RESEARCH

Game semantics of Martin-Löf type theory

Norihiro Yamada

Correspondence:
yamad041@umn.edu
University of Minnesota,
Minneapolis, MN 55455, USA
Full list of author information is available at the end of the article

Abstract

We present new *game semantics* of *Martin-Löf type theory (MLTT)* equipped with One-, Zero-, N-, Pi-, Sigma- and Id-types as well as universes. It interprets both Id-type and universes in the presence of N-type for the first time as game semantics. Its another advantage is its interpretation of Sigma-type that is *direct* and *compatible with the game semantics of linear logic*. Also, its mathematical structure is quite novel and useful; e.g., the category of our games has all *finite limits*, which is a key step for an extension of the present work to *homotopy type theory*. Finally, we provide a new, game-semantic proof of the *independence of Markov's principle* from MLTT, which demonstrates an advantage of our game semantics over extensional models of MLTT such as realisability models.

Keywords: game semantics; Martin-Löf type theory; constructive mathematics

AMS Subject Classification: primary 03F65; secondary 03B70

Contents

1 Introduction

1.1 Martin-Löf type theory and the meaning explanation

Martin-Löf type theory (MLTT) [1, 2, 3] is one of the best-known formal systems for *constructive mathematics* [4], which is comparable to set theory [5, 6] for classical mathematics. MLTT is also a functional programming language [7] that is a generalisation of the *simply-typed lambda-calculus (STLC)* [8] along the generalisation of (intuitionistic) propositional logic to predicate logic under the *Curry-Howard isomorphisms* [9]. By this *computational* nature, MLTT enables computer formalisations of mathematics and its applications to programming [10].

Like set theory is explained informally by *sets*, the conceptual foundation of MLTT is *computations* in an informal sense. That is, the fundamental idea of MLTT is to regard objects and proofs in constructive mathematics uniformly as computations, and MLTT is a syntactic formalisation of this foundational idea [7]. Hence, objects and proofs in MLTT are unified into *terms*, while formulas are called *types*. This standard, informal semantics of MLTT is called the *meaning explanation* [11, §5].

However, MLTT is not always the best formalisation of this conceptual foundation of constructive mathematics since it is an intricate formal system that inevitably contains superficial syntactic details.[1] In other words, the intuition behind MLTT is often blurred by the complexity and the syntactic nature of MLTT. In addition, the syntactic complexity makes it difficult to study the meta-theory of MLTT.

Accordingly, *mathematical semantics* [12] of MLTT that faithfully formalises the meaning explanation is strongly desired since such semantics would accurately and directly (or non-inductively) describe the intuition behind MLTT, abstracting the syntactic details. It would not only deepen our understanding of MLTT in this way but also suggest improvements and extensions of MLTT just like *coherence spaces* by Girard [13] led to *linear logic* [13], and the *groupoid model* by Hofmann and Streicher [14] to *homotopy type theory (HoTT)* [15]. Besides, mathematical semantics has been highly effective for the meta-theoretic study of MLTT; for instance, see [16].

[1] A syntactic formalisation is also unsatisfactory from the *syntax-first-view*, i.e., the view that semantic concepts come first, and syntax merely provides notations.

1.2 Game semantics

Game semantics [17, 18] is a particular class of mathematical semantics of logic and computation that models types and terms by *games* and *strategies*, respectively.

A strong point of game semantics is its conceptual naturality: One may think of logic as consisting of *dialogical arguments* between *Player* and *Opponent*, and game semantics formalises this idea, providing a conceptually very natural, computational explanation of logic. This game-semantic view on logic is also in harmony with the meaning explanation since dialogical arguments are a certain kind of *computations*. Another advantage of game semantics is its precision in modelling syntax as various *full completeness/abstraction* results [19] in the literature have demonstrated; initial results are [20, 21, 22, 23]. Moreover, the computational nature of game semantics enables its algorithmic applications to program analysis and verification [24].

1.3 Main results

To summarise the points so far, mathematical semantics of MLTT for advancing our understanding of MLTT, promoting its improvements and extensions, and studying its meta-theory is strongly desired, and game semantics seems perfect for this role by its conceptual naturality, harmony with the meaning explanation, precision in modelling syntax and applications in program analysis and verification. Also, game semantics models *effects* [25] and *linear typing* [13] (see [26] for the details); thus, game semantics of MLTT may lead to MLTT with effects and linear typing.

However, although game semantics of various logics and computations has been given, it is difficult to establish game semantics of MLTT. In fact, this problem had been open for more than twenty years, and even today its definitive solution is yet to emerge though a few candidates have arisen recently [27, 28]; see §1.4.

Hence, we aim to provide another candidate for game semantics of MLTT with the hope that it would shed new light on this problem and eventually lead to a definitive solution. Motivated in this way, we prove the following main result:

Theorem (Game semantics of MLTT) *There is new game semantics of MLTT with One-, Zero-, N-, Pi-, Sigma- and Id-types as well as universes (§4.4–4.5).*

See §1.4 for the advantages of this game semantics over existing methods [27, 28].

By this theorem, we also give a new proof of the independence result of [29]:

Corollary (Independence of Markov's principle [29]) *Markov's principle [30] is independent from MLTT equipped with the types listed in Theorem 1.3 (§4.7).*

The method used by Mannaa and Coquand [29] is *syntactic*, while our semantic approach provides a new, intuitive argument on why the independence holds. This corollary also illustrates an advantage of game semantics over other computational models since, e.g., Hyland's *effective topos* [31] cannot show the independence.

1.4 Related work and our contributions

Abramsky et al. have established the first game semantics of MLTT equipped with One-, Pi-, Sigma-, Id- and finite inductive types [27, 32]. Its significance is that it is the first *intensional* model of MLTT and hence stands in sharp contrast with other computational models such as realisability and domain models [16, 33] since they are extensional. This game semantics is based on the classic *AJM-games* [21], and their main result is a certain kind of *full completeness*. On the other hand, they do not interpret universes. Notably, although they present game semantics of Pi-type (by simply adapting the game semantics of universal quantification [34] to Pi-type), they merely interpret Sigma-type in terms of the interpretation of Pi-type inductively via a formal adjunction between the two types. In other words, they do not directly achieve game semantics of Sigma-type. As a result, they interpret types $\Pi_{\Sigma_{x:A}B}C$ and $\Pi_{x:A}\Pi_{y:B}C$ (or $A \times B \Rightarrow C$ and $A \Rightarrow B \Rightarrow C$) *by the same game* even though they are clearly distinct (yet isomorphic) types. Another undesirable feature of their approach is that they interpret types and terms respectively by *lists* of games and *lists* of strategies due to the interpretation of Sigma-type, for which it is somewhat misleading to call them games and strategies. This use of lists drops to some degree the conceptual naturality and the mathematical elegance of game semantics. In particular, if one wonders *what is the game-semantic counterpart of the generalisation of STLC to MLTT*, then this approach of generalising games and strategies to their respective lists does not appear to be very insightful.

Another related work is the denotational model of MLTT [28] by Blot and Laird based on *sequential algorithms* and *concrete date structures* [35], *graph games* [36] and *event structures* [37]. They propose an *intensional* universe for the construction of recursive types, and relate it with the ordinary *extensional* universe. They interpret these universes, and Boolean-, Pi- and Sigma-types, but not Id-types. Their main results are certain *full completeness* and *full abstraction*. Notably, they *directly* interpret Sigma-type in terms of intrinsic rules of graph games without recourse to the list construction, overcoming the problem of the preceding work. On the other hand, it is possible to play on *both* of the first and the second components of their interpretation of Sigma-type within a single play, which is far from the graph game semantics [36] or the traditional game semantics [21, 22, 38] of product type. Hence, it seems arguable if their model corresponds to the game-semantic counterpart of the generalisation of product type to Sigma-type. As a more pragmatic aspect, their model *validates* Markov's principle (§4.7). That is, there is a *gap* between the semantics and the syntax of MLTT. More generally, their model admits *non-local control operators* or *classical reasonings*, while the logical part of MLTT is *intuitionistic*. Also, their model does not achieve the *linear decomposition* of function type [13] or the *characterisation of effects* in terms of constraints on strategies, which are both strong advantages of game semantics [26]. Finally, their method of interpreting Id-type by finite tuples of Boolean-type sketched in [28, §9] would not work in the presence of N-type since the set of all natural numbers is unbounded.

Thus, each of the existing approaches to game semantics of MLTT has pros and cons, and we have not reached a consensus on which option should be a definitive solution. In this context, we offer the third method with the novel features listed below, hoping that it would eventually lead to a definitive solution.

First, we achieve game semantics of *both* Id-type and universes in the presence of N-type for the first time. In particular, it is nontrivial to model the introduction rule of universes with respect to Id-type, but we attain it by a novel technique (§4.5.6).

Second, our games are a modest generalisation of a standard variant, *McCusker games* [38], and we model types and terms by such games and strategies, not lists of them. Also, our interpretation of Sigma-type is *direct* and *generalises that of product type* by McCusker games, which overcomes the aforementioned shortcomings of the preceding methods. Consequently, our method clarifies a game-semantic counterpart of the generalisation of STLC to MLTT; see the beginning of §3 and §3.1.

Third, our game semantics inherits the linear decomposition of function type and the characterisation of effects by constraints on strategies in McCusker games [26].

Finally, strategies are traditionally defined in terms of games underlying them, but we *reverse* this relation between games and strategies similarly to Girard's *ludics* [39]. This is our key idea on how to achieve game semantics of MLTT; again, see the beginning of §3. The resulting mathematical structure is quite novel and useful. For instance, the category of our games has all *finite limits* (Corollary 3.22), while that of existing games does not. This structure enables us to internalise a certain notion of ∞-groupoids in the category of our games, which is a key step to extend the present work to HoTT [40]. Besides, the present work has led to the *consistency of MLTT with Church's thesis* [41], which is a long-standing problem in constructive mathematics open for fifteen years [42, 43]. That is, our biggest contribution is to establish the mathematical foundation of these innovative results.

1.5 Concluding remarks

Due to the ludics-like structure, one might misunderstand that our method is rather close to the realisability, domain, or even set model. However, it is *not* the case for the highly *intensional* nature of our game semantics. For instance, the category of our games is not well-pointed. Also, the intensional features of the preceding game semantics [27, §1] are all applied to our game semantics as well (§4.6); e.g., it refutes function extensionality. Finally, our model refutes Markov's principle by its intensionality, showing the independence (§4.7). In contrast, the effective topos validates Markov's principle due to its extensionality, unable to verify the independence.

1.6 Structure of the present article

The rest of this article proceeds as follows. We first recall games and strategies in §2 and generalise them in §3. We then interpret MLTT by these generalised games and strategies in §4, where we analyse the intensionality of our model in §4.6, and provide a new proof of the independence of Markov's principle from MLTT in §4.7.

2 Games and strategies for simple type theories

We first recall McCusker's games and strategies for simple type theories [38], which the present work is based on. We select this variant for the following reasons. First, it combines the strong points of the two best-known variants: the *linear decomposition* of function type [13] achieved by *AJM-games* [21] and the characterisation of *effects* by constraints on strategies [26] that utilises *pointers* in *HO-games* [22] (originally introduced in [44]). Our games inherit these advantages so that it would shed new light on the problem of combining MLTT with linear logic and/or effects. Second, pointers enable us to refine game semantics into a *model of computation* [45], which is highly desirable as a mathematical foundation of *constructive* mathematics (§1.1).

We assume that the reader is familiar with McCusker's games and strategies, and leave more expositions and examples to the gentle introduction [26]. We first recall two preliminary concepts in §2.1, and then games and strategies in §2.2. We finally recall standard constructions on games and strategies in §2.3.

Remark McCusker's game semantics of *of-course* ! [13] is *ad-hoc* since it does not form a comonad [38, pp. 47–48]. We can, however, remedy this problem by adding an equivalence relation between positions to each game [38, §3.6]. We use this method in the preprint of this article [46] and show that the resulting games have elegant categorical structures. Nevertheless, although we prefer this mathematical elegance, the equivalence relations bring significant technical overheads, and the preprint is lengthy. Hence, this article employs the simpler variant of McCusker games.

Notation We use the following notations throughout the present article:

- We use bold small letters $\boldsymbol{s}, \boldsymbol{t}, \boldsymbol{u}, \boldsymbol{v}$, etc. for sequences, in particular $\boldsymbol{\epsilon}$ for the *empty sequence*, and small letters a, b, m, n, x, y, etc. for elements of sequences;
- We define $\overline{n} := \{ 1, 2, \dots, n \}$ for each $n \in \mathbb{N}^+ := \mathbb{N} \setminus \{0\}$, and $\overline{0} := \emptyset$;
- We abbreviate a sequence $\boldsymbol{s} = (x_1, x_2, \dots, x_{|\boldsymbol{s}|})$ as $x_1 x_2 \dots x_{|\boldsymbol{s}|}$, where $|\boldsymbol{s}|$ is the *length* (i.e., the number of elements) of $\boldsymbol{s}$, and write $\boldsymbol{s}(i)$ for x_i ($i \in \overline{|\boldsymbol{s}|}$);
- A *concatenation* of sequences $\boldsymbol{s}$ and $\boldsymbol{t}$ is represented by the juxtaposition $\boldsymbol{st}$ of them, but we often write $a\boldsymbol{s}$, $\boldsymbol{t}b$, $\boldsymbol{u}c\boldsymbol{v}$ for $(a)\boldsymbol{s}$, $\boldsymbol{t}(b)$, $\boldsymbol{u}(c)\boldsymbol{v}$, and so on;
- We write $\boldsymbol{s}.\boldsymbol{t}$ for $\boldsymbol{st}$ if it increases readability, and $\boldsymbol{s}^n := \underbrace{\boldsymbol{ss}\cdots\boldsymbol{s}}_{n}$ for each $n \in \mathbb{N}$;
- We write $\mathrm{Even}(\boldsymbol{s})$ (resp. $\mathrm{Odd}(\boldsymbol{s})$) if $\boldsymbol{s}$ is of even- (resp. odd-) length, and given a set S of sequences and $\mathsf{P} \in \{\mathrm{Even}, \mathrm{Odd}\}$, we define $S^{\mathsf{P}} := \{ \boldsymbol{s} \in S \mid \mathsf{P}(\boldsymbol{s}) \}$;
- We write $\boldsymbol{s} \preceq \boldsymbol{t}$ if $\boldsymbol{s}$ is a *prefix* of $\boldsymbol{t}$, and given a set S of sequences, $\mathrm{Pref}(S)$ for the set of all prefixes of sequences in S, i.e., $\mathrm{Pref}(S) := \{ \boldsymbol{s} \mid \exists \boldsymbol{t} \in S.\, \boldsymbol{s} \preceq \boldsymbol{t} \}$;
- We define $X^* := \{ x_1 x_2 \dots x_n \mid n \in \mathbb{N}, \forall i \in \overline{n}.\, x_i \in X \}$ for each set X;
- We write $f \upharpoonright S$ for the *restriction* of a map $f : A \to B$ to a subset $S \subseteq A$;
- Given sets $X_1, X_2, \dots, X_n$, and an index $i \in \overline{n}$, we write $\pi_i^{(n)}$ or π_i for the i^{th}*-projection map* $X_1 \times X_2 \times \dots \times X_n \to X_i$.

2.1 Arenas and legal positions

A *game* is a certain kind of a dag whose branches represent possible 'developments' or *(valid) positions* in the 'game in the ordinary sense' (such as chess and poker). These branches are finite sequences of vertices or *moves* connected by edges; a play of the game proceeds as its participants alternately (and separately) perform moves along a branch. We focus on standard *two-person* games between ***Player (P)*** (or a 'prover') and ***Opponent (O)*** (or a 'refuter') in which *O always starts a play.*

More technically, games are based on two preliminary concepts: *arenas* and *legal positions*. An arena defines the basic components of a game, which in turn induces legal positions of the arena that specify the basic rules of the game in the sense that each position of the game must be legal. Let us first recall these two concepts.

Definition 2.1 (Moves) Let us fix, throughout the present work, arbitrary pairwise distinct symbols O, P, Q and A, and call them ***labels***. A ***move*** is any triple

$m^{xy} := (m, x, y)$ such that $x \in \{\mathsf{O}, \mathsf{P}\}$ and $y \in \{\mathsf{Q}, \mathsf{A}\}$, for which we often abbreviate m^{xy} as m, and instead define $\lambda(m) := xy$, $\lambda^{\mathsf{OP}}(m) := x$ and $\lambda^{\mathsf{QA}}(m) := y$.

We call a move m an ***O-move*** if $\lambda^{\mathsf{OP}}(m) = \mathsf{O}$, a ***P-move*** if $\lambda^{\mathsf{OP}}(m) = \mathsf{P}$, a ***question*** if $\lambda^{\mathsf{QA}}(m) = \mathsf{Q}$, and an ***answer*** if $\lambda^{\mathsf{QA}}(m) = \mathsf{A}$.

Definition 2.2 (Arenas [22, 38]) An ***arena*** is a pair $G = (M_G, \vdash_G)$ such that

- M_G is a set of moves;
- $\vdash_G$ is a subset of the cartesian product $(\{\star\} \cup M_G) \times M_G$, where $\star$ (or represented more precisely by $\star_G$) is an arbitrarily fixed element such that $\star \notin M_G$, called the ***enabling relation***, that satisfies
  - (E1) If $\star \vdash_G m$, then $\lambda(m) = \mathsf{OQ}$;
  - (E2) If $m \vdash_G n$ and $\lambda^{\mathsf{QA}}(n) = \mathsf{A}$, then $\lambda^{\mathsf{QA}}(m) = \mathsf{Q}$;
  - (E3) If $m \vdash_G n$ and $m \neq \star$, then $\lambda^{\mathsf{OP}}(m) \neq \lambda^{\mathsf{OP}}(n)$.

We call moves $m \in M_G$ ***initial*** if $\star \vdash_G m$, and set $M_G^{\mathrm{Init}} := \{\, m \in M_G \mid \star \vdash_G m \,\}$.

An arena G is ***well-founded (w.f.)*** if $\vdash_G$ is well-founded, i.e., there is no sequence $(m_i)_{i \in \mathbb{N}}$ of moves $m_i \in M_G$ such that $\star \vdash_G m_0$ and $m_i \vdash_G m_{i+1}$ for all $i \in \mathbb{N}$.

Remark In the original article [38], an arena is a triple $G = (M_G, \lambda_G, \vdash_G)$, and labels are *assigned* to moves by the *labelling function* $\lambda_G : M_G \to \{\mathsf{O}, \mathsf{P}\} \times \{\mathsf{Q}, \mathsf{A}\}$. Instead, we *embed* labels into moves in an arena in Definition 2.2; this modification is convenient for our generalisation of games in §3 since then labels on each move are unambiguous *globally without underlying arenas*. Also, the axiom E1 in [38] further requires $n \vdash_G m \Leftrightarrow n = \star$ whenever $\star \vdash_G m$. We discard this condition again for the generalisation of games. See §3, especially Footnote 3 and Definition 3.3.

An arena G specifies moves in a game, each of which is O's/P's question/answer, and which move n can be performed for each move m during a play in the game by the relation $m \vdash_G n$ (cf. Definition 2.3), where $\star \vdash_G m$ means that O can initiate a play by m in the game. The axioms E1, E2 and E3 are then to be read as follows:

- E1 sets the convention that an initial move must be O's question;
- E2 states that an answer must be performed for a question;
- E3 says that an O-move must be performed for a P-move, and vice versa.

We next review *legal positions*, a certain class of finite sequences of moves equipped with *pointers* from later to earlier occurrences in the sequences. The idea is that each non-initial occurrence in a legal position must be made for a specific previous occurrence, and pointers specify such pairs of occurrences. Technically, pointers enable us to distinguish similar yet different plays [26, §2.4] and define *views* (Definition 2.4) crucial for legal positions and constraints on strategies (Definition 2.9).

We call a finite sequence of moves together with a pointer a *justified (j-) sequence*. A legal position is then a particular kind of a j-sequence.

Definition 2.3 (Justified sequences [44, 38]) An ***occurrence*** in a finite sequence $\boldsymbol{s}$ is a pair $(\boldsymbol{s}(i), i)$ such that $i \in \overline{|\boldsymbol{s}|}$. A ***justified (j-) sequence*** is a pair $\boldsymbol{s} = (\boldsymbol{s}, \mathcal{J}_{\boldsymbol{s}})$ of a finite sequence $\boldsymbol{s}$ of moves and a map $\mathcal{J}_{\boldsymbol{s}} : \overline{|\boldsymbol{s}|} \to \{0\} \cup \overline{|\boldsymbol{s}| - 1}$ such that $0 \leqslant \mathcal{J}_{\boldsymbol{s}}(i) < i$ for all $i \in \overline{|\boldsymbol{s}|}$, called the ***pointer*** of the j-sequence. An occurrence $(\boldsymbol{s}(i), i)$ is ***initial*** in $\boldsymbol{s}$ if $\mathcal{J}_{\boldsymbol{s}}(i) = 0$. We say that the occurrence $(\boldsymbol{s}(\mathcal{J}_{\boldsymbol{s}}(i)), \mathcal{J}_{\boldsymbol{s}}(i))$ is the ***justifier*** of a non-initial one $(\boldsymbol{s}(i), i)$ in $\boldsymbol{s}$, and $(\boldsymbol{s}(i), i)$ is ***justified*** by $(\boldsymbol{s}(\mathcal{J}_{\boldsymbol{s}}(i)), \mathcal{J}_{\boldsymbol{s}}(i))$ in $\boldsymbol{s}$.

A j-sequence $\boldsymbol{s}$ is ***in an arena*** G if its elements are moves in G, and its pointer respects the enabling in G, i.e., $\boldsymbol{s} \in M_G^* \wedge \forall i \in \overline{|\boldsymbol{s}|}. \Big(\mathcal{J}_{\boldsymbol{s}}(i) = 0 \Rightarrow \star \vdash_G \boldsymbol{s}(i)\Big) \wedge \Big(\mathcal{J}_{\boldsymbol{s}}(i) \neq 0 \Rightarrow \boldsymbol{s}(\mathcal{J}_{\boldsymbol{s}}(i)) \vdash_G \boldsymbol{s}(i)\Big)$. We write $\mathscr{J}_G$ for the set of all j-sequences in G.

A ***justified (j-) subsequence*** of a j-sequence $\boldsymbol{s}$ is a j-sequence $\boldsymbol{t}$, written $\boldsymbol{t} \sqsubseteq \boldsymbol{s}$, such that $\boldsymbol{t}$ is a subsequence of $\boldsymbol{s}$, and $\mathcal{J}_{\boldsymbol{t}}(i) = j \Leftrightarrow \exists n \in \mathbb{N}^+. \mathcal{J}_{\boldsymbol{s}}^n(i) = j$.

Remark Unlike the original formulation [38], we have defined j-sequences in such a way that they make sense *even without underlying arenas*. This point becomes important for Definition 2.4–2.7, and in turn for the generalisation of games in §3.

Convention Henceforth, we are casual about the distinction between moves and occurrences, and by abuse of notation, we frequently keep the pointer $\mathcal{J}_{\boldsymbol{s}}$ of each j-sequence $\boldsymbol{s} = (\boldsymbol{s}, \mathcal{J}_{\boldsymbol{s}})$ implicit since it is mostly obvious, and abbreviate occurrences $(\boldsymbol{s}(i), i)$ in $\boldsymbol{s}$ as $\boldsymbol{s}(i)$. Besides, we often write $\mathcal{J}_{\boldsymbol{s}}(\boldsymbol{s}(i)) = \boldsymbol{s}(j)$ if $\mathcal{J}_{\boldsymbol{s}}(i) = j > 0$.

Next, the following is crucial for legal positions and constraints on strategies:

Definition 2.4 (Views [44, 22, 38]) The ***P-view*** $\lceil \boldsymbol{s} \rceil$ and the ***O-view*** $\lfloor \boldsymbol{s} \rfloor$ of a j-sequence $\boldsymbol{s}$ are the j-subsequences of $\boldsymbol{s}$ defined by the following induction:

- $\lceil \boldsymbol{\epsilon} \rceil := \boldsymbol{\epsilon}$;
- $\lceil \boldsymbol{s}m \rceil := \lceil \boldsymbol{s} \rceil . m$ if m is a P-move;
- $\lceil \boldsymbol{s}m \rceil := m$ if m is initial;
- $\lceil \boldsymbol{s}m\boldsymbol{t}n \rceil := \lceil \boldsymbol{s} \rceil . mn$ if n is an O-move such that m justifies n;
- $\lfloor \boldsymbol{\epsilon} \rfloor := \boldsymbol{\epsilon}$;
- $\lfloor \boldsymbol{s}m \rfloor := \lfloor \boldsymbol{s} \rfloor . m$ if m is an O-move;
- $\lfloor \boldsymbol{s}m\boldsymbol{t}n \rfloor := \lfloor \boldsymbol{s} \rfloor . mn$ if n is a P-move such that m justifies n.

A ***P-view*** (resp. an ***O-view***) refers to that of some j-sequence, and a ***view*** (of a j-sequence) to a P- or O-view (of the j-sequence).

The idea on views is as follows. Given a nonempty j-sequence $\boldsymbol{s}m$ such that m is a P- (resp. O-) move, the P-view $\lceil \boldsymbol{s} \rceil$ (resp. O-view $\lfloor \boldsymbol{s} \rfloor$) is the currently 'relevant part' of the previous occurrences in $\boldsymbol{s}$ for P (resp. O). I.e., P (resp. O) is concerned only with the last occurrence of an O- (resp. P-) move, its justifier and that justifier's P- (resp. O-) view, which then recursively proceeds. See [44] for an explanation of views in terms of their counterparts in logical calculi, and [47, 48] in lambda-calculi.

We are now ready to recall *legal positions*:

Definition 2.5 (Legal positions [38, 26]) A ***legal position*** is a j-sequence $\boldsymbol{s}$ that satisfies

- (Alternation) If $\boldsymbol{s} = \boldsymbol{s_1}mn\boldsymbol{s_2}$, then $\lambda^{\mathsf{OP}}(m) \neq \lambda^{\mathsf{OP}}(n)$;
- (Visibility) If $\boldsymbol{s} = \boldsymbol{t}m\boldsymbol{u}$ with m non-initial, then $\mathcal{J}_{\boldsymbol{s}}(m)$ occurs in the P-view $\lceil \boldsymbol{t} \rceil$ if m is a P-move, and in the O-view $\lfloor \boldsymbol{t} \rfloor$ otherwise.

A legal position is ***in an arena*** G if it is a j-sequence in G (Definition 2.3). We write $\mathscr{L}_G$ for the set of all legal positions in G.

Remark For the same reason as the case of j-sequences, we have generalised legal positions in such a way that they make sense *even without underlying arenas*.

As already noted, legal positions are to specify the basic rules of a game in the sense that each position in the game must be legal (Definition 2.6) so that

- During a play in the game, O makes the first move by a question (by E1), and then P and O alternately make moves (by alternation), where each non-initial move is made for a specific previous occurrence, viz., its justifier;
- The justifier of each non-initial occurrence belongs to the 'relevant part' or view of the previous occurrences (by visibility).

2.2 Games and strategies

We are now ready to recall *games* and deterministic games called *strategies*, which are slight modifications of those given in the original article [38].

Definition 2.6 (Games) A ***game*** is a set G of legal positions, called ***positions*** in G that is nonempty and *prefix-closed* (i.e., $\boldsymbol{s}m \in G \Rightarrow \boldsymbol{s} \in G$). It is ***well-founded (w.f.)*** if the arena $\mathrm{Arn}(G) := (M_G, \vdash_G)$ is w.f., where $M_G := \{\, \boldsymbol{s}(i) \mid \boldsymbol{s} \in G, i \in \overline{|\boldsymbol{s}|}\,\}$ and $\vdash_G := \{\, (\star, \boldsymbol{s}(j)) \mid \boldsymbol{s} \in G, \mathcal{J}_{\boldsymbol{s}}(j) = 0\,\} \cup \{\, (\boldsymbol{s}(i), \boldsymbol{s}(j)) \mid \boldsymbol{s} \in S, \mathcal{J}_{\boldsymbol{s}}(j) = i\,\}$.

A ***subgame*** of G is a game H such that $H \subseteq G$.

Remark The original article [38, p. 27] also imposes *thread-closure* on games G: The *thread* $\boldsymbol{s} \upharpoonright \mathcal{I} \sqsubseteq \boldsymbol{s}$ of a given position $\boldsymbol{s} \in G$ with respect to a given set $\mathcal{I}$ of initial occurrences in $\boldsymbol{s}$, which consists of occurrences hereditarily justified by elements in $\mathcal{I}$, must be in G. This axiom is to ensure that positions in G are in the *exponential* $!G$ (Definition 2.12), i.e., $G \subseteq !G$, which matches the intuition on exponential ! [13]. However, $!G$ is a well-defined game even if G is not thread-closed. Moreover, we shall soon focus on *well-opened* games (Definition 2.15), which are trivially thread-closed. For these reasons, we have omitted the thread-closure axiom in Definition 2.6.

Each game G is nonempty and prefix-closed since conceptually each nonempty position or 'moment' in G has a previous 'moment.' As mentioned before, positions in G are automatically *legal in the arena* $\mathrm{Arn}(G)$. We shall later focus on w.f. games because identities in the categories of w.f. games are *noetherian* (§2.3).

The tuple $\mathcal{M}(G) := (M_G, \lambda_G, \vdash_G, G)$, where $\lambda_G : m \mapsto \lambda(m)$, forms a game in the sense of [38][2] whose labels are embedded into moves, which we call an ***MC-game***. $\mathcal{M}(G)$ satisfies: Each move $m \in M_G$ occurs in a position, and each pair $m_1 \vdash_G m_2$ occurs as an occurrence m_2 and its justifier m_1 in a position. Conversely, given an MC-game H that satisfies these conditions, for which we call H ***economical***, the set of all positions in H forms a game. Besides, these operations are inverses to each other. Hence, $\mathcal{M}$ is a *bijection* between games and economical MC-games.

Since the economical axioms only exclude unused structures from MC-games, our simplification of MC-games into games made in Definition 2.6 is *harmless*. Moreover, this simplification will be very convenient when we generalise games in §3.

Definition 2.7 (Strategies) A ***strategy*** is a game σ that is *deterministic*, i.e., $\boldsymbol{s}mn, \boldsymbol{s}mn' \in \sigma^{\mathrm{Even}} \Rightarrow \boldsymbol{s}mn = \boldsymbol{s}mn'$. A strategy σ is ***on a game*** G, written $\sigma : G$, if $\sigma \subseteq G$ and $(\boldsymbol{s}m \in G^{\mathrm{Odd}} \wedge \boldsymbol{s} \in \sigma) \Rightarrow \boldsymbol{s}m \in \sigma$.

[2] Except that the axiom E1 is slightly weakened, and the thread-closure is omitted; see the remarks after Definitions 2.2 and 2.6, respectively.

Thus, a strategy on a game G is a deterministic subgame $\sigma \subseteq G$ such that possible plays by O in σ coincide with those in G. I.e., σ describes for P *how to play on* G.

Given a strategy $\sigma : G$, the subset $\mathcal{M}(\sigma) := \sigma^{\text{Even}} \subseteq \sigma$ is a strategy on the MC-game $\mathcal{M}(G)$ in the sense of [38], which we call an ***MC-strategy***. Conversely, given an MC-strategy τ on $\mathcal{M}(G)$, the union $\tau \cup \{ \boldsymbol{s}m \in G \mid \boldsymbol{s} \in \tau \}$ is a strategy on G. Again, these operations are inverses to each other, and thus strategies and MC-strategies are in a *bijective correspondence* $\mathcal{M}$. Their crucial difference is, however, that MC-strategies need underlying (MC-) games, but strategies do not. This point enables us to generalise games into certain *sets of strategies* later; see §3.

Example 2.8 The ***terminal game*** $T := \{\boldsymbol{\epsilon}\}$ only has the strategy $\top := \{\boldsymbol{\epsilon}\}$.

The ***flat game*** on a set S is the game $\text{flat}(S) := \text{Pref}(\{ q^{\mathsf{OQ}}.m^{\mathsf{PA}} \mid m \in S \})$, where q is an arbitrarily fixed element with $q \notin S$, and q^{OQ} justifies m^{PA}. For instance, there are the ***empty game*** $\mathbf{0} := \text{flat}(\emptyset)$, the ***unit game*** $\mathbf{1} := \text{flat}(\{*\})$, where $\{*\}$ is any singleton set, and the ***natural number game*** $N := \text{flat}(\mathbb{N})$. The flat game $\text{flat}(S)$ has strategies $\bot := \{\boldsymbol{\epsilon}, q\}$ and $\underline{m} := \{\boldsymbol{\epsilon}, q, qm\}$ for each $m \in S$.

Next, recall that not every strategy corresponds to a *proof*. For example, the empty game $\mathbf{0}$ models *falsity*, and thus the strategy $\bot : \mathbf{0}$ should not be an interpretation of a proof. We therefore carve out strategies for proofs as *winning* ones:

Definition 2.9 (Constraints on strategies [44, 38, 26]) A strategy σ is

- ***Total*** if it always responds: $\forall \boldsymbol{s}m \in \sigma^{\text{Odd}}. \exists \boldsymbol{s}mn \in \sigma^{\text{Even}}$;
- ***Innocent*** if it depends only on P-views: $\forall \boldsymbol{s}mn \in \sigma^{\text{Even}}, \boldsymbol{s'}m' \in \sigma^{\text{Odd}}. \lceil \boldsymbol{s}m \rceil = \lceil \boldsymbol{s'}m' \rceil \Rightarrow \exists \boldsymbol{s'}m'n' \in \sigma^{\text{Even}}. \lceil \boldsymbol{s}mn \rceil = \lceil \boldsymbol{s'}m'n' \rceil$;
- ***Noetherian*** if there is no strictly increasing infinite sequence of elements (with respect to $\preceq$) in the set $\lceil \sigma \rceil := \{ \lceil \boldsymbol{s} \rceil \mid \boldsymbol{s} \in \sigma \}$ of P-views in σ;
- ***Winning*** if it is total, innocent and noetherian;
- ***Well-bracketed (w.b.)*** if its 'question-answering' in P-views is in the 'last-question-first-answered' fashion (see [26, §3.2.4] for the precise definition).

We leave it to the reader to show that these constraints on strategies are equivalent to those on MC-strategies [26, 38] under the bijection $\mathcal{M}$.

Example 2.10 The strategies $\top$ and $\underline{n}$ for all $n \in \mathbb{N}$ are winning and w.b., while the strategy $\bot$ is not even total, let alone winning.

We think of winning strategies as *proofs in classical logic* as follows. First, proofs should not get 'stuck,' and so strategies for proofs must be *total*. Next, imposing *innocence* on strategies corresponds to excluding *stateful* terms [26, §2.9]. Since logic is concerned with *truths*, which are independent of 'passage of time,' proofs should not depended on 'states of arguments.' Hence, we impose innocence on strategies for proofs. Besides, we need *noetherianity* on strategies to handle infinite plays: If a play by an innocent, noetherian strategy keeps growing infinitely, then it cannot be P's 'intention,' and so the play must be *win* for P. Technically, noetherianity is crucial for closure of winning strategies under *composition* (Definition 2.14) [44].

Further, *well-bracketing* bans classical reasoning or *control operators* [26, §2.10]. Hence, we regard winning, w.b. strategies as *proofs in intuitionistic logic*.

2.3 Constructions on games and strategies

In this section, we briefly recall constructions on games and strategies. Since they are standard in the literature, we leave expositions and examples to [26, §3.2].

Convention We omit 'tags' for disjoint union $\uplus$. For instance, we write $x \in A \uplus B$ if $x \in A$ or $x \in B$; given relations $R_A \subseteq A \times A$ and $R_B \subseteq B \times B$, we write $R_A \uplus R_B$ for the relation on $A \uplus B$ such that $(x, y) \in R_A \uplus R_B \stackrel{\mathrm{df.}}{\Leftrightarrow} (x, y) \in R_A \vee (x, y) \in R_B$.

Definition 2.11 (Constructions on arenas [38]) Given arenas A and B, we define

- $A \uplus B := (M_A \uplus M_B, \vdash_A \uplus \vdash_B)$;
- $A \multimap B := (\{ a^{\overline{x}y} \mid a^{xy} \in M_A \} \uplus M_B, \vdash_{A \multimap B})$, where $\overline{\mathsf{O}} := \mathsf{P}$, $\overline{\mathsf{P}} := \mathsf{O}$, $\star \vdash_{A \multimap B} m :\Leftrightarrow \star \vdash_B m$ and $m \vdash_{A \multimap B} n :\Leftrightarrow m \vdash_A n \vee m \vdash_B n \vee (\star \vdash_B m \wedge \star \vdash_A n)$.

Definition 2.12 (Constructions on games [38]) Given games G and H, we define

- $G \otimes H := \{ \boldsymbol{s} \in \mathscr{L}_{\mathrm{Arn}(G) \uplus \mathrm{Arn}(H)} \mid \forall X \in \{G, H\}. \boldsymbol{s} \upharpoonright X \in X \}$, called the ***tensor*** of G and H, where $\boldsymbol{s} \upharpoonright X \sqsubseteq \boldsymbol{s}$ consists of moves in X;
- $!G := \{ \boldsymbol{s} \in \mathscr{L}_{\mathrm{Arn}(G)} \mid \forall m \in M_G^{\mathrm{Init}}. \boldsymbol{s} \upharpoonright \{m\} \in G \}$, called the ***exponential*** of G, where $\{m\}$ ranges over the singleton set of an initial occurrence in $\boldsymbol{s}$ whose move is m if there is any, and the empty set $\emptyset$ otherwise, and $\boldsymbol{s} \upharpoonright \{m\} \sqsubseteq \boldsymbol{s}$ consists of occurrence in $\boldsymbol{s}$ hereditarily justified by the initial occurrence m;
- $G \mathbin{\&} H := \{ \boldsymbol{s} \in \mathscr{L}_{\mathrm{Arn}(G) \uplus \mathrm{Arn}(H)} \mid (\boldsymbol{s} \upharpoonright G \in G \wedge \boldsymbol{s} \upharpoonright H = \boldsymbol{\epsilon}) \vee (\boldsymbol{s} \upharpoonright G = \boldsymbol{\epsilon} \wedge \boldsymbol{s} \upharpoonright H \in H) \}$, called the ***product*** of G and H;
- $G \multimap H := \{ \boldsymbol{s} \in \mathscr{L}_{\mathrm{Arn}(G) \multimap \mathrm{Arn}(H)} \mid \forall X \in \{G, H\}. \boldsymbol{s} \upharpoonright X \in X \}$ and $G \Rightarrow H := {!G} \multimap H$, called the ***linear implication*** and the ***implication*** from G to H, respectively.

Lemma 2.13 (Well-defined constructions on games) *Games (resp. w.f. games) are closed under $\otimes$, $\multimap$, $\&$ and $!$, and w.o. games are closed under $\&$, $\multimap$ and $\Rightarrow$.*

Proof See [38] for closure of MC-games under these constructions. It is essentially the same for games. The preservation of w.f. (resp. w.o.) games is obvious. □

We leave it to the reader to verify that these constructions on games correspond to those on economical MC-games [38] under the bijection $\mathcal{M}$ (except the case analysis for linear implication between *economical* MC-games [46, Definition 20]).

Definition 2.14 (Constructions on strategies [38]) Given strategies $\phi : A \multimap B$, $\sigma : C \multimap D$, $\tau : A \multimap C$, $\psi : B \multimap C$ and $\theta : {!A} \multimap B$, we define

- $\mathrm{cp}_A := \{ \boldsymbol{s} \in A_{[0]} \multimap A_{[1]} \mid \forall \boldsymbol{t} \preceq \boldsymbol{s}. \mathrm{Even}(\boldsymbol{t}) \Rightarrow \boldsymbol{t} \upharpoonright A_{[0]} = \boldsymbol{t} \upharpoonright A_{[1]} \}$, called the ***copy-cat*** on A, where the subscripts $(\_)_{[i]}$ $(i = 0, 1)$ are 'tags' for clarity;
- $\mathrm{der}_A := \{ \boldsymbol{s} \in {!A} \multimap A \mid \forall \boldsymbol{t} \preceq \boldsymbol{s}. \mathrm{Even}(\boldsymbol{t}) \Rightarrow \boldsymbol{t} \upharpoonright {!A} = \boldsymbol{t} \upharpoonright A \}$, called the ***dereliction*** on A;
- $\phi \otimes \sigma := \{ \boldsymbol{s} \in A \otimes C \multimap B \otimes D \mid \boldsymbol{s} \upharpoonright A, B \in \phi, \boldsymbol{s} \upharpoonright C, D \in \sigma \}$, called the ***tensor*** of ϕ and σ, where $\boldsymbol{s} \upharpoonright A, B \sqsubseteq \boldsymbol{s}$ (resp. $\boldsymbol{s} \upharpoonright C, D \sqsubseteq \boldsymbol{s}$) consists of moves in A or B (resp. C or D);
- $\langle \phi, \tau \rangle := \{ \boldsymbol{s} \in A \multimap B \mathbin{\&} C \mid (\boldsymbol{s} \upharpoonright A, B \in \phi \wedge \boldsymbol{s} \upharpoonright C = \boldsymbol{\epsilon}) \vee (\boldsymbol{s} \upharpoonright A, C \in \tau \wedge \boldsymbol{s} \upharpoonright B = \boldsymbol{\epsilon}) \}$, called the ***pairing*** of ϕ and τ;

- $\phi;\psi := \{ \boldsymbol{s} \upharpoonright A, C \mid \boldsymbol{s} \in \phi \parallel \psi \}$, called the ***composition*** of ϕ and ψ, where $\phi \parallel \psi := \{ \boldsymbol{s} \in \mathscr{J} \mid \boldsymbol{s} \upharpoonright A, B_{[0]} \in \phi, \boldsymbol{s} \upharpoonright B_{[1]}, C \in \psi, \boldsymbol{s} \upharpoonright B_{[0]}, B_{[1]} \in \mathrm{cp}_B \}$, $\mathscr{J} := \mathscr{J}_{\mathrm{Arn}(((A \multimap B_{[0]}) \multimap B_{[1]}) \multimap C)}$, and $\phi;\psi$ is also written $\psi \circ \phi$;
- $\theta^\dagger := \{ \boldsymbol{s} \in {!A} \multimap {!B} \mid \forall m \in M^{\mathrm{Init}}_{!B}.\, \boldsymbol{s} \upharpoonright \{m\} \in \theta \}$, called the ***promotion*** of θ, where $\boldsymbol{s} \upharpoonright \{m\} \sqsubseteq \boldsymbol{s}$ is the thread of $\boldsymbol{s}$ with respect to the singleton set $\{m\}$.

It is well-known that derelictions are in general not well-defined; see [38, pp. 42–43] for the details. To remedy this problem, we have to focus on:

Definition 2.15 (Well-opened games [38]) A game A is ***well-opened (w.o.)*** if the conjunction of $\boldsymbol{s}m \in A$ and $m \in M^{\mathrm{Init}}_A$ implies $\boldsymbol{s} = \boldsymbol{\epsilon}$.

In other words, a game is w.o. if its position has at most one initial occurrence. Although w.o. games are not closed under exponential !, it does not matter for the present work, just like [38], since what we need is implication $\Rightarrow$, not exponential ! itself, and w.o. games are closed under implication $\Rightarrow$ and product & [38, p. 43].

Lemma 2.16 (Well-defined constructions on strategies) *If $\phi : A \multimap B$, $\sigma : C \multimap D$, $\tau : A \multimap C$, $\psi : B \multimap C$ and $\theta : {!A} \multimap B$, then $\mathrm{cp}_A : A \multimap A$, $\phi \otimes \sigma : A \otimes C \multimap B \otimes D$, $\langle \phi, \tau \rangle : A \multimap B \mathbin{\&} C$, $\phi;\psi : A \multimap C$ and $\theta^\dagger : {!A} \multimap {!B}$; $\mathrm{der}_B : {!B} \multimap B$ if B is w.o. Also, cp_B (resp. der_B) is winning and w.b. if B is w.f. (resp. w.o. and w.f.), and the constructions $\otimes$, $\langle \_, \_ \rangle$, $\circ$ and $(\_)^\dagger$ preserve winning and well-bracketing.*

Proof Similar to the case of MC-strategies [38], and so we only show that cp_A is noetherian if A is w.f. (the case of der_B is the same). Note that cp_A is total, innocent and w.b. even if A is not w.f. Given $\boldsymbol{s}mm \in \mathrm{cp}_A$, we see by induction on $|\boldsymbol{s}|$ that the P-view $\lceil \boldsymbol{s}m \rceil$ is of the form $m_1 m_1 m_2 m_2 \dots m_k m_k m$, and so there is a sequence $\star \vdash_A m_1 \vdash_A m_2 \cdots \vdash_A m_{k-1} \vdash_A m_k \vdash_A m$. Thus, cp_A is noetherian if A is w.f. $\square$

Again, we leave it to the reader to verify that these constructions on strategies correspond to those on MC-strategies [38] under the bijection $\mathcal{M}$.

Definition 2.17 (Categories of games) The category $\mathbb{G}$ consists of

- W.o. games as objects;
- Strategies on the implication $A \Rightarrow B$ as morphisms $A \to B$;
- The composition $\psi \bullet \phi := \psi \circ \phi^\dagger : A \Rightarrow C$ of strategies $\phi^\dagger : {!A} \multimap {!B}$ and $\psi : {!B} \multimap C$ as the composition of morphisms $\phi : A \to B$ and $\psi : B \to C$;
- The dereliction der_A as the identity on each object A.

The subcategory $\mathbb{LG}$ (resp. $\mathbb{WG}$) of $\mathbb{G}$ consists of w.f., w.o. games as objects, and winning (resp. winning, w.b.) strategies as morphisms.

Games in $\mathbb{G}$ (resp. $\mathbb{LG}$ and $\mathbb{WG}$) are *w.o.* (resp. *w.o.* and *w.f.*) for identities to be well-defined (Lemma 2.16). Strategies in $\mathbb{G}$ are unconstrained, embodying *programs in computation*. In contrast, strategies in $\mathbb{LG}$ (resp. $\mathbb{WG}$) are winning (resp. winning and w.b.), embodying *proofs in classical logic* (resp. *proofs in intuitionistic logic*).

These categories are *cartesian closed*, where a terminal object, product and exponential objects are the terminal game T, product & and implication $\Rightarrow$, respectively.

We leave it to the reader to verify, based on Lemmata 2.13 and 2.16, that the categories satisfy, similarly to [38, §3.5], the required axioms to be cartesian closed.

Finally, we employ the following notations:

- Given a strategy $\sigma : G$, we write $\sigma^T : T \multimap G$ and $\sigma^{!T} : T \Rightarrow G$ for the evident strategies that coincide with σ up to 'tags';
- Given strategies $\phi : T \multimap G$ and $\phi' : T \Rightarrow G$, we write $\phi_T, \phi'_{!T} : G$ for the evident strategies that coincide with ϕ and ϕ' up to 'tags' respectively;
- Given strategies $\psi : A \multimap B$ and $\alpha : A$, we define $\psi \circ \alpha := (\psi \circ \alpha^T)_T : B$;
- Given strategies $\alpha : A$ and $\beta : B$, we define $\alpha \otimes \beta := ((\alpha^T \otimes \beta^T) \circ \iota)_T : A \otimes B$, where ι is the unique strategy on $T \multimap T \otimes T$, and $\langle \alpha, \beta \rangle := \langle \alpha^T, \beta^T \rangle_T : A \& B$;
- Given a strategy $\alpha : A$, we define $\alpha^\dagger := ((\alpha^{!T})^\dagger)_{!T} : !A$.

3 Predicate games

Having reviewed games and strategies in §2, we initiate our contributions in this section. Before going into details, let us sketch our main idea as follows. In short, the challenge in game semantics of MLTT is not in dependent types but in Sigma-type.

Naively, we can interpret each dependent type $\mathsf{x} : \mathsf{C} \vdash \mathsf{D}(\mathsf{x})$ type by a family $D = (D(\sigma))_{\sigma : C}$ of games $D(\sigma)$ indexed by strategies σ on the game C that models the simple type C. In the presence of Sigma-type, dependent types with a single variable covers those with more than one variable, and hence we can focus on the former.

In light of product & (Definition 2.12), which models a particular kind of Sigma-type, viz., product type, it seems a natural idea to model the Sigma-type $\Sigma_{\mathsf{x}:\mathsf{C}}\mathsf{D}(\mathsf{x})$ by a game $\Sigma(C, D)$ such that $\Sigma(C, D) \subseteq C \uplus \bigcup_{\sigma : C} D(\sigma)$, where $\uplus$ denotes disjoint union, and strategies on $\Sigma(C, D)$ are precisely the pairings $\langle \sigma, \tau \rangle$ of strategies $\sigma : C$ and $\tau : D(\sigma)$. However, this idea does not work due to the following two problems:

1. Each game G, by definition, determines the set of all strategies on G;
2. It is impossible for P, when playing on such a game $\Sigma(C, D)$, if any, to *fix* a strategy $\sigma : C$, let alone a game $D(\sigma)$, at the beginning of a play.

As an example of the first problem, consider a dependent type $\mathsf{x} : \mathsf{N} \vdash \mathsf{N_b}(\mathsf{x})$ type such that canonical terms of the simple type $\mathsf{N_b}(\underline{\mathsf{k}})$ $(k \in \mathbb{N})$ are the numerals $\underline{\mathsf{n}}$ such that $n \leqslant k$. However, there is no game G such that $G \subseteq N \uplus N$, and $\langle \underline{k}, \underline{n} \rangle : G$ if and only if $n \leqslant k$ for all $k, n \in \mathbb{N}$ since if such a game G existed, then $\langle \underline{0}, \underline{0} \rangle, \langle \underline{1}, \underline{1} \rangle : G$, which implies $\langle \underline{0}, \underline{1} \rangle : G$ by the definition of strategies on a game (Definition 2.7), a contradiction. Consequently, no game can properly model the Sigma-type $\Sigma_{\mathsf{x}:\mathsf{N}}\mathsf{N_b}(\mathsf{x})$.

Let us next give an example of the second problem. Let $\mathsf{x} : \mathsf{N} \vdash \mathsf{List_N}(\mathsf{x})$ type be the dependent type such that canonical terms of the simple type $\mathsf{List_N}(\underline{\mathsf{k}})$ $(k \in \mathbb{N})$ are k-lists of numerals, and assume that we model $\mathsf{List_N}$ as the family List_N of games such that $\mathrm{List}_N(\underline{0}) := T$, $\mathrm{List}_N(\underline{n+1}) := \mathrm{List}_N(\underline{n}) \otimes N$ $(n \in \mathbb{N})$ and $\mathrm{List}_N(\bot) := \bigcup_{n \in \mathbb{N}} \mathrm{List}_N(\underline{n})$. If there were a game that models the Sigma-type $\Sigma_{\mathsf{x}:\mathsf{N}}\mathsf{List_N}(\mathsf{x})$, then the pairings $\langle \underline{k}, \underline{n_1} \otimes \underline{n_2} \otimes \cdots \otimes \underline{n_k} \rangle$ for all $k, n_1, n_2, \dots, n_k \in \mathbb{N}$ would be *total* on the game since strategies that interpret proofs must be winning (§2.2). However, there is no such a game since O may select, by his first move on the component game on the right-hand side, e.g., the $(k+1)$-*ary* tensor $\otimes$ of N on the right hand side.

We have sketched the two fundamental limitations of games in modelling Sigma-type. Our main idea on solving this problem is to generalise games to certain *sets of strategies*, not sets of positions, called *predicate (p-) games*. We say that a strategy is *on* a p-game if it is an element of the p-game. To implement this idea smoothly, we have reformulated MC-games by games, which dispense with underlying arenas, and MC-strategies by strategies, which dispense with underlying games (§2).

This generalisation of games to p-games solves the first problem. For instance, the p-game $\Sigma(N, N_b) := \{ \langle \sigma, \tau \rangle \mid \sigma, \tau : N, \forall n \in \mathbb{N}. \sigma = \underline{n} \Rightarrow \tau : N_b(\underline{n}) \}$ models the Sigma-type $\Sigma_{\mathsf{x:N}} \mathsf{N_b}(\mathsf{x})$, where we define $N_b = (N_b(\sigma))_{\sigma : N}$ by $N_b(\bot) := N$ and $N_b(\underline{n}) := \{\bot\} \cup \{ \underline{k} \mid k \leqslant n \}$. Typical plays by the strategy $\langle \underline{7}, \underline{3} \rangle \in \Sigma(N, N_b)$ are

$$\begin{array}{ccc} \Sigma(N, & & N_b) \\ \hline & q_{\Sigma(N,N_b)} & \\ & \langle \underline{7}, \underline{3} \rangle & \\ q & & \\ 7 & & \end{array} \qquad \begin{array}{ccc} \Sigma(N, & & N_b) \\ \hline & q_{\Sigma(N,N_b)} & \\ & \langle \underline{7}, \underline{3} \rangle & \\ & & q \\ & & 3 \end{array}$$

where *Judge (J)* first asks P the question $q_{\Sigma(N,N_b)}$ ('What is your strategy?') and P answers it by the strategy $\langle \underline{7}, \underline{3} \rangle \in \Sigma(N, N_b)$ ('I declare $\langle \underline{7}, \underline{3} \rangle$!'), and then a play in the declared strategy $\langle \underline{7}, \underline{3} \rangle$ between P and O follows. The arrows in the diagram represent pointers in j-sequences (Definition 2.3). Although the declaration of a strategy is not strictly necessary in this example, it is clear why P cannot play by the strategy $\langle \underline{0}, \underline{1} \rangle$ on the p-game $\Sigma(N, N_b)$: $\langle \underline{0}, \underline{1} \rangle \notin \Sigma(N, N_b)$ by the definition of $\Sigma(N, N_b)$. This definition of the p-game $\Sigma(N, N_b)$ is made possible by reversing the traditional relation between games and strategies: P-games are defined by strategies. In this way, p-games solve the first problem of games in modelling Sigma-type.

Moreover, the declaration of a strategy solves the second problem: The p-game $\Sigma(N, \mathrm{List}_N) := \{ \langle \sigma, \tau \rangle \mid \sigma : N, \tau : \bigcup_{k \in \mathbb{N}} \mathrm{List}_N(\underline{k}), \forall n \in \mathbb{N}. \sigma = \underline{n} \Rightarrow \tau : \mathrm{List}_N(\underline{n}) \}$ models the Sigma-type $\Sigma_{\mathsf{x:N}} \mathsf{List_N}(\mathsf{x})$. Typical plays in $\Sigma(N, \mathrm{List}_N)$ look like

$$\begin{array}{ccc} \Sigma(N, & & \mathrm{List}_N) \\ \hline & q_{\Sigma(N,\mathrm{List}_N)} & \\ & \langle \underline{2}, \underline{1} \otimes \underline{3} \rangle & \\ q & & \\ 2 & & \end{array} \qquad \begin{array}{cccc} \Sigma(N, & & & \mathrm{List}_N) \\ \hline & q_{\Sigma(N,\mathrm{List}_N)} & & \\ & \langle \underline{2}, \underline{1} \otimes \underline{3} \rangle & & \\ & & & q \\ & & & 3 \\ & & q & \\ & & 1 & \end{array}$$

where the declaration of the strategy $\langle \underline{2}, \underline{1} \otimes \underline{3} \rangle \in \Sigma(N, \mathrm{List}_N)$ *fixes* the underlying game on the right-hand side List_N (n.b., a strategy is a particular game, specifying its odd-length positions too) so that O must play on the 2-ary tensor $N \otimes N$ there. In this way, the strategy $\langle \underline{2}, \underline{1} \otimes \underline{3} \rangle$ is *total* on the p-game $\Sigma(N, \mathrm{List}_N)$.

Finally, we justify p-games, in particular the use of J, as a generalisation of games as follows. First, J is assumed at least implicitly in conventional games too since there must be someone other than P or O to check if j-sequences made by P and O are *valid* positions in the underlying game. Hence, the use of J for p-games is not a significant departure from games. However, this argument is only *conceptual*, and the following *mathematical* arguments matter much more. Second, the initial two moves played in p-games are by J and P, not O and P, for the *spirit* of game semantics: Computations in games are revealed *only gradually* along the development of plays. That is, the declaration of a strategy by P is hidden from O and visible only to J, so that O can see the strategy *only gradually via his play against the strategy* (as in the case of games). Technically, we implement this idea by keeping the initial two moves *outside of O-views*. Dually, O's play on the domain Γ of each linear implication $\Gamma \multimap \Delta$ between p-games is revealed *only gradually* to P, by which the extensions of strategies on $\Gamma \multimap \Delta$ are *continuous* maps like each linear implication between games; see the paragraph right after Definition 3.13. Besides, p-games are as *intensional* as games and so in contrast with extensional models; see §4.6–4.7.

The rest of the present section proceeds as follows. We first make the limitations of games precise in §3.1 and based on this observation define the fundamental concept of p-games in §3.2. We then generalise constructions on games to p-games in §3.3.

Remark Strictly speaking, we shall interpret dependent types by families of p-games indexed by *winning, w.b.* strategies, not mere strategies, in §4.1. Accordingly, we shall slightly modify the examples List_N and N_b; see Example 4.2.

3.1 Consistency and completeness on sets of strategies

For convenience, we define a nonempty set $\mathcal{S}$ of strategies to be *consistent* if there is a game G such that every element of $\mathcal{S}$ is a strategy on G, or equivalently:

Definition 3.1 (Consistency) A nonempty set $\mathcal{S}$ of strategies is ***consistent*** if $\forall \sigma, \tau \in \mathcal{S}, \boldsymbol{s}m \in (\sigma \cup \tau)^{\mathrm{Odd}}. \boldsymbol{s} \in (\sigma \cap \tau) \Rightarrow \boldsymbol{s}m \in (\sigma \cap \tau)$.

Strategies σ and τ are ***consistent***, written $\sigma \asymp \tau$, if the set $\{\sigma, \tau\}$ is consistent.

Definition 3.1 formulates the intended meaning of consistency: The union $\bigcup \mathcal{S}$ of a consistent set $\mathcal{S}$ of strategies forms a game such that each element of $\mathcal{S}$ is a strategy on $\bigcup \mathcal{S}$, and conversely the set of all strategies on a game is consistent.[3] Besides:

Definition 3.2 (Completeness) A consistent set $\mathcal{S}$ is ***complete*** if every subset $\mathcal{A} \subseteq P_{\mathcal{S}} := \bigcup \mathcal{S}$ is an element of $\mathcal{S}$ whenever it is a strategy on the game $P_{\mathcal{S}}$.

It is not hard to show that the map $G \stackrel{\sim}{\mapsto} \{\, \sigma \mid \sigma : G \,\}$ is a bijection between games G and complete sets $\{\, \sigma \mid \sigma : G \,\}$ of strategies with the inverse $\bigcup$ [46, Theorem 84]. Hence, we can identify games with complete sets of strategies by this bijection.

Now, observe that the problem in interpreting the Sigma-type $\Sigma_{\mathsf{x}:\mathsf{N}}\mathsf{List}_{\mathsf{N}}(\mathsf{x})$ (resp. $\Sigma_{\mathsf{x}:\mathsf{N}}\mathsf{N}_{\mathsf{b}}(\mathsf{x})$) by games sketched at the beginning of §3 is due to the *consistency* (resp. *completeness*) of games. In this way, we have identified the fundamental limitations of games in interpreting Sigma-type: consistency and completeness.

3.2 Predicate games

Our idea is then to relax completeness and even consistency of games as follows:

Definition 3.3 (Predicate games) A ***predicate (p-) game*** is a nonempty set Γ of strategies that is directed complete with respect to the partial order $\leqslant$ defined by

[3]For the union $\bigcup \mathcal{S}$ to be a game, the weakening of the axiom E1 (Definition 2.6) and the embedding of labels into moves (Definition 2.1) are crucial.

$\sigma \leqslant \tau :\Leftrightarrow \sigma \asymp \tau \wedge \sigma \subseteq \tau$. A strategy γ is ***on*** Γ if $\gamma : \Gamma :\Leftrightarrow \gamma \in \Gamma$. Γ is ***well-founded (w.f.)*** (resp. ***well-opened (w.o.)***) if the game $P_\Gamma := \bigcup \Gamma$ is w.f. (resp. w.o.).

A ***position*** in Γ is a prefix of a sequence $q_\Gamma \gamma \boldsymbol{s}$ such that $\gamma \in \Gamma$ and $\boldsymbol{s} \in \gamma$, where q_Γ is any fixed element, $q_\Gamma \gamma$ is an ***initial protocol***, and $\boldsymbol{s}$ is an ***actual position***.

We relax completeness of games to directed completeness of p-games so that p-games can model Sigma-type (§4.5.2); recall, for instance, that the p-game $\Sigma(N, N_b)$ (resp. $\Sigma(N, \mathrm{List}_N)$) given at the beginning of §3 is incomplete (resp. inconsistent).

A play in a p-game Γ proceeds as follows. First, ***Judge (J)*** asks P a question q_Γ 'What is your strategy?,' and P answers it by some strategy $\gamma : \Gamma$. Then, a play between O and P follows, in which P must play by γ. I.e., after an initial protocol $q_\Gamma \gamma$ and given an even-length position $q_\Gamma \gamma \boldsymbol{s}$ in Γ, O may only make an O-move m such that $\boldsymbol{s}m \in \gamma$, and P must make the unique P-move n such that $\boldsymbol{s}mn \in \gamma$, if any, and none otherwise, and so on. Our naming of *predicate* games is motivated by this point: In p-games, P can specify strategies just like individuals in predicates.

Crucially, $\gamma : \Gamma$ may range over *strategies on different games* (Definition 2.7) as Γ may be inconsistent. Hence, P *selects* an underlying game γ when she answers J.

As already explained, initial protocols played in a p-game Γ are 'kept secret' from O, i.e., they are out of the O-view of any position in Γ, so that p-games are a natural generalisation of games. In fact, the moves occurring in the initial protocols are not counted as moves in the underlying game P_Γ since $M_{P_\Gamma} = M_{\bigcup \Gamma}$, and the concepts of pointers and views are applied only to *actual* positions in Γ since $\vdash_{P_\Gamma} = \vdash_{\bigcup \Gamma}$.

Example 3.4 The ***predicate (p-) game induced by a game*** G is the p-game $\mathscr{P}(G) := \{ \sigma \mid \sigma : G \}$. We abbreviate the p-games $\mathscr{P}(T)$, $\mathscr{P}(\mathbf{0})$, $\mathscr{P}(\mathbf{1})$ and $\mathscr{P}(N)$ as T, $\mathbf{0}$, $\mathbf{1}$ and N, and call them the ***terminal p-game***, the ***empty p-game***, the ***unit p-game*** and the ***natural number p-game***, respectively (cf. Example 2.8).

3.3 Cartesian closed categories of predicate games

Next, we lift constructions on games (§2.3) to p-games. For this task, we need the corresponding constructions on strategies since p-games are defined by strategies.

The cases of product & and tensor $\otimes$ are just straightforward:

Definition 3.5 (Product on predicate games) The ***product*** of p-games Γ and Δ is the p-game $\Gamma \mathbin{\&} \Delta := \{ \langle \gamma, \delta \rangle \mid \gamma : \Gamma, \delta : \Delta \}$.

Definition 3.6 (Tensor on predicate games) The ***tensor*** of p-games Γ and Δ is the p-game $\Gamma \otimes \Delta := \{ \gamma \otimes \delta \mid \gamma : \Gamma, \delta : \Delta \}$.

We proceed to generalise exponential ! on games to p-games, for which there is an obstacle: A strategy on the exponential $!G$ of a game G may not be obtained as the promotion of a single strategy on G. For instance, consider $\mathrm{Pref}(\{q0q1q2\dots\}) : !N$.

However, it is not a difficult problem; it suffices to introduce:

Notation Let Γ be a p-game. Given $\boldsymbol{s} \in P_{!\Gamma}$ and $i \in \mathbb{N}$, we write $\boldsymbol{s} \upharpoonright i$ for the j-subsequence of $\boldsymbol{s}$ that consists of moves hereditarily justified[4] by the $(i+1)$-st

[4] An occurrence n in a j-sequence $\boldsymbol{s}$ is *hereditarily justified* by another occurrence m in $\boldsymbol{s}$ if $\mathcal{J}_{\boldsymbol{s}}^i(n) = m$ for some $i \in \mathbb{N}^+$ [38, p. 22].

initial occurrence in $\boldsymbol{s}$. For instance, consider $\boldsymbol{s} := q2q1q0 \in P_{!N}$; then $\boldsymbol{s} \upharpoonright 0 = q2$, $\boldsymbol{s} \upharpoonright 1 = q1$ and $\boldsymbol{s} \upharpoonright 2 = q0$. We are interested in the case where Γ is w.o. since in this case $\boldsymbol{s} \upharpoonright i$ is the actual position in Γ that occurs in $\boldsymbol{s}$ for the $(i+1)$-st time.

Definition 3.7 (Countable tensor) The ***countable tensor*** of a family $(\gamma_i)_{i \in \mathbb{N}}$ of strategies γ_i on a p-game Γ is the strategy $\otimes_{i \in \mathbb{N}} \gamma_i := \{ \boldsymbol{s} \in P_{!\Gamma} \mid \forall j \in \mathbb{N}. \boldsymbol{s} \upharpoonright j \in \gamma_j \}$.

We then define exponential ! on p-games by:

Definition 3.8 (Exponential of predicate games) The ***exponential*** of a p-game Γ is the p-game $!\Gamma := \{ \otimes_{i \in \mathbb{N}} \gamma_i \mid \forall j \in \mathbb{N}. \gamma_j : \Gamma \}$.

We next define the linear implication $\Gamma \multimap \Delta$ between p-games Γ and Δ. Unlike other constructions, we cannot define it by $\Gamma \multimap \Delta := \{ \gamma \multimap \delta \mid \gamma : \Gamma, \delta : \Delta \}$ since P and O have to be *switched* in Γ. For instance, any strategy of the form $\underline{n} \multimap \underline{m}$ for any fixed $n, m \in \mathbb{N}$ does not compute the successor function. The point here is that a strategy on Γ is to be *chosen by O* and thus must *range over all strategies on* Γ, but on the contrary the formula $\Gamma \multimap \Delta := \{ \gamma \multimap \delta \mid \gamma : \Gamma, \delta : \Delta \}$ fixes one particular $\gamma : \Gamma$ for each strategy $\gamma \multimap \delta$. Hence, this formula does not work.

For this technical challenge, we introduce the following novel concept:

Definition 3.9 (FoPLIs) A ***family of pointwise linear implications (FoPLI)*** between p-games Γ and Δ is a family $\Phi = (\Phi_\gamma)_{\gamma:\Gamma}$ of strategies Φ_γ, called the ***pointwise linear implication (PLI)*** of Φ at γ, that satisfies

1 $\forall \gamma : \Gamma. \exists \delta : \Delta. \Phi_\gamma : \gamma \multimap \delta \wedge \delta = \Phi_\gamma \circ \gamma$;

2 $\forall \gamma, \gamma' : \Gamma, \boldsymbol{s}mn \in (\Phi_\gamma \cup \Phi_{\gamma'})^{\mathrm{Even}}. \boldsymbol{s}m \in \Phi_\gamma \cap \Phi_{\gamma'} \Rightarrow \boldsymbol{s}mn \in \Phi_\gamma \cap \Phi_{\gamma'}$.

We call the union $\bigcup \Phi := \bigcup_{\gamma:\Gamma} \Phi_\gamma$ a ***union of PLIs (UoPLI)*** between Γ and Δ.

We write $\mathscr{F}(\Gamma, \Delta)$ for the set of all FoPLIs between Γ and Δ, and $\boldsymbol{t} \upharpoonright \Gamma := \boldsymbol{t} \upharpoonright P_\Gamma$, where $\boldsymbol{t}$ ranges over j-sequences for which the operation $\boldsymbol{t} \upharpoonright P_\Gamma$ makes sense.

Example 3.10 Define an FoPLI Φ from the natural number p-game $N_{[0]}$ to itself $N_{[1]}$, where the subscripts $(\_)_{[i]}$ $(i = 0, 1)$ are to distinguish the two copies of N, by

$$\Phi_\sigma := \begin{cases} \mathrm{Pref}(\{q_{[1]}q_{[0]}n_{[0]}n+1_{[1]}\}) : \underline{n}_{[0]} \multimap \underline{n+1}_{[1]} & \text{if } \sigma = \underline{n}_{[0]}; \\ \mathrm{Pref}(\{q_{[1]}q_{[0]}\}) : \bot_{[0]} \multimap \bot_{[1]} & \text{otherwise (i.e., if } \sigma = \bot_{[0]}). \end{cases}$$

The UoPLI $\bigcup \Phi$ coincides with the standard strategy on the game $N \multimap N$ for the successor [26, §2.2]. Besides, the map $\sigma : N \mapsto (\bigcup \Phi) \circ \sigma = \Phi_\sigma \circ \sigma : N$ is continuous.

The UoPLI $\bigcup \Phi$ on an FoPLI Φ between p-games Γ and Δ is to serve as a strategy on the *linear implication* $\Gamma \multimap \Delta$ between these p-games; see Definition 3.13. The first axiom on FoPLIs specifies UoPLIs in the *pointwise* fashion. Further, the second axiom brings UoPLIs *determinacy* and their *computational nature*, e.g., their input-output behaviours induce *continuous* maps between DCPOs of Corollary 3.21.

If we dropped the second axiom and took, instead of UoPLIs, the *disjoint union* (to retain determinacy) on each FoPLI (without the second axiom), then its input-output behaviours may give rise to an ordinary (possibly non-continuous, even non-monotone) map, though it would be still a strategy, losing the computational nature. Conceptually, if P has such a disjoint union, then she could *foresee* O's computation on the domain *before playing on the domain* (since O would have to select a 'tag' for the disjoint union on the very first move), which is close to ordinary maps and unnatural from the computational view of game semantics. Hence, we impose the second axiom on FoPLIs and take their *unions*. Let us illustrate this point by:

Example 3.11 As a nonexample, define a family $\Psi = (\Psi_\sigma)_{\sigma : N}$ of strategies $\Psi_\sigma : N_{[0]} \multimap N_{[1]}$ from the natural number p-game N to itself by

$$\Psi_\sigma := \begin{cases} \underline{0}_{[1]} & \text{if } \sigma = \bot_{[0]}; \\ \underline{n}_{[1]} & \text{if } \sigma = \underline{n}_{[0]} \text{ for some } n \in \mathbb{N}^+; \\ \bot_{[1]} & \text{otherwise (i.e., if } \sigma = \underline{0}_{[0]}). \end{cases}$$

This family is *not* an FoPLI because it does not satisfy the second axiom. As a result, the union $\bigcup \Psi$ is not a strategy since it does not satisfy determinacy.

On the other hand, the disjoint union $\uplus\Psi := \bigcup_{\sigma:N} \{ \boldsymbol{s}_{[\sigma]} \mid \boldsymbol{s} \in \Psi_\sigma \}$ is a strategy, where $(\_)_{[\sigma]}$ is the 'tag' for the disjoint union. However, its computation on the codomain $N_{[1]}$ depends not on the computation on the domain $N_{[0]}$ but on the 'tag' $(\_)_{[\sigma]}$. Thus, it is not surprising that the map $\sigma : N \mapsto \Psi_\sigma \circ \sigma : N$ is not monotone.

Lemma 3.12 (Well-defined UoPLIs) *Let Γ and Δ be p-games.*

1. *Each UoPLI between Γ and Δ is a strategy;*
2. *Each UoPLI between Γ and Δ has exactly one FoPLI between Γ and Δ whose union coincides with the UoPLI.*

Proof Let us first verify the first clause. By the two axioms on FoPLIs, UoPLs are clearly strategies, where the embedding of labels into moves (Definition 2.1) and the weakening of the axiom E1 (Definition 2.2) are essential.

We next show the second clause. Let $\Phi, \Phi' \in \mathscr{F}(\Gamma, \Delta)$ and $\phi := \bigcup \Phi$, and assume $\bigcup \Phi' = \phi$; we have to show $\Phi = \Phi'$. By the first axiom in Definition 3.9 on Φ and Φ', $\Phi_\gamma \circ \gamma = \phi \circ \gamma = \Phi'_\gamma \circ \gamma$ for all $\gamma : \Gamma$. This equation together with the second axiom on Φ and Φ' implies $\Phi_\gamma = \phi_\gamma = \Phi'_\gamma$ for all $\gamma : \Gamma$, where $\phi_\gamma := \{ \boldsymbol{s} \in \phi \mid \boldsymbol{s} \upharpoonright \Gamma \in \gamma \}$. We have shown the required equation $\Phi = \Phi'$. □

Thanks to the second clause of Lemma 3.12, we can regard the *existence* of the unique FoPLI Φ between p-games Γ and Δ such that $\bigcup \Phi = \phi$ as the *axiom* for a given strategy ϕ to be a UoPLI from Γ to Δ (rather than Φ as a *structure* on ϕ).

Remark One might wonder why we do not define a UoPLI between p-games Γ and Δ to be a strategy ϕ such that $\forall \gamma : \Gamma. \exists \delta : \Delta. \{ \boldsymbol{s} \in \phi \mid \boldsymbol{s} \upharpoonright \Gamma \in \gamma \} : \gamma \multimap \delta \wedge \delta = \phi \circ \gamma$. However, the operation $\boldsymbol{s} \upharpoonright \Gamma$ does not make sense since a priori there is no known arena underlying ϕ, which is why we define UoPLIs via FoPLIs (Definition 3.9).

Note also that in general $\phi : P_\Gamma \multimap P_\Delta$ does not hold; e.g., see Example 4.6.

As we have already indicated, UoPLIs between p-games Γ and Δ are to serve as strategies on the *linear implication* $\Gamma \multimap \Delta$ between Γ and Δ:

Definition 3.13 (Linear implication on predicate games) The ***linear implication between p-games*** Γ and Δ is the p-game $\Gamma \multimap \Delta := \{ \bigcup \Phi \mid \Phi \in \mathscr{F}(\Gamma, \Delta) \}$. $\Gamma \Rightarrow \Delta := {!\Gamma} \multimap \Delta$ is the ***function space*** or ***implication*** from Γ to Δ.

At this point, recall that in predicate games P declares a strategy which is invisible to O, and then O has to play in a way that is compatible with that strategy. Then, since P and O are *switched* in the domain of each linear implication between games (Definition 2.12), one expects that in the domain Γ of each linear implication $\Gamma \multimap \Delta$ between p-games O declares a strategy which is invisible to P, and then P has to play in a way that is compatible with that strategy. In fact, this is achieved by the second axiom on FoPLIs: Each UoPLI $\phi : \Gamma \multimap \Delta$ must be compatible with *any* strategy $\gamma : \Gamma$. This point also plays a crucial role in the proof of Lemma 3.18.

Lemma 3.14 (Well-defined constructions on predicate games) *P-games (w.f. ones) are closed under* $\spadesuit \in \{\&, \otimes, !, \multimap\}$*, and w.o. ones under* $\clubsuit \in \{\&, \multimap, \Rightarrow\}$*. Further, the map* $\mathscr{P}$ *(Example 3.4) preserves* $\clubsuit$ *on games.*

Proof Assume that Γ and Δ are p-games, and G and H are games. It is easy to see that the product $\Gamma \,\&\, \Delta$, the tensor $\Gamma \otimes \Delta$ and the exponential $!\Gamma$ are p-games, and the equation $\mathscr{P}(G \,\&\, H) = \mathscr{P}(G) \,\&\, \mathscr{P}(H)$ holds; we leave the details to the reader.

Next, we consider linear implication $\multimap$. By Lemma 3.12, it suffices to show that the linear implication $\Gamma \multimap \Delta$ satisfies the two axioms of p-games. Then, the axioms on $\Gamma \multimap \Delta$ follow immediately from those on Δ and the definition of UoPLIs.

Also, well-foundedness is clearly preserved under $\spadesuit$, and well-openness under $\clubsuit$.

Let us next show $\mathscr{P}(G \multimap H) = \mathscr{P}(G) \multimap \mathscr{P}(H)$. Given $\phi : G \multimap H$, the family $(\phi_\sigma)_{\sigma : G}$ of strategies $\phi_\sigma := \{ \boldsymbol{s} \in \phi \mid \boldsymbol{s} \upharpoonright G \in \sigma \}$ satisfies the two axioms on FoPLIs, whence $\phi = \bigcup_{\sigma : G} \phi_\sigma : \mathscr{P}(G) \multimap \mathscr{P}(H)$. Conversely, any UoPLI $\varphi : \mathscr{P}(G) \multimap \mathscr{P}(H)$ is clearly a strategy on $G \multimap H$. We have shown $\mathscr{P}(G \multimap H) = \mathscr{P}(G) \multimap \mathscr{P}(H)$.

Finally, since each $\psi : G \Rightarrow H$ satisfies $\psi = \bigcup_{\otimes_{i \in \mathbb{N}} \sigma_i : !G} \psi_{\otimes_{i \in \mathbb{N}} \sigma_i}$, where $\psi_{\otimes_{i \in \mathbb{N}} \sigma_i} := \{ \boldsymbol{s} \in \psi \mid \boldsymbol{s} \upharpoonright !G \in \otimes_{i \in \mathbb{N}} \sigma_i \}$, we prove the equation $\mathscr{P}(G \Rightarrow H) = \mathscr{P}(G) \Rightarrow \mathscr{P}(H)$ just in the same way as the equation $\mathscr{P}(G \multimap H) = \mathscr{P}(G) \multimap \mathscr{P}(H)$. $\square$

Remark The map $\mathscr{P}$ does not preserve tensor $\otimes$ or exponential !. For instance, the p-game $\mathscr{P}(N)_{[0]} \otimes \mathscr{P}(N)_{[1]}$ does not have $\mathrm{Pref}(\{q_{[0]} 1_{[0]} q_{[1]} 0_{[1]}, q_{[1]} 1_{[1]} q_{[0]} 0_{[0]}\})$ as its strategy, but the p-game $\mathscr{P}(N_{[0]} \otimes N_{[1]})$ does, where we use the 'tags' $(\_)_{[i]}$ $(i = 0, 1)$ for clarity. However, *innocent* strategies on $\mathscr{P}(G \otimes H)$ and those on $\mathscr{P}(G) \otimes \mathscr{P}(H)$ coincide for any games G and H, and similarly for the case of exponential !.

In contrast, $\mathscr{P}$ preserves the other constructions $(T, \&, \multimap, \Rightarrow)$. It is notable that $\mathscr{P}$ preserves implication $\Rightarrow$ even though it does not preserve exponential !.

Let us next introduce constructions on UoPLIs:

Definition 3.15 (Constructions on UoPLIs) Given p-games Γ, Δ, Θ and Ξ, and UoPLIs $\phi : \Gamma \multimap \Delta$, $\psi : \Delta \multimap \Theta$, $\sigma : \Theta \multimap \Xi$, $\tau : \Gamma \multimap \Theta$ and $\theta : !\Gamma \multimap \Delta$, we define

- $\mathrm{cp}_\Gamma := \mathrm{cp}_{P_\Gamma}$, called the ***copy-cat*** on Γ;
- $\mathrm{der}_\Gamma := \mathrm{der}_{P_\Gamma}$, called the ***dereliction*** on Γ;
- $\phi;\psi := \{\, \boldsymbol{s} \upharpoonright \Gamma, \Theta \mid \boldsymbol{s} \in \phi \parallel \psi \,\}$ (cf. Definition 2.14 and Lemma 3.12), called the ***composition*** of ϕ and ψ, where $\phi;\psi$ is also written $\psi \circ \phi$;
- $\phi \otimes \sigma := \{\, \boldsymbol{s} \in \Gamma \otimes \Theta \multimap \Delta \otimes \Xi \mid \boldsymbol{s} \upharpoonright \Gamma, \Delta \in \phi, \boldsymbol{s} \upharpoonright \Theta, \Xi \in \sigma \,\}$, called the ***tensor*** of ϕ and σ;
- $\langle \phi, \tau \rangle := \{\, \boldsymbol{s} \in \Gamma \multimap \Delta \,\&\, \Theta \mid (\boldsymbol{s} \upharpoonright \Gamma, \Delta \in \phi \wedge \boldsymbol{s} \upharpoonright \Theta = \boldsymbol{\epsilon}) \vee (\boldsymbol{s} \upharpoonright \Gamma, \Theta \in \tau \wedge \boldsymbol{s} \upharpoonright \Delta = \boldsymbol{\epsilon}) \,\}$, called the ***pairing*** of ϕ and τ;
- $\theta^\dagger := \{\, \boldsymbol{s} \in {!\Gamma} \multimap {!\Delta} \mid \forall m \in M^{\mathrm{Init}}_{!\Delta}.\, \boldsymbol{s} \upharpoonright \{m\} \in \theta \,\}$, called the ***promotion*** of θ.

Lemma 3.16 (Well-defined copy-cats and derelictions on predicate games) *Suppose that Γ is a p-game, and Δ is a w.o. p-game.*

1. *The copy-cat cp_Γ is a w.b. strategy on $\Gamma \multimap \Gamma$, and winning if Γ is w.f.;*
2. *Copy-cats are the unit with respect to composition $\circ$ on strategies;*
3. *The dereliction der_Δ is a w.b. strategy on $\Delta \Rightarrow \Delta$, and winning if Γ is w.f.*

Proof For the first clause, define $\Phi = (\Phi_\gamma)_{\gamma:\Gamma} \in \mathscr{F}(\Gamma, \Gamma)$ by $\Phi_\gamma := \mathrm{cp}_\gamma$, where cp_γ is the copy-cat on the game γ (Definition 2.14). Because $\forall \boldsymbol{s} \in P_\Gamma.\, \exists \gamma : \Gamma.\, \boldsymbol{s} \in \gamma$, we have $\bigcup_{\gamma:\Gamma} \Phi_\gamma = \mathrm{cp}_\Gamma$, showing that cp_Γ is a strategy on $\Gamma \multimap \Gamma$.

Next, cp_Γ is clearly total, innocent and w.b., and noetherian if Γ is w.f., where the noetherianity follows from the well-foundedness of P_Γ as in the proof of Lemma 2.16.

The second clause clearly holds by the definitions of cp_Γ and the composition $\circ$ just like the case of copy-cats between games.

Finally, on the third clause, the case of der_Δ is essentially the same as that of cp_Γ except that Δ must be w.o. as in the case of derelictions between games (§2.3). □

Lemma 3.17 (Promotion lemma on UoPLIs) *Let $\phi : {!\Gamma} \multimap \Delta$, $\psi : {!\Delta} \multimap \Theta$ and $\varphi : {!\Theta} \multimap \Xi$ be strategies between p-games. Then, we have*

1. *$\mathrm{der}_\Gamma^\dagger = \mathrm{cp}_{!\Gamma} : {!\Gamma} \multimap {!\Gamma}$ and $\mathrm{der}_\Delta \circ \phi^\dagger = \phi$ if Γ and Δ are w.o.;*
2. *$\varphi \circ (\psi \circ \phi^\dagger)^\dagger = (\varphi \circ \psi^\dagger) \circ \phi^\dagger : {!\Gamma} \multimap \Xi$.*

Proof Essentially the same as the case of MC-strategies [38]. □

Lemma 3.18 (Well-defined constructions on UoPLIs) *Given UoPLIs $\phi : \Gamma \multimap \Delta$, $\psi : \Delta \multimap \Theta$, $\sigma : \Theta \multimap \Xi$, $\tau : \Gamma \multimap \Theta$ and $\theta : {!\Gamma} \multimap \Delta$, we have UoPLIs $\psi \circ \phi : \Gamma \multimap \Theta$ $\phi \otimes \sigma : \Gamma \otimes \Theta \multimap \Delta \otimes \Xi$, $\langle \phi, \tau \rangle : \Gamma \multimap \Delta \& \Theta$ and $\theta^\dagger : {!\Gamma} \multimap {!\Delta}$, and these constructions preserve winning and well-bracketing of UoPLIs.*

Proof First, observe that the constructions on UoPLIs are essentially the same as those on strategies (Definition 2.14), and therefore they preserve winning and well-bracketing of UoPLIs just as in the case of Lemma 2.16.

Let us next show $\psi \circ \phi : \Gamma \multimap \Theta$. Let $\Phi \in \mathscr{F}(\Gamma, \Delta)$ and $\Psi \in \mathscr{F}(\Delta, \Theta)$ be the unique FoPLIs that satisfy $\bigcup \Phi = \phi$ and $\bigcup \Psi = \psi$, respectively. We first need to show that the composition $\psi \circ \phi$ is well-defined. For this task, we have:

> (Claim) Given $\boldsymbol{s}mn \in \psi^{\mathrm{Even}}$, if $\boldsymbol{s}m \upharpoonright \Delta \in \Phi_\gamma \circ \gamma$ for some $\gamma : \Gamma$, then $\boldsymbol{s}mn \upharpoonright \Delta \in \Phi_\gamma \circ \gamma$ (n.b., the nontrivial case is when n is a move in Δ).

This claim follows from the second axiom on Ψ. By the claim, ψ plays on Δ *always in the scope of* ϕ *on* Δ: $\boldsymbol{smn} \in \psi^{\mathrm{Even}} \wedge \boldsymbol{sm} \upharpoonright \Delta \in \phi \upharpoonright \Delta = \{\, \boldsymbol{t} \upharpoonright \Delta \mid \boldsymbol{t} \in \phi \,\}$ implies $\boldsymbol{smn} \upharpoonright \Delta \in \phi \upharpoonright \Delta$. It then follows that the composition $\psi \circ \phi$ is well-defined.

Define a family $\Psi \circ \Phi := (\Psi_{\Phi_\gamma \circ \gamma} \circ \Phi_\gamma)_{\gamma : \Gamma}$ of strategies. It is then easy to verify $\Psi \circ \Phi \in \mathscr{F}(\Gamma, \Theta)$, which we leave to the reader. We have to show $\bigcup \Psi \circ \Phi = \psi \circ \phi$.

Then, the inclusion $\psi \circ \phi \subseteq \bigcup(\Psi \circ \Phi)$ holds since each element of $\psi \circ \phi$ is of the form $\boldsymbol{s} \upharpoonright \Gamma, \Theta$ with $\boldsymbol{s} \in \phi \parallel \psi$ (cf. Definition 2.14), so that $\boldsymbol{s} \in \Phi_\gamma \parallel \Psi_{\Phi_\gamma \upharpoonright \Delta}$ for some (not necessarily unique) $\gamma : \Gamma$, whence $\boldsymbol{s} \upharpoonright \Gamma, \Theta \in \Psi_{\Phi_\gamma \upharpoonright \Delta} \circ \Phi_\gamma$. Finally, the other inclusion $\bigcup(\Psi \circ \Phi) \subseteq \psi \circ \phi$ clearly holds, showing $\psi \circ \phi = \bigcup(\Psi \circ \Phi) : \Gamma \multimap \Theta$.

In the following, we focus on showing $\phi \otimes \sigma : \Gamma \otimes \Theta \multimap \Delta \otimes \Xi$ since the cases of the remaining constructions are similar. Let $\Sigma \in \mathscr{F}(\Theta, \Xi)$ be the unique UoPLI such that $\sigma = \bigcup \Sigma$. Define $\Phi \otimes \Sigma := (\Phi_\gamma \otimes \Sigma_\theta)_{\gamma \otimes \theta : \Gamma \otimes \Theta}$. It is then easy to verify $\Phi \otimes \Sigma \in \mathscr{F}(\Gamma \otimes \Theta, \Delta \otimes \Xi)$ and $\phi \otimes \sigma = \bigcup(\Phi \otimes \Sigma)$. Thus, $\phi \otimes \sigma : \Gamma \otimes \Theta \multimap \Delta \otimes \Xi$. $\square$

Definition 3.19 (Categories of predicate games) The category $\mathbb{PG}$ consists of

- W.o. p-games as objects;
- Strategies on the implication $\Gamma \Rightarrow \Delta$ as morphisms $\Gamma \to \Delta$;
- The composition $\psi \bullet \phi := \psi \circ \phi^\dagger : \Gamma \Rightarrow \Theta$ of strategies as the composition of morphisms $\phi : \Gamma \to \Delta$ and $\psi : \Delta \to \Theta$;
- The dereliction $\mathrm{der}_\Gamma : \Gamma \Rightarrow \Gamma$ as the identity id_Γ on each object Γ.

The subcategory $\mathbb{LPG}$ (resp. $\mathbb{WPG}$) of $\mathbb{PG}$ consists of w.f., w.o. p-games as objects, and winning (resp. winning, w.b.) strategies as morphisms.

Just like the categories of games (Definition 2.17), p-games in $\mathbb{PG}$ (resp. $\mathbb{LPG}$ and $\mathbb{WPG}$) are *w.o.* (resp. *w.f.* and *w.o.*) for identities to be well-defined.

As in the case of MC-games [38, Lemma 3.4.7], each morphism $\Gamma \to !\Delta$ in $\mathbb{LPG}$ or $\mathbb{WPG}$ is, by innocence, the promotion $\phi^\dagger$ of a unique morphism $\phi : \Gamma \to \Delta$. We often use this property, e.g., write $\phi^\dagger$ for an *arbitrary* morphism $\Gamma \to !\Delta$ in $\mathbb{WPG}$.

Theorem 3.20 (Well-defined CCCs of predicate games) *$\mathbb{PG}$, $\mathbb{LPG}$ and $\mathbb{WPG}$ are well-defined categories with finite products $(\mathbf{1}, \&)$ and exponential objects $\Rightarrow$.*

Proof By Lemma 3.18, it suffices to focus on $\mathbb{PG}$. First, composition is well-defined by Lemma 3.18, and so are identities by Lemma 3.16. Next, the associativity and the unit law follow from Lemmata 3.16–3.17. Finally, cartesian closure of $\mathbb{PG}$ is by Lemmata 3.14 and 3.18, where the required equations on strategies between p-games hold as in the case of the cartesian closed categories of games (§2.3). $\square$

Corollary 3.21 (DCPO-enrichment) *The CCCs $\mathbb{PG}$, $\mathbb{LPG}$ and $\mathbb{WPG}$ are DCPO-enriched with respect to the partial order $\leqslant$ on p-games, and they are algebraic.*

Proof Similar to the case of MC-games [38, §3.5.2] by the directed completeness of p-games, where directed joins are given by unions, except that some p-games are not pointed (e.g., the p-game $\Sigma(N, \mathrm{List}_N)$ is not pointed since it is inconsistent). $\square$

Corollary 3.22 (Game-semantic limits) *The lluf subcategory $\mathbb{PG}^\sharp \hookrightarrow \mathbb{PG}$ (resp. $\mathbb{LPG}^\sharp \hookrightarrow \mathbb{LPG}$, $\mathbb{WPG}^\sharp \hookrightarrow \mathbb{WPG}$), in which morphisms $\phi : \Gamma \to \Delta$ are all strict [49], i.e., $xy \in \phi$ implies $y \in P_\Gamma$, has all finite limits.*

Proof The argument below holds for all the three subcategories. By Theorem 3.20, it suffices to establish the equaliser of given morphisms $\phi_1, \phi_2 : \Gamma \rightrightarrows \Delta$, but it is the p-game $\Theta := \{ \gamma : \Gamma \mid \phi_1 \bullet \gamma = \phi_2 \bullet \gamma \}$ together with the dereliction $\mathrm{der}_\Theta : \Theta \hookrightarrow \Gamma$. □

We have to focus on *strict* strategies for Corollary 3.22 since otherwise the finite completeness does not hold; e.g., there is no equaliser of the non-strict strategies $\underline{0}, \underline{1} : N \rightrightarrows N$. This categorical structure is innovative as game semantics and quite useful. For instance, it enables us to internalise a certain notion of ∞-groupoids in the category $\mathbb{WPG}^\sharp$, which is a key step to extend the present work to HoTT [40].

The categories of games (Definition 2.17) do not have finite limits even if we focus on strict strategies. For example, we have no equaliser of the projections $\Omega \& \Omega \rightrightarrows \Omega$, where Ω is the flat game $\mathrm{flat}(\{0, 1\})$, since $\{ \langle \omega, \omega \rangle \mid \omega : \Omega \}$ is not complete (§3.1).

In summary, p-games are innovative as they provide novel game-semantic finite limits but also a reasonable generalisation of games that inherits the computational, intensional nature of games. We see further intensionality of p-games in §4.6–4.7.

Convention We write $\mathbb{PG}(\Gamma)$, $\mathbb{LPG}(\Gamma)$ and $\mathbb{WPG}(\Gamma)$ for the hom-DCPOs $\mathbb{PG}(T, \Gamma)$, $\mathbb{LPG}(T, \Gamma)$ and $\mathbb{WPG}(T, \Gamma)$, respectively, for each w.o., w.f. p-game Γ, and do not distinguish Γ and $\mathbb{PG}(\Gamma)$; e.g., $\gamma : \Gamma$ is winning and w.b. if and only if $\gamma \in \mathbb{WPG}(\Gamma)$.

4 Game semantics of Martin-Löf type theory

We are now ready to present our game semantics of MLTT. Concretely, we shall show that the CCC $\mathbb{WPG}$ forms abstract semantics of MLTT: a *category with families (CwF)* [50]. CwFs are much closer to the syntax than other abstract semantics, so that we can directly see their semantic counterparts of syntax. In fact, we even regard CwFs as another presentation of MLTT just as [51] does, and so we shall only show that $\mathbb{WPG}$ forms a CwF, leaving how a CwF interprets MLTT to [52].

Specifically, we shall show that $\mathbb{WPG}$ gives rise to a CwF equipped with *semantic type formers* [52] for One-, Zero-, N-, Pi-, Sigma- and Id-types as well as universes, therefore achieving game semantics of MLTT equipped with these types.

The rest of this section proceeds as follows. We model (dependent) types in §4.1, Pi-type in §4.2, and Sigma-type in §4.3. We then show that $\mathbb{WPG}$ gives rise to a CwF in §4.4 and equip it with all the semantic type formers in §4.5. Finally, we analyse the intensionality of our model of MLTT in §4.6 and prove the independence of Markov's principle from MLTT in §4.7 (for which employ $\mathbb{WPG}$, not $\mathbb{PG}$ or $\mathbb{LPG}$).

4.1 Dependent predicate games

We shall interpret *dependent types* by *w.o., w.f. dependent p-games*:

Definition 4.1 (Dependent predicate games) A ***linearly dependent predicate (p-) game*** over a p-game Γ is a pair $L = (|L|, \|L\|)$ of a p-game $|L|$ and a family $\|L\| = (L(\gamma_0))_{\gamma_0 \in \mathbb{WPG}(\Gamma)}$ of p-games $L(\gamma_0)$ with $P_{L(\gamma_0)} \subseteq P_{|L|}$. It is ***well-opened (w.o.)*** (resp. ***well-founded (w.f.)***) if $|L|$ is w.o. (resp. w.f.). A ***dependent predicate (p-) game*** over a p-game Γ is a linearly dependent p-game over $!\Gamma$.

We write $\mathscr{D}_\ell(\Gamma)$ (resp. $\mathscr{D}_\ell^{\mathrm{w}}(\Gamma)$) for the set of all linearly dependent p-games (resp. w.o., w.f. ones) over Γ, and $\{\Gamma'\}_\Gamma$ for the *constant* one valued at Γ', i.e., $\{\Gamma'\}_\Gamma := (\Gamma', (\Gamma')_{\gamma_0 \in \mathbb{WPG}(\Gamma)})$. We define $\mathscr{D}(\Gamma) := \mathscr{D}_\ell(!\Gamma)$ and $\mathscr{D}^{\mathrm{w}}(\Gamma) := \mathscr{D}_\ell^{\mathrm{w}}(!\Gamma)$.

One might think of another definition of dependent p-games. Hence, we explain the motivations for Definition 4.1 as follows. First, we shall define p-games $\mathcal{U}$ that model universes and encode dependent p-games A over a p-game Γ by morphisms $\phi_A : \Gamma \to \mathcal{U}$ in $\mathbb{WPG}$ for the introduction rule of universes in §4.5.6. Then, a strategy $\gamma : !\Gamma$ must be winning and w.b. to ensure $\phi_A \circ \gamma \in \mathbb{WPG}(\mathcal{U})$, i.e., $\phi_A \circ \gamma$ is a *valid* encoding of a p-game. For instance, if γ is *partial*, then $\phi_A \circ \gamma$ can be also partial or *incomplete* as an encoding. Note that the map $\gamma_0 \in \mathbb{WPG}(!\Gamma) \mapsto \mathrm{El}(\phi_A \circ \gamma_0) \in \mathbb{WPG}$ is A, where $\mathrm{El}(\mu)$ is the p-game encoded by each $\mu \in \mathbb{WPG}(\mathcal{U})$. This is why we index the second component $\|A\|$ by *winning, w.b.* strategies, not mere strategies, on $!\Gamma$. See the remark right after Example 4.2 on more technical reasons for this point.

Next, the second component $\|A\|$ is *not continuous or even monotone*, in contrast with the interpretation of dependent types by Blot and Laird [28], since the partial order $\leqslant$ becomes equality $=$ on $\mathscr{D}^{(\mathrm{w})}(\Gamma)$ (n.b., elements in $\mathscr{D}^{(\mathrm{w})}(\Gamma)$ are *total*). This point matches the remark by Abramsky et al. [32, Footnote 5] that the continuity of their interpretation of dependent types does not play any roles for their results. From another angle: If $\gamma_0 \in \mathbb{WPG}(!\Gamma)$, $\gamma_0 \geqslant \gamma$ and $\gamma_0 \neq \gamma$, then $\phi_A \circ \gamma$ may be *partial*, i.e., not a completed encoding of a p-game, and $A(\gamma)$ may not be well-defined; thus, it does not make much sense to consider a partial order between $A(\gamma_0)$ and $A(\gamma)$.

Besides, we need the first component $|A|$ too as the second component $\|A\|$ can be *empty*. To explain this point, note that without $|A|$ we would have to model a term $\mathsf{\Gamma \vdash a : A}$ by a morphism $\alpha : \Gamma \to \int A$, where $\int A$ is the p-game obtained from the union $\bigcup \|A\|$ by taking the closure of directed joins (so that it is directed complete), such that α satisfies the expected type dependency (§4.2). However, the constant one $\{\Gamma\}_{\mathbf{0}}$ has $\|\{\Gamma\}_{\mathbf{0}}\| = \emptyset$ for any p-game Γ; thus, e.g., $\int \{N\}_{\mathbf{0}}$ is not a p-game, and so we cannot model terms $\mathsf{x : 0 \vdash \underline{n} : N}$ by morphisms $\Gamma \to \int \{N\}_{\mathbf{0}}$. Hence, in place of $\int A$ we introduce the ambient p-game $|A|$ similarly to Abramsky et al. [27, 32].

Finally, one might consider $A(\gamma_0) \subseteq |A|$ in place of $P_{A(\gamma_0)} \subseteq P_{|A|}$ ($\gamma_0 \in \mathbb{WPG}(\Gamma)$). However, it is *too strong* in the sense that it is not preserved under our interpretation of Pi-type (§4.5.1), and Lemma 4.15 would fail if we adopted it; see Example 4.16.

Example 4.2 Define $\mathrm{List}_N \in \mathscr{D}^{\mathrm{w}}(N)$ by $\mathrm{List}_N(\underline{0}^\dagger) := T$ and $\mathrm{List}_N(\underline{n+1}^\dagger) := \mathrm{List}_N(\underline{n}^\dagger) \otimes N$ for each $n \in \mathbb{N}$, and $|\mathrm{List}_N| := \bigcup_{n \in \mathbb{N}} \mathrm{List}_N(\underline{n}^\dagger)$. Similarly, define $N_b \in \mathscr{D}^{\mathrm{w}}(N)$ by $N_b(\underline{n}^\dagger) := \{\bot\} \cup \{\, \underline{k} \mid k \leqslant n \,\}$ for each $n \in \mathbb{N}$, and $|N_b| := N$.

Remark The refinement of games into p-games provides us with another reason why we should not define dependent p-games over a p-game Γ as continuous maps from $!\Gamma$ to the class of all p-games: There is no appropriate partial order between p-games. In fact, the partial order $\trianglelefteq$ between p-games defined as in the case of games by $\Gamma \trianglelefteq \Delta :\Leftrightarrow P_\Gamma \subseteq P_\Delta$ has $N \mathbin{\&} N \trianglelefteq \Sigma(N, N_b)$; i.e., $\trianglelefteq$ is *too coarse* to capture the refinement of games into p-games. On the other hand, the subset relation $\subseteq$ is finer than $\trianglelefteq$, but not preserved under our interpretation of Pi-type; see Example 4.16.

Nevertheless, we shall regard dependent p-games A as a certain class of continuous maps $\overline{A}$ without defining a partial order between p-games (Definition 4.3). This view on dependent p-games will turn out to be accurate for various technical reasons.

4.2 Pi between predicate games

Let us next interpret *Pi-type*. Our idea is best explained by the set-theoretic analogy as follows. Given a dependent type $\mathsf{x} : \mathsf{C} \vdash \mathsf{D}(\mathsf{x})$ type, the Pi-type $\vdash \Pi_{\mathsf{x}:\mathsf{C}}\mathsf{D}(\mathsf{x})$ type is something like the set of all functions f from C to $\bigcup_{x:C} D(x)$ such that $f(x) \in D(x)$ for all $x \in C$, called *dependent maps* from C to D, where recall that the set-theoretic semantics interprets simple types C and terms $\mathsf{x} : \mathsf{C}$ as sets C and elements $x \in C$, respectively, and dependent types D over C as families $D = (D(x))_{x \in C}$ of sets $D(x)$.

Based on this idea, we interpret Pi-type by the following *pi* Π (Definition 4.5):

Definition 4.3 (Closure) The ***closure*** of a linearly dependent p-game L over a p-game Γ is the map $\overline{L}$ from Γ to the power set $\mathcal{P}(P_{|L|})$ of $P_{|L|}$ defined inductively by $\overline{L}(\gamma) := \{\boldsymbol{\epsilon}\} \cup \{\, \boldsymbol{s}m \mid \boldsymbol{s} \in \overline{L}(\gamma)^{\mathrm{Even}}, \exists \gamma_0 \in \mathbb{WPG}(\Gamma).\, \gamma \leqslant \gamma_0 \wedge \boldsymbol{s}m \in P_{L(\gamma_0)} \,\} \cup \{\, \boldsymbol{t}l r \mid \boldsymbol{t}l \in \overline{L}(\gamma)^{\mathrm{Odd}}, \forall \gamma_0' \in \mathbb{WPG}(\Gamma).\, \gamma \leqslant \gamma_0' \wedge \boldsymbol{t}l \in P_{L(\gamma_0')} \Rightarrow \boldsymbol{t}lr \in P_{L(\gamma_0')} \,\}$ for all $\gamma : \Gamma$.

Definition 4.4 (FoDPLIs) A ***family of dependently pointwise linear implications (FoDPLI)*** from a p-game Γ to a linearly dependent p-game L over Γ is an FoPLI Φ from Γ to $|L|$, where PLIs of Φ are called ***dependent***, that satisfies

$$\forall \gamma : \Gamma.\, \Phi_\gamma \circ \gamma \subseteq \overline{L}(\gamma). \tag{1}$$

We call the UoPLI $\bigcup \Phi$ a ***union of DPLIs (UoDPLI)*** from Γ to L, and write $\mathscr{F}(\Gamma, L)$ for the set of all FoDPLIs from Γ to L.

Definition 4.5 (Linear-pi and pi) The ***linear-pi*** from a p-game Γ to a linearly dependent p-game L over Γ is the p-game $\Pi_\ell(\Gamma, L) := \{\, \bigcup \Phi \mid \Phi \in \mathscr{F}(\Gamma, L) \,\}$, and the ***pi*** from Γ to a dependent p-game A over Γ is the linear-pi $\Pi(\Gamma, A) := \Pi_\ell(!\Gamma, A)$.

Remark One may wonder if another, more naive axiom $\forall \gamma_0 \in \mathbb{WPG}(\Gamma).\, \Phi_{\gamma_0} \circ \gamma_0 : L(\gamma_0)$ works in place of (1). The answer is, however, 'no' since Lemma 4.15 does not hold for it. The point is that (1) permits $\gamma_0 \in \mathbb{WPG}(\Gamma)$ and $\boldsymbol{s}m \in (\Phi_{\gamma_0} \circ \gamma_0)^{\mathrm{Odd}}$ with $\boldsymbol{s} \in P_{L(\gamma_0)}$ and $\boldsymbol{s}m \notin P_{L(\gamma_0)}$, if any, but the other one may not; see Example 4.16.

We leave it to the reader to show that the axiom (1) is equivalent to

$$\forall \gamma : \Gamma.\, \Phi_\gamma \subseteq \gamma \multimap \overline{L}(\gamma), \tag{2}$$

and further (1) particularly implies $\forall \gamma_0 \in \mathbb{WPG}(\Gamma).\, \Phi_{\gamma_0} \circ \gamma_0 \subseteq P_{L(\gamma_0)}$. We also leave it to the reader to verify that the slightly stronger axiom

$$\forall \gamma : \Gamma.\, \Phi_\gamma : \gamma \multimap \overline{L}(\gamma), \tag{3}$$

is equivalent to the axiom on *pi-games* given by Abramsky et al. [32, Definition 4.4]:

1. If $\boldsymbol{s} \in \bigcup \Phi^{\mathrm{Even}}$ and $\boldsymbol{s}m \in P^{\gamma_0}_{L(\gamma_0)}$ for some $\gamma_0 \in \mathbb{WPG}(\Gamma)$, then $\boldsymbol{s}m \in \bigcup \Phi$;
2. If $\boldsymbol{t}lr \in \bigcup \Phi^{\mathrm{Even}}$ and $\boldsymbol{t}l \in P^{\gamma_0}_{L(\gamma_0')}$ for some $\gamma_0' \in \mathbb{WPG}(\Gamma)$, then $\boldsymbol{t}lr \in P^{\gamma_0}_{L(\gamma_0')}$,

where $P^{\gamma_0}_{L(\gamma_0)} := \gamma_0 \multimap P_{L(\gamma_0)}$. The first condition states that, at a given even-length position $\boldsymbol{s} \in \bigcup \Phi$, O *can* play as in the game $P^{\gamma_0}_{L(\gamma_0)}$ for *any* $\gamma_0 \in \mathbb{WPG}(\Gamma)$ that is not yet excluded (i.e., $\boldsymbol{s} \in P^{\gamma_0}_{L(\gamma_0)}$), and the second condition stipulates that, at a given odd-length position $\boldsymbol{tl} \in \bigcup \Phi$, the next P's move by $\bigcup \Phi$ *must* be compatible with the game $P^{\gamma_0'}_{L(\gamma_0')}$ for *any* $\gamma_0' \in \mathbb{WPG}(\Gamma)$ that is not yet excluded (i.e., $\boldsymbol{tl} \in P^{\gamma_0'}_{L(\gamma_0')}$).

We have to slightly weaken the axiom (3) into the one (2) for our generalisation of strategies on linear implication between games to UoPLIs (n.b., as we shall see in §4.3, this generalisation plays a crucial role for our improvement of the semantics of Pi- and Sigma-types by Abramsky et al. [27, 32]). Specifically, this modification is necessary because UoPLIs may *control possible plays by O on the codomain*; e.g., the UoPLI $\bigcup_{\sigma : !N} \top_\sigma = \top : \Pi(N, \mathrm{List}_N)$, where $\top_\sigma := \top$, does not satisfy $\top_{\underline{1}^\dagger} : \underline{1}^\dagger \multimap N$, where $\overline{\mathrm{List}_N}(\underline{1}^\dagger) = N$, i.e., the axiom (3) fails for the UoPLI $\top$.

In summary, pi Π is based on the same idea as that of pi-games given by Abramsky et al., but we adjust their axiom along the generalisation of games to p-games.

The closure $\overline{L}$ is a *continuous* map $(\Gamma, \leqslant) \to (\mathcal{P}(P_{|L|}), \preccurlyeq)$, where $\preccurlyeq$ is the *liveness ordering* between sets of positions introduced by Chroboczek [53]. Roughly, $S \preccurlyeq S'$ if P (resp. O) is more (resp. less) restricted in S than in S'; see [53, Definition 8] for the precise definition.[5] The axiom (1) means that our interpretation of the *type dependency* on Pi-type is with respect to these *induced* continuous maps $\overline{L}$.[6]

Linear-pi Π_ℓ generalises linear implication $\multimap$: Given p-games Γ and Γ', $\Pi_\ell(\Gamma, \{\Gamma'\}_\Gamma) = \Gamma \multimap \Gamma'$. Similarly, pi generalises implication $\Rightarrow$: $\Pi(\Gamma, \{\Gamma'\}_{!\Gamma}) = \Gamma \Rightarrow \Gamma'$.

Example 4.6 $\zeta : \Pi(N, \mathrm{List}_N)$ plays as the dependent map $n \in \mathbb{N} \mapsto (0, 0, \dots, 0) \in \mathbb{N}^n$ as follows. If O makes the first move $q_{[k]}$ ($k \in \mathbb{N}^+$) on the codomain $|\mathrm{List}_N|$, where $(\_)_{[k]}$ is the 'tag' for the iterated tensor $\otimes$ on the codomain, then ζ asks a question q on the domain $!N$; then, if O plays on the domain $!N$ by $q \mapsto n$ ($n \in \mathbb{N}^+$ and $k \leqslant n$), then ζ makes the move $0_{[k]}$ on the codomain $\mathrm{List}_N(\underline{n}^\dagger)$, and so on.

The family $Z = (Z_\sigma)_{\sigma : !N}$ of strategies $Z_\sigma := \{\, \boldsymbol{s} \in \zeta \mid \boldsymbol{s} \upharpoonright !N \in \sigma \,\}$ satisfies $Z \in \mathscr{F}(N, \mathrm{List}_N)$ and $\bigcup Z = \zeta$. Hence, we indeed have $\zeta : \Pi(N, \mathrm{List}_N)$.

In a pi $\Pi(\Gamma, A)$, O specifies his strategy on the domain $!\Gamma$ *explicitly* by his play on the domain $!\Gamma$, and *implicitly* by his play on the codomain $|A|$. On the strategy $\zeta : \Pi(N, \mathrm{List}_N)$ in Example 4.6, for instance, after O makes any first move on the codomain $|\mathrm{List}_N|$, he cannot play by $\underline{0}^\dagger$ on the domain $!N$ since $\mathrm{List}_N(\underline{0}^\dagger) = T$; i.e., *implicit*. In contrast, after O plays by $q \mapsto n + 1$ ($n \in \mathbb{N}$) on the domain $!N$, his strategy there cannot be, e.g., $\underline{n+2}^\dagger$; i.e., *explicit*. This *gradual* specification of O's strategy on the domain *never completes* in general since positions are *finite*; e.g., any position in ζ cannot completely specify O's strategy on the domain $!N$, where note that O may play on the domain $!N$ by a strategy that is not a promotion.

Theorem 4.7 (Well-defined linear-pi) *Given a (w.o., w.f.) linearly dependent p-game L over a (w.o., w.f.) p-game Γ, the linear-pi $\Pi_\ell(\Gamma, L)$ is a (w.o., w.f.) p-game.*

[5] The directed join with respect to $\preccurlyeq$ is given by taking the intersection (resp. union) on odd-length (resp. even-length) positions.

[6] However, we cannot directly interpret dependent types by continuous functions $(\Gamma, \leqslant) \to (\mathcal{P}(P_{|L|}), \preccurlyeq)$ since the domain of our interpretation of Pi-type (Theorem 4.17) is *contravariant*, and hence the monotonicity is not preserved under the interpretation.

Proof Immediate from Lemma 3.14 since (1) is preserved under directed joins. □

This theorem also implies that, given a (w.o., w.f.) dependent p-game A over a (w.o., w.f.) p-game Γ, the pi $\Pi(\Gamma, A)$ is a (w.o., w.f.) p-game. However, we have to handle the case where Γ is a dependent p-game. We address this problem in §4.5.1.

4.3 Sigma on predicate games

We next interpret *Sigma-type*. Recall that by the set-theoretic analogy the Sigma-type $\vdash \Sigma_{\mathsf{x:C}}\mathsf{D(x)}\ \mathsf{type}$ represents the set of all pairs $\langle x, y\rangle$ such that $x \in C$ and $y \in D(x)$, called *dependent pairs* on C and D. Hence, we model Sigma-type by:

Definition 4.8 (Sigma) The ***sigma*** of a p-game Γ and a dependent p-game A over Γ is the p-game $\Sigma(\Gamma, A) := \{\, \langle \gamma, \alpha\rangle : \Gamma \,\&\, |A| \mid \alpha \subseteq \overline{L}(\gamma)\,\}$.

When A is a constant dependent p-game $\{\Gamma'\}_\Gamma$, the sigma $\Sigma(\Gamma, \{\Gamma'\}_{!\Gamma})$ coincides with the product $\Gamma \,\&\, \Gamma'$. Hence, sigma Σ generalises product & on p-games.

The concise axiom $\alpha \subseteq \overline{L}(\gamma)$ on the sigma $\Sigma(\Gamma, A)$ achieves the following highly nontrivial point. First, the categorical view on semantics of MLTT (Definition 4.11) tells us that there is a bijection between strategies $\psi \in \mathbb{WPG}(\Delta, \Sigma(\Gamma, A))$ and pairs (ϕ, α) of strategies $\phi \in \mathbb{WPG}(\Delta, \Gamma)$ and $\alpha \in \mathbb{WPG}(\Delta, A\{\phi\})$, where $A\{\phi\} \in \mathscr{D}(\Delta)$ is defined by $A\{\phi\}(\delta) := A(\phi^\dagger \circ \delta)$ for all $\delta \in \mathbb{WPG}(!\Delta)$. Next, recall that strategies on the pi $\Pi(\Delta, A\{\phi\})$ satisfy the axiom (1), which *gradually* specify the constraint on the codomain $A\{\phi\}$ along the gradual specification of a strategy on the domain $!\Delta$ by O. However, since the sigma $\Sigma(\Gamma, A)$ has nothing to do with Δ, a main challenge is to define $\Sigma(\Gamma, A)$ in such a way that strategies $\psi : \Delta \Rightarrow \Sigma(\Gamma, A)$ accomplish the bijection. The upshot is that the axiom $\alpha \subseteq \overline{L}(\gamma)$, combined with the first axiom on FoPLIs, meets this requirement; see the remark right after Theorem 4.13.

Example 4.9 The sigma $\Sigma(N, \mathrm{List}_N)$ is a game-semantic counterpart of the set of dependent pairs $\{\, (k, (n_1, n_2, \dots, n_k)) \mid k, n_1, n_2, \dots, n_k \in \mathbb{N}^* \,\}$, and the sigma $\Sigma(N, N_b)$ is that of the set of dependent pairs $\{\, (k, n) \mid k \leqslant n \,\}$.

Note in particular that the pairings $\langle \underline{k}, \underline{n_1} \otimes \underline{n_2} \otimes \cdots \otimes \underline{n_k}\rangle$ $(k, n_1, n_2, \dots, n_k \in \mathbb{N}^*)$ are *total* on the sigma $\Sigma(N, \mathrm{List}_N)$ since strategies on p-games can control possible plays by O. In contrast, even by their use of *lists* of games and strategies, the list $(\underline{k}, \underline{n_1} \otimes \underline{n_2} \otimes \cdots \otimes \underline{n_k})$ of strategies is *partial* on the list (N, List_N) of (dependent) games in the framework of Abramsky et al. [27, 32] since unlike the case of pi-games there is no way to prevent O from playing *arbitrarily* on the second component $|\mathrm{List}_N|$. For instance, the list $(\underline{0}, \top)$ (i.e., $k = 0$) is partial on the list (N, List_N) since O can play as in N on the right-hand side List_N of the list. To overcome this problem, Abramsky et al. define the lists $(\underline{k}, \tau)$ of strategies $\underline{k} : N$ and $\tau : |\mathrm{List}_N|$ such that $\tau : \mathrm{List}_N(\underline{k}^\dagger)$ to be *winning* on (N, List_N) if they are winning on N and $|\mathrm{List}_N|$, respectively. Note, however, that τ contains an infinite amount of *redundant* information since it is total on $|\mathrm{List}_N|$; i.e., τ is an infinite iteration of tensor $\otimes$ of strategies on N. Even worse, for making their interpretation of Pi-type compatible with that of Sigma-type (e.g., the aforementioned bijection holds), they apply the operation O-sat to pi-games for their interpretation of Pi-type [32, Remark 4.5 and Theorem 5.6], which allows O to play *arbitrarily* on the codomain of pi-games. In this way, the elegant interpretation of Pi-type as pi-games, which comes from the game semantics of universal quantification [34], is lost in Abramsky et al. [27, 32].

Remark The aforementioned redundancy of winning lists on the interpretation of Sigma-type by Abramsky et al. [27, 32] appears opposing to full completeness since terms $\langle \mathsf{a}, \mathsf{b} \rangle : \Sigma(\mathsf{A}, \mathsf{B})$ satisfy $\mathsf{a} : \mathsf{A}$ and $\mathsf{b} : \mathsf{B}(\mathsf{a})$, not $\mathsf{x} : \mathsf{A} \vdash \mathsf{b} : \mathsf{B}(\mathsf{x})$ [52]. Then, how does their full completeness holds? The answer is that they focus on a nonstandard class of finite inductive types $\mathsf{x} : \mathsf{A} \vdash \mathsf{B}(\mathsf{x})\ \mathsf{type}$, which are freely generated only by *closed* terms, and there is no type conversion on $\mathsf{B}(\mathsf{a})$ $(\mathsf{a} : \mathsf{A})$ [32, Figure 7]. That is, they focus on dependent types for which the redundancy does not matter. However, the full completeness clearly fails as soon as they include standard dependent types such as those given by the elimination rule of N-type with respect to a universe.

In summary, the novel mathematical structure of p-games enables us to not only dispense with the list construction but also extend the elegant game semantics of universal quantification [34] to Pi- and Sigma-types (without the operation O-sat). In addition, our full completeness result (§**??**) is with respect to a standard class of dependent types, which indicates some *canonicity* of our game semantics of MLTT.

Theorem 4.10 (Well-defined sigma) *Given a (w.o., w.f.) dependent p-game A over a (w.o., w.f.) p-game Γ, the sigma $\Sigma(\Gamma, A)$ is a (w.o., w.f.) p-game.*

Proof We just show that $\Sigma(\Gamma, A)$ is directed complete since it is the only nontrivial point. Let $\mathcal{D} \subseteq \Sigma(\Gamma, A)$ be directed; we have to show $\bigcup \mathcal{D} : \Sigma(\Gamma, A)$. Let $\pi_1 \mathcal{D} := \{ \gamma : \Gamma \mid \exists \alpha : |A|. \langle \gamma, \alpha \rangle \in \mathcal{D} \}$ and $\pi_2 \mathcal{D} := \{ \alpha : |A| \mid \exists \gamma : \Gamma. \langle \gamma, \alpha \rangle \in \mathcal{D} \}$. Note that $\pi_1 \mathcal{D}$ and $\pi_2 \mathcal{D}$ are both directed since so is $\mathcal{D}$, and $\bigcup \mathcal{D} = \langle \gamma_\star, \alpha_\star \rangle$, where $\gamma_\star := \bigcup \pi_1 \mathcal{D}$ and $\alpha_\star := \bigcup \pi_2 \mathcal{D}$. It then suffices to show $\alpha_\star \subseteq \overline{A}(\gamma_\star)$ by induction on the lengths of the elements of $\alpha_\star$. The base case is trivial: $\top \in \alpha_\star$ and $\top \in \overline{A}(\gamma_\star)$.

For one of the induction steps, assume $\boldsymbol{sm} \in \alpha_\star^{\mathrm{Odd}}$; we have to show $\boldsymbol{sm} \in \overline{A}(\gamma_\star)$. By $\boldsymbol{sm} \in \alpha_\star$, the set $\{ \gamma \mid \langle \gamma, \alpha \rangle \in \mathcal{D}, \boldsymbol{sm} \in \alpha \}$ is nonempty. Further, any $\langle \gamma, \alpha \rangle \in \mathcal{D}$ with $\boldsymbol{sm} \in \alpha$ has $\boldsymbol{sm} \in \alpha \subseteq \overline{A}(\gamma)$. Since $\mathcal{D}$ is directed, it follows from these two points that there is some $\gamma_0 \in \mathbb{WPG}(\Gamma)$ with $\boldsymbol{sm} \in P_{A(\gamma_0)}$ and $\gamma \leqslant \gamma_0$ for all $\gamma \in \pi_1 \mathcal{D}$, thus $\gamma_\star \leqslant \gamma_0$. Then, $\boldsymbol{sm} \in \overline{A}(\gamma_\star)$ follows from $\boldsymbol{sm} \in P_{A(\gamma_0)}$ and $\gamma_\star \leqslant \gamma_0$.

For the other induction step, let $\boldsymbol{tlr} \in \alpha_\star^{\mathrm{Even}}$; we have to show $\boldsymbol{tlr} \in \overline{A}(\gamma_\star)$. There is some $\langle \gamma', \alpha' \rangle \in \mathcal{D}$ with $\boldsymbol{tlr} \in \alpha' \subseteq \overline{A}(\gamma')$. Besides, $\boldsymbol{tl} \in \overline{A}(\gamma_\star)$ by the induction hypothesis. Then, $\boldsymbol{tlr} \in \overline{A}(\gamma_\star)$ follows from $\gamma' \leqslant \gamma_\star$, $\boldsymbol{tlr} \in \overline{A}(\gamma')$ and $\boldsymbol{tl} \in \overline{A}(\gamma_\star)$. $\square$

Theorem 4.10 is another technical highlight. Unlike Abramsky et al. [27, 32], our game semantics Σ of Sigma-type does not rely on the list construction or the O-sat construction. Also, recall that the model of Sigma-type by Blot and Laird [28] does not preserve the linearity of product & on games (§1.4). In contrast, sigma Σ preserves the linearity of product &. The challenge we have overcome is that the linearity requires that a play in $\Sigma(\Gamma, A)$ is *either* a play in Γ or A, but in turn we have to specify the constraint on A in $\Sigma(\Gamma, A)$ *gradually yet without playing on* Γ.

Again, however, this construction of sigma Σ is not general enough for the same reason as the case of pi Π (§4.2); we generalise it in §4.5.2.

4.4 Game-semantic category with families

We are now ready to give our game-semantic CwF. Let us first recall the general definition of CwFs introduced by Dybjer [50]:

Definition 4.11 (CwFs [50, 52]) A ***category with families (CwF)*** is a tuple $\mathcal{C} = (\mathcal{C}, \mathrm{Ty}, \mathrm{Tm}, \_\{\_\}, T, \_.\_, \mathrm{p}, \mathrm{v}, \langle \_, \_ \rangle_\_)$, where

- $\mathcal{C}$ is a category with a terminal object $T \in \mathcal{C}$;
- Ty assigns, to each object $\Gamma \in \mathcal{C}$, a set $\mathrm{Ty}(\Gamma)$ of ***types*** in the ***context*** Γ;
- Tm assigns, to each pair (Γ, A) of an object $\Gamma \in \mathcal{C}$ and a type $A \in \mathrm{Ty}(\Gamma)$, a set $\mathrm{Tm}(\Gamma, A)$ of ***terms*** of type A in the context Γ;
- To each $f : \Delta \to \Gamma$ in $\mathcal{C}$, $\_\{\_\}$ assigns a map $\_\{f\} : \mathrm{Ty}(\Gamma) \to \mathrm{Ty}(\Delta)$, called the ***substitution on types***, and a family $(\_\{f\}_A)_{A \in \mathrm{Ty}(\Gamma)}$ of maps $\_\{f\}_A : \mathrm{Tm}(\Gamma, A) \to \mathrm{Tm}(\Delta, A\{f\})$, called the ***substitutions on terms***;
- $\_.\_$ assigns, to each pair (Γ, A) of a context $\Gamma \in \mathcal{C}$ and a type $A \in \mathrm{Ty}(\Gamma)$, a context $\Gamma.A \in \mathcal{C}$, called the ***comprehension*** of A;
- p (resp. v) associates each pair (Γ, A) of a context $\Gamma \in \mathcal{C}$ and a type $A \in \mathrm{Ty}(\Gamma)$ with a morphism $\mathrm{p}(A) : \Gamma.A \to \Gamma$ in $\mathcal{C}$ (resp. a term $\mathrm{v}_A \in \mathrm{Tm}(\Gamma.A, A\{\mathrm{p}(A)\})$), called the ***first projection*** on A (resp. the ***second projection*** on A);
- $\langle \_, \_ \rangle_\_$ assigns, to each triple (f, A, g) of a morphism $f : \Delta \to \Gamma$ in $\mathcal{C}$, a type $A \in \mathrm{Ty}(\Gamma)$ and a term $g \in \mathrm{Tm}(\Delta, A\{f\})$, a morphism $\langle f, g \rangle_A : \Delta \to \Gamma.A$ in $\mathcal{C}$, called the ***extension*** of f by g,

that satisfies, for any $\Gamma, \Delta, \Theta \in \mathcal{C}$, $A \in \mathrm{Ty}(\Gamma)$, $f : \Delta \to \Gamma$, $e : \Theta \to \Delta$, $h \in \mathrm{Tm}(\Gamma, A)$ and $g \in \mathrm{Tm}(\Delta, A\{f\})$, the equations

- (Ty-Id) $A\{\mathrm{id}_\Gamma\} = A$;
- (Ty-Comp) $A\{f \circ e\} = A\{f\}\{e\}$;
- (Tm-Id) $h\{\mathrm{id}_\Gamma\}_A = h$;
- (Tm-Comp) $h\{f \circ e\}_A = h\{f\}_A\{e\}_{A\{f\}}$;
- (Cons-L) $\mathrm{p}(A) \circ \langle f, g \rangle_A = f$;
- (Cons-R) $\mathrm{v}_A\{\langle f, g \rangle_A\} = g$;
- (Cons-Nat) $\langle f, g \rangle_A \circ e = \langle f \circ e, g\{e\}_{A\{f\}} \rangle_A$;
- (Cons-Id) $\langle \mathrm{p}(A), \mathrm{v}_A \rangle_A = \mathrm{id}_{\Gamma.A}$.

Roughly, judgements of MLTT are interpreted in a CwF $\mathcal{C}$ by

$$\vdash \mathsf{\Gamma}\ \mathsf{ctx} \mapsto [\![\mathsf{\Gamma}]\!] \in \mathcal{C} \qquad \mathsf{\Gamma} \vdash \mathsf{A}\ \mathsf{type} \mapsto [\![\mathsf{A}]\!] \in \mathrm{Ty}([\![\mathsf{\Gamma}]\!])$$

$$\mathsf{\Gamma} \vdash \mathsf{a} : \mathsf{A} \mapsto [\![\mathsf{a}]\!] \in \mathrm{Tm}([\![\mathsf{\Gamma}]\!], [\![\mathsf{A}]\!]) \qquad \vdash \mathsf{\Gamma} = \mathsf{\Delta}\ \mathsf{ctx} \Rightarrow [\![\mathsf{\Gamma}]\!] = [\![\mathsf{\Delta}]\!] \in \mathcal{C}$$

$$\mathsf{\Gamma} \vdash \mathsf{A} = \mathsf{B}\ \mathsf{type} \Rightarrow [\![\mathsf{A}]\!] = [\![\mathsf{B}]\!] \qquad \mathsf{\Gamma} \vdash \mathsf{a} = \mathsf{a}' : \mathsf{A} \Rightarrow [\![\mathsf{a}]\!] = [\![\mathsf{a}']\!],$$

where $[\![\_]\!]$ denotes the *semantic map* or *interpretation*. See [52] for the details.

Let us now turn to introducing our game-semantic CwF:

Definition 4.12 (Game-semantic CwF) We define the CwF $\mathbb{WPG}$ as follows:

- The category $\mathbb{WPG}$ is given in Definition 3.19, and T is the terminal p-game;
- $\mathrm{Ty}(\Gamma) := \mathscr{D}^{\mathrm{w}}(\Gamma)$ $(\Gamma \in \mathbb{WPG})$ and $\mathrm{Tm}(\Gamma, A) := \mathbb{WPG}(\Pi(\Gamma, A))$ $(A \in \mathscr{D}^{\mathrm{w}}(\Gamma))$;

- Given $\phi : \Delta \to \Gamma$ in $\mathbb{WPG}$, the map $\_\{\phi\} : \mathrm{Ty}(\Gamma) \to \mathrm{Ty}(\Delta)$ is defined by $A\{\phi\}(\delta) := A(\phi^\dagger \circ \delta)$ for all $A \in \mathrm{Ty}(\Gamma)$ and $\delta : !\Delta$, and the map $\_\{\phi\}_A : \mathrm{Tm}(\Gamma, A) \to \mathrm{Tm}(\Delta, A\{\phi\})$ by $\alpha\{\phi\}_A := \iota_A(\phi) \bullet \alpha \bullet \phi$ for all $\alpha \in \mathrm{Tm}(\Gamma, A)$, where $\iota_A(\phi)$ is the evident dereliction $\mathscr{P}\big(\bigcup_{\gamma:!\Gamma} \overline{A}(\gamma)\big) \hookrightarrow \mathscr{P}\big(\bigcup_{\delta:!\Delta} \overline{A\{\phi\}}(\delta)\big)$;
- $\Gamma.A := \Sigma(\Gamma, A)$, $\mathrm{p}(A) := \mathrm{der}_\Gamma : \Sigma(\Gamma, A) \to \Gamma$ (up to 'tags'), $\mathrm{v}_A := \mathrm{der}_{|A|} : \Pi(\Sigma(\Gamma, A), A\{\mathrm{p}(A)\})$ (up to 'tags') and $\langle \phi, \alpha \rangle_A := \langle \phi, \alpha \rangle : \Delta \to \Sigma(\Gamma, A)$.

Remark The dereliction $\iota_A(\phi)$ is crucial for the substitution $\alpha\{\phi\}_A$ to satisfy the axiom (1). For instance, consider the strategy $\zeta : \Pi(N, \mathrm{List}_N)$ in Example 4.6 and the constant strategy $\underline{1} : T \to N$. The composition $\zeta \bullet \underline{1} : T \to |\mathrm{List}_N|$ does not satisfy (1) since in $(\zeta \bullet \underline{1}) \bullet \top : |\mathrm{List}_N|$ O can perform any first move in $|\mathrm{List}_N|$.

Notation Given $\Gamma \in \mathbb{WPG}$ and $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)$, we write $\mathbb{WPG}(\Gamma, A)$ for $\mathrm{Tm}(\Gamma, A)$.

We write $\mathrm{fst}_{\Sigma(\Gamma,A)}$ and $\mathrm{snd}_{\Sigma(\Gamma,A)}$ for the projections $\mathrm{p}(A)$ and $\mathrm{v}(A)$, respectively, and frequently omit the subscripts $(\_)_A$ and $(\_)_{\Sigma(\Gamma,A)}$ on the components of $\mathbb{WPG}$.

Theorem 4.13 (Well-defined game-semantic CwF) *The CCC $\mathbb{WPG}$ together with the structures defined in Definition 4.12 gives rise to a CwF.*

Proof Let us focus on the substitutions on terms and the extensions since the other structures of the CwF $\mathbb{WPG}$ are simpler to verify. Let $\Gamma, \Delta \in \mathbb{WPG}$, $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)$, $\phi \in \mathbb{WPG}(\Delta, \Gamma)$, $\alpha \in \mathbb{WPG}(\Gamma, A)$ and $\breve{\alpha} \in \mathbb{WPG}(\Delta, A\{\phi\})$.

By Lemma 3.18, the substitution $\alpha\{\phi\}$ is winning and w.b. on $\Delta \Rightarrow |A\{\phi\}|$. To show $\alpha\{\phi\} \in \mathbb{WPG}(\Delta, A\{\phi\})$, let $\delta^\dagger : !\Delta$; we have to show $\alpha\{\phi\} \circ \delta \subseteq \overline{A\{\phi\}}(\delta)$. By Lemma 3.17, $\alpha\{\phi\} \bullet \delta = \iota_A(\phi) \bullet \alpha \circ (\phi^\dagger \bullet \delta)$, where $\alpha \circ (\phi^\dagger \bullet \delta) \subseteq \overline{A}(\phi^\dagger \bullet \delta)$ by the axiom (1) on α. Hence, we conclude $\alpha\{\phi\} \circ \delta = \iota_A(\phi) \bullet \alpha \circ (\phi^\dagger \bullet \delta) \subseteq \overline{A\{\phi\}}(\delta)$.

Similarly, the extension $\langle \phi, \breve{\alpha} \rangle$ is winning and w.b. on $\Delta \Rightarrow \Gamma \,\&\, |A|$. The family $(\langle \phi, \breve{\alpha} \rangle_\delta)_{\delta:!\Delta}$ of strategies $\langle \phi, \breve{\alpha} \rangle_\delta := \{\, \boldsymbol{s} \in \langle \phi, \breve{\alpha} \rangle \mid \boldsymbol{s} \upharpoonright !\Delta \in \delta \,\}$ is an FoPLI from $!\Delta$ to $\Gamma \& |A|$ whose union is $\langle \phi, \breve{\alpha} \rangle$. We have to show that this family meets the first axiom on FoPLIs from $!\Delta$ to $\Sigma(\Gamma, A)$. Fix $\delta : !\Delta$; by $\langle \phi, \breve{\alpha} \rangle_\delta \circ \delta = \langle \phi \circ \delta, \breve{\alpha} \circ \delta \rangle$, it suffices to verify $\breve{\alpha} \subseteq \overline{A}(\phi \circ \delta)$, which follows from $\breve{\alpha} \circ \delta \subseteq \overline{A\{\phi\}}(\delta)$ and $\overline{A}(\phi \circ \delta) \preccurlyeq \overline{A\{\phi\}}(\delta)$. $\square$

Remark The function $\theta \mapsto (\pi_1 \bullet \theta, \pi_2\{\theta\})$ is the bijection between strategies $\theta \in \mathbb{WPG}(\Delta, \Sigma(\Gamma, A))$ and pairs $(\phi, \breve{\alpha})$ of strategies $\phi \in \mathbb{WPG}(\Delta, \Gamma)$ and $\breve{\alpha} \in \mathbb{WPG}(\Delta, A\{\phi\})$ mentioned in a paragraph between Definition 4.8 and Example 4.9. Its inverse is the extension $(\phi, \breve{\alpha}) \mapsto \langle \phi, \breve{\alpha} \rangle$. Seeing the bijection closely, it satisfies the crucial axiom (1) for $\pi_2\{\theta\} \in \mathbb{WPG}(\Delta, A\{\pi_1 \bullet \theta\})$ by the axiom on the sigma $\Sigma(\Gamma, A)$ with the first axiom of FoPLIs on θ, which constitutes a particular instance of the substitution of terms shown to be well-defined in the proof of Theorem 4.13.

4.5 Game-semantic type formers

CwFs only interpret the fragment of MLTT common to all types. Hence, in this section, we equip the CwF $\mathbb{WPG}$ with *semantic type formers* [52] that interpret One-, Zero-, N-, Pi-, Sigma- and Id-types as well as universes (à la Tarski).

4.5.1 Game semantics of Pi-type

We begin with *Pi-type*. Recall first the semantic type former of Pi-type:

Definition 4.14 (CwFs with Pi-type [52]) A CwF $\mathcal{C}$ ***supports pi*** if

- (Π-Form) Given $\Gamma \in \mathcal{C}$, $A \in \mathrm{Ty}(\Gamma)$ and $B \in \mathrm{Ty}(\Gamma.A)$, there is a type $\Pi(A,B) \in \mathrm{Ty}(\Gamma)$;
- (Π-Intro) Given $\beta \in \mathrm{Tm}(\Gamma.A, B)$, there is a term $\lambda_{A,B}(\beta) \in \mathrm{Tm}(\Gamma, \Pi(A,B))$;
- (Π-Elim) Given $\kappa \in \mathrm{Tm}(\Gamma, \Pi(A,B))$ and $\alpha \in \mathrm{Tm}(\Gamma, A)$, there is a term $\mathrm{App}_{A,B}(\kappa,\alpha) \in \mathrm{Tm}(\Gamma, B\{\overline{\alpha}\})$, where $\overline{\alpha} := \langle \mathrm{id}_\Gamma, \alpha\rangle_A : \Gamma \to \Gamma.A$;
- (Π-Comp) $\mathrm{App}_{A,B}(\lambda_{A,B}(\beta),\alpha) = \beta\{\overline{\alpha}\}$;
- (Π-Subst) Given $\Delta \in \mathcal{C}$ and $\phi : \Delta \to \Gamma$ in $\mathcal{C}$, $\Pi(A,B)\{\phi\} = \Pi(A\{\phi\}, B\{\phi^+\})$, where $\phi^+ := \langle f \circ \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{\phi\}}\rangle_A : \Delta.A\{\phi\} \to \Gamma.A$;
- (λ-Subst) $\lambda_{A,B}(b)\{\phi\} = \lambda_{A\{\phi\},B\{f^+\}}(b\{\phi^+\}) \in \mathrm{Tm}(\Delta, \Pi(A\{\phi\}, B\{\phi^+\}))$;
- (App-Subst) $\mathrm{App}_{A,B}(\kappa,\alpha)\{\phi\} = \mathrm{App}_{A\{\phi\},B\{\phi^+\}}(\kappa\{\phi\},\alpha\{\phi\}) \in \mathrm{Tm}(\Delta, B\{\overline{\alpha}\}\{\phi\})$.

Furthermore, $\mathcal{C}$ ***supports pi in the strict sense*** if it additionally satisfies

- (λ-Uniq) $\lambda_{A,B} \circ \mathrm{App}_{A\{\mathrm{p}(A)\},B\{\mathrm{p}(A)^+\}}(\kappa\{\mathrm{p}(A)\}, \mathrm{v}_A) = \kappa$.

Pi-type (with η-rule) is modelled in a CwF that supports pi (in the strict sense) [52]. We now proceed to show that the CwF $\mathbb{WPG}$ supports pi in the strict sense.

Lemma 4.15 (Pi-sigma correspondence) *Given $\Gamma \in \mathbb{WPG}$, $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)$ and $B \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Gamma,A))$, there is a bijection $\lambda_{A,B} : \mathbb{WPG}(\Sigma(\Gamma,A),B) \xrightarrow{\sim} \mathbb{WPG}(\Gamma,\Pi(A,B))$, where $\Pi(A,B) \in \mathscr{D}^{\mathrm{w}}(\Gamma)$ is defined by $|\Pi(A,B)| := |A| \Rightarrow |B|$ and $\Pi(A,B)(\gamma_0) := \Pi(A(\gamma_0), B_{\gamma_0})$ for all $\gamma_0 \in \mathbb{WPG}(\Gamma)$, and $B_{\gamma_0} \in \mathscr{D}^{\mathrm{w}}(A(\gamma_0))$ by $|B_{\gamma_0}| := |B|$ and $B_{\gamma_0}(\alpha) := B(\langle \gamma_0, \alpha_0\rangle)$ for all $\alpha_0 \in \mathbb{WPG}(A(\gamma_0))$.*

Proof It is easy to verify $B_{\gamma_0} \in \mathscr{D}^{\mathrm{w}}(A(\gamma_0))$ ($\gamma_0 \in \mathbb{WPG}(\Gamma)$) and $\Pi(A,B) \in \mathscr{D}^{\mathrm{w}}(\Gamma)$; let us leave the details to the reader. It then suffices to show that the p-games $\Pi(\Sigma(\Gamma,A),B)$ and $\Pi(\Gamma,\Pi(A,B))$ coincide up to 'tags' for disjoint union as well-bracketing and winning are both invariant with respect to the 'tags.' Let $\phi : \Pi(\Sigma(\Gamma,A),B)$; we obtain $\phi' : \Pi(\Gamma,\Pi(A,B))$ from ϕ by adjusting 'tags' as follows.

Let $\boldsymbol{s} \in \phi$ and $\boldsymbol{t} := \boldsymbol{s} \upharpoonright !\Sigma(\Gamma,A)$. Then, for each $i \in \mathbb{N}$, $\boldsymbol{t} \upharpoonright i$ is in some $\gamma : \Gamma$ or α such that $\langle \gamma, \alpha \rangle : \Sigma(\Gamma,A)$. Hence, $\boldsymbol{t} \in (\otimes_{i\in\mathbb{N}}\gamma_i) \otimes (\otimes_{j\in\mathbb{N}}\alpha_j)$ for some $(\gamma_i : \Gamma)_{i\in\mathbb{N}}$ and $(\alpha_j : |A|)_{j\in\mathbb{N}}$. Thus, we can adjust 'tags' or *curry* ϕ with respect to the adjunction between tensor $\otimes$ and linear implication $\multimap$ [38], obtaining $\phi' : \Gamma \Rightarrow (|A| \Rightarrow |B|)$.

Next, fix $\gamma : !\Gamma$; for $\phi' : \Pi(\Gamma,\Pi(A,B))$, it remains to show $\phi' \circ \gamma \subseteq \overline{\Pi(A,B)}(\gamma)$. We can prove it by induction on the length of elements of $\phi' \circ \gamma$, where the assumption $\phi : \Pi(\Sigma(\Gamma,A),B)$ provides the conditions for the induction to go through; we leave the details to the reader. Therefore, $\phi' : \Pi(\Gamma,\Pi(A,B))$, defining the required map $\lambda_{A,B} : \phi \mapsto \phi'$ which only adjusts 'tags.' The inverse is given in the same vein. $\square$

Example 4.16 Define $N_b^+ \in \mathscr{D}(\Sigma(N,N_b))$ by $N_b^+(\langle \underline{n}, \underline{k}\rangle^\dagger) := N_b(\underline{n+k}^\dagger)$ for all $n,k \in \mathbb{N}$ with $n \leqslant k$ and $|N_b^+| := N$. For each $\phi : \Pi(\Gamma,A)$, if we adopted another axiom $\forall \gamma_0 \in \mathbb{WPG}(!\Gamma).\, \phi \circ \gamma_0 : A(\gamma_0)$ in place of (1) in Definition 4.4, then the bijection $\mathbb{WPG}(\Sigma(N,N_b),N_b^+) \cong \mathbb{WPG}(N,\Pi(N_b,N_b^+))$ would fail as der_N between $\Sigma(N,N_b)$

and N_b^+ is in $\mathbb{WPG}(\Sigma(N, N_b), N_b^+)$, but $\lambda_{N_b, N_b^+}(\mathrm{der}_N) \notin \mathbb{WPG}(N, \Pi(N_b, N_b^+))$ as $\lambda_{N_b, N_b^+}(\mathrm{der}_N) \bullet \underline{0} = \mathrm{der}_N \notin \mathbb{WPG}(N_b(\underline{0}^\dagger), (N_b^+)_{\underline{0}^\dagger}) = \mathbb{WPG}(\underline{0}, \underline{0})$.

This example also explains why Definition 3.3 requires $P_{A(\gamma_0)} \subseteq P_{|A|}$, not $A(\gamma_0) \subseteq |A|$, for all $\gamma_0 \in \mathbb{WPG}(!\Gamma)$: $N_b(\underline{n}^\dagger) \subseteq N = |N_b|$ and $N_b^+(\langle \underline{n}, \underline{k} \rangle^\dagger) = N = |N_b^+|$ for all $n, k \in \mathbb{N}$ with $n \leqslant k$; however, $\Pi(N_b, N_b^+)(\underline{0}^\dagger) = \underline{0} \Rightarrow \underline{0} \not\subseteq N \Rightarrow N = |\Pi(N_b, N_b^+)|$.

Theorem 4.17 (Game semantics of Pi-type) *The CwF* $\mathbb{WPG}$ *strictly supports pi.*

Proof Let $\Gamma \in \mathbb{WPG}$, $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)$, $B \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Gamma, A))$ and $\beta \in \mathbb{WPG}(\Sigma(\Gamma, A), B)$.

- (Π-Form) $\Pi(A, B) \in \mathscr{D}^{\mathrm{w}}(\Gamma)$ is defined in Lemma 4.15.
- (Π-Intro) By Lemma 4.15, we obtain $\lambda_{A,B}(\beta) \in \mathbb{WPG}(\Gamma, \Pi(A, B))$. We often omit the subscripts $(\_)_{A,B}$ on $\lambda_{A,B}$ and the evident inverse $\lambda_{A,B}^{-1}$.
- (Π-Elim) Define $\mathrm{App}_{A,B}(\kappa, \alpha) := \lambda_{A,B}^{-1}(\kappa)\{\overline{\alpha}\}$ for all $\kappa \in \mathbb{WPG}(\Gamma, \Pi(A, B))$ and $\alpha \in \mathbb{WPG}(\Gamma, A)$. We can show $\lambda_{A,B}^{-1}(\kappa)\{\overline{\alpha}\} : \Pi(\Gamma, B\{\overline{\alpha}\})$ as in the proof of Theorem 4.13, whence $\mathrm{App}_{A,B}(\kappa, \alpha) \in \mathbb{WPG}(\Gamma, B\{\overline{\alpha}\})$ by Lemma 3.18. We often omit the subscripts $(\_)_{A,B}$ on $\mathrm{App}_{A,B}$.
- (Π-Comp) $\mathrm{App}_{A,B}(\lambda_{A,B}(\beta), \alpha) = \lambda_{A,B}^{-1}(\lambda_{A,B}(\beta))\{\overline{\alpha}\} = \beta\{\overline{\alpha}\}$.
- (Π-Subst) Given $\Delta \in \mathbb{WPG}$ and $\phi : \Delta \to \Gamma$ in $\mathbb{WPG}$,

$$\begin{aligned}
\Pi(A, B)\{\phi\} &= (\Pi(A(\gamma_0), B_{\gamma_0}))_{\gamma_0 \in \mathbb{WPG}(!\Gamma)}\{\phi\} \\
&= (\Pi(A(\phi^\dagger \circ \delta_0), B_{\phi^\dagger \circ \delta_0}))_{\delta_0 \in \mathbb{WPG}(!\Delta)} \\
&= (\Pi(A\{\phi\}(\delta_0), B\{\phi^+\}_{\delta_0}))_{\delta_0 \in \mathbb{WPG}(!\Delta)} \\
&= \Pi(A\{\phi\}, B\{\phi^+\}),
\end{aligned}$$

  where the third equation holds because for all $\hat{\alpha}_0 \in \mathbb{WPG}(!A(\phi \bullet \delta))$ we have

$$\begin{aligned}
B\{\phi^+\}_{\delta_0}(\hat{\alpha}_0) &= B\{\phi^+\}(\langle \delta_0, \hat{\alpha}_0 \rangle) \\
&= B(\langle \phi \bullet \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{\phi\}} \rangle^\dagger \circ \langle \delta_0, \hat{\alpha}_0 \rangle) \\
&= B(\langle \phi^\dagger \circ \delta_0, \hat{\alpha}_0 \rangle) \\
&= B_{\phi^\dagger \circ \delta_0}(\hat{\alpha}_0).
\end{aligned}$$

- (λ-Subst) We clearly have

$$\begin{aligned}
\lambda_{A,B}(\beta)\{\phi\} &= \iota \bullet \lambda_{A,B}(\beta) \bullet \phi \\
&= \lambda_{A\{\phi\}, B\{\phi^+\}}(\iota \bullet \beta^\dagger \circ \langle \phi \bullet \mathrm{fst}_{\Sigma(\Delta, A\{\phi\})}, \mathrm{snd}_{\Sigma(\Delta, A\{\phi\})} \rangle^\dagger) \quad \text{(by the definition of } \lambda) \\
&= \lambda_{A\{\phi\}, B\{\phi^+\}}(\iota \bullet \beta^\dagger \bullet \langle \phi \bullet \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{f\}} \rangle) \\
&= \lambda_{A\{\phi\}, B\{\phi^+\}}(\beta\{\phi^+\}).
\end{aligned}$$

- (App-Subst) Moreover, it is easy to see that

$$\begin{aligned}\mathrm{App}_{A,B}(\kappa,\alpha)\{\phi\} &= \lambda^{-1}_{A,B}(\kappa)\{\langle \mathrm{der}_\Gamma, \alpha\rangle\}\{\phi\} \\ &= \iota \bullet \lambda^{-1}_{A,B}(\kappa) \bullet (\langle \mathrm{der}_\Gamma, \alpha\rangle \bullet \phi) \\ &= \iota \bullet \lambda^{-1}_{A,B}(\kappa) \bullet \langle \phi, \alpha\{\phi\}\rangle \\ &= \iota \bullet \lambda^{-1}_{A,B}(\kappa) \bullet (\langle \phi \bullet \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{\phi\}}\rangle \bullet \langle \mathrm{der}_\Delta, \alpha\{\phi\}\rangle) \\ &= (\lambda^{-1}_{A,B}(\kappa)\{\phi^+\}) \bullet \overline{(\alpha\{\phi\})} \\ &= \lambda^{-1}_{A\{\phi\},B\{\phi^+\}}(\kappa\{\phi\})\{\overline{(\alpha\{\phi\})}\} \quad (\text{by } \lambda\text{-Subst}) \\ &= \mathrm{App}_{A\{\phi\},B\{\phi^+\}}(\kappa\{\phi\}, \alpha\{\phi\}) \quad (\text{by } \Pi\text{-Comp}).\end{aligned}$$

- (λ-Uniq) Finally, we have

$$\begin{aligned}\lambda_{A,B}(\mathrm{App}_{A\{\mathrm{p}(A)\},B\{\mathrm{p}(A)^+\}}(\kappa\{\mathrm{p}(A)\}, \mathrm{v}_A)) &= \lambda_{A,B}(\lambda^{-1}_{A\{\mathrm{p}(A)\},B\{\mathrm{p}(A)^+\}}(\kappa\{\mathrm{p}(A)\})\{\overline{\mathrm{v}_A}\}) \\ &= \lambda_{A,B}((\lambda^{-1}_{A,B}(\kappa)\{\mathrm{p}(A)^+\})\{\overline{\mathrm{v}_A}\}) \quad (\text{by } \lambda\text{-Subst}) \\ &= \lambda_{A,B}(\lambda^{-1}_{A,B}(\kappa) \bullet (\mathrm{p}(A)^+ \bullet \overline{\mathrm{v}_A})) \\ &= \lambda_{A,B}(\lambda^{-1}_{A,B}(\kappa) \bullet \mathrm{der}_{\Sigma(\Gamma,A)}) \\ &= \lambda_{A,B}(\lambda^{-1}_{A,B}(\kappa)) \\ &= k,\end{aligned}$$

where $\mathrm{p}(A)^+ := \langle \mathrm{p}(A) \bullet \mathrm{p}(A\{\mathrm{p}(A)\}), \mathrm{v}_{A\{\mathrm{p}(A)\}}\rangle : \Sigma(\Sigma(\Gamma,A), A\{\mathrm{p}(A)\}) \to \Sigma(\Gamma, A)$ and $\overline{\mathrm{v}_A} := \langle \mathrm{der}_{\Sigma(\Gamma,A)}, \mathrm{v}_A\rangle : \Sigma(\Gamma,A) \to \Sigma(\Sigma(\Gamma,A), A\{\mathrm{p}(A)\})$,

which completes the proof. □

4.5.2 Game semantics of Sigma-type

Next, we consider *Sigma-type*. Again, we first recall its semantic type former:

Definition 4.18 (CwFs with Sigma-type [52]) A CwF $\mathcal{C}$ ***supports sigma*** if

- (Σ-Form) Given $\Gamma \in \mathcal{C}$, $A \in \mathrm{Ty}(\Gamma)$ and $B \in \mathrm{Ty}(\Gamma.A)$, there is a type $\Sigma(A,B) \in \mathrm{Ty}(\Gamma)$;
- (Σ-Intro) There is a morphism $\mathrm{Pair}_{A,B} : \Gamma.A.B \to \Gamma.\Sigma(A,B)$ in $\mathcal{C}$;
- (Σ-Elim) Given $P \in \mathrm{Ty}(\Gamma.\Sigma(A,B))$ and $\rho \in \mathrm{Tm}(\Gamma.A.B, P\{\mathrm{Pair}_{A,B}\})$, there is a term $\mathcal{R}^\Sigma_{A,B,P}(\rho) \in \mathrm{Tm}(\Gamma.\Sigma(A,B), P)$;
- (Σ-Comp) $\mathcal{R}^\Sigma_{A,B,P}(\rho)\{\mathrm{Pair}_{A,B}\} = \rho$;
- (Σ-Subst) Given $\Delta \in \mathcal{C}$ and $\phi : \Delta \to \Gamma$ in $\mathcal{C}$, $\Sigma(A,B)\{\phi\} = \Sigma(A\{\phi\}, B\{\phi^+\})$, where $\phi^+ := \langle \phi \circ \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{\phi\}}\rangle_A : \Delta.A\{\phi\} \to \Gamma.A$;
- (Pair-Subst) $\mathrm{p}(\Sigma(A,B)) \circ \mathrm{Pair}_{A,B} = \mathrm{p}(A) \circ \mathrm{p}(B)$ and $\phi^\star \circ \mathrm{Pair}_{A\{\phi\},B\{\phi^+\}} = \mathrm{Pair}_{A,B} \circ \phi^{++}$, where $\phi^\star := \langle \phi \circ \mathrm{p}(\Sigma(A,B)\{\phi\}), \mathrm{v}_{\Sigma(A,B)\{\phi\}}\rangle_{\Sigma(A,B)} : \Delta.\Sigma(A,B)\{\phi\} \to \Gamma.\Sigma(A,B)$ and $\phi^{++} := \langle \phi^+ \circ \mathrm{p}(B\{\phi^+\}), \mathrm{v}_{B\{\phi^+\}}\rangle_B : \Delta.A\{\phi\}.B\{\phi^+\} \to \Gamma.A.B$;
- ($\mathcal{R}^\Sigma$-Subst) $\mathcal{R}^\Sigma_{A,B,P}(p)\{\phi^\star\} = \mathcal{R}^\Sigma_{A\{f\},B\{f^+\},P\{f^\star\}}(p\{\phi^{++}\})$.

Moreover, $\mathcal{C}$ ***supports sigma in the strict sense*** if it also satisfies

- ($\mathcal{R}^\Sigma$-Uniq) If $\rho \in \mathrm{Tm}(\Gamma.A.B, P\{\mathrm{Pair}_{A,B}\})$, $\sigma \in \mathrm{Tm}(\Gamma.\Sigma(A,B), P)$ and $\sigma\{\mathrm{Pair}_{A,B}\} = \rho$, then $\sigma = \mathcal{R}^\Sigma_{A,B,P}(\rho)$.

Sigma-types with (η-rule) are interpreted in a CwF that supports sigma (in the strict sense) [52]. Now, we describe our game-semantic interpretation of Sigma-type:

Theorem 4.19 (Game semantics of Sigma-type) *The CwF $\mathbb{WPG}$ supports sigma in the strict sense.*

Proof Let $\Gamma \in \mathbb{WPG}$, $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)$ and $B \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Gamma, A))$.

- (Σ-Form) Similarly to pi Π, define $\Sigma(A, B) := (\Sigma(A(\gamma_0), B_{\gamma_0}))_{\gamma_0 \in \mathbb{WPG}(\Gamma)}$.
- (Σ-Intro) By the obvious bijection $\Sigma(\Sigma(\Gamma, A), B) \cong \Sigma(\Gamma, \Sigma(A, B))$, define $\mathrm{Pair}_{A,B} : \Sigma(\Sigma(\Gamma, A), B) \to \Sigma(\Gamma, \Sigma(A, B))$ to be the evident dereliction up to 'tags,' i.e., $\mathrm{Pair}_{A,B} := \langle \mathrm{fst} \bullet \mathrm{fst}, \langle \mathrm{snd}\{\mathrm{fst}\}, \mathrm{snd} \rangle \rangle$. Note that the inverse is $\mathrm{Pair}_{A,B}^{-1} = \langle \langle \mathrm{fst}, \mathrm{snd}_1 \rangle, \mathrm{snd}_2 \rangle$, where $\mathrm{snd}_1 : \Sigma(\Gamma, \Sigma(A, B)) \to A\{\mathrm{fst}\}$ and $\mathrm{snd}_2 : \Sigma(\Gamma, \Sigma(A, B)) \to B\{\langle \mathrm{fst}, \mathrm{snd}_1 \rangle\}$ are the evident derelictions, respectively.
- (Σ-Elim) Given $P \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Gamma, \Sigma(A, B)))$ and $\xi \in \mathbb{WPG}(\Pi(\Sigma(\Sigma(\Gamma, A), B), P\{\mathrm{Pair}_{A,B}\}))$, define $\mathcal{R}^{\Sigma}_{A,B,P}(\xi) \in \mathbb{WPG}(\Pi(\Sigma(\Gamma, \Sigma(A, B)), P))$ to be $\xi\{\mathrm{Pair}_{A,B}^{-1}\}$.
- (Σ-Comp) We have

$$\begin{aligned}
\mathcal{R}^{\Sigma}_{A,B,P}(\xi)\{\mathrm{Pair}_{A,B}\} &= \mathcal{R}^{\Sigma}_{A,B,P}(\xi)\{\mathrm{Pair}_{A,B}\} \\
&= \xi\{\mathrm{Pair}_{A,B}^{-1}\}\{\mathrm{Pair}_{A,B}\} \\
&= \xi\{\mathrm{Pair}_{A,B}^{-1} \bullet \mathrm{Pair}_{A,B}\} \\
&= \xi\{\mathrm{id}_{\Sigma(\Sigma(\Gamma,A),B)}\} \\
&= \xi.
\end{aligned}$$

- (Σ-Subst) Similar to the case of pi Π.
- (Pair-Subst) Under the same assumption, we have

$$\begin{aligned}
\mathrm{p}(\Sigma(A, B)) \bullet \mathrm{Pair}_{A,B} &= \mathrm{fst} \bullet \langle \mathrm{fst} \bullet \mathrm{fst}, \langle \mathrm{snd}\{\mathrm{fst}\}, \mathrm{snd} \rangle \rangle \\
&= \mathrm{fst} \bullet \mathrm{fst} \\
&= \mathrm{p}(A) \bullet \mathrm{p}(B),
\end{aligned}$$

and also we have

$$\begin{aligned}
\phi^{\star} \bullet \mathrm{Pair}_{A\{\phi\},B\{\phi^+\}} &= \langle \phi \bullet \mathrm{p}(\Sigma(A, B)\{\phi\}), \mathrm{v}_{\Sigma(A,B)\{\phi\}} \rangle \bullet \mathrm{Pair}_{A\{\phi\},B\{\phi^+\}} \\
&= \langle \phi \bullet \mathrm{p}(\Sigma(A\{\phi\}, B\{\phi^+\})) \bullet \mathrm{Pair}_{A\{\phi\},B\{\phi^+\}}, \mathrm{v}_{\Sigma(A,B)\{\phi\}}\{\mathrm{Pair}_{A\{\phi\},B\{\phi^+\}}\} \rangle \\
&= \langle \phi \bullet \mathrm{p}(A\{\phi\}) \bullet \mathrm{p}(B\{\phi^+\}), \mathrm{v}_{\Sigma(A\{\phi\},B\{\phi^+\})}\{\mathrm{Pair}_{A\{\phi\},B\{\phi^+\}}\} \rangle \\
&\text{(by the above equations)} \\
&= \langle \phi \bullet \mathrm{fst} \bullet \mathrm{fst}, \mathrm{snd}\{\langle \mathrm{fst} \bullet \mathrm{fst}, \langle \mathrm{snd}\{\mathrm{fst}\}, \mathrm{snd} \rangle \rangle\} \rangle \\
&= \langle \phi \bullet \mathrm{fst} \bullet \mathrm{fst}, \langle \mathrm{snd}\{\mathrm{fst}\}, \mathrm{snd} \rangle \rangle \\
&= \langle \mathrm{fst} \bullet \mathrm{fst}, \langle \mathrm{snd}\{\mathrm{fst}\}, \mathrm{snd} \rangle \rangle \bullet \langle \langle \phi \bullet \mathrm{fst} \bullet \mathrm{fst}, \mathrm{snd}\{\mathrm{fst}\} \rangle, \mathrm{snd} \rangle \\
&= \langle \mathrm{fst} \bullet \mathrm{fst}, \langle \mathrm{snd}\{\mathrm{fst}\}, \mathrm{snd} \rangle \rangle \bullet \langle \langle \phi \bullet \mathrm{fst}, \mathrm{snd} \rangle \bullet \mathrm{fst}, \mathrm{snd} \rangle \\
&= \mathrm{Pair}_{A,B} \bullet \langle \langle \phi \bullet \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{\phi\}} \rangle \bullet \mathrm{p}(B\{\phi^+\}), \mathrm{v}_{B\{\phi^+\}} \rangle \\
&= \mathrm{Pair}_{A,B} \bullet \langle \phi^+ \bullet \mathrm{p}(B\{\phi^+\}), \mathrm{v}_{B\{\phi^+\}} \rangle \\
&= \mathrm{Pair}_{A,B} \bullet \phi^{++}.
\end{aligned}$$

- ($\mathcal{R}^{\Sigma}$-Subst) We have

$$\begin{aligned}
\mathcal{R}^{\Sigma}_{A,B,P}(\xi)\{\phi^\star\} &= \xi\{\mathrm{Pair}^{-1}_{A,B}\}\{\langle\phi \bullet \mathrm{p}(\Sigma(A,B)\{\phi\}, \mathrm{v}_{\Sigma(A,B)\{\phi\}}\rangle\} \\
&= \xi\{\langle\langle \mathrm{fst}, \mathrm{snd}_1\rangle, \mathrm{snd}_2\rangle \bullet \langle\phi \bullet \mathrm{fst}, \mathrm{snd}\rangle\} \\
&= \xi\{\langle\langle\phi \bullet \mathrm{fst}, \mathrm{snd}_1\{\langle\phi \bullet \mathrm{fst}, \mathrm{snd}\rangle\}\rangle, \mathrm{snd}_2\{\langle\phi \bullet \mathrm{fst}, \mathrm{snd}\rangle\}\rangle\} \\
&= \xi\{\langle\langle\phi \bullet \mathrm{fst}, \mathrm{snd}_1\rangle, \mathrm{snd}_2\rangle\} \\
&= \xi\{\langle\langle\phi \bullet \mathrm{fst}, \mathrm{snd}\rangle \bullet \mathrm{fst}, \mathrm{snd}\rangle \bullet \langle\langle \mathrm{fst}, \mathrm{snd}_1\rangle, \mathrm{snd}_2\rangle\} \\
&= \xi\{\langle\phi^+ \bullet \mathrm{fst}, \mathrm{snd}\rangle \bullet \langle\langle \mathrm{fst}, \mathrm{snd}_1\rangle, \mathrm{snd}_2\rangle\} \\
&= \xi\{\langle\phi^+ \bullet \mathrm{p}(B\{\phi^+\}), \mathrm{v}_{B\{\phi^+\}}\rangle \bullet \mathrm{Pair}^{-1}_{A\{\phi\}, B\{\phi^+\}}\} \\
&= \mathcal{R}^{\Sigma}_{A\{\phi\}, B\{\phi^+\}, P\{\phi^\star\}}(\xi\{\langle\phi^+ \bullet \mathrm{p}(B\{\phi^+\}), \mathrm{v}_{B\{\phi^+\}}\rangle\}) \\
&= \mathcal{R}^{\Sigma}_{A\{\phi\}, B\{\phi^+\}, P\{\phi^\star\}}(\xi\{\phi^{++}\}).
\end{aligned}$$

- ($\mathcal{R}^{\Sigma}$-Uniq) Given $\rho \in \mathbb{WPG}(\Pi(\Sigma(\Gamma, \Sigma(A,B)), P))$ and $\rho\{\mathrm{Pair}_{A,B}\} = \xi$,

$$\rho = \rho\{\mathrm{id}_{\Sigma(\Gamma,\Sigma(A,B))}\} = \rho\{\mathrm{Pair}_{A,B}\}\{\mathrm{Pair}^{-1}_{A,B}\} = \xi\{\mathrm{Pair}^{-1}_{A,B}\} = \mathcal{R}^{\Sigma}_{A,B,P}(\xi),$$

which completes the proof. □

4.5.3 Game semantics of N-type

We proceed to give our game semantics of the *N-type*. Again, let us first present the semantic type former of N-type:

Definition 4.20 (CwFs with the N-type [52]) A CwF $\mathcal{C}$ ***supports N*** if:

- (N-Form) Given $\Gamma \in \mathcal{C}$, there is a type $N_{[\Gamma]} \in \mathrm{Ty}(\Gamma)$, called ***natural number (N-) type*** in Γ, which we often abbreviate as N;
- (N-Intro) There are a term $\underline{0}_\Gamma \in \mathrm{Tm}(\Gamma, N)$ and a morphism $\mathrm{succ}_\Gamma : \Gamma.N \to \Gamma.N$ in $\mathcal{C}$ that satisfy for any morphisms $f : \Delta \to \Gamma$ and $g : \Delta.N \to \Gamma$ in $\mathcal{C}$

$$\underline{0}_\Gamma\{f\} = \underline{0}_\Delta \in \mathrm{Tm}(\Delta, N) \qquad \mathrm{p}(N) \circ \mathrm{succ}_\Gamma = \mathrm{p}(N) : \Gamma.N \to \Gamma$$

$$\mathrm{succ}_\Gamma \circ \langle g, \mathrm{v}_N\rangle_N = \langle g, \mathrm{v}_N\{\mathrm{succ}_\Delta\}\rangle_N : \Delta.N \to \Gamma.N;$$

  Notation Let $\mathrm{zero}_\Gamma := \langle \mathrm{id}_\Gamma, \underline{0}_\Gamma\rangle_N : \Gamma \to \Gamma.N$ for each $\Gamma \in \mathcal{C}$; it satisfies $\mathrm{zero}_\Gamma \circ f = \langle f, \underline{0}_\Delta\rangle_N = \langle f, \mathrm{v}_N\{\mathrm{zero}_\Delta\}\rangle_N : \Delta \to \Gamma.N$ for any $f : \Delta \to \Gamma$ in $\mathcal{C}$. We often omit the subscript $(\_)_\Gamma$ on $\underline{0}$, zero and succ. Define $\underline{n}_\Gamma \in \mathrm{Tm}(\Gamma, N)$ for each $n \in \mathbb{N}$ by: $\underline{0}_\Gamma$ is already given, and $\underline{n+1}_\Gamma := \mathrm{v}_N\{\mathrm{succ}_\Gamma \circ \langle \mathrm{id}_\Gamma, \underline{n}_\Gamma\rangle\}$.

- (N-Elim) Given a type $P \in \mathrm{Ty}(\Gamma.N)$, and terms $c_{\mathrm{z}} \in \mathrm{Tm}(\Gamma, P\{\mathrm{zero}\})$ and $c_{\mathrm{s}} \in \mathrm{Tm}(\Gamma.N.P, P\{\mathrm{succ} \circ \mathrm{p}(P)\})$, there is a term $\mathcal{R}^N_P(c_{\mathrm{z}}, c_{\mathrm{s}}) \in \mathrm{Tm}(\Gamma.N, P)$;
- (N-Comp) We have the following equations:

$$\begin{aligned}
&\mathcal{R}^N_P(c_{\mathrm{z}}, c_{\mathrm{s}})\{\mathrm{zero}\} = c_{\mathrm{z}} \in \mathrm{Tm}(\Gamma, P\{\mathrm{zero}\}); \\
&\mathcal{R}^N_P(c_{\mathrm{z}}, c_{\mathrm{s}})\{\mathrm{succ}\} = c_{\mathrm{s}}\{\langle \mathrm{id}_{\Gamma.N}, \mathcal{R}^N_P(c_{\mathrm{z}}, c_{\mathrm{s}})\rangle_P\} \in \mathrm{Tm}(\Gamma.N, P\{\mathrm{succ}\});
\end{aligned}$$

- (N-Subst) $N^\Gamma\{f\} = N^\Delta \in \mathrm{Ty}(\Delta)$;

- ($\mathcal{R}^N$-Subst) $\mathcal{R}^N_P(c_z, c_s)\{\phi^+\} = \mathcal{R}^N_{P\{\phi^+\}}(c_z\{\phi\}, c_s\{\phi^{++}\}) \in \mathrm{Tm}(\Delta.N, P\{\phi^+\})$, where $\phi^+ := \langle \phi \circ \mathrm{p}(N), \mathrm{v}_N\rangle_N : \Delta.N \to \Gamma.N$ and $\phi^{++} := \langle \phi^+ \circ \mathrm{p}(P\{\phi^+\}), \mathrm{v}_{P\{\phi^+\}}\rangle_P : \Delta.N.P\{\phi^+\} \to \Gamma.N.P$.

The following is basically the *winning* part of the game semantics of PCF [26]:

Theorem 4.21 (Game semantics of the N-type) *The CwF* $\mathbb{WPG}$ *supports N.*

Proof Let $\Gamma, \Delta \in \mathbb{WPG}$, $\phi : \Delta \to \Gamma$ and $\psi : \Sigma(\Delta, \{N\}_\Delta) \to \Gamma$ in $\mathbb{WPG}$.

- (N-Form) N^Γ is the constant dependent p-game $\{N\}_{!\Gamma}$ (Example 3.4).
- (N-Intro) $\underline{0}_\Gamma \in \mathbb{WPG}(\Gamma, \{N\})$ is $\underline{0}$ up to 'tags,' and $\mathrm{succ}_\Gamma : \Sigma(\Gamma, \{N_{[0]}\}) \to \Sigma(\Gamma, \{N_{[1]}\})$ is $\langle \mathrm{p}(\{N\}), \mathrm{s}_\Gamma\rangle$, where $\mathrm{s}_\Gamma : \Pi(\Sigma(\Gamma, \{N_{[0]}\}), \{N_{[1]}\})$ is defined by $\mathrm{s}_\Gamma := \mathrm{Pref}(\{\, q_{[1]} q_{[0]} n_{[0]} (n+1)_{[0]} \mid n \in \mathbb{N}\,\})$. Clearly, $\underline{0}_\Gamma \bullet \phi = \underline{0}_\Delta$ and $\mathrm{s}_\Gamma \bullet \langle g, \mathrm{v}_{\{N\}_\Delta}\rangle = \mathrm{s}_\Delta = \mathrm{v}_{\{N\}_\Delta}\{\mathrm{succ}_\Delta\}$, and thus the required equations hold.
- (N-Elim) Given $P \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Gamma, \{N\}))$, $c_z \in \mathbb{WPG}(\Gamma, P\{\mathrm{zero}\})$ and $c_s \in \mathbb{WPG}(\Sigma(\Sigma(\Gamma, \{N\}), P), P\{\mathrm{succ} \bullet \mathrm{p}(P)\})$, there are two terms

$$\widetilde{c}_z \in \mathbb{WPG}(\Sigma(\Pi(\Sigma(\Gamma, \{N\}), P), \{\Sigma(\Gamma, \{N\})\}), P\{\mathrm{zero} \bullet \mathrm{fst} \bullet \mathrm{snd}\});$$
$$\widetilde{c}_s \in \mathbb{WPG}(\Sigma(\Pi(\Sigma(\Gamma, \{N\}), P), \{\Sigma(\Gamma, \{N\})\}), P\{\mathrm{succ} \bullet \mathrm{pred} \bullet \mathrm{snd}\})$$

defined respectively by:

$$\widetilde{c}_z : \Pi(\Sigma(\Gamma, \{N\}), P) \,\&\, \Sigma(\Gamma, \{N\}) \xrightarrow{\mathrm{snd}} \Sigma(\Gamma, \{N\}) \xrightarrow{\mathrm{fst}} \Gamma \xrightarrow{c_z} |P\{\mathrm{zero}\}|;$$
$$\widetilde{c}_s : \Pi(\Sigma(\Gamma, \{N\}), P) \,\&\, \Sigma(\Gamma, \{N\}) \xrightarrow{\langle \mathrm{pred} \bullet \mathrm{snd}, \mathrm{ev}_P\{\langle \mathrm{fst}, \mathrm{pred} \bullet \mathrm{snd}\rangle\}\rangle} \Sigma(\Gamma, \{N\}) \,\&\, |P| \xrightarrow{c_s} |P\{\mathrm{succ} \bullet \mathrm{fst}\}|$$

where $\mathrm{ev}_P \in \mathbb{WPG}(\Sigma(\Pi(\Sigma(\Gamma, \{N\}), P), \{\Sigma(\Gamma, \{N\})\}), P\{\mathrm{snd}\})$ is the *evaluation* on P [17], i.e., $\mathrm{ev}_P := \lambda^{-1}(\mathrm{der}_{\Pi(\Sigma(\Gamma,\{N\}),P)})$, and $\mathrm{pred}_\Gamma : \Sigma(\Gamma, \{N_{[0]}\}) \to \Sigma(\Gamma, \{N_{[1]}\})$ is $\mathrm{Pref}(\{q_{[1]} q_{[0]} 0_{[0]} 0_{[0]}\} \cup \{\, q_{[1]} q_{[0]} (n+1)_{[0]} n_{[0]} \mid n \in \mathbb{N}\,\})$.
In addition, define

$$P_z := P\{\mathrm{zero} \bullet \mathrm{p}(N)\} \in \mathrm{Ty}(\Sigma(\Gamma, \{N\}));$$
$$P_s := P\{\mathrm{succ} \bullet \mathrm{pred} \bullet \mathrm{p}(P_z)\} \in \mathrm{Ty}(\Sigma(\Sigma(\Gamma, \{N\}), P_z)).$$

We also have $\mathrm{cond}_P \in \mathbb{WPG}(\Sigma(\Sigma(\Sigma(\Gamma, \{N\}), P_z), P_s), P\{\mathrm{p}(P_z) \bullet \mathrm{p}(P_s)\})$, which is the standard interpretation of *conditionals* on P [21, 22, 26, 38]: It first asks an input natural number in the component N of the domain, and plays as the dereliction between P_z and $P\{\mathrm{p}(P_z) \bullet \mathrm{p}(P_s)\}$ if the answer is $\underline{0}$, and as the dereliction between P_s and $P\{\mathrm{p}(P_z) \bullet \mathrm{p}(P_s)\}$ otherwise.
We then define $\mathcal{F}^N_P(c_z, c_s) : \Pi(\Sigma(\Gamma, \{N\}), P) \to \Pi(\Sigma(\Gamma, \{N\}), P)$ by:

$$\mathcal{F}^N_P(c_z, c_s) := \lambda_{\{\Sigma(\Gamma,\{N\})\},\{P\{\mathrm{snd}\}\}}(\mathrm{cond}_P\{\langle\langle \mathrm{snd}, \widetilde{c}_z\rangle, \widetilde{c}_s\rangle\}).$$

Finally, we define $\mathcal{R}^N_P(c_z, c_s) \in \mathbb{WPG}(\Sigma(\Gamma, \{N\}), P)$ to be the least upper bound of the following chain of $(\mathcal{R}^N_P(c_z, c_s)_n : \Pi(\Sigma(\Gamma, \{N\}), P))_{n \in \mathbb{N}}$:

$$\mathcal{R}^N_P(c_z, c_s)_0 := \top \qquad \mathcal{R}^N_P(c_z, c_s)_{n+1} := \mathcal{F}^N_P(c_z, c_s) \bullet \mathcal{R}^N_P(c_z, c_s)_n.$$

- (N-Comp) By the definition of $\mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})$, we have

$$\mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})\{\mathrm{zero}\} = c_\mathrm{z} \in \mathbb{WPG}(\Gamma, P\{\mathrm{zero}\});$$
$$\mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})\{\mathrm{succ}\} = c_\mathrm{s}\{\langle \mathrm{der}_{\Sigma(\Gamma,\{N\})}, \mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})\rangle\} \in \mathbb{WPG}(\Sigma(\Gamma, \{N\}), P\{\mathrm{succ}\}).$$

- (N-Subst) It is clear that $\{N\}_{!\Gamma}\{f\} = \{N\}_{!\Delta}$.
- ($\mathcal{R}^N$-Subst) Finally, by the definition of $\mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})$, we have

$$\mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})\{\phi^+\} = \mathcal{R}^N_{P\{\phi^+\}}(c_\mathrm{z}\{\phi\}, c_\mathrm{s}\{\phi^{++}\})$$

(or alternatively show $\mathcal{R}^N_P(c_\mathrm{z}, c_\mathrm{s})_n\{\phi^+\} = \mathcal{R}^N_{P\{\phi^+\}}(c_\mathrm{z}\{\phi\}, c_\mathrm{s}\{\phi^{++}\})_n$ for all $n \in \mathbb{N}$ by induction on n so that the above equation holds) which completes the proof. □

4.5.4 Game semantics of One- and Zero-types

Next, we interpret *One-* and *Zero-types*. Since it is trivial to interpret these types, we only sketch the proof. See [52] for the semantic type formers for these types.

Theorem 4.22 (Game semantics of one- and zero-types) *The CwF $\mathbb{WPG}$ supports the semantic type formers for One- (in the strict sense) and Zero-types.*

Proof (sketch) We interpret One- and Zero-types by the constant dependent p-games valued at the terminal p-game T and the empty p-game $\mathbf{0}$, respectively. □

4.5.5 Game semantics of Id-type

Let us proceed to our game semantics of *Id-types*. Again, we first review the semantic type former of Id-type:

Definition 4.23 (CwFs with Id-types [52]) A CwF $\mathcal{C}$ ***supports Id*** if

- (Id-Form) Given $\Gamma \in \mathcal{C}$ and $A \in \mathrm{Ty}(\Gamma)$, there is a type $\mathrm{Id}_A \in \mathrm{Ty}(\Gamma.A.A^+)$, where $A^+ := A\{\mathrm{p}(A)\} \in \mathrm{Ty}(\Gamma.A)$;
- (Id-Intro) There is a morphism $\mathrm{Refl}_A : \Gamma.A \to \Gamma.A.A^+.\mathrm{Id}_A$ in $\mathcal{C}$;
- (Id-Elim) Given $B \in \mathrm{Ty}(\Gamma.A.A^+.\mathrm{Id}_A)$ and $\beta \in \mathrm{Tm}(\Gamma.A, B\{\mathrm{Refl}_A\})$, there is a term $\mathcal{R}^{\mathrm{Id}}_{A,B}(\beta) \in \mathrm{Tm}(\Gamma.A.A^+.\mathrm{Id}_A, B)$;
- (Id-Comp) $\mathcal{R}^{\mathrm{Id}}_{A,B}(\beta)\{\mathrm{Refl}_A\} = \beta$;
- (Id-Subst) $\mathrm{Id}_A\{f^{++}\} = \mathrm{Id}_{A\{f\}} \in \mathrm{Ty}(\Delta.A\{f\}.A\{\phi\}^+)$ for all $\Delta \in \mathcal{C}$ and $\phi : \Delta \to \Gamma$ in $\mathcal{C}$, where $A\{\phi\}^+ := A\{\phi\}\{\mathrm{p}(A\{\phi\})\} \in \mathrm{Ty}(\Delta.A\{\phi\})$, $\phi^+ := \langle \phi \circ \mathrm{p}(A\{\phi\}), \mathrm{v}_{A\{\phi\}}\rangle_A : \Delta.A\{\phi\} \to \Gamma.A$ and $\phi^{++} := \langle \phi^+ \circ \mathrm{p}(A^+\{\phi^+\}), \mathrm{v}_{A^+\{\phi^+\}}\rangle_{A^+} : \Delta.A\{\phi\}.A^+\{\phi^+\} \to \Gamma.A.A^+$;
- (Refl-Subst) $\mathrm{Refl}_A \circ \phi^+ = \phi^{+++} \circ \mathrm{Refl}_{A\{\phi\}} : \Delta.A\{\phi\} \to \Gamma.A.A^+.\mathrm{Id}_A$, where $\phi^{+++} := \langle \phi^{++} \circ \mathrm{p}(\mathrm{Id}_A\{\phi^{++}\}), \mathrm{v}_{\mathrm{Id}_A\{\phi^{++}\}}\rangle_{\mathrm{Id}_A} : \Delta.A\{\phi\}.A^+\{\phi^+\}.\mathrm{Id}_{A\{\phi\}} \to \Gamma.A.A^+.\mathrm{Id}_A$;
- ($\mathcal{R}^{\mathrm{Id}}$-Subst) $\mathcal{R}^{\mathrm{Id}}_{A,B}(\beta)\{\phi^{+++}\} = \mathcal{R}^{\mathrm{Id}}_{A\{\phi\},B\{\phi^{+++}\}}(\beta\{\phi^+\})$.

Theorem 4.24 (Game semantics of Id-type) *The CwF $\mathbb{WPG}$ supports Id.*

Proof Let $\Gamma \in \mathbb{WPG}$ and $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)$.

- (Id-Form) Define $\mathsf{Id}_A := (\mathsf{Id}_{A(\gamma_0)}(\alpha_0, \alpha'_0))_{\langle\langle\gamma_0,\alpha_0\rangle,\alpha'_0\rangle \in \mathbb{WPG}(\Sigma(\Sigma(\Gamma,A),A^+))}$, where $\mathrm{Id}_{A(\gamma_0)}(\alpha_0, \alpha'_0) := \begin{cases} T & \text{if } \alpha_0 = \alpha'_0; \\ \mathbf{0} & \text{otherwise.} \end{cases}$
- (Id-Intro) Define $\mathrm{Refl}_A : \Sigma(\Gamma, A_{[1]}) \to \Sigma(\Sigma(\Sigma(\Gamma, A_{[2]}), A^+_{[3]}), \mathsf{Id}_A)$ to be the strategy that plays as the dereliction between $\Sigma(\Gamma, A_{[1]})$ and $\Sigma(\Gamma, A_{[2]})$, between $|A_{[1]}|$ and $|A^+_{[3]}|$, or as the canonical one $\Sigma(\Gamma, A_{[1]}) \to \mathbf{1}$. Note that there is the inverse $\mathrm{Refl}_A^{-1} : \Sigma(\Sigma(\Sigma(\Gamma, A_{[2]}), A^+_{[3]}), \mathsf{Id}_A) \to \Sigma(\Gamma, A_{[1]})$ which is the dereliction between $\Sigma(\Gamma, A_{[2]})$ and $\Sigma(\Gamma, A_{[1]})$ up to 'tags.'
- (Id-Elim) Given $B \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Sigma(\Sigma(\Gamma, A_{[2]}), A^+_{[3]}), \mathsf{Id}_A))$ and $\beta \in \mathbb{WPG}(\Sigma(\Gamma, A), B\{\mathrm{Refl}_A\})$, define $\mathcal{R}^{\mathsf{Id}}_{A,B}(\beta) := \beta\{\mathrm{Refl}_A^{-1}\} \in \mathbb{WPG}(\Sigma(\Sigma(\Sigma(\Gamma, A_{[1]}), A^+_{[2]}), \mathsf{Id}_A), B)$.

This interpretation of Id-type essentially coincides with that of the preceding work [27], and the remaining details are straightforward. Hence, we leave the remaining interpretation of Id-Comp, Id-Subst and Refl-Subst to the reader. □

4.5.6 Game semantics of universes

Definition 4.25 (Universe predicate games) For each $k \in \mathbb{N}$, we define a p-game $\mathcal{U}_k$, a full subcategory $\mathbb{WPG}_{\leqslant k} \hookrightarrow \mathbb{WPG}$ and a subset $\mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k} \subseteq \mathscr{D}^{\mathrm{w}}(\Gamma)$ for each $\Gamma \in \mathbb{WPG}_{\leqslant k}$ by the following mutual recursion:

- $\mathcal{U}_k := \{\, \underline{A} \circ \gamma_0 \mid \Gamma \in \mathbb{WPG}_{\leqslant k}, A \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}, \gamma_0 \in \mathbb{WPG}(!\Gamma) \,\}$, where we define the morphism $\underline{A} \in \mathbb{WPG}_{\leqslant k}(\Gamma, \mathcal{U}_k)$ by

$$\underline{A} := \mathrm{Pref}(\{q^{\mathsf{OQ}}.A^{\mathsf{PA}}\}) \quad (A \in \{\{\mathbf{1}\}_{!\Gamma}, \{\mathbf{0}\}_{!\Gamma}, \{N\}_{!\Gamma}, \{\mathcal{U}_i\}_{!\Gamma}\})$$

$$\underline{A\{\phi\}} := \underline{A} \bullet \phi \qquad\qquad \underline{\mathrm{El}(\phi)} := \phi$$

$$\underline{X(A,B)} := \mathrm{Pref}(\{\, q^{\mathsf{OQ}}.\sharp(X)^{\mathsf{PA}}.\boldsymbol{s} \mid \boldsymbol{s} \in \langle \underline{A}, \underline{B} \rangle \,\}) \quad (X \in \{\Pi, \Sigma\})$$

$$\underline{\mathrm{Id}_A(\alpha, \alpha')} := \mathrm{Pref}(\{\, q^{\mathsf{OQ}}.\sharp(\mathrm{Id})^{\mathsf{PA}}.\boldsymbol{s} \mid \boldsymbol{s} \in \langle \underline{A}, \langle \alpha, \alpha' \rangle \rangle \,\}),$$

  where $\sharp$ assigns arbitrary fixed natural numbers to Π, Σ and Id;
- $\mathbb{WPG}_{\leqslant k}$ is the least full subcategory of $\mathbb{WPG}$ that satisfies
  1. $\mathbf{1}, \mathbf{0}, N \in \mathbb{WPG}_{\leqslant k}$;
  2. $\mathcal{U}_i \in \mathbb{WPG}_{\leqslant k}$ if $i < k$;
  3. $\Sigma(\Gamma, A) \in \mathbb{WPG}_{\leqslant k}$ if $\Gamma \in \mathbb{WPG}_{\leqslant k}$ and $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$;
- $\mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ ($\Gamma \in \mathbb{WPG}_{\leqslant k}$) is the least subset of $\mathscr{D}^{\mathrm{w}}(\Gamma)$ that satisfies
  1. $\{G\}_{!\Gamma} \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ if $G \in \mathbb{WPG}_{\leqslant k}$;
  2. $A\{\phi\} \in \mathscr{D}^{\mathrm{w}}(\Delta)_{\leqslant k}$ if $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$, $\Delta \in \mathbb{WPG}_{\leqslant k}$ and $\phi \in \mathbb{WPG}(\Delta, \Gamma)$;
  3. $\Pi(A, B), \Sigma(A, B) \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ if $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ and $B \in \mathscr{D}^{\mathrm{w}}(\Sigma(\Gamma, A))_{\leqslant k}$;
  4. $\mathrm{Id}_A(\alpha, \alpha') \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ if $A \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ and $\alpha, \alpha' \in \mathbb{WPG}(\Pi(\Gamma, A))$;
  5. $\mathrm{El}(\phi) \in \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$ if $\phi \in \mathbb{WPG}(\Gamma, \mathcal{U}_j)$ such that $j \leqslant k$

  where $\mathrm{El}(\mu) \in \mathbb{WPG}$ for each $\mu \in \mathbb{WPG}(\mathcal{U})$ is the unique p-game that satisfies $\underline{\mathrm{El}(\mu)} = \mu$, and $\mathrm{El}(\phi) \in \mathscr{D}^{\mathrm{w}}(\Gamma)$ for each $\phi \in \mathbb{WPG}(\Gamma, \mathcal{U})$ is the unique dependent p-game over Γ given by $\mathrm{El}(\phi)(\gamma) := \mathrm{El}(\phi^{\dagger}\gamma)$ for all $\gamma : !\Gamma$.

Note that it is in general impossible to encode strategies α and α' by natural numbers. For instance, consider $A = \{N \Rightarrow N\}_{!T}$ and $\alpha, \alpha' : N \Rightarrow N$, where α

and α' are uncomputable ones given by O. Hence, we instead define the strategy $\underline{\mathrm{Id}_A(\alpha,\alpha')} := \mathrm{Pref}(\{\, q^{\mathsf{OQ}}.\sharp(\mathrm{Id})^{\mathsf{PA}}.\boldsymbol{s} \mid \boldsymbol{s} \in \langle \underline{A}, \langle \alpha, \alpha' \rangle \rangle \,\})$ by incorporating α and α' *themselves*. This novel technique enables us to encode our game semantics of Id-types *effectively* as strategies whose codomains are universes.

Definition 4.26 (The subcategory $\mathbb{UPG} \hookrightarrow \mathbb{WPG}$) The full subcategory $\mathbb{UPG} \hookrightarrow \mathbb{WPG}$ is defined by $\mathrm{ob}(\mathbb{UPG}) := \bigcup_{k \in \mathbb{N}} \mathrm{ob}(\mathbb{WPG}_{\leqslant k})$, and the subset $\mathscr{D}_{\mathbb{U}}(\Gamma) \subseteq \mathscr{D}^{\mathrm{w}}(\Gamma)$ $(\Gamma \in \mathbb{UPG})$ by $\mathscr{D}_{\mathbb{U}}(\Gamma) := \bigcup_{k \in \mathbb{N}} \mathscr{D}^{\mathrm{w}}(\Gamma)_{\leqslant k}$. We also write $\mathbb{UPG}(\Gamma)$ for $\mathbb{WPG}(\Gamma)$.

Definition 4.27 (CwFs with universes) A CwF $\mathcal{C}$ ***supports (the cumulative hierarchy of) universes*** if

- (U-Form) Given $\Gamma \in \mathcal{C}$, there is a type $\mathcal{U}_k^\Gamma \in \mathrm{Ty}(\Gamma)$ for each $k \in \mathbb{N}$, called the ***k^{th} universe*** in the context Γ, where we often omit the superscript $(\_)^\Gamma$ and/or the subscript $(\_)_k$ on $\mathcal{U}_k^\Gamma$;
- (U-Intro) Given $A \in \mathrm{Ty}(\Gamma)$, there is a term $\mathrm{En}(A) \in \mathrm{Tm}(\Gamma, \mathcal{U}_k)$ for some $k \in \mathbb{N}$, and in particular $\mathrm{En}(\mathcal{U}_k^\Gamma) \in \mathrm{Tm}(\Gamma, \mathcal{U}_{k+1}^\Gamma)$ for all $k \in \mathbb{N}$;
- (U-Elim) Each term $c \in \mathrm{Tm}(\Gamma, \mathcal{U})$ induces a type $\mathrm{El}(c) \in \mathrm{Ty}(\Gamma)$;
- (U-Comp) $\mathrm{El}(\mathrm{En}(A)) = A$ for all $A \in \mathrm{Ty}(\Gamma)$;
- (U-Cumul) If $c \in \mathrm{Tm}(\Gamma, \mathcal{U}_k)$, then $c \in \mathrm{Tm}(\Gamma, \mathcal{U}_{k+1})$;
- (U-Subst) Given $\phi : \Delta \to \Gamma$ in $\mathcal{C}$, $\mathcal{U}_k^\Gamma\{\phi\} = \mathcal{U}_k^\Delta \in \mathrm{Ty}(\Delta)$ for all $k \in \mathbb{N}$;
- (En-Subst) $\mathrm{En}(A)\{\phi\} = \mathrm{En}(A\{\phi\}) \in \mathrm{Tm}(\Delta, \mathcal{U})$ for all $A \in \mathrm{Ty}(\Gamma)$.

Theorem 4.28 (Game semantics of universes) *The CwF $\mathbb{UPG}$ supports universes.*

Proof Let $\Gamma, \Delta \in \mathbb{UPG}$, $\phi : \Delta \to \Gamma$ in $\mathbb{UPG}$ and $A \in \mathscr{D}_{\mathbb{U}}(\Gamma)$.

- (U-Form) $\mathcal{U}_k^\Gamma := \{\mathcal{U}_k\}_{!_\Gamma}$ for each $k \in \mathbb{N}$, where $\mathcal{U}_k$ is given in Definition 4.25.
- (U-Intro) $\mathrm{En}(A) := \underline{A} \in \mathbb{UPG}(\Pi(\Gamma, \{\mathcal{U}\}_{!_\Gamma}))$ (Definition 4.25);
- (U-Elim) The operation El is given in Definition 4.25;
- (U-Comp) We see that $\mathrm{El} \circ \mathrm{En}(A) = A$ holds by induction on A;
- (U-Cumul) Immediate from Definition 4.25;
- (U-Subst) As in the case of N-type;
- (En-Subst) $\mathrm{En}(A)\{\phi\} = \underline{A}\{\phi\} = \underline{A\{\phi\}} = \mathrm{En}(A\{\phi\})$,

which complete the proof. $\square$

Finally, it is straightforward to see that the CwF $\mathbb{UPG}$ supports the other semantic type formers considered so far in the same way the CwF $\mathbb{WPG}$ does:

Corollary 4.29 (Game semantics of MLTT) *The CwF $\mathbb{UPG}$ strictly supports One-, Zero-, N-, Pi-, Sigma- and Id-types as well as universes.*

4.6 Intensionality

The CwF $\mathbb{UPG}$ is not well-pointed since $\underline{0} \neq \underline{0}' : N \to N$, where $\underline{0} := \mathsf{Pref}(\{q0\})$ and $\underline{0}' := \mathsf{Pref}(\{\, qqn0 \mid n \in \mathbb{N} \,\})$, and $\underline{0} \bullet \sigma = \underline{0}' \bullet \sigma$ for all $\sigma \in \mathbb{UPG}(N)$. More generally, p-games are as *intensional* as games, which Corollary **??** shows.

In this section, we illustrate this intensionality of our game semantics in terms of the following principles within MLTT.

4.6.1 Equality reflection

Equality reflection (EqRefl) [11] states that propositionally equal terms are judgmentally equal, i.e., $\Gamma \vdash \mathsf{p} : \mathsf{Id}_\mathsf{A}(\mathsf{a}, \mathsf{a}')$ implies $\Gamma \vdash \mathsf{a} = \mathsf{a}' : \mathsf{A}$.

EqRefl is invalid in our game semantics. For example, terms $\mathsf{x} : \mathsf{N}, \mathsf{y} : \mathsf{0} \vdash \mathsf{x} : \mathsf{N}$ and $\mathsf{x} : \mathsf{N}, \mathsf{y} : \mathsf{0} \vdash \mathsf{succ}(\mathsf{x}) : \mathsf{N}$ in MLTT are interpreted respectively as morphisms $N \,\&\, \mathbf{0} \xrightarrow{\text{fst}} N \xrightarrow{\text{snd}} N$ and $N \,\&\, \mathbf{0} \xrightarrow{\text{fst}} N \xrightarrow{\text{succ}} N \xrightarrow{\text{snd}} N$ in $\mathbb{UPG}$. These morphisms are not equal, but there is the interpretation of a term $\mathsf{x} : \mathsf{N}, \mathsf{y} : \mathsf{0} \vdash \mathsf{p} : \mathsf{Id}_\mathsf{N}(\mathsf{succ}(\mathsf{x}), \mathsf{x})$ in $\mathbb{UPG}$ that plays in the component game $\mathbf{0}$ in the domain by the second move.

4.6.2 Function extensionality

Next, *function extensionality (FunExt)* [11, 15] states that given types $\Gamma \vdash \mathsf{A}\ \mathsf{type}$ and $\Gamma, \mathsf{x} : \mathsf{A} \vdash \mathsf{B}\ \mathsf{type}$, and terms $\Gamma \vdash \mathsf{f} : \Pi_{\mathsf{x}:\mathsf{A}}\mathsf{B}(\mathsf{x})$ and $\Gamma \vdash \mathsf{g} : \Pi_{\mathsf{x}:\mathsf{A}}\mathsf{B}(\mathsf{x})$, the type $\Gamma \vdash \Pi_{\mathsf{x}:\mathsf{A}}\mathsf{Id}_{\mathsf{B}(\mathsf{x})}(\mathsf{f}(\mathsf{x}), \mathsf{g}(\mathsf{x})) \Rightarrow \mathsf{Id}_{\Pi_{\mathsf{x}:\mathsf{A}}\mathsf{B}(\mathsf{x})}(\mathsf{f}, \mathsf{g})\ \mathsf{type}$ has a proof.

Our game semantics does not satisfy FunExt since strategies $f \in \mathbb{WPG}(\Pi(\Gamma, A))$ are in general not completely specified by the map $\gamma \in \mathbb{UPG}(!\Gamma) \mapsto f \circ \gamma : A(\gamma)$.

For instance, the morphisms $\underline{0}, \underline{0}' : N \to N$ given above serve as a concrete witness of our refutation of FunExt.

4.6.3 Uniqueness of identity proofs

Let us proceed to analyse *uniqueness of identity proofs (UIP)* [14], which states that there is a proof of the following type: $\Gamma \vdash \Pi_{\mathsf{a}_1,\mathsf{a}_2:\mathsf{A}}\Pi_{\mathsf{p},\mathsf{q}:\mathsf{Id}_\mathsf{A}(\mathsf{a}_1,\mathsf{a}_2)}\mathsf{Id}_{\mathsf{Id}_\mathsf{A}(\mathsf{a}_1,\mathsf{a}_2)}(\mathsf{p}, \mathsf{q})\ \mathsf{type}$.

Our game semantics validates UIP as follows. If the context Γ contains Zero-type, then the validation is trivial just as in the case of EqRefl. Hence, assume that Γ does not contain Zero-type. Then, by our interpretation of Id-types as either $\mathbf{1}$ or $\mathbf{0}$, our game semantics validates UIP in this case too.

4.6.4 Criteria of intensionality

There are Streicher's *Criteria of Intensionality* [54]:

1. $\mathsf{x}, \mathsf{y} : \mathsf{S}, \mathsf{z} : \mathsf{Id}_\mathsf{S}(\mathsf{x}, \mathsf{y}) \nvdash \mathsf{x} = \mathsf{y} : \mathsf{S}$;
2. $\mathsf{x}, \mathsf{y} : \mathsf{S}, \mathsf{z} : \mathsf{Id}_\mathsf{S}(\mathsf{x}, \mathsf{y}) \nvdash \mathsf{A}(\mathsf{x}) = \mathsf{A}(\mathsf{y})\ \mathsf{type}$;
3. If $\vdash \mathsf{p} : \mathsf{Id}_\mathsf{S}(\mathsf{t}, \mathsf{t}')$, then $\vdash \mathsf{t} = \mathsf{t}' : \mathsf{S}$

for some simple type S and dependent type $\mathsf{x} : \mathsf{S} \vdash \mathsf{A}\ \mathsf{type}$.

For the first criterion, take N-type N as S. There are derelictions on N (up to 'tags') as the interpretations of $\mathsf{x}, \mathsf{y} : \mathsf{N}, \mathsf{z} : \mathsf{Id}_\mathsf{N}(\mathsf{x}, \mathsf{y}) \vdash \mathsf{x} : \mathsf{N}$ and $\mathsf{x}, \mathsf{y} : \mathsf{N}, \mathsf{z} : \mathsf{Id}_\mathsf{N}(\mathsf{x}, \mathsf{y}) \vdash \mathsf{y} : \mathsf{N}$, respectively. They are distinct as O may behave differently for x and y. Hence, our game semantics of MLTT satisfies the first criterion of intensionality.

Next, our game semantics also satisfies the second criterion because dependent p-games are indexed by strategies, not necessarily winning ones.

Finally, our model satisfies the third criterion as the terms t and t' are *closed*.

4.7 Independence of Markov's principle

Markov's principle (MP) [30] is formulated within MLTT as the following type:

$$\Pi_{\mathsf{f}:\mathsf{N}\Rightarrow\mathsf{N}}\neg\neg\Sigma_{\mathsf{x}:\mathsf{N}}\mathsf{Id}_\mathsf{N}(\mathsf{f}(\mathsf{x}), \underline{0}) \Rightarrow \Sigma_{\mathsf{y}:\mathsf{N}}\mathsf{Id}_\mathsf{N}(\mathsf{f}(\mathsf{y}), \underline{0}), \tag{4}$$

where $\neg\mathsf{A}$ abbreviates $\mathsf{A} \Rightarrow \mathsf{0}$ for any type A.

There is *double negation elimination* in MP, and hence MP seems to be a *classical* principle. However, MP can be seen as a *constructive* principle as well since if we cannot show that a computable map $f : \mathbb{N} \to \mathbb{N}$ satisfies $\forall x \in \mathbb{N}. f(x) \neq 0$, then we can find a natural number $n \in \mathbb{N}$ such that $f(n) = 0$ by executing an algorithm for f to check the values $f(0), f(1), \dots$ until we encounter $f(n) = 0$. In fact, Hyland's *effective topos* [31], which gives a computational model of MLTT, validates MP.

Further, it is easy to see that the sequential algorithm model by Blot and Laird [28] validates MP as well. This phenomenon is hardly surprising since their model admits classical reasonings or *non-local controls* [28, §8].

On the other hand, MP is *independent* from MLTT [29]. Therefore, neither the effective topos nor the sequential algorithm model accurately captures MLTT.

In contrast, our game semantics *refutes* MP:

Theorem 4.30 (Game semantics refutes Markov's principle) *There is no game-semantic term on the interpretation of the type (4) in the CwF* $\mathbb{UPG}$*.*

Proof (sketch) Let us write

$$\Pi\Bigg(N_{[0]} \Rightarrow N_{[1]}, \Big(\big(\Sigma(N_{[2]}, \mathrm{Id}_N\{\langle \mathrm{App}(\pi_1, \pi_2), \underline{0}\rangle\}_{[3]}) \Rightarrow \mathbf{0}_{[4]}\big) \Rightarrow \mathbf{0}_{[5]}\Big) \Rightarrow \Sigma(N_{[6]}, \mathrm{Id}_N\{\langle \mathrm{App}(\pi_1, \pi_2), \underline{0}\rangle\}_{[7]})\Bigg) \tag{5}$$

for the interpretation of the type (4) in the CwF $\mathbb{UPG}$, and assume that there is a term $\langle \phi, \psi \rangle$ on this p-game in $\mathbb{UPG}$.

We first show that ϕ cannot be a constant one $\underline{n} : N_{[6]}$ (up to 'tags') for some $n \in \mathbb{N}$ as follows. Assume $\phi = \underline{n}$. Then, O may initiate a play on the p-game (5) by the question $q_{[7]}$ and on the domain play by any strategies $\varphi : N_{[0]} \Rightarrow N_{[1]}$ and $\langle \underline{m}, \top \rangle$ that satisfy $\varphi \bullet \underline{n} \neq \underline{0}$ and $\varphi \bullet \underline{m} = \underline{0}$, so that $\mathrm{Id}_N\{\langle \mathrm{App}(\pi_1, \pi_2), \underline{0}\rangle\}_{[7]} = \mathbf{0}$ and $\mathrm{Id}_N\{\langle \mathrm{App}(\pi_1, \pi_2), \underline{0}\rangle\}_{[3]} = T$. Consequently the p-game (5) is now of the form $\Delta \Rightarrow (\Gamma \Rightarrow \Theta)$ such that Δ and Γ have winning strategies, but Θ does not. Thus, it is easy to see that in this play any ψ cannot be winning, again a contradiction.

We have shown that ϕ cannot be constant, and therefore it must compute $q_{[6]} \mapsto q_{[5]}$ when O initiates a play by the question $q_{[6]}$. In this case, O may proceed by $q_{[6]} q_{[5]} \mapsto q_{[4]}$. At this point, note that ϕ must eventually provide an answer $k_{[7]} \in \mathbb{N}$ that satisfies $\varphi \bullet \underline{k} = \underline{0}$ since otherwise ψ would not be winning by the argument given in the preceding paragraph. However, such a computation by P would violate the well-bracketing of ϕ, a contradiction. □

Acknowledgements

The author acknowledges the financial support from Funai Overseas Scholarship, and is grateful to fruitful discussions with Samson Abramsky.

Conflict of interest statement

The author states that there is no conflict of interest.

References

1. Martin-Löf, P.: An Intuitionistic Theory of Types: Predicative Part. Studies in Logic and the Foundations of Mathematics **80**, 73–118 (1975)
2. Martin-Löf, P.: Intuitionistic Type Theory: Notes by Giovanni Sambin of a Series of Lectures Given in Padova, June 1980, (1984)
3. Martin-Löf, P.: An Intuitionistic Theory of Types. Twenty-five years of constructive type theory **36**, 127–172 (1998)
4. Troelstra, A.S., van Dalen, D.: Constructivism in mathematics. Vol. I, volume 121 of. Studies in Logic and the Foundations of Mathematics, 26 (1988)
5. Zermelo, E.: Untersuchungen über die grundlagen der mengenlehre. i. Mathematische Annalen **65**(2), 261–281 (1908)
6. Fraenkel, A.: Zu den grundlagen der cantor-zermeloschen mengenlehre. Mathematische annalen **86**(3-4), 230–237 (1922)
7. Martin-Löf, P.: Constructive Mathematics and Computer Programming. Studies in Logic and the Foundations of Mathematics **104**, 153–175 (1982)
8. Church, A.: A formulation of the simple theory of types. The journal of symbolic logic **5**(02), 56–68 (1940)
9. Sørensen, M.H., Urzyczyn, P.: Lectures on the Curry-Howard Isomorphism vol. 149. Elsevier, ??? (2006)
10. Constable, R., Allen, S., Bromley, H., Cleaveland, W., Cremer, J., Harper, R., Howe, D., Knoblock, T., Mendler, N., Panangaden, P., et al.: Implementing mathematics with the Nuprl proof development system (1986)
11. Dybjer, P., Palmgren, E.: Intuitionistic type theory. Stanford Encyclopedia of Philosophy (2016)
12. Amadio, R.M., Curien, P.-L.: Domains and Lambda-Calculi vol. 46. Cambridge University Press, Cambridge (1998)
13. Girard, J.-Y.: Linear logic. Theoretical Computer Science **50**(1), 1–101 (1987)
14. Hofmann, M., Streicher, T.: The Groupoid Interpretation of Type Theory. Twenty-five years of constructive type theory (Venice, 1995) **36**, 83–111 (1998)
15. Univalent Foundations Program, T.: Homotopy Type Theory: Univalent Foundations of Mathematics. https://homotopytypetheory.org/book, Institute for Advanced Study (2013)
16. Streicher, T.: Semantics of Type Theory: Correctness, Completeness and Independence Results. Springer, ??? (2012)
17. Abramsky, S., *et al.*: Semantics of interaction: An introduction to game semantics. Semantics and Logics of Computation **14**, 1–31 (1997)
18. Hyland, M.: Game semantics. In: Semantics and Logics of Computation vol. 14, p. 131. Cambridge University Press, New York (1997)
19. Curien, P.-L.: Definability and full abstraction. Electronic Notes in Theoretical Computer Science **172**, 301–310 (2007)
20. Abramsky, S., Jagadeesan, R.: Games and full completeness for multiplicative linear logic. The Journal of Symbolic Logic **59**(02), 543–574 (1994)
21. Abramsky, S., Jagadeesan, R., Malacaria, P.: Full abstraction for PCF. Information and Computation **163**(2), 409–470 (2000)
22. Hyland, J.M.E., Ong, C.-H.: On full abstraction for PCF: I, II, and III. Information and Computation **163**(2), 285–408 (2000)
23. Nickau, H.: Hereditarily sequential functionals. In: Logical Foundations of Computer Science, pp. 253–264. Springer, Berlin, Heidelberg (1994)
24. Abramsky, S.: Algorithmic game semantics. In: Proof and System-Reliability, pp. 21–47. Springer, ??? (2002)
25. Plotkin, G., Power, J.: Computational effects and operations: An overview. Electronic Notes in Theoretical Computer Science **73**, 149–163 (2004)
26. Abramsky, S., McCusker, G.: Game semantics. In: Computational Logic: Proceedings of the 1997 Marktoberdorf Summer School, pp. 1–55. Springer, Berlin, Heidelberg (1999)
27. Abramsky, S., Jagadeesan, R., Vákár, M.: Games for dependent types. In: Automata, Languages, and Programming, pp. 31–43. Springer, Berlin, Heidelberg (2015)
28. Blot, V., Laird, J.: Extensional and intensional semantic universes: A denotational model of dependent types. In: Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, pp. 95–104 (2018). ACM
29. Mannaa, B., Coquand, T.: The independence of markov's principle in type theory. Logical Methods in Computer Science **13** (2017)
30. Markov, A.A.: On constructive mathematics. Trudy Matematicheskogo Instituta imeni VA Steklova **67**, 8–14 (1962)
31. Hyland, J.M.E.: The effective topos. In: Studies in Logic and the Foundations of Mathematics vol. 110, pp. 165–216. Elsevier, ??? (1982)
32. Vákár, M., Jagadeesan, R., Abramsky, S.: Game semantics for dependent types. Information and Computation **261**, 401–431 (2018)
33. Palmgren, E., Stoltenberg-Hansen, V.: Domain interpretations of martin-löf's partial type theory. Annals of Pure and Applied Logic **48**(2), 135–196 (1990)
34. Abramsky, S., Jagadeesan, R.: A game semantics for generic polymorphism. Annals of Pure and Applied Logic **133**(1), 3–37 (2005)
35. Berry, G., Curien, P.-L.: Sequential algorithms on concrete data structures. Theoretical Computer Science **20**(3), 265–321 (1982)
36. Hyland, M., Schalk, A.: Games on graphs and sequentially realizable functionals. extended abstract. In: Proceedings 17th Annual IEEE Symposium on Logic in Computer Science, pp. 257–264 (2002). IEEE
37. Winskel, G.: Events in computation. PhD thesis, University of Edinburgh (1980)
38. McCusker, G.: Games and Full Abstraction for a Functional Metalanguage with Recursive Types. Springer,

London (1998)
39. Girard, J.-Y.: Locus solum: From the rules of logic to the logic of rules. Mathematical structures in computer science **11**(3), 301 (2001)
40. Yamada, N.: Game Semantics of Homotopy Type Theory. Talk presented at homotopy type theory electronic seminar talks (HoTTEST), Western University, Ontario, Canada, ??? (2021)
41. Yamada, N.: Game semantics of martin-löf type theory, part iii: its consistency with church's thesis. arXiv e-prints, 2007 (2020)
42. Maietti, M.E., Sambin, G.: Toward a minimalist foundation for constructive mathematics. From Sets and Types to Topology and Analysis: Practicable Foundations for Constructive Mathematics **48**, 91–114 (2005)
43. Ishihara, H., Maietti, M.E., Maschio, S., Streicher, T.: Consistency of the intensional level of the minimalist foundation with church's thesis and axiom of choice. Archive for Mathematical Logic **57**(7-8), 873–888 (2018)
44. Coquand, T.: A semantics of evidence for classical arithmetic. The Journal of Symbolic Logic **60**(1), 325–337 (1995)
45. Yamada, N.: A game-semantic model of computation. Research in the Mathematical Sciences **6**(1), 3 (2019)
46. Yamada, N.: Game semantics for Martin-Löf type theory. arXiv preprint arXiv:1610.01669 (2016)
47. Curien, P.-L.: Notes on game semantics. From the author?s web page (2006)
48. Curien, P.-L.: Abstract Böhm trees. Mathematical Structures in Computer Science **8**(06), 559–591 (1998)
49. Laurent, O.: Polarized games. In: Proceedings of the 17th Annual IEEE Symposium on Logic in Computer Science, pp. 265–274 (2002). IEEE
50. Dybjer, P.: Internal Type Theory. In: Types for Proofs and Programs, pp. 120–134. Springer, ??? (1996)
51. Clairambault, P., Dybjer, P.: The Biequivalence of Locally Cartesian Closed Categories and Martin-Löf Type Theories. Mathematical Structures in Computer Science **24**(05), 240501 (2014)
52. Hofmann, M.: Syntax and Semantics of Dependent Types. In: Extensional Constructs in Intensional Type Theory, pp. 13–54. Springer, ??? (1997)
53. Chroboczek, J.: Game semantics and subtyping. In: Proceedings Fifteenth Annual IEEE Symposium on Logic in Computer Science (Cat. No. 99CB36332), pp. 192–203 (2000). IEEE
54. Streicher, T.: Investigations Into Intensional Type Theory, (1993)