle\!\langle\Gamma(x)\rangle\!\rangle \end{aligned}$$

$$
\begin{aligned}
\langle\!\langle x : \sigma \rangle\!\rangle_\rho &\triangleq \rho(x) \\
\langle\!\langle \mathsf{z} : \mathsf{nat} \rangle\!\rangle_\rho &\triangleq \eta(0) \\
\langle\!\langle \mathsf{s}(m) : \mathsf{nat} \rangle\!\rangle_\rho &\triangleq \mathfrak{E}(1 + -)\, \langle\!\langle m : \mathsf{nat} \rangle\!\rangle_\rho \\
\langle\!\langle \mathsf{rec}_\sigma([x,y].s[x,y]; z; n) : \sigma \rangle\!\rangle_\rho &\triangleq \mathit{Rec}\left(\begin{array}{l} (a, b) \mapsto \langle\!\langle s[x,y] : \sigma \rangle\!\rangle_{\rho, x \mapsto a, y \mapsto b}, \\ \langle\!\langle z : \sigma \rangle\!\rangle_\rho, \\ - \end{array} \right)^{\circledast}_{\sigma} \langle\!\langle n : \mathsf{nat} \rangle\!\rangle_\rho \\
\langle\!\langle \lambda x. m[x] : \sigma \to \tau \rangle\!\rangle_\rho &\triangleq a \mapsto \langle\!\langle m[x] : \tau \rangle\!\rangle_{\rho, x \mapsto a} \\
\langle\!\langle m \bullet_\sigma n : \tau \rangle\!\rangle_\rho &\triangleq \langle\!\langle m : \sigma \to \tau \rangle\!\rangle_\rho\, \langle\!\langle n : \sigma \rangle\!\rangle_\rho
\end{aligned}
$$

Figure 7: The interpretation of **System** $\mathbb{T}$ terms into dialogues.

We will need to lift the Kleisli extension $(-)^\star$ to apply at higher type; for a type σ *type* and a map $f \in X \to \mathcal{V}\,\langle\!\langle \sigma \rangle\!\rangle$, we have the lifted Kleisli extension $f^{\circledast}_\sigma \in \mathfrak{E}^{\mathbb{N}}_{\mathbb{N}}(X) \to \mathcal{V}\,\langle\!\langle \sigma \rangle\!\rangle$, defined as follows:

$$
\begin{aligned}
f^{\circledast}_\sigma &\in \mathfrak{E}^{\mathbb{N}}_{\mathbb{N}}(X) \to \mathcal{V}\,\langle\!\langle \sigma \rangle\!\rangle \\
f^{\circledast}_\iota(d) &\triangleq f^\star(d) \\
f^{\circledast}_{\sigma \to \tau}(d) &\triangleq s \mapsto f(-, s)^{\circledast}_\tau(d)
\end{aligned}
$$

The interpretation $\langle\!\langle m : \sigma \rangle\!\rangle_\rho \in \mathcal{V}\,\langle\!\langle \sigma \rangle\!\rangle$ for an environment $\rho \in \mathcal{G}\,\langle\!\langle \Gamma \rangle\!\rangle$, presupposing $\Gamma \vdash m : \sigma$, is essentially a monadic or "effectful" version of the standard interpretation given in Figure 5; we summarize the dialogue interpretation in Figure 7.

3.4 Coherence of interpretations

The standard semantics and the interactive semantics cohere for closed terms of atomic type ι *atype* in the sense that the following diagram commutes for any $\alpha \in \mathbb{N}^{\mathbb{N}}$:

$$
\begin{tikzcd}
& \cdot \vdash \iota \arrow[dl, "\langle\!\langle - : \iota \rangle\!\rangle"'] \arrow[dr, "[\![- : \iota]\!]"] & \\
\mathcal{V}\,\langle\!\langle \iota \rangle\!\rangle \arrow[rr, "- \diamond \alpha"'] & & \mathcal{V}\,[\![\iota]\!]
\end{tikzcd}
\tag{3}
$$

In order to prove this, we will need to actually prove a stronger lemma for open terms at higher type using *logical relations*. Logical relations, also known as *Tait's method* or the *method of computability*, is a technique in which a predicate on closed terms at base type is extended uniformly over open terms at higher type. As with our current situation, usually the theorem that one wants to prove has to do with only closed terms of base type; however, in order to prove such a thing by induction, it is necessary to strengthen the motive in this way.

$$\boxed{v \,\mathcal{R}^{\alpha}_{\sigma}\, d \textit{ presupposing } \alpha \in \mathbb{N}^{\mathbb{N}},\ \sigma \textit{ type},\ v \in \mathcal{V}[\![\sigma]\!],\ d \in \mathcal{V}\langle\!\langle\sigma\rangle\!\rangle}$$

$$\boxed{\rho_0 \,\overline{\mathcal{R}^{\alpha}_{\Gamma}}\, \rho_1 \textit{ presupposing } \alpha \in \mathbb{N}^{\mathbb{N}},\ \Gamma \textit{ ctx},\ \rho_0 \in \mathcal{G}[\![\Gamma]\!],\ \rho_1 \in \mathcal{G}\langle\!\langle\Gamma\rangle\!\rangle}$$

$$\frac{v = d \diamond \alpha}{v \,\mathcal{R}^{\alpha}_{\iota}\, d} \qquad \frac{\forall v \in \mathcal{V}[\![\sigma]\!], e \in \mathcal{V}\langle\!\langle\sigma\rangle\!\rangle.\ v \,\mathcal{R}^{\alpha}_{\sigma}\, e \implies f(v) \,\mathcal{R}^{\alpha}_{\tau}\, d(e)}{f \,\mathcal{R}^{\alpha}_{\sigma\to\tau}\, d}$$

$$\frac{\forall x \in |\Gamma|.\ \rho_0(x) \,\mathcal{R}^{\alpha}_{\Gamma(x)}\, \rho_1(x)}{\rho_0 \,\overline{\mathcal{R}^{\alpha}_{\Gamma}}\, \rho_1}$$

Figure 8: Definition of the logical relations between our two interpretations.

We define our logical relation $\mathcal{R}^{\alpha}_{\sigma}$ for $\alpha \in \mathbb{N}^{\mathbb{N}}$ between the (values, environments) of each interpretation in Figure 8. It will be useful to prove an auxiliary lemma about $(-)^{\circledast}_{\sigma}$, following [7].

Lemma 3.3. *Fix* ι *atype,* σ *type,* $f \in \mathcal{V}[\iota] \to \mathcal{V}[\![\sigma]\!]$, $d \in \mathcal{V}[\iota] \to \mathcal{V}\langle\!\langle\sigma\rangle\!\rangle$, $v \in \mathcal{V}[\iota]$, *and* $e \in \mathcal{V}\langle\!\langle\iota\rangle\!\rangle$. *Then, we may infer:*

$$\frac{v \,\mathcal{R}^{\alpha}_{\iota}\, e \quad \forall k \in \mathcal{V}[\iota].\ f(k) \,\mathcal{R}^{\alpha}_{\sigma}\, d(k)}{f(v) \,\mathcal{R}^{\alpha}_{\sigma}\, d^{\circledast}_{\sigma}(e)}$$

Proof. By induction on σ *type*. If σ is atomic, then our goal holds by Lemma 3.2; otherwise, by the inductive hypothesis. □

Theorem 3.1. *The standard and interactive interpretations of each* ***System*** $\mathbb{T}$*-definable term are related by* $\mathcal{R}$*, assuming environments related by* $\overline{\mathcal{R}}$*. More precisely, for any* $\Gamma \vdash M : \sigma$, $\rho_0 \in \mathcal{G}[\![\Gamma]\!]$ *and* $\rho_1 \in \mathcal{G}\langle\!\langle\Gamma\rangle\!\rangle$ *such that* $\rho_0 \,\overline{\mathcal{R}^{\alpha}_{\Gamma}}\, \rho_1$, *we have* $[\![M : \sigma]\!]_{\rho_0} \,\mathcal{R}^{\alpha}_{\sigma}\, \langle\!\langle M : \sigma\rangle\!\rangle_{\rho_1}$.

Proof. By case on the term M.

Case $M \equiv x : \sigma$. Because $\rho_0 \,\overline{\mathcal{R}^{\alpha}_{\Gamma}}\, \rho_1$, we also have $\rho_0(x) \,\mathcal{R}^{\alpha}_{\sigma}\, \rho_1(x)$.

Case $M \equiv \mathsf{rec}_{\sigma}([x, y].s[x, y]; z; n) : \sigma$. Let us begin with some auxiliary definitions:

$$S_0 \triangleq (a, b) \mapsto [\![s : \sigma]\!]_{\rho_0, x\mapsto a, y\mapsto b}$$
$$S_1 \triangleq (a, b) \mapsto \langle\!\langle s : \sigma\rangle\!\rangle_{\rho_1, x\mapsto a, y\mapsto b}$$

$$\begin{aligned} Z_0 &\triangleq [\![z : \sigma]\!]_{\rho_0} & \qquad Z_1 &\triangleq \langle\!\langle z : \sigma\rangle\!\rangle_{\rho_1} \\ N_0 &\triangleq [\![n : \mathsf{nat}]\!]_{\rho_0} & N_1 &\triangleq \langle\!\langle n : \mathsf{nat}\rangle\!\rangle_{\rho_1} \\ R_0 &\triangleq \mathit{Rec}(S_0, Z_0, -) & R_1 &\triangleq \mathit{Rec}(S_1 \circ \eta, Z_1, -) \end{aligned}$$

By backward chaining through Lemma 3.3, letting $\iota \equiv \mathsf{nat}$, $f \equiv R_0$, $d \equiv R_1$, $v \equiv N_0$ and $e \equiv N_1$, it suffices to show the following:

1. $N_0 \; \mathcal{R}^{\alpha}_{\mathsf{nat}} \; N_1$: by the inductive hypothesis for n.
2. For any $k \in \mathbb{N}$, $R_0(k) \; \mathcal{R}^{\alpha}_{\sigma} \; R_1(k)$: by induction on k, applying the inductive hypotheses for z and s in the base case and inductive step respectively.

All remaining cases follow from their inductive hypotheses and Lemma 3.1. □

Corollary 3.1. *Diagram 3 commutes for any* ι *atype and* $\alpha \in \mathbb{N}^{\mathbb{N}}$.

Proof. By Theorem 3.1, instantiated at the type ι and the empty environment. □

4 Validity of the Bar Thesis for $\mathbb{T}$-definable bars

Recall the definition of a bar from Section 1 (Equation 1):

$$\vec{u} \lhd Q \triangleq \forall \alpha \succ \vec{u}.\ \exists k \in \mathbb{N}.\ \overline{\alpha}\,[k] \in Q$$

It is well-known that we cannot hope to prove the Bar Thesis (Proposition 1.3) for this definition of barhood, but our experience with the interactive semantics of **System** $\mathbb{T}$ suggests that we might prove a slightly weaker rule, by requiring the premise to be *realized* by a **System** $\mathbb{T}$-definable functional.

If we interpret the quantifiers constructively and functionally, this is the same as to say that we have a functional $\cdot \vdash f : \mathsf{seq} \to \mathsf{nat}$ which computes the length of an approximation that is in the bar (recall from Figure 4 that $\mathsf{seq} \triangleq \mathsf{nat} \to \mathsf{nat}$). To apply such a functional to a meta-level choice sequence, let us exploit the standard semantics defined in Section 3.1:

$$f\,\langle\alpha\rangle \triangleq [\![f : \mathsf{seq} \to \mathsf{nat}]\!]\,(\alpha)$$

Now, we can define a new **System** $\mathbb{T}$-centric notion of barhood:

$$\vec{u} \lhd_{\mathbb{T}} Q \triangleq \exists \cdot \vdash f : \mathsf{seq} \to \mathsf{nat}.\ \forall \alpha \succ \langle\rangle.\ \overline{\vec{u} \Cap \alpha}\,[f\,\langle\alpha\rangle + |\vec{u}|] \in Q$$

Proposition 4.1 (Bar Thesis for **System** $\mathbb{T}$). *The Bar Thesis for **System** $\mathbb{T}$ states that for any monotone species $Q \subseteq B$ of nodes in the Baire spread, the inductive definition of barhood is complete in the sense that we can conclude $\vec{u} \blacktriangleleft Q$ from $\vec{u} \lhd_{\mathbb{T}} Q$.*

4.1 Brouwer's ephemeral dialogues

The content of Brouwer's purported (but failed) proof of his Bar Thesis was to assert that one can analyze the evidence for barhood into a well-founded mental construction [24, 6]; Escardó's translation of **System** $\mathbb{T}$ terms into dialogue trees is essentially a formalization of Brouwer's insight.

However, Escardó's dialogues differ from Brouwer's mental constructions of barhood, which are captured precisely by the judgment $\vec{u} \blacktriangleleft Q$, in one crucial respect: whereas Escardó's trees branch on an arbitrary query to the ambient choice sequence, queries in Brouwer's mental constructions must be made in order, i.e. with respect to the current moment in ideal time; moreover, the Brouwerian dialogues are *ephemeral*—with each query, the head of the ambient choice sequence is consumed and the remainder of the dialogue is interpreted with respect to the tail of the choice sequence.

Our task, then, will be to normalize Escardó's dialogues into Brouwer's ephemeral mental constructions, and then show how to massage these into a derivation of $\vec{u} \blacktriangleleft Q$. Below we define the set of Brouwerian dialogues $\mathfrak{B}_Y(Z)$ (coding functionals $Y^{\mathbb{N}} \to Z$) as the least set closed under the rules in Figure 9.

$$\boxed{\mathfrak{B}_Y(-) : \mathbf{Set} \to \mathbf{Set}} \qquad \frac{z \in Z}{\eta_{\mathfrak{B}}(z) \in \mathfrak{B}_Y(Z)}\ \textbf{spit} \qquad \frac{b \in Y \to \mathfrak{B}_Y(Z)}{\digamma(b) \in \mathfrak{B}_Y(Z)}\ \textbf{bite}$$

Figure 9: Rules for forming Brouwerian ephemeral dialogues; compare with the inductive barhood judgment $\vec{u} \blacktriangleleft Q$ in Figure 2.

We will design an inductive/proof-theoretic characterization of the normalizable Escardó dialogues, and then show that all Escardó dialogues can be coded as such. This yields a constructive and structurally recursive normalization algorithm.

To this end, we define below two mutually inductive forms of judgment, whose rules are given in Figure 10:

1. $\vec{u} \Vdash d \rightsquigarrow b$, presupposing $\vec{u} \in Y^\star$ and $d \in \mathfrak{E}_Y^{\mathbb{N}}(Z)$, and guaranteeing $b \in \mathfrak{B}_Y(Z)$, means that the Escardó dialogue d normalizes to the Brouwerian dialogue b.
2. $\vec{u} \Vdash \beta\langle i\rangle\,(d) \ll \vec{v} \rightsquigarrow b$ presupposes $\vec{u}, \vec{v} \in Y^\star$, $i \in \mathbb{N}$ and $d \in Y \to \mathfrak{E}_Y^{\mathbb{N}}(Z)$, and guarantees $b \in \mathfrak{B}_Y(Z)$. This auxiliary form of judgment can be thought of as searching for the appropriate moment to insert a query to the oracle.

Lemma 4.1. *The inductive characterization of normalization $\vec{u} \Vdash d \rightsquigarrow b$ is functional, i.e. for any $\vec{u} \in Y^\star$ and $d \in \mathfrak{E}_Y^{\mathbb{N}}(Z)$ we can exhibit some unique $b \in \mathfrak{B}_Y(Z)$ such that $\vec{u} \Vdash d \rightsquigarrow b$.*

Proof. Simultaneously, we must also show that $\vec{u} \Vdash \beta\langle i\rangle\,(d) \ll \vec{v} \rightsquigarrow b$ is functional. We will begin with $\vec{u} \Vdash d \rightsquigarrow b$, proceeding by case on $d \in \mathfrak{E}_Y^{\mathbb{N}}(Z)$.

Case $d \equiv \eta(z)$. By $\mathbf{norm}_\eta$, we have $b \equiv \eta_{\mathfrak{B}}(z)$.

Case $d \equiv \beta\langle i\rangle(e)$. By induction, we have $\vec{u} \Vdash \beta\langle i\rangle\,(e) \ll \vec{u} \rightsquigarrow b$; apply $\mathbf{norm}_\beta$.

Next, we tackle $\vec{u} \Vdash \beta\langle i\rangle\,(d) \ll \vec{v} \rightsquigarrow b$ by simultaneous induction on $\vec{v} \in Y^\star$ (viewed as a *cons*-list) and $i \in \mathbb{N}$.

Case $\vec{v} \equiv \langle\rangle$, $i \equiv 0$. By induction, we have $\vec{u}\frown y \Vdash d(y) \rightsquigarrow b(y)$ for any $y \in Y$; apply $\mathbf{norm}_\beta^{\langle\rangle,z}$.

Case $\vec{v} \equiv \langle\rangle$, $i \equiv j+1$. By induction, we have $\vec{u}\frown y \Vdash \beta\langle j\rangle\,(d) \ll \langle\rangle \rightsquigarrow b(y)$ for any $y \in Y$; apply $\mathbf{norm}_\beta^{\langle\rangle,s}$.

Case $\vec{v} \equiv y :: w$, $i \equiv 0$. By induction, we have $\vec{u} \Vdash d(y) \rightsquigarrow b$; apply $\mathbf{norm}_\beta^{::,z}$.

$$\boxed{\vec{u} \Vdash d \rightsquigarrow b \ \textit{presupposing} \ \vec{u} \in Y^{\star},\ d \in \mathfrak{E}^{\mathbb{N}}_{Y}(Z),\ b \in \mathfrak{B}_{Y}(Z)}$$

$$\boxed{\vec{u} \Vdash \beta\langle i\rangle\,(d) \ll \vec{v} \rightsquigarrow b \ \textit{presupposing} \ \vec{u},\vec{v} \in Y^{\star},\ i \in \mathbb{N},\ d \in Y \to \mathfrak{E}^{\mathbb{N}}_{Y}(Z),\ b \in \mathfrak{B}_{Y}(Z)}$$

$$\frac{}{\vec{u} \Vdash \eta(z) \rightsquigarrow \eta_{\mathfrak{B}}(z)}\ \textbf{norm}_{\eta} \qquad \frac{\vec{u} \Vdash \beta\langle i\rangle\,(d) \ll \vec{u} \rightsquigarrow b}{\vec{u} \Vdash \beta\langle i\rangle(d) \rightsquigarrow b}\ \textbf{norm}_{\beta}$$

$$\frac{\vec{u} \Vdash d(y) \rightsquigarrow b}{\vec{u} \Vdash \beta\langle 0\rangle\,(d) \ll y :: \vec{v} \rightsquigarrow b}\ \textbf{norm}^{::,z}_{\beta} \qquad \frac{\vec{u} \Vdash \beta\langle i\rangle\,(d) \ll \vec{v} \rightsquigarrow b}{\vec{u} \Vdash \beta\langle i+1\rangle\,(d) \ll y :: \vec{v} \rightsquigarrow b}\ \textbf{norm}^{::,s}_{\beta}$$

$$\frac{\forall y \in Y.\ \vec{u} \frown y \Vdash d(y) \rightsquigarrow b(y)}{\vec{u} \Vdash \beta\langle 0\rangle\,(d) \ll \langle\rangle \rightsquigarrow \digamma(b)}\ \textbf{norm}^{\langle\rangle,z}_{\beta} \qquad \frac{\forall y \in Y.\ \vec{u} \frown y \Vdash \beta\langle i\rangle\,(d) \ll \langle\rangle \rightsquigarrow b(y)}{\vec{u} \Vdash \beta\langle i+1\rangle\,(d) \ll \langle\rangle \rightsquigarrow \digamma(b)}\ \textbf{norm}^{\langle\rangle,s}_{\beta}$$

Figure 10: Rules for dialogue normalization.

Case $\vec{v} \equiv y :: w$, $i \equiv j+1$. By induction, we have $\vec{u} \Vdash \beta\langle j\rangle\,(d) \ll \vec{v} \rightsquigarrow b$; apply $\textbf{norm}^{::,s}_{\beta}$.

□

Corollary 4.1 (Normalization Algorithm). *We have a structurally recursive function* $\mathit{norm}_{\vec{u}}(d)$ *such that for all* $\vec{u} \in Y^{\star}$ *and* $d \in \mathfrak{E}^{\mathbb{N}}_{Y}(Z)$, $\vec{u} \Vdash d \rightsquigarrow \mathit{norm}_{\vec{u}}(d)$.

Proof. This is the constructive content of Lemma 4.1. □

4.2 Execution Semantics for $\mathfrak{B}_{Y}(Z)$

Just as we showed how to execute an Escardó dialogue against a choice sequence in Section 3.2, we can do the same for the Brouwerian, ephemeral version. For $b \in \mathfrak{B}_{Y}(Z)$ and $\alpha \in Y^{\mathbb{N}}$, we define $b \diamond_{\mathfrak{B}} \alpha \in Z$ by recursion on b as follows:

$$\begin{aligned} - \diamond_{\mathfrak{B}} - &\in \mathfrak{B}_{Y}(Z) \times Y^{\mathbb{N}} \to Z \\ \eta_{\mathfrak{B}}(z) \diamond_{\mathfrak{B}} \alpha &\triangleq z \\ \digamma(b) \diamond_{\mathfrak{B}} \alpha &\triangleq b(\mathsf{head}(\alpha)) \diamond_{\mathfrak{B}} \mathsf{tail}(\alpha) \end{aligned}$$

Lemma 4.2. *Execution of dialogues coheres with normalization, as defined in Corollary 4.1; to be precise, the following diagram commutes for all* $\vec{u} \in Y^{\star}$ *and* $\alpha \in Y^{\mathbb{N}}$.

$$\begin{tikzcd} \mathfrak{E}^{\mathbb{N}}_{Y}(Z) \arrow[rr, "\mathit{norm}_{\vec{u}}(-)"] \arrow[dr, "- \diamond \vec{u} \mathrel{\in\!\!\!+} \alpha"'] & & \mathfrak{B}_{Y}(Z) \arrow[dl, "- \diamond_{\mathfrak{B}} \alpha"] \\ & Z & \end{tikzcd} \tag{4}$$

When $\vec{u} \equiv \langle\rangle$*, this becomes the statement that* $- \diamond \alpha = - \diamond_{\mathfrak{B}} \alpha \circ \mathit{norm}(-)$.

Proof. We have to show that for any $\vec{u} \in Y^\star$, $\alpha \in Y^{\mathbb{N}}$ and $d \in \mathfrak{E}^{\mathbb{N}}_{Y}(Z)$, we have $d \diamond \vec{u} \Subset \alpha = \mathit{norm}_{\vec{u}}(d) \diamond_{\mathfrak{B}} \alpha$. This follows by straightforward induction on the normalization of d. □

We can compose Diagrams 3, 4 to see a birds' view of the state of affairs concerning interpretation, normalization and execution. For any ι *atype* and $\alpha \in Y^{\mathbb{N}}$, the following diagram commutes:

$$\begin{tikzcd}
\cdot \vdash \iota \arrow[r, "{[\![- : \iota]\!]}"] \arrow[d, "{\langle\!\langle - : \iota \rangle\!\rangle}"'] & \mathcal{V}[\iota] \\
\mathfrak{E}^{\mathbb{N}}_{Y}(\mathcal{V}[\iota]) \arrow[ur, "- \diamond \alpha"] \arrow[r, "\mathit{norm}(-)"'] & \mathfrak{B}_{Y}(\mathcal{V}[\iota]) \arrow[u, "- \diamond_{\mathfrak{B}} \alpha"']
\end{tikzcd} \tag{5}$$

4.3 The Generic Point

In the dialogue model, we can define a so-called "generic point" which is not definable in **System** $\mathbb{T}$:

$$\begin{aligned}
&\mathbf{generic} \in \mathfrak{E}^{X}_{Y}(X) \to \mathfrak{E}^{X}_{Y}(Y) \\
&\mathbf{generic} \triangleq (\beta\langle - \rangle(\eta))^{\star}
\end{aligned}$$

In particular, note that we have $\mathbf{generic} \in \mathcal{V}\langle\!\langle \mathsf{nat} \to \mathsf{nat} \rangle\!\rangle$. Intuitively, by applying the dialogue interpretation of a functional $\cdot \vdash \phi : \mathsf{seq} \to \mathsf{nat}$ to this generic point, we get a dialogue tree $\langle\!\langle \phi : \mathsf{seq} \to \mathsf{nat} \rangle\!\rangle(\mathbf{generic}) \in \mathcal{V}\langle\!\langle \mathsf{nat} \rangle\!\rangle \equiv \mathfrak{E}^{\mathbb{N}}_{\mathbb{N}}(\mathbb{N})$ which is precisely the trace of ϕ's interaction with the ambient choice sequence. Then, assuming that ϕ witnesses $\vec{u} \lhd_{\mathbb{T}} Q$, we can compute the derivation of $\vec{u} \blacktriangleleft Q$ by induction on this trace.

Lemma 4.3. *The generic point commutes with dialogue execution in the following sense:*

$$\begin{tikzcd}
\mathfrak{E}^{X}_{Y}(X) \arrow[r, "\mathbf{generic}"] \arrow[d, "- \diamond \alpha"] & \mathfrak{E}^{X}_{Y}(Y) \arrow[d, "- \diamond \alpha"] \\
X \arrow[r, "\alpha"] & Y
\end{tikzcd}$$

Proof. Immediate by induction on the dialogue tree. □

4.4 Brouwer's Bar Theorem

We may now prove the Bar Theorem for **System** $\mathbb{T}$-definable bars.

Theorem 4.2 (Proposition 4.1)**.** *For any monotone species* $Q \subseteq B$ *of nodes in the Baire spread, we can conclude* $\vec{u} \blacktriangleleft Q$ *from* $\vec{u} \lhd_{\mathbb{T}} Q$.

Proof. By inversion on the premise, we must have some $\cdot \vdash f : \mathsf{seq} \to \mathsf{nat}$ such that for all $\alpha \succ \langle\rangle$, we know $\overline{\vec{u} \Subset \alpha}\,[[\![f : \mathsf{seq} \to \mathsf{nat}]\!]\ \alpha + |\vec{u}|] \in Q$. Let $d \triangleq \langle\!\langle f : \mathsf{seq} \to \mathsf{nat} \rangle\!\rangle\,(\mathbf{generic})$ and $b \triangleq \mathit{norm}(d)$; then, by coercing along Diagram 5 and Lemma 4.3, we have a proof $\mathfrak{P}(\alpha)$ that $\overline{\vec{u} \Subset \alpha}\,[b \diamond_{\mathfrak{B}} \alpha + |\vec{u}|] \in Q$. We proceed by induction on $b \in \mathfrak{B}_{\mathbb{N}}(\mathbb{N})$.

Case $b \equiv \eta_{\mathfrak{B}}(0)$. In this case, we are already in the bar. Let $0\cdots$ be the choice sequence $\alpha(i) \triangleq 0$; from $\mathfrak{P}(0\cdots)$, we know $\overline{\vec{u} \Subset 0\cdots}\,[0 + |\vec{u}|] \in Q$, which is the same as $\vec{u} \in Q$; therefore, apply η. Note that the choice of $0\cdots$ was completely arbitrary, since at this stage, we have stopped consuming from the choice sequence.

Case $b \equiv \eta_{\mathfrak{B}}(k+1)$. We have not yet reached the bar, and may step in any direction to approach it. Apply $\digamma$, fixing $y \in \mathbb{N}$; then, we want to apply our inductive hypothesis at $\eta_{\mathfrak{B}}(k)$. It suffices to show that for any $\alpha \succ \langle\rangle$, we have $\overline{\vec{u}\frown y \Subset \alpha}\,[k + 1 + |\vec{u}|] \in Q$; this follows from $\mathfrak{P}(y :: \alpha)$, and from the fact that $\vec{u}\frown y \Subset \alpha = \vec{u} \Subset y :: \alpha$.

Case $b \equiv \digamma(b')$. Apply $\digamma$, fixing $y \in \mathbb{N}$; apply the inductive hypothesis at $b'(y)$. Now, fixing $\alpha \succ \langle\rangle$, it suffices to show that $\overline{\vec{u}\frown y \Subset \alpha}\,[b'(y) \diamond_{\mathfrak{B}} \alpha + 1 + |\vec{u}|] \in Q$. By $\mathfrak{P}(y :: \alpha)$, we have $\overline{\vec{u} \Subset y :: \alpha}\,[b'(y) \diamond_{\mathfrak{B}} \alpha + |\vec{u}|] \in Q$; because $\vec{u} \Subset y :: \alpha = \vec{u}\frown y \Subset \alpha$, we only have to show that we will remain in Q if we take one more element from the composite choice sequence. But this is precisely what it means for Q to be monotone, and so we are done.

□

5 Formalization in Agda

This paper amounts to an "unformalization" of a completely formal development[1] in the Agda proof assistant [17], using Darin Morrison's alternative Agda basis library [16]. We owe a lot to Martín Escardó's original formalizations of effectful forcing in Agda [7]. It is important to emphasize that the proof has been completely effected within the intensional dialect of Martin-Löf's Constructive Type Theory as implemented in Agda, without postulating any further principles.

6 Related Work

Schwichtenberg's closure theorem In 1979, Schwichtenberg proved an even stronger result than what we have proved here, namely that for any closed **System** $\mathbb{T}$ term which codes a bar of type 0 and 1, the *bar recursor* can already be defined in **System** $\mathbb{T}$ [22]. In more recent work, Oliva and Steila give an elegant and *direct* proof of Schwichtenberg's result [18].

[1] The full development is available here: https://github.com/jonsterling/agda-effectful-forcing

It should be possible to replicate this result in our setting by using Church encodings of Escardó dialogues, which can easily be defined in **System** $\mathbb{T}$. Escardó has used this technique to exhibit the modulus of continuity of a **System** $\mathbb{T}$ functional as a program in **System** $\mathbb{T}$ [7].

Forcing in type theory Aside from Escardó, whose results we have discussed in detail already, there has been a great deal of work related to (traditional) forcing in type theory over the past several years. Coquand and Jaber have combined forcing with realizability to obtain a version of type theory which validates the uniform continuity of functionals on the Cantor space [4]; Coquand and Mannaa have also used a similar technique to demonstrate the independence of Markov's Principle from dependent type theory [5].

Howe's computational open-endedness In his remarkable paper, *On Computational Open-Endedness in Martin-Löf's Type Theory* [12], Douglas Howe demonstrated that the computation systems of Martin-Löf-style type theories may be extended with infinitary rules, for instance, to endow type theory with oracles or classical / set-theoretic functions, Brouwerian free choice sequences, etc.

Howe's result is crucial for justifying the adequacy of Type Theory as a semi-formal theory of constructions in which to realize Brouwer's vision of mathematical activity, in which the (at least) potential existence of non-constructive (non-algorithmic) operations is absolutely essential (see [8, 9, 10]).

Bar induction in Nuprl A weak form of the bar induction principle has been added as a basic axiom in Nuprl's proof theory; consequently, via a bootstrapping technique, stronger forms of the bar principle are also made to hold in Nuprl, including both the monotone and decidable bar principles [19, 21]. As a result, the Fan Theorem is true in Nuprl, and has been used to establish the uniform continuity of all functionals on the Cantor spread [20].

Traditionally, the soundness of Nuprl's proof theory with respect to its partial equivalence relation semantics has been established by completely constructive means. With the addition of the bar induction principle, however, Nuprl's semantics must be performed in a classical metatheory. We are curious whether the technique used here can be extended to a language that supports universal computation, like Nuprl; if so, this may provide a path toward recovering the constructive character of Nuprl's semantics.

In order to prove the validity of their bar induction rule, all (classical) sequences of numerals have been added directly to Nuprl's computation system; this is essentially an iteration of what has been done in the present work, and by Escardó [7], and is justified by Howe's open-endedness result [12]. It corresponds to the well-known fact that the bar principle cannot hold if functions are restricted to the computable (recursive, algorithmic) operations [23].

7 What is the significance of System $\mathbb{T}$?

The most common way to construct a model of a formal system in which some non-constructive principle holds is to use sheaf semantics, in which types are interpreted as sheaves on a topological space (in our case, the Baire space $\mathsf{B} \triangleq \mathbb{N}^{\mathbb{N}}$), and predicates are interpreted as subsheaves of the types which form their range of significance.[2] This is the semantical technique to which Escardó's *effectful forcing* is the syntactic analogue.

When considering our proof in context, it is reasonable to ask what the specific role of **System** $\mathbb{T}$ was, as opposed to some other theory of constructions (possibly incorporating general recursion and other computational effects). Our view is that the answer to these questions comes down essentially to the fact that programs in **System** $\mathbb{T}$ code their own termination proofs, and what's more, these termination arguments are valid in *any* constructive metatheory, including those in which the bar principle is refuted.

As a result, it is possible to consider quite concretely the extension of **System** $\mathbb{T}$ with bar induction, by simply adding an operator to the language which denotes the generic point which we defined in Section 4.3, **generic** $\in \mathcal{V}\,\langle\!\langle \mathsf{nat} \to \mathsf{nat} \rangle\!\rangle$. What about using **PCF** as a theory of constructions—or, more generally, interpreting predicates as relations defined over a *partial* programming language as is done in Martin-Löf's type theory [14, 1]?

Here the question becomes more complicated, since such languages can *already* code any computationally effective principle, including bar recursion and Markov's principle, which are both represented by unbounded search. Thus the matter which must be dealt with shifts from *definability* to *totality*, which once again depends entirely on the characteristics of the ambient metatheory. To force the bar recursor to be total a suitable domain (which is the same as to say that it shall realize the bar induction principle), then, we must ensure that the judgment T *terminates* is interpreted in a suitable metatheory or judgmental apparatus in which the bar principle already holds.

The above amounts to defining the ambient judgmental apparatus of our theory of constructions using sheaf semantics: in particular, the *derivations* that a particular program realizes a predicate will form a sheaf on the Baire space, and the inductive character of these derivations which is induced by the locality laws for sheaves will enable us to extract a suitable Escardó dialogue, which can then be processed into a Brouwerian dialogue in the way that we have described in Section 4.1, Figure 10.

This is in essence the technique used to construct a model of Martin-Löf's Type Theory which refutes Markov's principle in [5], though the authors do not explain fully the relationship with standard sheaf semantics. This method can also be seen as a direct formalization and modernization of Brouwer's "creating subject arguments" [6].

In future work, we intend to use this method to construct an interpretation of Martin-Löf's type theory in which Brouwer's Thesis is upheld on a purely constructive basis, an improvement on the current state of affairs in Nuprl [19, 21].

[2] Sheaf-theoretic forcing is not only useful for validating Brouwerian principles; if one has committed to a *constructive* ambient metatheory, classical principles can be interpreted by forcing over the double-negation topology, as explained in [13].

Acknowledgements

Thanks to Thierry Coquand, Martín Escardó, Mark van Atten, Bob Harper, Mark Bickford and Bob Constable for helpful conversations about the Bar Theorem; thanks to Vincent Rahli for reading an earlier draft of this paper and suggesting a significant simplification of the apparatus.

References

[1] R. L. Constable, S. F. Allen, H. M. Bromley, W. R. Cleaveland, J. F. Cremer, R. W. Harper, D. J. Howe, T. B. Knoblock, N. P. Mendler, P. Panangaden, J. T. Sasaki, and S. F. Smith. *Implementing Mathematics with the Nuprl Proof Development System*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1986. 18

[2] T. Coquand. About Brouwer's fan theorem. *Revue Internationale de Philosophie*, 58(230 (4)):483–489, 2004. 1

[3] T. Coquand. Personal correspondence, January 2016. 1

[4] T. Coquand and G. Jaber. A computational interpretation of forcing in type theory. In P. Dybjer, S. Lindstrm, E. Palmgren, and G. Sundholm, editors, *Epistemology versus Ontology*, volume 27 of *Logic, Epistemology, and the Unity of Science*, pages 203–213. Springer Netherlands, 2012. 17

[5] T. Coquand and B. Mannaa. The independence of Markov's principle in type theory. Accepted to Formal Structures for Computation and Deduction, 2016. 17, 18

[6] M. Dummett. *Elements of intuitionism*, volume 39 of *Oxford Logic Guides*. The Clarendon Press Oxford University Press, New York, second edition, 2000. 5, 6, 12, 18

[7] M. Escardó. Continuity of Gödel's System T definable functionals via effectful forcing. *Electronic Notes in Theoretical Computer Science*, 298:119–141, 2013. Agda development: `http://www.cs.bham.ac.uk/~mhe/dialogue/dialogue-lambda.html`. 1, 7, 9, 11, 16, 17

[8] M. P. Fourman. Notions of choice sequence. In D. van Dalen and A. Troelstra, editors, *L.E.J. Brouwer Centenary Symposium*, pages 91–105. North-Holland, 1982. 17

[9] M. P. Fourman. Continuous Truth I: Non-constructive objects. In G. L. G. Lolli and A. Marcja, editors, *Logic Colloquium '82*, volume 112 of *Studies in Logic and the Foundations of Mathematics*, pages 161 – 180. Elsevier, 1984. 17

[10] M. P. Fourman. Continuous Truth II: Reflections. In L. Libkin, U. Kohlenbach, and R. de Queiroz, editors, *Logic, Language, Information, and Computation: 20th International Workshop, WoLLIC 2013, Darmstadt, Germany, August 20-23, 2013.*

Proceedings, pages 153–167, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. 17

[11] K. Gödel. Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica*, 12(3-4):280–287, 1958. 6

[12] D. Howe. On computational open-endedness in Martin-Löf's type theory. In *Logic in Computer Science, 1991. LICS '91., Proceedings of Sixth Annual IEEE Symposium on*, pages 162–172, July 1991. 17

[13] S. Mac Lane and I. Moerdijk. *Sheaves in geometry and logic : a first introduction to topos theory*. Universitext. Springer, New York, 1992. 18

[14] P. Martin-Löf. *Intuitionistic type theory*, volume 1 of *Studies in Proof Theory*. Bibliopolis, 1984. 18

[15] P. Martin-Löf. Truth of a proposition, evidence of a judgement, validity of a proof. *Synthese*, 73(3):407–420, 1987. 5

[16] D. Morrison. `agda-prelude`: A simple prelude for Agda. https://github.com/freebroccolo/agda-prelude, 2016. 16

[17] U. Norell. Dependently typed programming in agda. In *Proceedings of the 4th International Workshop on Types in Language Design and Implementation*, TLDI '09, pages 1–2, New York, NY, USA, 2009. ACM. 1, 16

[18] P. Oliva and S. Steila. A direct proof of Schwichtenberg's bar recursion closure theorem. Under review, 2016. 16

[19] V. Rahli and M. Bickford. Coq as a metatheory for Nuprl with bar induction. In *Continuity, Computability, Constructivity – From Logic to Algorithms*, 2015. 17, 18

[20] V. Rahli and M. Bickford. A nominal exploration of Intuitionism. In *Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs*, CPP 2016, pages 130–141, New York, NY, USA, 2016. ACM. 17

[21] V. Rahli, M. Bickford, and R. Constable. A story of bar induction in Nuprl. http://www.nuprl.org/html/Nuprl2Coq/, 2016. 17, 18

[22] H. Schwichtenberg. On bar recursion of types 0 and 1. *Journal of Symbolic Logic*, 44(3):325fi?!329, Sep 1979. 1, 16

[23] A. Troelstra and van Dalen, Dirk. *Constructivism in mathematics: an introduction. Volume I.* Studies in logic and the foundations of mathematics. North-Holland, Amsterdam, New-York, Oxford, 1988. 17

[24] M. van Atten. *On Brouwer*. Wadsworth Philosophers Series. Thompson/Wadsworth, Toronto, Canada, 2004. 12