

Plays as Resource Terms via Non-idempotent Intersection Types

Takeshi Tsukada
University of Tokyo
tsukada@is.s.u-tokyo.ac.jp

C.-H. Luke Ong
University of Oxford
Luke.Ong@cs.ox.ac.uk

Abstract

A program is interpreted as a collection of resource terms by the Taylor expansion, as a collection of plays by game semantics, and as a collection of types by a non-idempotent intersection type assignment system. This paper investigates the connection between these models and aims to show that they are essentially the same in a certain sense. Technically we study the relational interpretations of resource terms and of plays, which can be seen as non-idempotent intersection type assignment systems for resource terms and plays, respectively. We show that both relational interpretations are injective, have the same image, and respect composition. This result allows us to study a property of the game model by using the syntax of a resource calculus and vice versa.

Categories and Subject Descriptors Theory of computation [Semantics and reasoning]: Program semantics

General Terms Theory

Keywords Game semantics, resource calculus, intersection types, relational models.

1. Introduction

The relationship between game-semantic and syntactic notions has been studied since the beginning of game semantics (or even before). For example, in the HO/N game model [26], the key notion of P-view corresponds to the notion of path of PCF terms in certain canonical form (see, e.g., [12, 13, 26]). This correspondence is important from both conceptual and technical points of view. Conceptually it provides us with an intuitive understanding of game-semantic notions (such as P-views and innocent strategies) through syntactic notions (such as paths of abstract syntax trees and terms in normal form), which are often more accessible. From the technical side, the correspondence is essential to the proof of the compact definability result, and hence that of the full abstraction result [26].

There are other examples. Danos, Herbelin and Regnier [16] have pointed out that interaction of innocent strategies in the HO/N game model can be seen as a sequence of *linear head reduction*, or a run of a *Pointer Abstract Machine*. They also developed an abstract machine for the AJM game model [4]. Mellies [35] has shown that a strategy in the *orbital game* model [32], which is a reformulation of the AJM game model, can be represented by a term (or Böhm tree) of an affine λ -calculus with tuples of length ω (named *non-uniform λ -calculus*).

However, the preceding results all concern themselves with the syntactic counterpart of *strategies*. To the best of our knowledge, no syntactic notion has been related to *plays* in the HO/N game model.

This is the problem we address in this paper. Specifically we answer the question:

What is the syntactic counterpart of plays in HO/N-games?

Fortunately we do not need to introduce a new calculus to answer the question: plays in HO/N-games are terms of a well-known and important calculus, the resource calculus [23].

Let us first point out similarities between plays and resource terms.

A play is a sequence of moves with an additional structure, called justification pointers. It is a *quantitative* notion in the sense that a play may contain one or more copies of the same move. Composition of plays s of type $A \rightarrow B$ and s' of type $B \rightarrow C$ is defined only if they match on the common component B . Hence composition is a *partial* operation on plays. An interpretation of a λ -term is a strategy, which is a *collection of plays*.

A term of the resource calculus is basically a term of the λ -calculus. The difference is that an application takes as the argument a multiset of terms (called a *bag*) and it is linear in the sense that each element of a bag must be used exactly once. For example, $f[\lambda x.x, \lambda x.x]$ is a resource term that expects f to call the argument twice. This is the *quantitative* aspect of the resource calculus. Consequently, substituting $\lambda h.h[z]$ for f is problematic since f calls its argument h only once. In fact, $(\lambda h.h[z])(\lambda x.x, \lambda x.x)$ is reduced to a meaningless value. In this sense, composition in the resource calculus is a *partial* operation. A λ -term can be interpreted as a *collection of resource terms* via the Taylor expansion [23].¹

There is a notable difference: composition of plays is a unique play if defined, but composition of terms, namely substitution followed by normalisation, results in a set of resource terms, which may have more than one element. This difference leads us to the idea that a resource term corresponds to a set of plays.

1.1 Contribution

The main result of this paper can be stated as follows (for precise statements, see Corollary 30, Theorem 39, and Theorem 44).

Theorem. *There exists a family of bijections parametrised by simple types T*

$$\varpi_T : (\text{plays of arena } T) / \sim \rightarrow (\text{resource terms of type } T)$$

that preserves composition (where \sim is the alternating homotopy relation of Mellies [35], see Definition 9). Via the bijections, the game-semantic interpretation of a simply-typed λ -term coincides with the Taylor expansion followed by normalisation.

As a by-product, we show that the category of HO/N games is isomorphic to a subCCC of the coKleisli category \mathbf{MRel} of the finite multiset comonad \mathcal{M}_{fin} on the category of sets and relations (Theorem 42). It is worth noting that the proofs of these results do not rely on any categorical structure of the game model; even the fact that HO/N games form a category is not used. On the contrary,

¹This paper considers only formal sums with coefficients in the ring $\{0, 1\}$.

we can show as a corollary that HO/N games form a category (and a CCC). Note that the purely game-semantic proof of this fact is quite involved.

This result has both technical and conceptual importance.

From the technical point of view, the coincidence provides us with a way to use the techniques of the Taylor expansion and/or MRel to study game semantics, and vice versa. An example application of such a technology transfer is discussed above: one can prove that the game model is a CCC (Theorem 43). Another is the characterisation of the image of the Taylor expansion. In game semantics, the image of the interpretation has been well-studied. One can exploit this idea in game semantics to characterise the image of the Taylor expansion. As an example, we give a complete characterisation of the image of the Taylor expansion of a nondeterministic simply-typed λ -terms (Theorem 48).

Conceptually this result suggests a deep connection between seemingly different research areas, game semantics and differential linear logic. It poses several questions that are worth investigating (see also Section 1.3).

1.2 Overview

A rough and informal sketch of the idea is shown in Figure 1. A resource term in normal form can be written as a tree, as in the middle of Figure 1. Each node is labelled by λ -abstraction followed by a head variable.² The edges express the relationship between functions and arguments: the child of a node is an argument of the head variable of the parent. Then we line up the nodes of the tree in such a way that every node is located to the left side of its children. The resulting sequence of λ -abstractions and variables is equipped with leftward pointers: the pointer from an abstraction (solid lines in Figure 1) comes from the parent-child relation in the tree, and that from a variable (dotted lines) is determined by the binder-bindee relation. The sequence with pointers is reminiscent of a play in the HO/N game model; indeed one can construct a play from it by replacing λ -abstraction and variables with O- and P-moves, respectively, in an appropriate way.

A resource term generates a set of plays because the process of lining up is nondeterministic. The main theorem states that the set of plays generated by a resource term is a \sim -equivalence class. Moreover every play is generated by a resource term.

The idea will now be intuitively clear. However the definition based on the above argument, which heavily depends on graphical operations, does not seem so easy to handle.

A tractable definition that we shall use in this paper is based on the interpretations on the category of sets and relations, or non-idempotent intersection assignment systems. In other words, we shall use non-idempotent intersection types as the bridge. This change is beneficial since the intersection type assignment systems for both the resource calculus and the game model have already been studied: a relational model is a common tool for studying resource terms as in [10], and game semantics for an intersection type assignment system has been studied in [36].

The definition of the bijection used in this paper is illustrated in Figure 2. It is divided into four steps.

Step 1: Colouring a play We assign a “colour” (written as α , β , γ and δ in Fig. 2) to each occurrence of moves in such a way that

- every consecutive O-P moves have the same colour, and
- different O-moves (reps. P-moves) have different colours.

Thus one needs n colours to annotate a play of length $2n$.

²We consider λ -abstraction that can bind a (possibly empty) sequence of variables, although only sequences of length 1 appear in Figure 1.

Step 2: Representing a coloured play by a tree This step simply forgets the sequential structure of the (coloured) play. The resulting structure is a tree whose edge is a justification pointer of the play and whose node is labelled by a pair of a move and a colour. For example, the move and colour of the node named l_7 in Figure 2 is o_{11} and δ . The node named l_i corresponds to the i th move in the original play. This structure is called a *valuated thick subtree* in [8], a *high-level arena* in [36], and a *partitioned position* in [20].

Step 3: Constructing an intersection type High-level arenas (or, valuated thick subtrees) bijectively correspond to intersection types that refine the simple type. Let us write τ_i for the type corresponding to the subtree whose root is l_i . For example, (1) τ_8 is an atomic type δ that refines o_{111} , (2) τ_3 is the type $\delta \rightarrow \beta$ that refines $o_{111} \rightarrow o_{11}$, (3) τ_5 is the type $\top \rightarrow \gamma$ that refines $o_{111} \rightarrow o_{11}$, where \top is the empty intersection type, and (4) τ_2 is the type $((\delta \rightarrow \beta) \wedge (\top \rightarrow \gamma)) \rightarrow \alpha$ that refines $(o_{111} \rightarrow o_{11}) \rightarrow o_1$. The type τ_1 is written in Figure 2.

Step 4: Computing the inhabitant The resource term corresponding to the play is the inhabitant of the intersection type. An inhabitant always exists uniquely (up to α -conversion) for an intersection type constructed in this way.

There are several difficulties in proving bijectivity of the map defined above. In particular, in the last step, the existence and uniqueness of an inhabitant are challenging to establish. Our strategy is to study the map defined by Steps 1-3 and the inverse of Step 4. We characterise the images of those maps, using game semantics, and show their coincidence. Then, given a type in the image, we construct a play and a resource term and prove their uniqueness.

Preservation of composition by the bijection is relatively easy to prove by applying known results.

1.3 Further considerations

This paper concerns only the simply-typed calculus, but the techniques are more generally applicable. To interpret an untyped calculus, one needs an arena with infinite moves. A problem here is that the notion of η -long terms does not make sense if a head variable expects an infinite arguments. To overcome the problem, one should use the *resource calculus with tests* [24] and then an analogy of the main theorem holds for this calculus.

The approach of this paper is not applicable to game models for stateful computation (e.g., [2]) since a strategy in such a model is not necessarily closed under the alternating homotopy \sim .

Several questions about the relationship between games and differential linear logic would naturally arise.

The first one concerns the possibility of extending the result to other calculi/logics. For example, can the game models of (fragments of) linear logic [1, 3, 34] be seen as the Taylor expansion in some sense? The converse also seems worth investigating; e.g. a game-semantic counterpart of the Taylor expansion of MELL proof-nets could turn out to be a known or new model. We are also interested in the π -calculus, which has interpretations based on game semantics [28] and on differential interaction nets [22].

The second question is about coefficients of the resource calculus. The Taylor expansion considered in this paper uses the resource calculus with coefficients taken from the Boolean algebra $\{0, 1\}$. This is often called the *support of the Taylor expansion*. It is natural to ask what is the game semantics of the Taylor expansion with more general coefficients as in [23]. The relationship to the probabilistic game model [14] would be worth studying.

1.4 Related work

The closest work to ours is that by Boudes [8], which concerns the connection between games and proof-nets for polarised linear

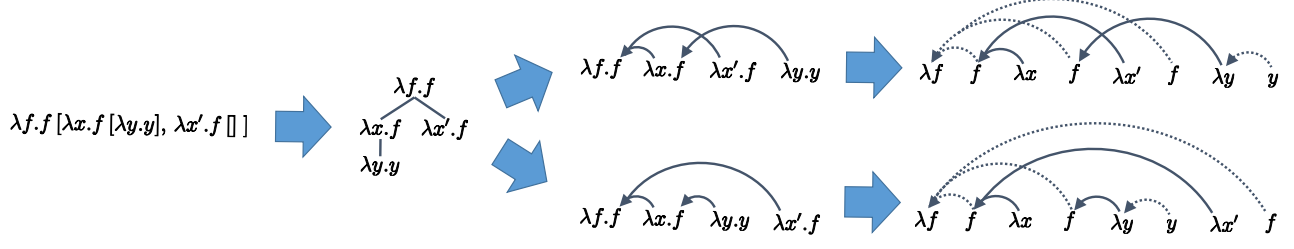


Figure 1. Idea of the correspondence

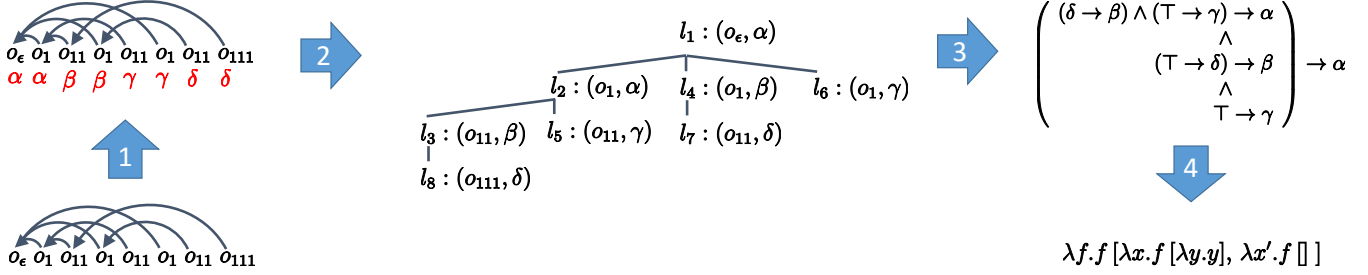


Figure 2. Illustration of the correspondence (where the simple type, or the arena, is $T = ((o_{111} \rightarrow o_{11}) \rightarrow o_1) \rightarrow o_\epsilon$)

logic [30]. He introduced the notion of *thick subtrees* (see Section 2.4) and observed correspondence between thick subtrees of an innocent strategy regarded as a tree of P-views and experiments of the corresponding proof-nets. The correspondence gives a functor from the category of polarised games and innocent strategies to the category of sets and relations. The proof of functoriality in [8] relies on a rich property of the game model, namely, correspondence between game semantics and syntax at the level of strategies/proof-nets. Our work can be seen as a refinement of his work, studying the correspondence in the same spirit but at the level of plays.

Another close result was given by Di Gianantonio and Lenisa [20], pointing out that game semantics can be seen as an intersection type assignment system. They first gave a reformulation of the HO/N game model in terms of *partitioned positions*, and then showed that a partitioned position can be seen as an intersection type. As they pointed out [20, Section 2.2], a partitioned position is almost identical to a *valuated thick subtree* in [8]. So Di Gianantonio and Lenisa's work and Boudes' work share the same idea.³

The preceding work can be placed in the series of work in the spirit of Timeless Games [5], which aims to relate a dynamic model such as the game model, and a static model such as the relational model. The idea is to forget the sequential structure of plays, and they showed that this time-forgetting map is a lax functor. This idea has subsequently been studied quite deeply, using relation-like static models such as coherence spaces [11] and hypercoherences [7, 33].

Among others, Melliès [35] showed that innocent strategies of asynchronous games are positional and thus can be seen as relations on positions. The present work is inspired by his work, and in fact uses the alternating homotopy relation \sim introduced in [35]. Unfortunately his approach is not directly applicable to the HO/N game model because of the exponential modality implicitly used in the function type $A \rightarrow B = !A \multimap B$. An approach based on *orbital games* [32] is mentioned in [35], but then composition is associative only up to the equivalence induced by the *group action*,

³We also proposed a closely related notion of *two-level arenas* [36] before [20] without noticing Boudes' work [8].

as in the AJM game model [4]. This is why we need to annotate colours to moves (see Figure 2).

As for the relationship between games and differential linear logic, Laird et al. [29] studied the relationship from the categorical perspective. They developed a way to construct a differential category [6] from a symmetric monoidal category, and then reconstructed a known category of games from a new category of *exhausting games*. As a consequence of this construction, one has a functor from games to relations. Compared to our work, this functor can be seen as the colourless version (i.e. the set of atomic types \mathcal{X} is singleton) of the functor in this paper.

The map from resource terms to relations has been well studied in the context of the relational interpretation of proof-nets, since a differential proof-net in the Taylor expansion of a proof-net is essentially an *injective experiment* of the original proof-net [18, 19]. Hence the map of the inverse direction of 4 in Fig. 2 is known.

2. Reviewing Category of Sets and Relations

Here we briefly discuss the category \mathbf{MRel} , which is the co-Kleisli category of the finite multiset comonad \mathcal{M}_{fin} on the category \mathbf{Rel} of sets and relations. This category plays an important rôle and all other categories in this paper can be seen as its subcategories. Since \mathbf{MRel} is a CCC, one can interpret a simple type as an object. For a simple type T , its interpretation $\llbracket T \rrbracket$ can be seen as the set of refinement non-idempotent intersection types for A and as two-level arenas for (the arena corresponding to) A .

Notation Given sets X_1 and X_2 , we write $X_1 + X_2$ for the tagged union, i.e. $X_1 + X_2 := \{(1, x_1) \mid x_1 \in X_1\} \cup \{(2, x_2) \mid x_2 \in X_2\}$. The function $\hat{\iota}_i : X_i \rightarrow X_1 + X_2$ is defined by $\hat{\iota}_i(x) := (i, x)$ for $i \in \{1, 2\}$. This connective is left-associative, i.e. $X + Y + Z = (X + Y) + Z$. By abuse of notation, we write $\iota_i : X_i \rightarrow X_1 + \dots + X_n$ for the embedding to the i th component, i.e. $\iota_1(x) := \hat{\iota}_1^{n-1}(x)$ and $\iota_i(x) = \hat{\iota}_1^{n-i}(\hat{\iota}_2(x))$ for $i > 1$.

2.1 Category \mathbf{MRel}

We write \mathbf{Rel} for the category of sets and relations. An object of \mathbf{Rel} is a set and a morphism from X to Y is a relation $R \subseteq X \times Y$.

The composite of $R_1 \subseteq X \times Y$ and $R_2 \subseteq Y \times Z$ is given by $R_2 \bullet_{\mathcal{T}} R_1 := \{ (x, z) \mid \exists y \in Y. (x, y) \in R_1 \text{ and } (y, z) \in R_2 \}$.

Given a set X , a *finite multiset over X* is a function $f : X \rightarrow \mathbf{N}$ to natural numbers such that $f(x) = 0$ for all but finitely many $x \in X$. Given finite multisets f and g on X , their *sum* $f \cdot g$ is defined by $(f \cdot g)(x) := f(x) + g(x)$. We write $[x_1, \dots, x_n]$ for the multiset $x \mapsto \#\{i \mid x_i = x\}$ (where $\#Y$ is the number of elements in Y). Then $[x_1, \dots, x_n] \cdot [y_1, \dots, y_k] = [x_1, \dots, x_n, y_1, \dots, y_k]$. In this notation, permutation of elements do not change the multiset, i.e. $[x_1, \dots, x_n] = [x_{\rho(1)}, \dots, x_{\rho(n)}]$ for every bijection $\rho : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$.

Let us write $\mathcal{M}_{\text{fin}}(X)$ for the set of all finite multisets on X . This is a functor on \mathbf{Rel} that acts on morphisms as $\mathcal{M}_{\text{fin}}(R) := \{ ([x_1, \dots, x_n], [y_1, \dots, y_n]) \mid \forall i \leq n. (x_i, y_i) \in R \}$. Furthermore \mathcal{M}_{fin} is a comonad with counit $\epsilon_x := \{ ([x], x) \mid x \in X \} : \mathcal{M}_{\text{fin}}(X) \rightarrow X$ and comultiplication $\delta_X := \{ (\theta_1 \cdot \dots \cdot \theta_n, [\theta_1, \dots, \theta_n]) \mid \forall i \leq n. \theta_i \in \mathcal{M}_{\text{fin}}(X) \} : \mathcal{M}_{\text{fin}}(X) \rightarrow \mathcal{M}_{\text{fin}}(\mathcal{M}_{\text{fin}}(X))$.

Then the coKleisli category \mathbf{MRel} of \mathcal{M}_{fin} is given as follows. An object of \mathbf{MRel} is a set and a morphism from X to Y is a subset of $\mathcal{M}_{\text{fin}}(X) \times Y$. Given $R \subseteq \mathcal{M}_{\text{fin}}(X) \times Y$, we define

$$R^! := \{ (\theta_1 \cdot \dots \cdot \theta_n, [y_1, \dots, y_n]) \mid \forall i \leq n. (\theta_i, y_i) \in R \}.$$

The composite of $R \subseteq \mathcal{M}_{\text{fin}}(X) \times Y$ and $S \subseteq \mathcal{M}_{\text{fin}}(Y) \times Z$ is defined as $S \bullet_{\mathcal{T}} R^!$.

\mathbf{MRel} is a CCC. The product of X and Y is given by the disjoint union $X + Y$ of sets and the exponential $X \Rightarrow Y$ is $\mathcal{M}_{\text{fin}}(X) \times Y$. The natural bijection $\Lambda_{X,Y,Z} : \mathbf{MRel}(X + Y, Z) \cong \mathbf{MRel}(X, Y \Rightarrow Z)$ is given by, for $R \subseteq \mathcal{M}_{\text{fin}}(X + Y) \times Z$,

$$\Lambda_{X,Y,Z}(R) := \{ ([x_1, \dots, x_n], ([y_1, \dots, y_k], z)) \mid ([\hat{l}_1(x_1), \dots, \hat{l}_1(x_n), \hat{l}_2(y_1), \dots, \hat{l}_2(y_k)], z) \in R \}.$$

2.2 Simple types and their interpretation

The set of *simple types* is given by the following grammar: $T ::= \circ \mid T \rightarrow T$, where \circ is the unique atomic type. Since \mathbf{MRel} is a CCC, a simple type can be interpreted as an object in \mathbf{MRel} if the interpretation of the atomic type \circ is determined. Let \mathcal{X} be a countably infinite set, fixed in this paper. Then $\llbracket T \rrbracket$ is inductively defined by: $\llbracket \circ \rrbracket := \mathcal{X}$ and $\llbracket T \rightarrow T' \rrbracket := \mathcal{M}_{\text{fin}}(\llbracket T \rrbracket) \times \llbracket T' \rrbracket$.

Let A be a (possibly empty) sequence (T_1, \dots, T_n) of simple types, which we consider as a product type $T_1 \times \dots \times T_n$. Its interpretation $\llbracket A \rrbracket$ in \mathbf{MRel} is defined as $\llbracket T_1 \rrbracket + \dots + \llbracket T_n \rrbracket$.

2.3 Elements of $\llbracket T \rrbracket$ as refinement intersection types

Given a simple type T , $\llbracket T \rrbracket$ can be seen as the set of all intersection types that refine T [17]. We use α and β for elements of \mathcal{X} . Then an element of $\llbracket \circ \rrbracket$ is an atomic type $\alpha \in \mathcal{X}$. Given a sequence τ_1, \dots, τ_n of elements in $\llbracket T \rrbracket$, we write $\tau_1 \wedge \dots \wedge \tau_n$ for the multiset $[\tau_1, \dots, \tau_n] \in \mathcal{M}_{\text{fin}}(\llbracket T \rrbracket)$. Then the intersection connective \wedge is associative and commutative but non-idempotent as in [17, 27]. An element of $\llbracket T \rightarrow T' \rrbracket = \mathcal{M}_{\text{fin}}(\llbracket T \rrbracket) \times \llbracket T' \rrbracket$ is a pair (θ, τ) , which we write as $\theta \multimap \tau$. Now $\llbracket T \rrbracket$ can be seen as the set of intersection types given by the grammar

$$\tau ::= \alpha \mid \theta \multimap \tau \quad \theta ::= \tau_1 \wedge \dots \wedge \tau_n \quad (n \geq 0)$$

that refines T (written as $\tau :: T$), where the *refinement relation* $\tau :: T$ and $\theta :: !T$ is defined by the following rules.

$$\frac{}{\alpha :: \circ} \quad \frac{\theta :: !S \quad \tau :: T}{(\theta \multimap \tau) :: (S \rightarrow T)} \quad \frac{\forall i \in \{1, \dots, n\}. \tau_i :: T}{\tau_1 \wedge \dots \wedge \tau_n :: !T}$$

In case $\tau :: T$, we say that τ is a refinement type of T ; similarly in case $\theta :: !T$, we say that θ is a refinement intersection of T . We

use \top to mean the empty intersection, that is, the empty bag $[\]$ of types.

Let $A = (T_1, \dots, T_n)$ be a sequence of simple types. A *refinement type binding of A* is of the form $\iota_i(\tau)$ with $\tau :: T_i$, i.e. $\iota_i(\tau) \in \llbracket A \rrbracket = \llbracket T_1 \rrbracket + \dots + \llbracket T_n \rrbracket$. We write $\iota_i(\tau) :: A$ if $\iota_i(\tau)$ is a refinement type binding of A . A *refinement type environment of A* , ranged over by Θ and Ξ , is a multiset of elements of the form $\iota_{i,j}(\tau_j)$, i.e. $\Theta \in \mathcal{M}_{\text{fin}}(\llbracket A \rrbracket)$. We write $\Theta :: A$ if Θ is a refinement type environment of A . They have type-based representation once one fixes variables x_i for each $1 \leq i \leq n$. Consider a simple type environment $\Gamma = (x_1 : T_1, \dots, x_n : T_n)$. Then a refinement type binding $\iota_i(\tau)$ can be written as $x_i : \tau$, and a refinement type environment $[\iota_{i_1}(\tau_1), \dots, \iota_{i_k}(\tau_k)]$ can be written as $x_{i_1} : \tau_1, \dots, x_{i_k} : \tau_k$. Note that a refinement type environment may have several type declaration for the same variable and it is not changed by a permutation of type bindings. The concatenation Θ, Ξ (or $\Theta \wedge \Xi$) of refinement type environments is given by concatenation as sequences or by multiset sum.

Let A and B be sequences of simple types. A *refinement type of (A, B)* is a pair $(\Theta, \iota_i(\tau))$ such that $\Theta :: A$ and $\iota_i(\tau) :: B$. A *refinement intersection of (A, B)* is a pair (Θ, Ξ) such that $\Theta :: A$ and $\Xi :: B$. We write $\mathbf{Ty}(A, B)$ (resp. $\mathbf{Ty}^1(A, B)$) for the set of all refinement types (resp. refinement intersections) of (A, B) .

A morphism in \mathbf{MRel} from $\llbracket A \rrbracket$ to $\llbracket B \rrbracket$ is a subset of $\mathbf{Ty}(A, B)$. Given a morphism $R \subseteq \mathcal{M}_{\text{fin}}(\llbracket A \rrbracket) \times \llbracket B \rrbracket$ from $\llbracket A \rrbracket$ to $\llbracket B \rrbracket$ in \mathbf{MRel} , $R^!$ can be seen as a subset of $\mathbf{Ty}^1(A, B)$.

2.4 Elements of $\llbracket A \rrbracket$ as trees or arenas

Sequences A of simple types as well as elements of $\llbracket A \rrbracket$ can be seen as forests. This view point, which can be found in [8] and [36], will play an important rôle in this paper.

A directed graph $A = (N_A, E_A \subseteq N_A \times N_A)$ is a *forest* if for every node n , there exists a unique path $n_0 E_A n_1 E_A \dots E_A n_k = n$ for some $k \geq 0$, where n_0 is a node that has no incoming edge, called a *root*. A *tree* is a forest with a unique root. A forest is *finite* if its node set is finite. A node $n \in N_A$ of a forest is *positive* if the length of the path from the root is even and it is *negative* otherwise.

A *forest homomorphism* from A to B is a function $f : N_A \rightarrow N_B$ on nodes that preserves edges (i.e. $(n, n') \in E_A$ implies $(f(n), f(n')) \in E_B$) and roots (i.e. if $n \in N_A$ is a root of A , then $f(n)$ is a root of B). A *forest isomorphism* is a forest homomorphism which is a bijection on nodes.

Let $\mathbf{N} = \{1, 2, \dots\}$ be the set of natural numbers and \mathbf{N}^+ be the set of (non-empty) finite sequences of natural numbers. We write ε for the empty sequence (which is not a member of \mathbf{N}^+). A subset $Z \subseteq \mathbf{N}^+$ is an *ordered forest* if

- $\xi \cdot i \in Z$ and $\xi \neq \varepsilon$ implies $\xi \in Z$, and
- $\xi \cdot i \in Z$ and $j < i$ implies $\xi \cdot j \in Z$,

An ordered forest Z defines a forest (Z, E_Z) , where $E_Z := \{(\xi, \xi \cdot i) \mid \xi \cdot i \in Z, \xi \neq \varepsilon\}$. An *ordered tree* is an ordered forest of which the corresponding forest is a tree. We shall use A, B, C as metavariables for ordered forests and S, T for ordered trees.

Let T_1, \dots, T_n be a sequence of ordered trees with $n \geq 0$. We write (T_1, \dots, T_n) for the ordered forest defined by $\{i \cdot \xi \mid 1 \leq \xi \in T_i\}$. An ordered forest can be canonically described in this form.

There is a bijective correspondence between simple types and finite ordered trees: \circ corresponds to the tree with a unique node and $T_1 \rightarrow \dots \rightarrow T_n \rightarrow \circ$ to the tree with n children of which the i th child is the tree associated to T_i . We shall identify the simple types with finite ordered trees. Similarly a finite sequence of simple types, which can be seen as a simple type environment, corresponds to a finite ordered forest.

In this paper, we shall consider only finite trees and forests (which are ordered or unordered). We call them simply trees and forests.

We introduce another presentation of elements in $\llbracket A \rrbracket$ that we call the two-level representation [36]. This representation is essentially the same as *thick subtrees*, as studied by Boudes [8]. The work [36] is based on game semantics, in which a forest representing a simple type is called an *arena*. Henceforth we use arena as a synonym of (ordered or unordered) forest.

Given an ordered forest $A = (N_A, E_A)$, a *high-level arena* of A is a tuple $\Theta = (N_\Theta, E_\Theta, h_\Theta, c_\Theta)$, where

- $|\Theta| = (N_\Theta, E_\Theta)$ is an unordered forest,
- $h_\Theta : |\Theta| \rightarrow A$ is a forest homomorphism, and
- $c_\Theta : N_\Theta \rightarrow \mathcal{X}$ is a colouring function.

A high-level arena is *finite* if the underlying forest is finite. A *two-level arena* is a pair $\Theta :: A$ of an arena A and a high-level arena Θ of A . It is *pointed* if $|\Theta|$ is a tree. We use metavariable θ for high-level arenas of a tree T , and τ for pointed high-level arenas of a tree. (See Lemma 1 for justifications of these notations.)

A *homomorphism of high-level arenas* from $\Theta :: A$ to $\Xi :: A$ is a function $f : N_\Theta \rightarrow N_\Xi$ such that

- f is a forest homomorphism from $|\Theta|$ to $|\Xi|$,
- $h_\Theta(n) = h_\Xi(f(n))$ for every $n \in N_\Theta$, and
- $c_\Theta(n) = c_\Xi(f(n))$ for every $n \in N_\Theta$.

It is an isomorphism of high-level arenas if f is a bijection. We identify isomorphic high-level arenas in the sequel.

Lemma 1. *Let T be an ordered tree (or a simple type) and A be an ordered forest (or a sequence of simple types). Then (i) pointed high-level arenas of T bijectively correspond to refinement types of T , (ii) high-level arenas of T bijectively correspond to refinement intersections of T , and (iii) high-level arenas of A bijectively correspond to sequences of refinement intersections of A .*

We shall use the two representations interchangeably.

2.5 Renaming preorder and equivalence

The set $\mathbf{Ty}(A, B)$ and $\mathbf{Ty}^1(A, B)$ can be regarded as a preorder by what we call the *renaming preorder*. Let $\chi : \mathcal{X} \rightarrow \mathcal{X}$ be a function on \mathcal{X} . Its action on elements in $\llbracket A \rrbracket$ is defined inductively by: $\chi(x)$ is already defined for $x \in \llbracket \mathbf{o} \rrbracket$, $\chi([a_1, \dots, a_n]) := [\chi(a_1), \dots, \chi(a_n)]$ for $[a_1, \dots, a_n] \in \mathcal{M}_{\text{fin}}(\llbracket T \rrbracket)$, $\chi((a, b)) := (\chi(a), \chi(b))$ for $(a, b) \in \llbracket T \rightarrow T' \rrbracket$, and $\chi(\iota_i(a)) := \iota_i(\chi(a))$ for $\iota_i(a) \in \llbracket (T_1, \dots, T_n) \rrbracket$. In the two-level presentation, given a high-level arena Θ of A , we define $\chi(\Theta) := (N_\Theta, E_\Theta, h_\Theta, \chi \circ c_\Theta)$. Given $(\Theta, \iota_i(\tau))$, $(\Theta', \iota_{i'}(\tau')) \in \mathbf{Ty}(A, B)$, we write $(\Theta, \iota_i(\tau)) \approx (\Theta', \iota_{i'}(\tau'))$ if there exists χ such that $\Theta = \chi(\Theta')$ and $\iota_i(\tau) = \chi(\iota_{i'}(\tau'))$ (hence $i = i'$). We define a preorder on $\mathbf{Ty}^1(A, B)$ similarly. The equivalence induced by this preorder is written as \approx . Given a set $f \subseteq \mathbf{Ty}(A, B)$ (or $f \subseteq \mathbf{Ty}^1(A, B)$), we write $\Downarrow f$ for its downward closure with respect to \approx .

3. Resource Terms, Plays and Refinement Non-idempotent Intersection Types

This section introduces a bijection between plays and resource terms. More precisely, given ordered forests $A = (T_1, \dots, T_n)$ and $B = (S_1, \dots, S_k)$, we shall establish bijections between the following sets:

- The set $\mathcal{T}_P(A, B)/\approx$ of *playful* refinement intersection types, which is a subset of refinement types, modulo \approx .
- The set $\{(j, M) \mid M \in \text{NF}, v_1 : T_1, \dots, v_n : T_n \vdash M : S_j\}$ of (tagged) terms of the resource calculus in normal form.
- The set $\mathbf{P}(A, B)/\sim$ of well-opened plays modulo alternating homotopy [35].

We first give two functions ϕ and ψ from terms and plays, respectively, to refinement intersection types (which are not necessarily playful). These functions are injective but not surjective. To capture the precise images of ϕ and ψ , we introduce the notion of *playful types* and show that the images of ϕ and ψ are the set of playful types. Then $\varpi := \phi^{-1} \circ \psi$ is a bijection between plays and resource terms.

3.1 Plays in game semantics

This subsection defines the notion of plays in the HO/N game model [26], as well as the notions of alternating homotopy [35] and of morphisms between plays [31, 37].

Remark 2. The plays in this paper are not necessarily O-visible. This is crucial. (This is the same as in [8, 20, 37].)

Definition 3. Let A be an ordered forest. In the context of game semantics, A is called an *arena* and a node $m \in N_A$ is called a *move*, ranged over by the metavariable m . A move is an *O-move* if the unique path from a root to the node is of even length. Otherwise it is called a *P-move*. We write \mathcal{M}_A for the set of moves, that is, N_A . The set of O-moves (resp. P-moves) is written as \mathcal{M}_A^O (resp. \mathcal{M}_A^P). The *enabling relation* $(\vdash_A) \subseteq (N_A + \{\star\}) \times N_A$ is defined by the following rules:

- If $(m, m') \in E_A$, then $m \vdash_A m'$.
- If m' is a root, i.e. $\neg \exists m. (m, m') \in E_A$, then $\star \vdash_A m'$.

A root is called an *initial move*.

Definition 4. An *arena pair* is a pair (A, B) of arenas. The move-set $\mathcal{M}_{A,B}$ of the pair is the disjoint union of moves, i.e. $\mathcal{M}_{A,B} := \mathcal{M}_A + \mathcal{M}_B$. We define $\mathcal{M}_{A,B}^O := \mathcal{M}_A^O + \mathcal{M}_B^O$ and $\mathcal{M}_{A,B}^P := \mathcal{M}_A^P + \mathcal{M}_B^P$. We write $m \vdash_{A,B} m'$ if either $m, m' \in \mathcal{M}_A$ and $m \vdash_A m'$ or $m, m' \in \mathcal{M}_B$ and $m \vdash_B m'$. Similarly $\star \vdash_{A,B} m$ if $\star \vdash_A m$ or $\star \vdash_B m$. Note that an arena pair (A, B) is the same as the disjoint union $(N_A + N_B, E_A + E_B)$ of the forests except for the polarity of moves: the polarity of moves in A is inverted.

Definition 5. Let (A, B) be an arena pair. A *justified sequence* over (A, B) is a sequence of moves of (A, B) equipped with justification pointers. Formally it is a triple $s = (\#s, s, \rho_s)$, where

- $\#s = \{1, 2, \dots, n\}$ for some $n \geq 0$,
 - $s : \#s \rightarrow \mathcal{M}_{A,B}$ is a move function and
 - $\rho_s : \#s \rightarrow (\#s \cup \{0\})$ is a justification pointer
- subject to the following conditions for every $i \in \#s$:
- $\rho_s(i) < i$.
 - $\rho_s(i) = 0$ implies $\star \vdash_{A,B} s(i)$.
 - $\rho_s(i) \neq 0$ implies $s(\rho_s(i)) \vdash_{A,B} s(i)$.

We define $\#_0 s := \{0\} \cup \#s$. An *alternating justified sequence* is a justified sequence s such that $s(i) \in \mathcal{M}_{A,B}^O$ if and only if i is odd. A justified sequence s is *well-opened* if it has exactly one initial O-move (namely $s(1)$).

Definition 6 (Morphism between plays). Let s and t be alternating justified sequences of (A, B) . A morphism $f : s \rightarrow t$ between alternating justified sequences is an injective function $f : \#_0 s \rightarrow \#_0 t$ subject to the following conditions: for every $i \in \#s$,

$$f(0) = 0 \quad s(i) = t(f(i)) \quad \text{and} \quad f(\rho_s(i)) = \rho_t(f(i)),$$

and

$$f(i+1) = f(i) + 1 \quad \text{for every odd index } i \in \#s.$$

The image of f is defined by $\text{img}(f) := \{f(i) \mid i \in \#s\} \subseteq \#t$.

Definition 7 (P-view/P-visibility). An even-length, alternating justified sequence s is a *P-view* if it is not empty and $\rho_s(i) = i - 1$ for every odd $i \in \#s$. In other words, if every O-move in s points to the previous move (and s has a unique initial O-move, namely $s(1)$). A morphism $f : p \rightarrow s$ from a P-view p to an alternating

justified sequence s is called a P-view of s . An alternating justified sequence s is P -visible if for every $i \in \#s$, there is a P-view $f : p \rightarrow s$ such that $i \in \text{img}(f)$.

Definition 8 (Play). A play over arena pair (A, B) is an even-length, P-visible, alternating justified sequence over (A, B) . We write $\mathbf{P}(A, B)$ for the set of well-opened plays of (A, B) and $\mathbf{P}^1(A, B)$ for the set of plays of (A, B) .

Definition 9 (Alternating homotopy [35]). Alternating homotopy $\sim_{A, B}$ is the least equivalence relation on $\mathbf{P}^1(A, B)$ generated by

$$s_1 \cdot m_1 \cdot n_1 \cdot m_2 \cdot n_2 \cdot s_2 \sim_{A, B} s_1 \cdot m_2 \cdot n_2 \cdot m_1 \cdot n_1 \cdot s_2,$$

where m_1 and m_2 are O-moves, n_1 and n_2 are P-moves and m_2 does not point to n_1 . Hence $s \sim_{A, B} s'$ means that s is a “rearrangement” of s' . This relation can be defined using morphism between plays (see [37] for the proof): $s \sim_{A, B} s'$ if and only if there exists an isomorphism $f : s \rightarrow s'$ (i.e. $\text{img}(f) = \#s'$).

3.2 From plays to types

We define a map from plays to types by introducing an intersection type assignment for plays, following the idea in [8, 20, 36].

Definition 10 (Colouring, types of a play). Let $s \in \mathbf{P}^1(A, B)$ be a play. A colouring of s is a function $c : \#s \rightarrow \mathcal{X}$ such that $c(i) = c(i+1)$ for every odd index $i \in \#s$. We write (s, c) for the high-level arena $(\#s, E_s, s, c)$ of (A, B) , where $E_s := \{(i, j) \in \#s \times \#s \mid i = \rho_s(j)\}$. The set of intersection types for a play $s \in \mathbf{P}^1(A, B)$ is given by:

$$\llbracket s \rrbracket := \{(s, c) \in \mathbf{Ty}^1(A, B) \mid c: \text{colouring of } s\}.$$

It is easy to see that, if s is well-opened (i.e. $s \in \mathbf{P}(A, B)$), then $\llbracket s \rrbracket \subseteq \mathbf{Ty}(A, B)$.

Example 11. Let s be the play in Figure 2, which belongs to $\mathbf{P}^1(A, B)$ where A is the empty arena and $B = (T)$ with $T = ((o_{111} \rightarrow o_{11}) \rightarrow o_1) \rightarrow o_e$. Consider the colouring c on s defined by $c(1) = c(2) = \alpha$, $c(3) = c(4) = \beta$, $c(5) = c(6) = \gamma$, and $c(7) = c(8) = \delta$. Then the associated high-level arena (s, c) is the one in Figure 2.

There is a canonical type from which other types can be obtained: it is the type corresponding to a colouring $c : \#s \rightarrow \mathcal{X}$ that is injective on odd indexes.

Lemma 12. Let $s \in \mathbf{P}^1(A, B)$ be a play and $c_0 : \#s \rightarrow \mathcal{X}$ be a colouring of s . Suppose for every distinct odd indexes $i, j \in \#s$, one has $c_0(i) \neq c_0(j)$. Then $\llbracket s \rrbracket = \downarrow\{(s, c_0)\}$.

Lemma 13. If $s_1 \sim s_2$, then $\llbracket s_1 \rrbracket = \llbracket s_2 \rrbracket$.

The function $\psi_{A, B} : \mathbf{P}(A, B)/\sim \rightarrow \mathbf{Ty}(A, B)/\approx$ maps a play to its canonical type, i.e.,

$$\psi_{A, B}(s) := \llbracket (s, c_0) \rrbracket,$$

where $c_0 : \#s \rightarrow \mathcal{X}$ is the canonical colouring. By Lemmas 12 and 13, it is a well-defined function. The function $\psi_{A, B}^1 : \mathbf{P}^1(A, B)/\sim \rightarrow \mathbf{Ty}^1(A, B)/\approx$ is defined similarly.

3.3 Terms of the resource calculus

Here we give the standard definitions (see [21, 23, 24]).

Terms and bags The syntax of terms and bags is given by the following grammar:

$$M, N := x \mid \lambda x. M \mid M P \quad P, Q := [M_1, \dots, M_n] \quad (n \geq 0).$$

M is called a *term* and P is a finite multiset of terms, called a *bag*. Since a bag is a multiset, a bag is identified with a permutation of its elements. Given bags $P = [M_1, \dots, M_n]$ and $Q = [N_1, \dots, N_k]$,

we write $P \cdot Q$ for $[M_1, \dots, M_n, N_1, \dots, N_k]$. Henceforth we allow tacit renaming of bound variables. So α -equivalent terms (resp. bags) are identified.

Simple type assignment A simple type environment, $\Gamma = x_1 : T_1, \dots, x_n : T_n$, is a finite sequence of pairs of variables and simple types such that $x_i \neq x_j$ if $i \neq j$. We write $x : T \in \Gamma$ to mean $x = x_i$ and $T = T_i$ for some $1 \leq i \leq n$. The typing rules are listed below.

$$\frac{x : T \in \Gamma}{\Gamma \vdash x : T} \quad \frac{\Gamma, x : S \vdash M : T}{\Gamma \vdash \lambda x. M : S \rightarrow T}$$

$$\frac{\Gamma \vdash M : S \rightarrow T \quad \Gamma \vdash P : S}{\Gamma \vdash M P : T} \quad \frac{\forall i \in \{1, \dots, n\}. \Gamma \vdash M_i : T}{\Gamma \vdash [M_1, \dots, M_n] : T}$$

In what follows, we consider simply-typed terms, although the simple type is often omitted.

Normal terms and bags A typed term $\Gamma \vdash M : T$ is β -normal if it does not have a subterm of the form $(\lambda x. M) P$. It is η -long if it is typable by the type system in which the variable and the application rules are replaced with the following rules:

$$\frac{x : T_1 \rightarrow \dots \rightarrow T_n \rightarrow \circ \in \Gamma \quad \Gamma \vdash M : T_1 \rightarrow \dots \rightarrow T_n \rightarrow \circ}{\Gamma \vdash P_1 : T_1} \quad \frac{\Gamma \vdash M : T_1 \rightarrow \dots \rightarrow T_n \rightarrow \circ}{\Gamma \vdash P_1 : T_1}$$

$$\frac{\Gamma \vdash P_n : T_n}{\Gamma \vdash x P_1 \dots P_n : \circ} \quad \frac{\Gamma \vdash P_n : T_n}{\Gamma \vdash M P_1 \dots P_n : \circ}$$

In words, every application and variable in a η -long term is fully applied. A β -normal η -long term is called a *normal term*. The notion of *normal bags* are defined similarly. We write $M \in \mathbf{NF}$ and $P \in \mathbf{NF}$ if M and P are normal.

Let $A = (S_1, \dots, S_n)$ and $B = (T_1, \dots, T_k)$ be ordered forests, which can be seen as sequences of simple types. We shall define sets $\mathbf{Tm}(A, B)$ and $\mathbf{Tm}^1(A, B)$ of normal terms and bags, respectively. Assume an infinite sequence v_1, v_2, \dots of distinct variables, fixed in this paper.

An element of $\mathbf{Tm}^1(A, B)$ is a tuple of bags (P_1, \dots, P_k) such that $v_1 : S_1, \dots, v_n : S_n \vdash P_i : T_i$ and $P_i \in \mathbf{NF}$ for every $1 \leq i \leq k$. This is reminiscent of a morphism in the syntactic category of, for example, the simply-typed lambda calculus, which is a sequence of terms.

An element of $\mathbf{Tm}(A, B)$ is a pair (i, M) of an index i and a term M such that $v_1 : S_1, \dots, v_n : S_n \vdash M : T_i$ and $M \in \mathbf{NF}$. We write $\iota_i(M)$ for the pair and $A \vdash \iota_i(M) : B$ to mean the above simple-type judgement. Then there is an obvious bijection between $\mathbf{Tm}^1(A, B)$ and the set of finite multisets of $\mathbf{Tm}(A, B)$.

3.4 From terms to types

We define the families of functions $\mathbf{Tm}(A, B) \rightarrow \mathcal{P}(\mathbf{Ty}(A, B))$ and $\mathbf{Tm}^1(A, B) \rightarrow \mathcal{P}(\mathbf{Ty}^1(A, B))$, based on an intersection type assignment system. The typing rules are listed below:

$$\frac{}{x : \tau \vdash x : \tau} \quad \frac{\Delta_1 \vdash M : \theta \multimap \tau \quad \Delta_2 \vdash P : \theta}{\Delta_1, \Delta_2 \vdash M P : \tau}$$

$$\frac{\Delta, x : \tau_1, x : \tau_2, \dots, x : \tau_n \vdash M : \tau \quad x \notin \Delta}{\Delta \vdash \lambda x. M : (\tau_1 \wedge \dots \wedge \tau_n) \multimap \tau}$$

$$\frac{\forall i \in \{1, \dots, n\}. \Delta_i \vdash M_i : \tau_i}{\Delta_1, \dots, \Delta_n \vdash [M_1, \dots, M_n] : \tau_1 \wedge \dots \wedge \tau_n}$$

Here Δ_1, Δ_2 is concatenation. The operation $\bigwedge_{1 \leq i \leq n} \Delta_i$ is defined by the same way. Here in the rule for variables, we implicitly assume that τ respects the simple type of x_i , i.e. if x_i is a variable of simple type T , it must be the case that $\tau :: T$. Then every provable judgement respects the simple types of the terms and bags. For

a sequence of intersections $\Theta = (\theta_1, \dots, \theta_n)$, we write $\Theta \vdash M : \tau$ (resp. $\Theta \vdash P : \delta$) to mean $v_1 : \theta_1, \dots, v_n : \theta_n \vdash M : \tau$ (resp. $v_1 : \theta_1, \dots, v_n : \theta_n \vdash P : \delta$).

Given a term $\iota_i(M) \in \mathbf{Tm}(A, B)$, we define

$$\llbracket \iota_i(M) \rrbracket := \{ (\Theta, \iota_i(\tau)) \in \mathbf{Ty}(A, B) \mid \Theta \vdash M : \tau \}.$$

Similarly given a bag $(P_1, \dots, P_k) \in \mathbf{Tm}^1(A, B)$, we define

$$\llbracket (P_1, \dots, P_k) \rrbracket := \{ (\bigwedge_{1 \leq i \leq n} \Theta_i, \bigwedge_{1 \leq i \leq n} \iota_i(\theta_i)) \mid \forall i \in \{1, \dots, n\}. \Theta_i \vdash P_i : \theta_i \}$$

where $\iota_i(\tau_1 \wedge \dots \wedge \tau_k)$ means $\iota_i(\tau_1) \wedge \dots \wedge \iota_i(\tau_k)$.

Let M be a normal term. A judgement $\Theta \vdash M : \tau$ is *canonical* if each atomic type has at most one positive occurrence. A *canonical judgement for a bag* is defined similarly.

Example 14. Let M be the normal term in Figure 2, i.e. $M = \lambda f.f [\lambda x.f [\lambda y.y], \lambda x'.f []]$. Then $\llbracket M \rrbracket$ contains (\top, τ_1) and (\top, τ_2) where $\tau_1 = (((\delta \rightarrow \beta) \wedge (\top \rightarrow \delta)) \rightarrow \alpha) \wedge ((\top \rightarrow \gamma) \rightarrow \beta) \wedge (\top \rightarrow \gamma) \rightarrow \alpha$, and $\tau_2 = (((\beta \rightarrow \beta) \wedge (\top \rightarrow \beta)) \rightarrow \alpha) \wedge ((\top \rightarrow \alpha) \rightarrow \beta) \wedge (\top \rightarrow \alpha) \rightarrow \alpha$. The former is canonical though the latter is not.

Lemma 15. *Every normal term $\iota_i(M) \in \mathbf{Tm}(A, B)$ has a derivable canonical judgement $\Theta \vdash M : \tau$. Furthermore, for every canonical judgement $\Theta \vdash M : \tau$, we have $\llbracket \iota_i(M) \rrbracket = \downarrow \{ (\Theta, \iota_i(\tau)) \}$. A similar statement holds for bags.*

Remark 16. For Lemma 15, the assumption that M is normal (since $\mathbf{Tm}(A, B)$ is a set of normal terms) is essential. For example, consider $x : o \rightarrow o \vdash x : o \rightarrow o$, which is not normal. Then $x : \top \rightarrow \alpha \vdash x : \top \rightarrow \alpha$ and $x : \beta \rightarrow \alpha \vdash x : \beta \rightarrow \alpha$ are canonical but not \approx -equivalent.

The function $\phi_{A,B} : \mathbf{Tm}(A, B) \rightarrow \mathbf{Ty}(A, B)/\approx$ is defined as the mapping from a term to the canonical typing, i.e. for a normal term $\iota_i(M) \in \mathbf{Tm}(A, B)$ with the canonical typing $\Theta \vdash M : \tau$, we define $\phi_{A,B}(\iota_i(M)) := [(\Theta, \iota_i(\tau))]$. Then it is a well-defined function since the canonical typing is unique up to \approx by Lemma 15. The function $\phi_{A,B}^1 : \mathbf{Tm}^1(A, B) \rightarrow \mathbf{Ty}^1(A, B)/\approx$ is defined similarly.

3.5 Playful types and bijectivity

This section characterises the images of ϕ and ψ . We employ ideas from game semantics and call a type in the image a *playful type*.

3.5.1 Playful types

Let (A, B) be a pair of ordered forests, fixed in this subsection. A refinement intersection $(\Theta, \Xi) \in \mathbf{Ty}^1(A, B)$ is *involutive* just if

$$\{ (m, m') \in \mathcal{M}_{\Theta, \Xi}^O \times \mathcal{M}_{\Theta, \Xi}^P \mid c_{\Theta, \Xi}(m) = c_{\Theta, \Xi}(m') \}$$

is a bijection from $\mathcal{M}_{\Theta, \Xi}^O$ to $\mathcal{M}_{\Theta, \Xi}^P$, which induces a fixpoint free involution on $\mathcal{M}_{\Theta, \Xi}$. This condition simply says that every colour $\alpha \in \mathcal{X}$ appearing in (Θ, Ξ) has exactly two occurrences, one positive and the other negative.

Given an involutive type $(\Theta, \Xi) \in \mathbf{Ty}^1(A, B)$, let $R_{\Theta, \Xi}$ be the relation on colours $\mathcal{X}_{\Theta, \Xi} = \{c_{\Theta, \Xi}(a) \mid a \in N_{\Theta, \Xi}\} \subseteq \mathcal{X}$ that appear in (Θ, Ξ) defined by:

$$R_{\Theta, \Xi} = \{ (c_{\Theta, \Xi}(a), c_{\Theta, \Xi}(b)) \mid a \in \mathcal{M}_{\Theta, \Xi}^P, (a, b) \in E_{\Theta, \Xi} \}.$$

The pair (Θ, Ξ) is said to be *P-visible* if

- $(c_{\Theta, \Xi}(a), c_{\Theta, \Xi}(b)) \in R_{\Theta, \Xi}^*$ for every $(a, b) \in E_{\Theta, \Xi}$ (including the case that $a \in \mathcal{M}_{\Theta, \Xi}^O$, and

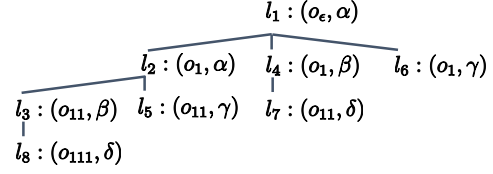


Figure 3. High-level arena in Figure 2

- for every root b of Θ , there exists a root a of Ξ such that $(c_{\Theta, \Xi}(a), c_{\Theta, \Xi}(b)) \in R_{\Theta, \Xi}^*$,

where $R_{\Theta, \Xi}^*$ is the reflexive and transitive closure of $R_{\Theta, \Xi}$.

A type is *playful* if it is involutive and P-visible. We write $\mathbf{PTy}(A, B) \subseteq \mathbf{Ty}(A, B)$ and $\mathbf{PTy}^1(A, B) \subseteq \mathbf{Ty}^1(A, B)$ for the respective restrictions to playful types.

Example 17. Let τ be the type corresponding to the high-level arena in Figure 3. Consider $(\top, \iota_1(\tau)) \in \mathbf{Ty}^1(A, B)$, where A is the empty arena and $B = (T)$ with $T = (((o_{111} \rightarrow o_{11}) \rightarrow o_1) \rightarrow o_e)$. This is involutive and $R_{\top, \iota_1(\tau)}$ is $\{ (\alpha, \beta), (\alpha, \gamma), (\beta, \delta) \}$. To see that this type is playful, one needs to check if, for each edge, the colours of the parent and the child are related by $R_{\top, \iota_1(\tau)}^*$. For example, focusing on the edge between l_3 and l_8 , we should check if $(\beta, \delta) \in R_{\top, \iota_1(\tau)}^*$. It is not difficult to see that this type is playful.

Remark 18. An involutive type induces a proof structure and P-visibility is equivalent to the correctness criterion of multiplicative linear logic. Consider the involutive type $((\top \multimap \gamma) \multimap \beta) \wedge ((\gamma \multimap \beta) \multimap \alpha) \multimap \alpha$ that refines $((o \rightarrow o) \rightarrow o) \rightarrow o$. One can associate it to the formula of multiplicative linear logic $(\gamma \otimes \beta^\perp) \wp ((\gamma^\perp \wp \beta) \otimes \alpha^\perp) \wp \alpha$, by interpreting \wedge as the tensor product and $A \multimap B$ as $A^\perp \wp B$. This formula induces a unique proof structure. The proof structure satisfies the correctness criterion [15, 25] if and only if the type satisfies P-visibility (and thus playfulness).

We need some auxiliary lemmas used in the proof of bijectivity. A consequence of the first result is that $R_{\Theta, \Xi}^*$ is a partial order.

Lemma 19. *Let $(\Theta, \Xi) \in \mathbf{Ty}^1(A, B)$ be a playful type. Then $(\mathcal{X}_{\Theta, \Xi}, R_{\Theta, \Xi})$ is a forest with roots $\{c_{\Theta, \Xi}(a) \mid a : \text{root of } \Xi\}$.*

Corollary 20. *Let (Θ, Ξ) be a playful type and b be a root node of Θ . Then a root a of Ξ s.t. $(c_{\Theta, \Xi}(a), c_{\Theta, \Xi}(b)) \in R_{\Theta, \Xi}^*$ is unique.*

Lemma 21. *Let $(\Theta, \Xi) \in \mathbf{Ty}^1(A, B)$ and assume (Θ, Ξ) is playful. Let $b, b' \in \mathcal{M}_{\Theta, \Xi}^O$ be distinct O-nodes such that either (1) both b and b' are roots of Ξ , or (2) b and b' has the same parent. Then $c_{\Theta, \Xi}(b)$ and $c_{\Theta, \Xi}(b')$ are incomparable by $R_{\Theta, \Xi}^*$.*

3.5.2 Bijectivity of $\psi_{A,B}$

Lemma 22. *The canonical colouring induces a playful type.*

Lemma 23. *For every playful type $(\Theta, \Xi) \in \mathbf{PTy}^1(A, B)$, there is a play $s \in \mathbf{P}^1(A, B)$ unique up to \sim such that $(\Theta, \Xi) \in \llbracket s \rrbracket$. Furthermore such a play is well-opened if Ξ is a tree.*

Proof. (Sketch) Let $(\Theta, \Xi) \in \mathbf{Ty}^1(A, B)$ be a playful type. Then $(\mathcal{X}_{\Theta, \Xi}, R_{\Theta, \Xi})$ is a tree (Lemma 19). Let us sort the elements in $\mathcal{X}_{\Theta, \Xi}$ as $\alpha_1, \alpha_2, \dots, \alpha_n$ in such a way that $(\alpha_i, \alpha_j) \in R_{\Theta, \Xi}^*$ implies $i \leq j$.

We first define a play $\hat{s} = (\#\hat{s}, \hat{s}, \rho_{\hat{s}})$ of $(|\Theta|, |\Xi|)$ as follows. The maximum index is the number of moves in $(|\Theta|, |\Xi|)$. So $\#\hat{s} = \{1, 2, \dots, 2n\}$. For $1 \leq i \leq n$, let us write O_i (resp. P_i) for the unique O-move (resp. P-move) in $(|\Theta|, |\Xi|)$ with colour α_i . We define the sequence \hat{s} as $O_1 P_1 O_2 P_2 \dots O_n P_n$. Then each move in

$(|\Theta|, |\Xi|)$ has exactly one occurrence in s . Thus the tree structure of $(|\Theta|, |\Xi|)$ induces the unique pointing structure, which determines $\rho_{\hat{s}}$.

The play s of (A, B) is given by mapping moves of $(|\Theta|, |\Xi|)$ to those of (A, B) by the tree homomorphism h equipped with a high-level arena. That means, s of (A, B) is defined by $s = (\#\hat{s}, s, \rho_{\hat{s}})$, where $s(i) = h(\hat{s}(i))$. \square

Example 24. Recall the playful type $(\top, \iota_1(\tau))$ in Example 17, for which $R_{\top, \iota_1(\tau)} = \{(\alpha, \beta), (\alpha, \gamma), (\beta, \delta)\}$. Let us compute the play that corresponds to this type. Consider a linear order on $\{\alpha, \beta, \gamma, \delta\}$ that respects $R_{\top, \iota_1(\tau)}$, e.g. $\alpha < \beta < \gamma < \delta$. Then $\hat{s}(1)$ is the O-node with colour α , that is, l_1 . Similarly $\hat{s}(2)$ is the P-node with colour α , that is, l_2 , and $\hat{s}(3)$ is the O-node with colour β , which is l_3 . We have $\hat{s}(i) = l_i$ for all i . The tree homomorphism h equipped with the two-level arena maps a node to its move name, e.g. $h(l_1) = o_\epsilon$ and $h(l_5) = o_{11}$. Hence $s = h \circ \hat{s}$ is given by $o_\epsilon o_{10} o_{11} o_{10} o_{11} o_{10} o_{11} o_{11}$, which is the same as the play in Figure 2. It is not difficult to see that the pointers are the same as well.

The next theorem is a corollary of Lemmas 12, 13, 22 and 23.

Theorem 25. $\psi_{A,B} : \mathbf{P}(A, B)/\sim \rightarrow \mathbf{PTy}(A, B)/\approx$ and $\psi_{A,B}^1 : \mathbf{P}^1(A, B)/\sim \rightarrow \mathbf{PTy}^1(A, B)/\approx$ are bijections.

3.5.3 Bijectivity of $\phi_{A,B}$

Lemma 26. Let M be a normal term with $v_1 : S_1, \dots, v_n : S_n \vdash \iota_i(M) : B$. Assume that $\Theta \vdash M : \tau$, where $\Theta :: A$ and $\iota_i(\tau) :: B$. If this judgement is canonical, then $(\Theta, \tau) :: (A, T)$ is playful. A similar statement holds for bags $\Theta \vdash P : \theta$.

Lemma 27. For every playful type $(\Theta, \iota_i(\tau)) \in \mathbf{PTy}(A, B)$, there is a unique term $\iota_i(M) \in \mathbf{Tm}(A, B)$ such that $(\Theta, \iota_i(\tau)) \in \llbracket \iota_i(M) \rrbracket$. A similar statement holds for $(\Theta, \Xi) \in \mathbf{PTy}^1(A, B)$.

We give an illustrative example instead of a sketch of the proof.

Example 28. Let $\tau = (\delta \rightarrow \beta) \wedge (\top \rightarrow \gamma) \rightarrow \alpha$. Then $\tau' = \tau \wedge ((\top \rightarrow \delta) \rightarrow \beta) \wedge (\top \rightarrow \gamma) \rightarrow \alpha$ is the type in Figure 2, which is playful. Let us compute the unique normal term M such that $\vdash M : \tau'$. Since M is normal, we have $M = \lambda f. M_1$ for some f and M_1 . Hence the derivation must be of the form

$$\frac{f : \tau, f : ((\top \rightarrow \delta) \rightarrow \beta), f : \top \rightarrow \gamma \vdash M_1 : \alpha}{\vdash \lambda f. M_1 : \tau'}$$

Since M_1 is normal, M_1 must be of the form $z P_1 \dots P_n$. Now we have an applicable assumption, namely $f : \tau$, which is indeed the unique applicable assumption. (In general, if (Θ, τ) is involutive, there exists at most one assumption applicable to derive $\Theta \vdash N : \tau$ for some N .) Hence the subderivation must be

$$\frac{f : \tau \vdash f : \tau \quad \frac{\Theta_1 \vdash M_{11} : \delta \rightarrow \beta \quad \Theta_2 \vdash M_{12} : \top \rightarrow \gamma}{\Theta \vdash [M_{11}, M_{12}] : (\delta \rightarrow \beta) \wedge (\top \rightarrow \gamma)}}{f : \tau, f : ((\top \rightarrow \delta) \rightarrow \beta), f : \top \rightarrow \gamma \vdash f [M_{11}, M_{12}] : \alpha}$$

where $\Theta = f : ((\top \rightarrow \delta) \rightarrow \beta), f : \top \rightarrow \gamma$ and $\Theta_1 \wedge \Theta_2 = \Theta$. The type environments Θ_1 and Θ_2 can be uniquely determined as follows. We have $(f : (\top \rightarrow \delta) \rightarrow \beta) \in \Theta_1$ since β appears in the first judgement, and $(f : \top \rightarrow \gamma) \in \Theta_2$ since γ appears in the second judgement. (In general, the partition of the type environment in the conclusion is uniquely determined by this argument because of Corollary 20.) By iterating this process, we can compute the inhabitant, which is unique by construction.

The bijectivity is a consequence of Lemmas 26, 27 and 15.

Theorem 29. $\phi_{A,B} : \mathbf{Tm}(A, B) \rightarrow \mathbf{PTy}(A, B)/\approx$ and $\phi_{A,B}^1 : \mathbf{Tm}^1(A, B) \rightarrow \mathbf{PTy}^1(A, B)/\approx$ are bijections.

Corollary 30. For every (A, B) , the map $\varpi_{A,B}^1 := (\phi_{A,B}^1)^{-1} \circ \psi_{A,B}^1 : (\mathbf{P}^1(A, B)/\sim) \rightarrow \mathbf{Tm}^1(A, B)$ is a bijection, which can be restricted to a bijection $(\mathbf{P}(A, B)/\sim) \rightarrow \mathbf{Tm}(A, B)$.

4. Compositions

We have seen the connection between plays and resource terms. This section focuses on the respective notions of composition. The composition of plays is defined by using interaction sequences, via ‘‘parallel composition plus hiding’’, whereas the composite of resource terms (or bags) in normal form is given by substitution followed by normalisation. We shall prove that these quite different notions of compositions coincide, via the composition of intersection types (i.e. composition in the relational model).

Following the composition in \mathbf{MRel} , in which we compose $f \subseteq \mathbf{Ty}^1(A, B) = \mathcal{M}_{\text{fin}}(\llbracket A \rrbracket) \times \mathcal{M}_{\text{fin}}(\llbracket B \rrbracket)$ and $g \subseteq \mathbf{Ty}(B, C) = \mathcal{M}_{\text{fin}}(\llbracket B \rrbracket) \times \llbracket C \rrbracket$, this section studies the composition of $s \in \mathbf{P}^1(A, B)$ and $s' \in \mathbf{P}(B, C)$ and that of $(P_1, \dots, P_n) \in \mathbf{Tm}^1(A, B)$ and $\iota_i(M) \in \mathbf{Tm}(B, C)$.

4.1 Composition of plays

To define the composite of plays, we need to define the notion of interaction sequences, which are plays on a triple of arenas.

Definition 31 (Interaction sequence). Let (A, B, C) be a triple of arenas. We define $\mathcal{M}_{A,B,C}$ as the disjoint union $\mathcal{M}_A + \mathcal{M}_B + \mathcal{M}_C$ of moves. The enabling relation $\vdash_{A,B,C}$ on $\mathcal{M}_{A,B,C}$ is defined by $a \vdash_{A,B,C} b$ (where $a \in \mathcal{M}_{A,B,C} + \{\star\}$ and $b \in \mathcal{M}_{A,B,C}$) if and only if $a \vdash_X b$ for some $X \in \{A, B, C\}$. A justified sequence on (A, B, C) is defined in the same way as in Definition 5. The projection $u|_{A,B}$ onto (A, B) component of a justified sequence u of (A, B, C) is defined as the subsequence obtained by removing all C -moves. The other projections $u|_{B,C}$ and $u|_{A,C}$ are defined similarly. An interaction sequence on (A, B, C) is a justified sequence on (A, B, C) such that $u|_{A,B}, u|_{B,C}$ and $u|_{A,C}$ are all plays on (A, B) , (B, C) and (A, C) , respectively. We write $\mathbf{Intr}(A, B, C)$ for the set of all interaction sequences on (A, B, C) .

Definition 32 (Composition of Plays). Let $s \in \mathbf{P}^1(A, B)$ and $s' \in \mathbf{P}(B, C)$ be plays. Their composition $s' \bullet_G s$ is a subset of plays in $\mathbf{P}(A, C)$ defined by

$$s' \bullet_G s := \{u|_{A,C} \mid \exists u \in \mathbf{Intr}(A, B, C). u|_{A,B} \sim s \text{ and } u|_{B,C} \sim s'\}.$$

Remark 33. Unlike the standard composition in game semantics (as in [26]), the projection of the interaction sequence u onto the (A, B) component is not necessarily the same as s but equivalent with respect to the alternating homotopy (Definition 9). So it might be better to regard \bullet_G as a composition of \sim -equivalence classes of plays, resulting in a set of \sim -equivalence class.

Recall the type-based interpretation $\llbracket s \rrbracket$ of a play s (Definition 10). We extend this to a set of plays J by $\llbracket J \rrbracket = \bigcup_{s \in J} \llbracket s \rrbracket$. The following result can be proved by an adaptation of the technique in [36].

Lemma 34. $\llbracket s' \bullet_G s \rrbracket = \llbracket s' \rrbracket \bullet_T \llbracket s \rrbracket$ for every plays $s \in \mathbf{P}(A, B)$ and $s' \in \mathbf{P}(B, C)$.

4.2 Composition of resource terms

Composition of resource terms is given by substitution followed by normalisation.

Definition 35 (Differential substitution). Given a resource term M and a variable x , let us write $\text{deg}_x(M)$ for the number of free occurrences of x in M . The differential substitution $\partial_x M \cdot P$ of

a variable x to a bag P in a resource term M is defined by

$$\partial_x M \cdot [N_1, \dots, N_k] = \begin{cases} \{ M \langle N_{f(1)}/x_1, \dots, N_{f(k)}/x_k \rangle \mid f \in \mathfrak{G}_k \} \\ \quad \text{(if } \deg_x(M) = k) \\ \emptyset \quad \quad \quad \text{(otherwise),} \end{cases}$$

where \mathfrak{G}_k is the group of permutations over $\{1, \dots, k\}$ and x_1, \dots, x_k is an enumeration of the free occurrences of x in M (the choice of the enumeration does not matter), and $\langle \dots \rangle$ is the simultaneous and capture-avoiding substitution.

Definition 36 (Reduction). We define a relation \longrightarrow_1 between terms and sets of terms by the following base rule

$$(\lambda x.M)P \longrightarrow_1 \partial_x M \cdot P$$

and the congruence rules such as

$$MP \longrightarrow_1 \{M'P \mid M' \in \mathbb{M}'\} \quad \text{if } M \longrightarrow_1 M',$$

where \mathbb{M}' is a (possibly infinite) set of resource terms. The reduction relation \longrightarrow on sets of terms is defined by

$$\mathbb{M}_0 \cup \{M\} \longrightarrow \mathbb{M}_0 \cup \mathbb{M}' \quad \text{if } M \longrightarrow_1 M'.$$

Lemma 37 ([23]). *The relation \longrightarrow is confluent and weakly normalising for finite sets of resource terms.*

Hence, given a resource term M , it has a unique (β -)normal form $\mathbf{nf}(M)$ such that $\{M\} \longrightarrow^* \mathbf{nf}(M)$. If M is η -long, then every term in the normal form $\mathbf{nf}(M)$ is also η -long. We extend this function to (possibly infinite) sets of resource terms by $\mathbf{nf}(\mathbb{M}) := \bigcup_{M \in \mathbb{M}} \mathbf{nf}(M)$.

Now we define the composite of $(P_1, \dots, P_k) \in \mathbf{Tm}(A, B)$ and $\iota_i(M) \in \mathbf{Tm}(B, C)$:

$$\begin{aligned} \iota_i(M) \bullet_C (P_1, \dots, P_k) &:= \\ &\{ \iota_i(N) \mid N \in \mathbf{nf}((\lambda v_1 \dots \lambda v_k.M)P_1 \dots P_k) \}. \end{aligned}$$

Given a set $\mathbb{M} \subseteq \mathbf{Tm}(A, C)$, we define $\llbracket \mathbb{M} \rrbracket := \bigcup \{ \iota_i(M) \mid \iota_i(M) \in \mathbb{M} \}$. Roughly speaking, this means that \mathbb{M} has type τ if and only if M has type τ , for some $M \in \mathbb{M}$.

The following lemma is a well-known result, which claims that \mathbf{MRel} is a model of the resource calculus (or intersection types of a resource term is invariant under reduction). See [10] for the proof.

Lemma 38. *Let $(P_1, \dots, P_n) \in \mathbf{Tm}^1(A, B)$ and $\iota_i(M) \in \mathbf{Tm}^1(B, C)$. Then $\llbracket \iota_i(M) \bullet_C (P_1, \dots, P_n) \rrbracket = \llbracket \iota_i(M) \rrbracket \bullet_{\mathcal{T}} \llbracket (P_1, \dots, P_n) \rrbracket$.*

Preservation of composition by the bijection ϖ is a consequence of Lemmas 34 and 38 and the existence of canonical colouring and typing (Lemmas 12 and 15).

Theorem 39. *$t \in s \bullet_G s'$ iff $\varpi_{A,C}(t) \in \varpi_{B,C}(s) \bullet_C \varpi_{A,B}(s')$.*

5. Applications

5.1 Categories of games and of playful relations

We give an isomorphism between the category of games and a subCCC of \mathbf{MRel} , by which we conclude that the category of games is a CCC.

The “category” \mathcal{G} of games is defined by the following data: (i) An object is an ordered forest A . (ii) A morphism from A to B is a strategy, which is a subset $\mathbf{P}(A, B)$ of well-opened plays closed under \sim . (iii) Composition defined below. Given a strategy $\sigma \subseteq \mathbf{P}(A, B)$, we define $\sigma^1 \subseteq \mathbf{P}^1(A, B)$ by

$$\sigma^1 := \{ t \mid \exists s_1, \dots, s_n \in \sigma. s_1 \dots s_n \sim t \}.$$

The composite of strategies $\sigma_1 \subseteq \mathbf{P}(A, B)$ and $\sigma_2 \subseteq \mathbf{P}(B, C)$ is given by $\sigma_2 \circ \sigma_1 := \bigcup \{ s_2 \bullet_G s_1 \mid s_1 \in \sigma_1^1, s_2 \in \sigma_2 \}$. This

definition of composition coincides with the standard one, since strategies are \sim -closed.

Lemma 40. *Let $\sigma \subseteq \mathbf{P}(A, B)$ be a strategy. Then $\llbracket \sigma^1 \rrbracket = \llbracket \sigma \rrbracket^1$.*

Lemma 41. *Let $\sigma_1 \subseteq \mathbf{P}(A, B)$ and $\sigma_2 \subseteq \mathbf{P}(B, C)$ be strategies. Then $\sigma_2 \circ \sigma_1 = \{ u \upharpoonright_{A,C} \mid u \in \mathbf{Intr}(A, B, C), u \upharpoonright_{A,B} \in \sigma_1^1, u \upharpoonright_{B,C} \in \sigma_2 \}$.*

In fact, \mathcal{G} is not yet a category, since we do not specify identities and do not show associativity. We shall prove that \mathcal{G} is indeed a category by using the bijection ψ .

Let \mathcal{T} be the full subcategory of \mathbf{MRel} whose objects are sets $\llbracket A \rrbracket := \{ \iota_i(\tau) \mid \iota_i(\tau) :: A \}$ for ordered forests A . The category \mathcal{T} is not equivalent to the category of games \mathcal{G} since \mathcal{T} has more morphisms. Let us consider the “subcategory” \mathcal{T}_P of \mathcal{T} in which morphisms are generated by a set of playful types, i.e. a morphism in \mathcal{T}_P is of the form $\Downarrow f$ for some $f \in \mathbf{PTy}(A, B)$.

The next result is a consequence of Lemmas 12, 22 and 34.

Theorem 42. *\mathcal{T}_P is isomorphic to \mathcal{G} , i.e., there is a family of composition-preserving bijections $\mathcal{G}(A, B) \cong \mathcal{T}_P(A, B)$.*

By Theorem 42, \mathcal{T}_P is closed under composition. Since the identities $\text{id}_{\llbracket A \rrbracket}$ of \mathbf{MRel} are in \mathcal{T}_P , it is indeed a subcategory of \mathbf{MRel} . It is easy to verify that \mathcal{T}_P is a CCC, and thus so is \mathcal{G} .

Theorem 43. *\mathcal{G} is indeed a category. Moreover it is a CCC.*

5.2 Taylor expansion as game-semantic interpretation

We prove that the Taylor expansion (followed by normalisation) of a simply-typed nondeterministic λ -term in η -long form coincides with its game-semantic interpretation via ϖ . We write $L_1 + L_2$ for nondeterministic branching and \perp for divergence. For a term L of the simply-typed nondeterministic λ -calculus, we define L^* by:

$$\begin{aligned} x^* &:= \{ x \} & (\lambda x.L)^* &:= \{ \lambda x.M \mid M \in L^* \} \\ (L L')^* &:= \{ M [N_1, \dots, N_k] \mid \\ &\quad M \in L^*, k \geq 0, \forall i \leq k. N_i \in (L')^* \} \\ (L_1 + L_2)^* &:= L_1^* \cup L_2^* & \perp^* &:= \emptyset \end{aligned}$$

We call L^* the Taylor expansion of L . If $v_1 : T_1, \dots, v_n : T_n \vdash L : S$ and L is η -long, then $\mathbf{nf}(L^*) \subseteq \mathbf{Tm}((T_1, \dots, T_n), S)$.

Recall that $\varpi_{A,B}^1$ is the function from $\mathbf{P}^1(A, B)/\sim$ to $\mathbf{Tm}^1(A, B)$, which can be restricted to $\varpi_{A,B} : (\mathbf{P}(A, B)/\sim) \rightarrow \mathbf{Tm}(A, B)$. Given a subset $\sigma \subseteq \mathbf{P}(A, B)$, we write $\varpi(\sigma) := \{ \varpi([s]_{\sim}) \mid s \in \sigma \}$, where $[s]_{\sim}$ is the \sim -equivalence class that s belongs to. We write the interpretation of L in the CCC \mathcal{G} as $\llbracket L \rrbracket_{\mathcal{G}}$, where $\llbracket L_1 + L_2 \rrbracket_{\mathcal{G}} := \llbracket L_1 \rrbracket_{\mathcal{G}} \cup \llbracket L_2 \rrbracket_{\mathcal{G}}$ and $\llbracket \perp \rrbracket_{\mathcal{G}} := \emptyset$ (cf. [37]).

Theorem 44. *For every η -long term $v_1 : T_1, \dots, v_n : T_n \vdash L : S$ of the simply-type λ -calculus, one has $\varpi(\llbracket L \rrbracket_{\mathcal{G}}) = \mathbf{nf}(L^*)$.*

5.3 Characterisation of the image

Let us again consider the nondeterministic simply-typed λ -calculus. The image of its game-semantic interpretation is given by [31, 37]. By translating the characterisation in [31, 37] via ϖ , we obtain the characterisation of the image of the Taylor expansion.

Definition 45 (Lower strategies [31]). Given $s, s' \in \mathbf{P}(A, B)$, we write $s \triangleleft s'$ if there exists a Levy-morphism from s to s' , which is a map in Definition 6 but not necessarily injective. A strategy $\sigma \subseteq \mathbf{P}(A, B)$ is lower if $s' \in \sigma$ and $s \triangleleft s'$ implies $s \in \sigma$.

Theorem 46 ([37]). *Let $\sigma \subseteq \mathbf{P}(A, B)$. Then $\sigma = \llbracket L \rrbracket_{\mathcal{G}}$ for some L if and only if σ is lower and has finitely many P -views.⁴*

⁴The statement is a bit different from [37] since a strategy is a collection of well-opened plays in this paper.

It is easy to see that $s \sim s' \triangleleft s'' \sim s'''$ implies $s \triangleleft s'''$. Hence \triangleleft is well-defined on $\mathbf{P}(A, B)/\sim$. If we define $\iota_i(M) \triangleleft \iota_{i'}(M')$ to mean $\varpi(\iota_i(M)) \triangleleft \varpi(\iota_{i'}(M'))$, an analogy of Theorem 46 immediately holds by Theorem 44. Now what we need is a purely syntactic characterisation of \triangleleft on resource terms.

Lemma 47. *Let $\iota_i(M), \iota_{i'}(M') \in \mathbf{Tm}(A, B)$. Then $\varpi(\iota_i(M)) \triangleleft \varpi(\iota_{i'}(M'))$ if and only if $i = i'$ and $M \triangleleft M'$ is derivable by the following rules:*

$$\frac{}{x \triangleleft x} \qquad \frac{M \triangleleft M'}{\lambda x.M \triangleleft \lambda x.M'}$$

$$\frac{M \triangleleft M' \quad P \triangleleft P'}{MP \triangleleft M'P'} \qquad \frac{\forall i. \exists j. M_i \triangleleft M'_j}{[M_1, \dots, M_n] \triangleleft [M'_1, \dots, M'_k]}$$

Theorem 48. *Let $\mathbb{M} \subseteq \mathbf{Tm}(A, B)$.⁵ Then $\mathbb{M} = \mathbf{nf}(L^*)$ for some L if and only if the height⁶ of resource terms in \mathbb{M} is bounded and \mathbb{M} is lower (i.e. $M \triangleleft M' \in \mathbb{M}$ implies $M \in \mathbb{M}$).*

The order \triangleleft induced from the above game-semantic argument resembles to a preorder given by Boudes et al. [9], which is used to characterise the image of the Taylor expansion of the λ -calculus (without nondeterminism). It is worth discussing a relationship in more detail, but it is left for future work.

Acknowledgement This work is partially supported by EPSRC EP/M023974/1, JSPS Kakenhi Grant Number 15H05706 and JSPS Kakenhi Grant Number 16K160004.

References

- [1] S. Abramsky and R. Jagadeesan. Games and full completeness for multiplicative linear logic. *The Journal of Symbolic Logic*, 59(2):543, 1994.
- [2] S. Abramsky and G. McCusker. Full abstraction for Idealized Algol with passive expressions. *Theoretical Computer Science*, 227(1-2): 3–42, 1999.
- [3] S. Abramsky and P.-A. Mellies. Concurrent games and full completeness. In *Proceedings of the 14th Symposium on Logic in Computer Science (LICS 1999)*, 431–442, 1999.
- [4] S. Abramsky, R. Jagadeesan, and P. Malacaria. Full abstraction for PCF. *Information and Computation*, 163(2):409–470, 2000.
- [5] P. Baillot, V. Danos, T. Ehrhard, and L. Regnier. Timeless games. In *Proceedings of the 11th International Workshop on Computer Science Logic (CSL 97)*, 56–77, 1997.
- [6] R. F. Blute, J. R. B. Cockett, and R. A. G. Seely. Differential categories. *Mathematical Structures in Computer Science*, 16(6):1049–1083, 2006.
- [7] P. Boudes. Projecting games on hypercoherences. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP 04)*, 257–268, 2004.
- [8] P. Boudes. Thick subtrees, games and experiments. In *Proceedings of the 9th International Conference on Typed Lambda Calculi and Applications (TLCA 09)*, 65–79, 2009.
- [9] P. Boudes, F. He, and M. Pagani. A characterization of the Taylor expansion of lambda-terms. In *Proceedings of the 22nd EACSL Annual Conference on Computer Science Logic (CSL 13)*, 101–115, 2013.
- [10] A. Bucciarelli, A. Carraro, T. Ehrhard, and G. Manzonetto. Full abstraction for the resource lambda calculus with tests, through Taylor expansion. *LMCS*, 8:1–44, Oct. 2012.
- [11] A. C. Calderon and G. McCusker. Understanding game semantics through coherence spaces. *Electronic Notes in Theoretical Computer Science*, 265:231–244, 2010.
- [12] P.-L. Curien. Abstract Böhm trees. *Mathematical Structures in Computer Science*, 8:559–591, 1998.
- [13] P.-L. Curien. Notes on game semantics. 2006.
- [14] V. Danos and R. S. Harmer. Probabilistic game semantics. *ACM Transactions on Computational Logic*, 3(3):359–382, 2002.
- [15] V. Danos and L. Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28(3):181–203, 1989.
- [16] V. Danos, H. Herbelin, and L. Regnier. Game semantics & Abstract Machine. In *Proceeding of the 11th Annual IEEE Symposium on Logic in Computer Science (LICS 96)*, 309–405, 1996.
- [17] D. de Carvalho. Execution time of lambda-terms via denotational semantics and intersection types. arXiv, preprint page 36, 2009.
- [18] D. de Carvalho. The relational model is injective for multiplicative exponential linear logic. arXiv, preprint, 163(9):1210–1236, 2015.
- [19] D. de Carvalho and L. Tortora de Falco. The relational model is injective for multiplicative exponential linear logic (without weakenings). *Annals of Pure and Applied Logic*, 163:1210–1236, 2012.
- [20] P. Di Gianantonio and M. Lenisa. Innocent game semantics via intersection type assignment systems. In *Proceedings of the 22nd EACSL Annual Conference on Computer Science Logic (CSL 13)*, 231–247, 2013.
- [21] T. Ehrhard. A finiteness structure on resource terms. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science*, pages 402–410, 2010.
- [22] T. Ehrhard and O. Laurent. Interpreting a finitary pi-calculus in differential interaction nets. In *Proceedings of the 18th International Conference on Concurrency Theory (CONCUR 07)*, 333–348, 2007.
- [23] T. Ehrhard and L. Regnier. Uniformity and the Taylor expansion of ordinary lambda-term. *Theoretical Computer Science*, 403(2-3):347–372, 2008.
- [24] T. Ehrhard, A. Bucciarelli, A. Carraro, and G. Manzonetto. Full abstraction for the resource lambda calculus with tests, through Taylor expansion. *Logical Methods in Computer Science*, 8(4):1–43, 2012.
- [25] J.-Y. Girard. Linear Logic. *Theoretical Computer Science*, 50:1–102, 1987.
- [26] J. M. E. Hyland and C.-H. L. Ong. On full abstraction for PCF: I, II, and III. *Information and Computation*, 163(2):285–408, 2000.
- [27] A. J. Kfoury. A linearization of the Lambda-calculus and consequences. *Journal of Logic and Computation*, 10(3):411–436, 2000.
- [28] J. Laird. A game semantics of the asynchronous π -calculus. *CONCUR*, pages 51–65, 2005.
- [29] J. Laird, G. Manzonetto, and G. McCusker. Constructing differential categories and deconstructing categories of games. *Information and Computation*, 222:247–264, 2013.
- [30] O. Laurent. Syntax vs. semantics: A polarized approach. *Theoretical Computer Science*, 343(1-2):177–206, 2005.
- [31] P. Levy. Morphisms between plays. GaLoP lecture Slides, 2013.
- [32] P.-A. Mellies. Asynchronous games 1: Uniformity by group invariance. 2003.
- [33] P.-A. Mellies. Sequential algorithms and strongly stable functions. *Theoretical Computer Science*, 343(1-2):237–281, 2005.
- [34] P.-A. Mellies. Asynchronous games 4: A fully complete model of propositional linear logic. In *Proceedings of the 20th Annual IEEE Symposium on Logic in Computer Science (LICS 05)*, 386–395, 2005.
- [35] P.-A. Mellies. Asynchronous games 2: The true concurrency of innocence. *Theoretical Computer Science*, 358(2-3):200–228, 2006.
- [36] C.-H. L. Ong and T. Tsukada. Two-level game semantics, intersection types, and recursion schemes. In *Proceedings of the 39th International Colloquium on Automata, Languages and Programming (ICALP 12)*, 325–336, 2012.
- [37] T. Tsukada and C.-H. L. Ong. Nondeterminism in game semantics via sheaves. In *Proceedings of the 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 15)*, 220–231, 2015.

⁵Hence the set of free variables of terms in \mathbb{M} is finite.

⁶The height of a resource term is that of its tree representation as in Fig. 1, e.g. $\text{height}([M_1, \dots, M_n]) = \max\{\text{height}(M_i) \mid 1 \leq i \leq n\}$.