Order algebras: a quantitative model of interaction

Emmanuel Beffara

Order algebras: a quantitative model of interaction

Emmanuel Beffara

Institut de Mathématiques de Luminy
UMR6206, Université Aix-Marseille II & CNRS

July 7, 2011

Abstract. A quantitative model of concurrent interaction is introduced. The basic objects are linear combinations of partial order relations, acted upon by a group of permutations that represents potential non-determinism in synchronisation. This algebraic structure is shown to provide faithful interpretations of finitary process algebras, for an extension of the standard notion of testing semantics, leading to a model that is both denotational (in the sense that the internal workings of processes are ignored) and non-interleaving. Constructions on algebras and their subspaces enjoy a good structure that make them (nearly) a model of differential linear logic, showing that the underlying approach to the representation of non-determinism as linear combinations is the same.

1 Introduction

The theory of concurrency has developed several very different models for interactive processes, focusing on different aspects of computation. Among those, process calculi are an appealing framework, because the formal language approach is well suited to modular reasoning, allowing to study sophisticated systems by means of abstract programming primitives for which powerful theoretical tools can be developed. They are also the setting of choice for extending the vast body of results of proof theory to less sequential settings. However, the vast majority of the semantic studies on process calculi like the π-calculus have focused on the so-called interleaving operational semantics, which is the basic definition of the dynamic of a process: the interaction of a program with its environment is reduced to possible sequences of transitions, thus considering that parallel composition of program components is merely an abstraction that represents all possible ways of combining several sequential processes into one. In Hoare's seminal work on Communicating Sequential Processes [20], this is even an explicit design choice.

There is clearly something unsatisfactory in this state of things. Although sophisticated theories have been established for interleaving semantics, most of which are based on various forms of bisimulation, they fundamentally forget the crucial (and obvious) fact that concurrent processes are intended to model situations where some events may occur independently, and event explicitly in parallel. This fact is well known, and the search for non interleaving semantics for process calculi is an active field of research, with fruitful interaction with proof theory and denotational semantics. Recently, the old idea of Winskel's interpretation of CCS in event structures [35, 36] has been revisited by Crafa, Varacca and Yoshida to provide an actually non-interleaving operational semantics for the π-calculus, using extensions of event structures [11]. Event structures are also one of the starting points of extensions of game semantics to non-sequential frameworks, for instance in asynchronous games [25] and concurrent extensions of ludics [18]. In a neighbouring line of research, the recent differential extension of linear logic is known to be expressive enough to represent the dynamics of the π-calculus [17, 15]. However the implications of this fact in the search for denotational semantics of the π-calculus are still unclear, in particular the quantitative contents of differential linear logic lacks a proper status in concurrency.

This paper presents a new semantic framework that addresses this question, following previous work by the author [5] on the search for algebraically pleasant denotational semantics of process calculi. The first step was to introduce in the π-calculus an additive structure (a formal sum with zero) that represents pure non-determinism, and this technique proved efficient enough to provide a readiness trace semantics [29] with a complete axiomatization of equivalence for finite terms. The second step presented here further extends the space of processes with arbitrary linear combinations, giving a meaning to these combinations in terms of quantitative testing. This introduction of scalar coefficients was not possible in the interleaving case, because of the combinatorial explosion that arose even when simply composing independent traces; moving to a non-interleaving setting through a quotient by homotopy of executions is the solution to this problem. Growing the space of processes to get more algebraic structure is also motivated by the idea that better structured semantics gives cleaner mathematical foundations for the object of study, in the hope that the obtained theory will be reusable for different purposes and that it will benefit from existing mathematical tools.

Informal description An order algebra is defined on an *arena*, which represents the set of all observable events that may occur in the execution of a process. Basic interaction scenarii, named *plays*, are partial order relations over finite subsets of the arena. We then postulate two principles:

- Linear combinations are used to represent non-determinism, which, although not the defining feature, is an unavoidable effect in concurrent interaction. Coefficients form the quantitative part of the model, the first thing they represent is how many times a given play may occur in a given situation. They can also represent more subtle things, like under which conditions a given play is relevant. This allows for the representation of features such as probabilistic choice, in which case coefficients will be random variables. In general, coefficients are taken in an arbitrary semiring with some additional properties. This use of linear combinations is a novelty of the differential λ-calculus and subsequent work [16], although a decomposition of processes as formal linear combinations was first proposed by Boreale and Gadducci [7], albeit without the quantitative aspect we develop here.

- The fact that some events may be indistinguishable by the environment of a process, typically different inputs (or outputs) on the same channel, is represented by a group action over the arena. Each element of the group acts as a permutation that represents a possible way of rearranging the events. A comparable approach was used in particular in AJM game semantics [2, 4] to represent the interchangeability of copies in the exponentials of linear logic.

Some words are borrowed from game semantics, since our objects have similarities with games, but this is not a “game” semantics, at most a degenerate one. In particular, there is no real notion of player and opponent interacting, since there is no polarity that could distinguish them or distinguish inputs and outputs. The term “strategy” does not really apply either since there is no notion of choosing the next move in a given situation. Under these circumstances, calling anything a “game” is kind of far fetched.

Outline Section 2 defines order algebras from these ideas. Arenas, plays and linear combinations of plays (simply called vectors) are defined, with the two basic operations on vectors: *synchronisation*, which extends the merging of orders to take permutations into account, and *outcome*, which is a scalar that acts as the “result” of a process. Two vectors are equivalent if they are indistinguishable by synchronisation and outcome, and the order algebra is the quotient of the vectors by this equivalence.

Section 3 describes constructs involving order algebras and their subspaces. Cartesian and tensor products are described in terms of interaction, and the symmetric algebra is constructed in the framework. This algebra is of particular interest because it represents the basic source of non-determinism in interaction, namely the fact that any number of interchangeable actions may occur at a given synchronisation point.

Section 4 shows how order algebras can be used to provide fully abstract models of process calculi, with the example of the πI-calculus. The crucial ingredient is a quantitative extension of the standard notion of testing, from which the present work stems. Standard forms of testing are obtained as particular choices of the semiring of scalars.

Future work Order algebras as defined and studied in the present work are very finitary in nature, because vectors are finite linear combinations of finite plays. This setting already has an interesting structure, as this paper illustrates, but it is unable to represent any kind of potentially infinitary behaviour. This includes identity functions over types that are not finite dimensional, and as a consequence we do not get a model of differential linear logic. Handling infinity is the natural next step, and for this we need to add topology to the structure, in order to get a sensible notion of convergence. Order algebras will then appear not only as the quotient of combinations of plays by equivalence, but as the separated and completed space generated by plays. In this line of thought, the dual space should play an important role, in order to define duality in the logical sense.

Another direction is to exploit the fact that the semiring of scalars is a parameter of the construction. In particular, going from a semiring $\mathbb{S}$ to the semiring of $\mathbb{S}$-valued random variables over a given probabilistic space properly extends the model to a probabilistic one. Similarly, using complex numbers and unitary transformations could provide a way to represent quantum computation in the same framework. Developing these ideas correctly is a line of research by itself, as the question of denotational models for these aspects of computation is known to be a difficult matter.

Related work Part of the construction of order algebras is concerned with modelling of features like name binding or creation of fresh names. The topic of proper formal handling of binders in syntax is a vast topic known as *nominal techniques* (see for instance Gabbay's survey [19]), and it has been applied in particular to construct operational semantics for process calculi in a generic way [28, 10]. We feel that our approach is orthogonal: arenas present a flattened version of the name structure, in which remains no notion of name creation or binding (or only indirectly); permutations are used only to relate different *occurrences* of names. Moreover, local names, by essence, are absent from order algebras, since our intent is to build a denotational model, in which internal behaviour is forgotten.

Our work aims in particular at constructing models of interaction that are not interleaving, a featured sometimes referred to as "true concurrency". This objective, of course, is not new, and the reference model in this respect is that of event structures. A relationship between our framework and event structures can be formulated: using the simplest semiring of coefficients, namely $\{0,1\}$ with $1+1=1$ (thus losing any "quantitative" content), linear combinations of plays are simply finite sets of plays. The set of plays interpreting a given process turns out to be exactly the set of configurations of the event structure interpreting this process, forgetting any internal events. We do not develop this correspondence in the present paper, as it is of limited interest in the current state of development of order algebras, however it will certainly be of great interest in the development of the theory, notably when applying it to modelling probabilistic processes, for which event structure semantics has been developed [1, 34]. Besides, the use of symmetry in event structures [37, 33] has been recently identified as a crucial feature; we defer to future work the comparison with our approach based on group actions.

The shift from sets of configurations to formal linear combinations in the interpretation of processes has a notable precedent in Boreale and Gadducci's interpretation of CSP processes as formal power series [7, 8], building on Rutten's work relating coinduction and formal power series [31]. Boreale and Gadducci's work differs from the present paper in two respects. Firstly, their interpretation of the semiring of coefficient is of a different nature: sum and product are seen as internal and external choice respectively, while we interpret them as internal choice and parallel composition without interaction. Secondly, their technical development uses only idempotent semirings (where $x+x=x$ for all x), which does not handle quantitative features, and leads inevitably to interleaving semantics (as proved in our setting by Theorem 41 and remarks in Section 4.4). Nevertheless, Rutten's approach to coinduction, and the idea of coinductive definitions by behavioural differential equations is certainly relevant to our work and is a promising source of inspiration for the extension of the present setting to infinitary behaviours.

2 Order algebras

2.1 Arenas and plays

An order algebra is defined on an *arena*, which represents a fixed set of potential events. The arena is equipped with a permutation group that represents the non-determinism that arises when synchronising events, as described below. Then a play is a partial order relation over a finite subset of the arena.

1 **Definition.** An *arena* X is a pair $(|X|, \mathrm{G}^X)$ where $|X|$ is a countable set (the *web* of X) and G^X is a subgroup of the group $\mathfrak{S}(|X|)$ of permutations of $|X|$. If G^X is trivial, then X is called *static* and it is identified with its web.

The points in the web are called *events*, rather than moves, since there is no actual notion of players interacting. Permutations represent the fact that there may be several different ways for two processes to synchronise. In process calculus language, permutations can be seen as relating different occurrences of the same action label.

2 *Example.* When modelling a simple process algebra like CSP [20] over an alphabet A (with no value passing), we can use a web like $A \times \mathbb{N}$, where $\mathbb{N}$ is the set of natural numbers; (a, i) is interpreted as the i-th copy of a (any other infinite set than $\mathbb{N}$ would do: the actual values are irrelevant). The permutation group will consist of all permutations of $A \times \mathbb{N}$ that leave the first member unchanged in each pair: different occurrences of a given event can be freely permuted, but obviously they cannot be exchanged for events of a different name.

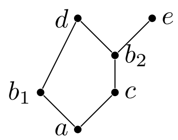
3 *Example.* When modelling a calculus like CCS [26], the same arena can be used as in CSP, taking for A the set of action labels, including polarities, that is $N \uplus \{\bar{u} \mid u \in N\}$ if N is the set of names.

4 *Example.* Things get more subtle when modelling a calculus with name passing like the π-calculus [27]. For the monadic case, the arena will consist of triples (ε, a, i) where ε is a polarity (input or output), a is a name (either a free name or a name bound by an action) and i is an occurrence number. Names bound by different input events will be considered different: in process terms, instead of $u(x).P \mid u(x).Q$, write $u(x_1).P[x_1/x] \mid u(x_2).P[x_2/x]$. The considered permutations are those that respect the name structure: if σ maps an event $u(x_1)$ to an event $u(x_2)$, then it must map any event involving x_1 to an event of the same type involving x_2 instead. Private names, like a in $(\boldsymbol{\nu}a)(a.P \mid \bar{a}.Q)$, will not be represented in arenas, since by definition they cannot be involved in interaction with the environment, unless they are communicated by scope extrusion, as in $(\boldsymbol{\nu}a)\bar{u}a$, in which case they will be modelled the same way as binding input prefixes. This construction is developed in more detail in Section 4.

5 **Definition.** A *play* over X is a pair $s = (|s|, \leqslant_s)$ where $|s|$ is a finite subset of $|X|$ (the support) and $\leqslant_s$ is a preorder over $|s|$; the set of plays over X is written $\mathcal{S}(X)$. A play s is called *consistent* if the relation $\leqslant_s$ is a partial order relation (i.e. if it is acyclic).

The intuition is that a play represents a possible way a process may act: the support contains the set of all events that will actually occur, the preorder represents scheduling constraints for these events. Consistency means that these constraints are not contradictory, i.e. that they do not lead to a deadlock. Synchronisation, defined below, consists in combining constraints from two plays, assuming they have the same events. The primitive definition of plays as pre-orders is a way to make it a total operator by separating it from the consistency condition: two plays can synchronise even if their scheduling constraints are not compatible, but then the result is inconsistent.

6 *Example.* We will represent a (consistent) play graphically as the Hasse diagram of its order relation, with each node labelled by the event's name. By convention, when two events are part of the same orbit under G^X, we use the same name with different indices:



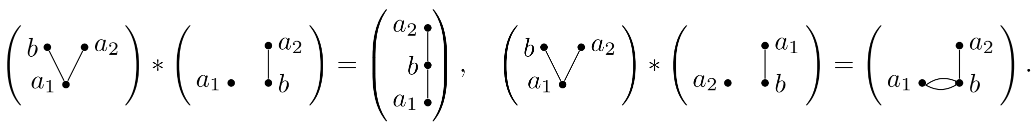
This represents a play with support $\{a, b_1, b_2, c, d, e\}$, with the order relation such that $a < b_1$, $a < b_2$, $b_1 < d$, $c < b_2$, $b_2 < d$ and $b_2 < e$, in an arena that has a permutation that swaps b_1 and b_2.

7 **Definition.** For $r, s \in \mathcal{S}(X)$ with $|r| = |s|$, the *synchronisation* of r and s is the play

$$r * s := \big(|r|, (\leqslant_r \cup \leqslant_s)^*\big),$$

where $(\cdot)^*$ denotes the reflexive transitive closure. Given a finite subset A of $|X|$, define the *A-neutral* play as $e_A := (A, \mathrm{id}_A)$ where id_A is the identity relation.

8 *Example.* We have the following synchronizations:



The second one leads to an inconsistent play, since the union of the order relations is cyclic.

Note that synchronisation is a very restrictive operator because it requires the event sets to be equal. The possibility of synchronising on some events while keeping the others independent, which is a natural notion, will be defined in Section 3.1 using this primitive form of total synchronisation.

Commutativity of $*$ is immediate from the definition. Associativity is also clear: for $r, s, t \in \mathcal{S}(X)$, $(r * s) * t$ and $r * (s * t)$ are defined if and only if $|r|, |s|, |t|$ are equal, and in this case we have $\leqslant_{(r*s)*t} = (\leqslant_r \cup \leqslant_s \cup \leqslant_t)^* = \leqslant_{r*(s*t)}$. Because of the constraint on supports, there cannot be a neutral element. However, among plays of a given support A, the neutral play e_A is actually neutral for synchronisation.

We now define the action of the permutation group G^X over the set of plays. Since there is usually no ambiguity, we overload the notation for group actions: given $\sigma \in \mathrm{G}^X$, for $x \in X$ we write σx for the image of x, for $A \subseteq X$ we write σA for the set of images $\{\sigma x \mid x \in A\}$, and similarly for $r \in \mathcal{S}(X)$ we write σr for the play r permuted by σ, as defined below.

9 **Definition.** Let X be an arena. The action of a permutation $\sigma \in \mathrm{G}^X$ on a play $r \in \mathcal{S}(X)$ is defined as

$$\sigma r := \big(\sigma|r|,\ \{(\sigma x, \sigma y) \mid (x, y) \in \leqslant_r\}\big).$$

The *orbit* of a play r in $\mathcal{S}(X)$ is the set

$$\mathrm{G}^X(r) := \big\{\sigma r \bigm| \sigma \in \mathrm{G}^X\big\}.$$

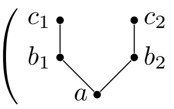
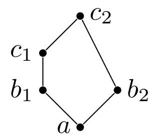
We refer the reader to some reference textbook (for instance Lang's *Algebra* [23]) for details on the standard group-theoretic notions in use here. For reference, given a group G acting on a set X, the *stabilizer* of a point $x \in X$ in the action of G is, by definition, the subgroup of G consisting of all the $\sigma \in G$ that leave x unchanged, i.e. $\sigma x = x$. The *pointwise* stabilizer of a set $A \subseteq X$ is the subgroup of those that leave each point in A unchanged, as opposed to the *setwise* stabilizer which includes all permutations that leave the set A unchanged as a whole (i.e. $\{\sigma x \mid x \in A\} = A$). The index of a subgroup H in a group G, written $(H : G)$, is the number of left cosets of H in G, that is the cardinal of $\{\sigma H \mid \sigma \in G\}$. When H is a normal subgroup of G, the index $(H : G)$ is the cardinal of the quotient group G/H.

10 **Definition.** Let X be an arena and r be a play in $\mathcal{S}(X)$. Let G^X_r be the stabilizer of r in the action of G^X over $\mathcal{S}(X)$ and let $\mathrm{G}^X_{|r|}$ be the pointwise stabilizer of $|r|$ in G^X, then the *multiplicity* of r in X is the index of $\mathrm{G}^X_{|r|}$ in G^X_r:

$$\mu_X(r) := (\mathrm{G}^X_r : \mathrm{G}^X_{|r|}).$$

Hence, the multiplicity of r is the number of different ways one can permute r into itself. Indeed, the definition as $(\mathrm{G}_r : \mathrm{G}_{|r|})$ exactly means the number of permutations of r into itself (elements of G_r), up to permutations that leave each point of $|r|$ invariant (elements of $\mathrm{G}_{|r|}$), in other words $\mu(r)$ is the order of the group $\{\sigma|_{|r|} \mid \sigma \in \mathrm{G}, \sigma r = r\}$, which is always finite since the support $|r|$ is finite.

11 *Example.* Using the same conventions as in Example 6, we have

μ () $= 2$ and μ () $= 1$

Both plays have the same support, there are 4 permutations of this support: b_1 and b_2 can be exchanged, idem for c_1 and c_2. In the first case if we exchange b_1 with b_2 and c_1 with c_2, we get the same play (permuting the b but not the c yields a different play). In the second case, no permutation can yield the same play.

2.2 Linear combinations

The set of plays of an arena X is independent of the group G, but our idea is that plays that are permutations of each other should be considered equivalent, since permutations exchange occurrences of indistinguishable actions. In the presence of permutations, however, there are several ways to synchronise two plays, so in order to extend the definition of synchronisation we have to be able to consider combinations of possible plays. For genericity, and because our aim is to get a quantitative account of interaction, we will use linear combinations, with coefficients in an unspecified commutative semiring.

12 **Definition.** A *commutative semiring* $\mathbb{S}$ is a tuple $(\mathbb{S}, +, \cdot, 0, 1)$ such that $(\mathbb{S}, +, 0)$ and $(\mathbb{S}, \cdot, 1)$ are commutative monoids and for all $x, y, z \in \mathbb{S}$ it holds that $x \cdot (y + z) = x \cdot y + x \cdot z$ and $x \cdot 0 = 0$. A *semimodule* over $\mathbb{S}$ is a commutative monoid $(M, +, 0)$ with an action $(\cdot) : \mathbb{S} \times M \to M$ that commutes with addition on both sides and satisfies $\lambda \cdot (\mu \cdot x) = (\lambda \cdot \mu) \cdot x$ for all $\lambda, \mu \in \mathbb{S}$ and $x \in M$. A *commutative semialgebra* over $\mathbb{S}$ is a semimodule M with a bilinear operation that is associative and commutative.

Terminology about semirings, semimodules and semialgebras is not standard, in particular some definitions do not require both neutrals. Sometimes, the neutrals are not required to be distinct (they are equal if and only if the semiring is a singleton, but this is a degenerate case that we will not consider). In the above definitions, if all elements of $\mathbb{S}$ have additive inverses, then $\mathbb{S}$ is a (commutative unitary) ring, and the semimodules and semialgebras are actually modules and algebras (indeed, the action of $\mathbb{S}$ imposes the existence of additive inverses in them too). If $\mathbb{S}$ is a field, we get the usual notions of vector space and algebra.

13 **Definition.** Let $\mathbb{S}$ be a commutative semiring. The *integers* of $\mathbb{S}$ are the finite sums of 1 including the empty sum 0, the *non-zero integers* are the finite non-empty sums. We call $\mathbb{S}$ *regular* if for every non-zero integer $n \in \mathbb{S}$, for all $x, y \in \mathbb{S}$, $nx = ny$ implies $x = y$. We call $\mathbb{S}$ *rational* if every non-zero integer has a multiplicative inverse.

In particular, regularity applied to $x = 1$ and $y = 0$ imposes that no non-empty sum of 1 can be equal to 0, in other words $\mathbb{S}$ has characteristic zero. The rationality condition means that it is possible to divide by non-zero natural numbers, or in more abstract terms that the considered semiring is a semimodule over the semiring of non-negative rationals. This obviously implies regularity.

Two important cases of rational semirings will be considered here. The first case is that of commutative algebras over the field $\mathbb{Q}$ of rational numbers, which includes fields of characteristic zero (among which rational, real and complex numbers) and commutative algebras over them. The second case is when addition is idempotent, which includes so-called *tropical* semirings [30], and the canonical examples are that of min-plus and max-plus semirings. Boolean algebras with disjunction as sum and conjunction as product are another typical example. In this case, all integers except 0 are equal to 1, so they obviously have multiplicative inverses.

Throughout this paper, unless explicitly stated otherwise, $\mathbb{S}$ is any semiring. Note that the semiring $\mathbb{N}$ of natural numbers and the ring $\mathbb{Z}$ of integers are regular but not rational. Indeed, when using natural numbers as scalars, some properties of order algebras will be lost, for instance the existence of bases. Hence some statements will explicitly require $\mathbb{S}$ to be regular or rational.

For an arbitrary set X, a formal linear combination over X is a function from X to $\mathbb{S}$ that has a value other than 0 on a finite number of points. Formal linear combinations over X, with sum and scalar product defined pointwise, form the free $\mathbb{S}$-semimodule over X and an element $x \in X$ is identified with its "characteristic" function $\delta_x : X \to \mathbb{S}$, such that $\delta_x(x) = 1$ and $\delta_x(y) = 0$ for all $y \neq x$. If X is finite, then the set of formal linear combinations is the $\mathbb{S}$-semimodule $\mathbb{S}^X$. For an arbitrary subset A of a $\mathbb{S}$-semimodule E, we denote by $\langle A \rangle_{\mathbb{S}}$, or simply $\langle A \rangle$, the submodule of E generated by A, i.e. the smallest submodule of E that contains A, that is the set of finite linear combinations of elements of A.

14 **Definition.** Let X be an arena. The *preliminary order algebra* $\mathcal{C}_{\mathbb{S}}(X)$ is the free $\mathbb{S}$-semimodule over $\mathcal{S}(X)$. The *outcome* is the linear form $\lfloor \cdot \rfloor$ over $\mathcal{C}_{\mathbb{S}}(X)$ such that $\lfloor r \rfloor = 1$ when r is consistent and $\lfloor r \rfloor = 0$ otherwise.

We usually keep the semiring $\mathbb{S}$ implicit in our notations. Vectors in $\mathcal{C}(X)$ are finite linear combination of plays in X, they represent the collection of all possible behaviours of a finite process. The coefficients can be understood as the amount of each behaviour that is present in the process. Examples in further sections also illustrate that $\mathbb{S}$ can be chosen to represent conditions on the availability of each behaviour. The outcome represents how relevant each play is, and by the intuition exposed in the previous section, plays with cyclic dependencies cannot happen, so they are considered irrelevant.

15 *Example.* Consider the CSP term $P = a{\to}(b \parallel c) \mid a{\to}c$ (remember that in CSP $\mid$ is the choice operator, and $\parallel$ is parallel composition). An interpretation of P in a preliminary order algebra containing only a, b, c as events could be

$$() + 2\,(\bullet\, a\,) + \begin{pmatrix} \bullet\, b \\ \bullet\, a \end{pmatrix} + 2 \begin{pmatrix} \bullet\, c \\ \bullet\, a \end{pmatrix} + \begin{pmatrix} b \bullet & & \bullet\, c \\ & \bullet\, a & \end{pmatrix}$$

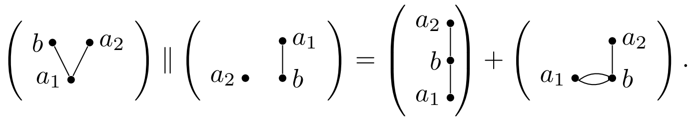
where we have a summand for each partial run of P. The coefficient 2 in the second and fourth summands represent the fact that there are two ways to perform only a, and two ways to perform a then c, depending on the choice one has done.

16 **Definition.** Let X be an arena. Permuted synchronisation in X is the bilinear operator $\parallel$ over $\mathcal{C}(X)$ such that for all plays $r, s \in \mathcal{S}(X)$,

$$r \parallel s := \mu_X(s) \sum_{\substack{s' \in \mathrm{G}^X(s) \\ |s'| = |r|}} r * s'$$

Observe that this sum is always finite. The reason is that each s' can be written σs for some $\sigma \in \mathrm{G}^X$, and $|\sigma s| = |r|$ implies $\sigma|s| = |r|$. Since σs is determined by the action of σ on $|s|$, there is at most one image of s for each bijection between $|s|$ and $|r|$. Since $|r|$ and $|s|$ are finite, the number of such bijections is finite.

17 *Example.* Considering again the plays in Example 8, we have



The first term in the sum corresponds to the identity permutation, the second one exchanges a_1 and a_2. Here, all plays involved have multiplicity 1.

18 Permuted synchronisation is similar to the parallel composition operator of CSP, in the case of processes defined on the same alphabet: in $r \parallel s$, every event of r must be synchronized with some event of the same name in s. There is a difference between plays and processes, however, in that in a play, every event must occur, whereas in a process, an action may be cancelled, for lack of a partner action to synchronize with.

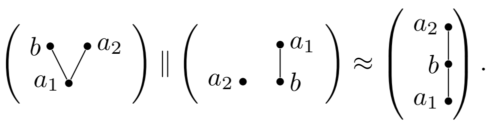
Any partial function $f : \mathcal{S}(X)^n \to \mathcal{S}(X)$ extends as an n-linear operator $\bar{f} : \mathcal{C}(X)^n \to \mathcal{C}(X)$, by setting $\bar{f}(r_1, \ldots, r_n) = 0$ when $f(r_1, \ldots, r_n)$ is undefined. This applies in particular to synchronisation, which yields a bilinear operator $\bar{*}$ over $\mathcal{C}(X)$. As a slight abuse of notations, we will write it simply as $*$ when there is no ambiguity.

Using this convention, permuted synchronisation can be seen as a generalisation of non-permuted synchronisation, since when the permutation group is trivial, all multiplicities are 1 and all orbits are singletons. Although $\parallel$ is a generalisation of $*$, we still use different notations, since both operators are of interest in a given non-static arena. The non-permuted version will be referred to as *static synchronisation* to avoid confusion.

19 **Definition.** Let X be an arena. Observational equivalence in $\mathcal{C}_{\mathbb{S}}(X)$ is defined as $u \approx_X u'$ when $\lfloor u \parallel v \rfloor = \lfloor u' \parallel v \rfloor$ for all $v \in \mathcal{C}_{\mathbb{S}}(X)$. The *order algebra* over X is $\mathcal{A}_{\mathbb{S}}(X) := \mathcal{C}_{\mathbb{S}}(X)/{\approx_X}$.

The scalar $\lfloor u \parallel v \rfloor$ is understood as the result of testing a process u against a process v. It linearly extends the basic case of single plays: $\lfloor r * s \rfloor$ is 1 if r and s are compatible and 0 otherwise; $\lfloor r \parallel s \rfloor$ is the number of different ways r and s can be permuted so that they become compatible. Hence the definition: $u \approx v$ if u and v are indistinguishable by this testing protocol.

20 *Example.* Any inconsistent play is observationally equivalent to 0, since synchronising it with any order yields an inconsistent play. Hence the synchronisation of Example 17 implies



21 **Lemma.** *Let X be an arena. For all $u \in \mathcal{C}(X)$ and $\sigma \in \mathrm{G}^X$, we have $u \approx \sigma u$.*

Proof. Since plays generate the module $\mathcal{C}(X)$, clearly $u \approx \sigma u$ if and only if $\lfloor u \parallel s \rfloor = \lfloor \sigma u \parallel s \rfloor$ for all play s. Since u is a finite linear combination of plays, the definition of synchronisation on plays extends as $\lfloor u \parallel s \rfloor = \mu(s) \sum_{s' \in \mathrm{G}(s)} \lfloor u * s' \rfloor$. It is clear that for all $\sigma \in G$ we have $\sigma(u * s') = \sigma u * \sigma s'$, moreover outcomes are preserved by permutations, so we have $\lfloor u * s' \rfloor = \lfloor \sigma u * \sigma s' \rfloor$ for all s', hence $\lfloor u \parallel s \rfloor = \mu(s) \sum_{s' \in \mathrm{G}(s)} \lfloor \sigma u * \sigma s' \rfloor$. Since σ acts as a permutation on the orbit $\mathrm{G}(s)$, $\sigma s'$ and s' range over the same set, so we have $\lfloor u \parallel s \rfloor = \lfloor \sigma u \parallel s \rfloor$, and finally $u \approx \sigma u$. $\square$

The fact that observational equivalence is preserved by linear combinations is immediate from the definition, since synchronisation and outcome are linear. As a consequence, in each orbit $\mathrm{G}(s)$, we can choose a representant $\underline{s}$ such that each vector in $\mathcal{C}(X)$ is equivalent to a linear combination of representants.

22 **Definition.** Let X be an arena. A *choice of representants* for X is a pair of an idempotent map $A \mapsto \underline{A}$ over $\mathcal{P}_f(|X|)$ and an idempotent map $r \mapsto \underline{r}$ over $\mathcal{S}(X)$ such that for all $r \in \mathcal{S}(X)$ $|\underline{r}| = \underline{|r|}$, and for all $r, s \in \mathcal{S}(X)$, $\underline{r} = \underline{s}$ if and only if $r = \sigma s$ for some $\sigma \in \mathrm{G}^X$.

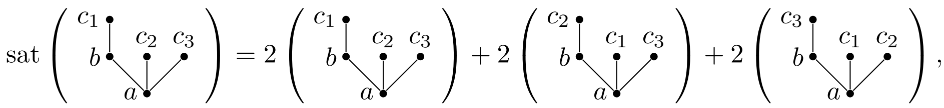
So a choice of representants picks one play in each orbit under G^X in such a way that representants have the same support if it is possible. There always exists such choices, and in the sequel we assume that each arena comes with a particular choice, written $r \mapsto \underline{r}_X$. The choice function over $\mathcal{S}(X)$ induces a projection in $\mathcal{C}(X)$ by linearity, and for all $u \in \mathcal{C}(X)$ we have $u \approx \underline{u}_X$ by Lemma 21.

23 **Definition.** Let X be an arena. *Saturation* in X is the linear map $\mathrm{sat}_X : \mathcal{C}(X) \to \mathcal{C}(X)$ such that for each play r,

$$\mathrm{sat}_X\, r := \sum_{\sigma \in \mathrm{G}^{|r|}} \sigma r \qquad \text{where} \qquad \mathrm{G}^A := \big\{ \sigma|_A \;\big|\; \sigma \in \mathrm{G}^X, \sigma A = A \big\}.$$

The set G^A is the group of permutations of A induced by G, it is isomorphic to the quotient $\mathrm{G}_{\{A\}}/\mathrm{G}_A$ where $\mathrm{G}_{\{A\}}$ is the setwise stabilizer of A in G and G_A is its pointwise stabilizer (it is easy to check that the latter is a normal subgroup of the former).

24 *Example.* Again using the conventions of Example 6, we have



where the factor 2 comes from the fact that exchanging c_2 and c_3 in the original play does not change it. Indeed, the multiplicity of this play is 2.

25 **Lemma.** *Let X be an arena. For all $r, s \in \mathcal{S}(X)$ such that $|r| = |s|$ we have $s \parallel r = s \mathbin{\bar{*}} \mathrm{sat}\, r$.*

Proof. We use the notations of Definition 23. As σ ranges over $\mathrm{G}^{|r|}$, σr ranges over all the elements of the orbit of r under G that have the same support as r. Moreover, each play in the orbit is hit a number of times equal to $\mu(r)$, so for any play s we have the expected equality. □

In particular, this implies the equivalence $u \parallel v \approx \underline{u} \mathbin{\bar{*}} \mathrm{sat}\, \underline{v}$ for all u and v. We will use this fact in Proposition 29 to get a representation of arbitrary order algebras in static ones.

26 **Proposition.** *Permuted synchronisation is compatible with observational equivalence. Up to observational equivalence, it is associative and commutative.*

Proof. We first prove that permuted synchronisation is strictly associative. Consider three plays r, s, t. For all $\sigma \in \mathrm{G}^X$ we have $r \parallel \sigma s = r \parallel s$, so we can assume that r, s, t have equal support (if no permutation can let them have the same support, then synchronisation in any order is zero). Then we have $r \parallel (s \parallel t) = r * \mathrm{sat}(s * \mathrm{sat}\, t)$ by Lemma 25, hence

$$r \parallel (s \parallel t) = \sum_{\sigma,\tau \in \mathrm{G}^{|r|}} r * \sigma(s * \tau t) = \sum_{\sigma,\tau \in \mathrm{G}^{|r|}} (r * \sigma s) * \sigma\tau t = \sum_{\sigma,\tau \in \mathrm{G}^{|r|}} (r * \sigma s) * \tau t = (r \parallel s) \parallel t$$

using associativity of strict synchronisation and the fact that, for a fixed σ, the permutations $\sigma\tau$ and τ range over the same set. This extends to all vectors by linearity.

Associativity implies compatibility with observational equivalence: for two equivalent vectors $u \approx u'$ and an arbitrary $v \in \mathcal{C}(X)$, for all $w \in \mathcal{C}(X)$ we have $\lfloor (u \parallel v) \parallel w \rfloor = \lfloor u \parallel (v \parallel w) \rfloor = \lfloor u' \parallel (v \parallel w) \rfloor = \lfloor (u' \parallel v) \parallel w \rfloor$ so $u \parallel v \approx u' \parallel v$.

Since observational equivalence is preserved by permutations, using commutativity of strict synchronisation we have

$$s \parallel r = \sum_{\sigma \in \mathrm{G}^{|r|}} s * \sigma r = \sum_{\sigma \in \mathrm{G}^{|r|}} \sigma(\sigma^{-1} s * r) \approx \sum_{\sigma \in \mathrm{G}^{|r|}} \sigma^{-1} s * r = \sum_{\sigma \in \mathrm{G}^{|r|}} r * \sigma^{-1} s = r \parallel s.$$

which proves commutativity of permuted synchronisation up to observational equivalence. ☐

27 **Lemma.** *Let X be an arena and let $(u_i)_{i \in I}$ be a finite family of vectors in $\mathcal{C}(X)$. There exists a vector e and an integer $n > 0$ such that for all $i \in I$, $u_i \parallel e = n\, u_i$.*

Proof. Observe that for all finite subset A of X, every setwise stabilizer of A is a stabilizer of the A-neutral play e_A, so we have $\operatorname{sat} e_A = \mu(e_A)\, e_A$, and subsequently for all play $s \in \mathcal{S}(X)$ such that $\underline{|s|} = \underline{A}$ we get $s \parallel e_A = \mu(e_A)\,(s * e_A) = \mu(e_A)\, s$. Call P the set of all $\underline{|r|}$ such that the play r has a non-zero coefficient in some u_i, then P is finite since each u_i is a finite linear combination of plays. Let n be the least common multiple of the $\mu(e_A)$ for A in P, then the vector

$$e := \sum_{A \in P} \frac{n}{\mu(e_A)} e_A$$

satisfies $e \parallel u_i = n u_i$ for each i by construction. Note that the coefficients $n/\mu(e_A)$ are all natural numbers, by construction. ☐

28 **Corollary.** *If $\mathbb{S}$ is regular then outcome is preserved by observational equivalence.*

Proof. Let $u \approx v$ be a pair of equivalent vectors in $\mathcal{C}(X)$. By Lemma 27 there is a vector e and an integer $n \neq 0$ such that $u \parallel e = nu$ and $v \parallel e = nv$, so we have $n\lfloor u \rfloor = \lfloor u \parallel e \rfloor = \lfloor v \parallel e \rfloor = n \lfloor v \rfloor$. By regularity, we can deduce $\lfloor u \rfloor = \lfloor v \rfloor$. ☐

As a consequence, the order algebra $\mathcal{A}(X)$, which is defined as the quotient of $\mathcal{C}(X)$ by observational equivalence, is a commutative semialgebra over $\mathbb{S}$ with synchronisation $\parallel$ as the product, and outcome $\lfloor \cdot \rfloor$ is a linear form over it. The choice of representants for orbits of finite sets and plays induces the following representation property of $\mathcal{A}(X)$ in the static algebra $\mathcal{A}(|X|)$.

29 **Proposition.** *Let X be an arena. Define the linear map $\Delta_X : \mathcal{C}(X) \to \mathcal{C}(X)$ as*

$$\Delta_X(u) := \operatorname{sat}_X \underline{u}_X.$$

For all $u, v \in \mathcal{C}(X)$,

$$u \approx_X v \qquad \text{if and only if} \qquad \Delta_X(u) \approx_{|X|} \Delta_X(v).$$

Hence Δ_X is an injective map from $\mathcal{A}(X)$ into $\mathcal{A}(|X|)$. For all $u, v \in \mathcal{A}(X)$,

$$\Delta_X(u \parallel v) = \Delta_X(u) * \Delta_X(v).$$

Proof. For compatibility with observational equivalences, first suppose that u and v are such that $\operatorname{sat} \underline{u} \approx_{|X|} \operatorname{sat} \underline{v}$. Consider a play $r \in \mathcal{S}(X)$, then we have $\lfloor u \parallel r \rfloor = \lfloor \operatorname{sat} \underline{u} * \underline{r} \rfloor = \lfloor \operatorname{sat} \underline{v} * \underline{r} \rfloor = \lfloor v \parallel r \rfloor$ using Lemma 25, so we have $u \approx_X v$.

Reciprocally suppose $u \approx_X v$, then by definition for all play $r \in \mathcal{S}(X)$ we have $\lfloor u \parallel r \rfloor = \lfloor v \parallel r \rfloor$. By the remarks above, the outcome $\lfloor u \parallel r \rfloor$ is equal to $\lfloor r \parallel u \rfloor = \lfloor \underline{r} * \operatorname{sat} \underline{u} \rfloor$, so we have $\lfloor \underline{r} * \operatorname{sat} \underline{u} \rfloor = \lfloor \underline{r} * \operatorname{sat} \underline{v} \rfloor$ for all r. Let s be an arbitrary play in $\mathcal{S}(X)$. Writing u as a linear combination $\sum_{i \in I} \lambda_i r_i$, we get $\lfloor \operatorname{sat} \underline{u} * s \rfloor = \sum_{i \in I} \lambda_i \left\lfloor \operatorname{sat} \underline{r_i} * \underline{s} \right\rfloor$. If $|s|$ is not a representant subset of $|X|$, then this sum is zero since $\operatorname{sat} r_i$ is a combination of plays whose supports are representant subsets. The same applies to v so we have $\lfloor \operatorname{sat} \underline{u} * s \rfloor = \lfloor \operatorname{sat} \underline{v} * s \rfloor = 0$. Now suppose that $|s|$ is a representant subset of $|X|$, then the representant $\underline{s}$ of s has the same support as s by definition, so there is a permutation σ such that $\sigma s = \underline{s}$ and $\sigma|s| = |s|$. For all $i \in I$, if $|\underline{r_i}| = |s|$, then by definition of saturation we have $\sigma \operatorname{sat} \underline{r_i} = \operatorname{sat} \underline{r_i}$, so we get $\lfloor \operatorname{sat} \underline{r_i} * s \rfloor = \lfloor \operatorname{sat} \underline{r_i} * \underline{s} \rfloor$. If $|\underline{r_i}| \neq |s|$, then the equality holds trivially since both sides are 0. By linearity, we can deduce $\lfloor \operatorname{sat} \underline{u} * s \rfloor = \lfloor \operatorname{sat} \underline{u} * \underline{s} \rfloor$, and applying the same reasoning to v, from our initial remarks we deduce $\lfloor \operatorname{sat} \underline{u} * s \rfloor = \lfloor \operatorname{sat} \underline{v} * s \rfloor$. Hence we get $\operatorname{sat} \underline{u} \approx_{|X|} \operatorname{sat} \underline{v}$.

For the commutation property with synchronisation, consider two plays $r, s \in \mathcal{S}(X)$. If the supports $|\underline{r}|$ and $|\underline{s}|$ are distinct, then clearly $\Delta_X(r \parallel s) = \Delta_X(r) * \Delta_X(s) = 0$. Otherwise, let A be this support. If ρ is a permutation in G^X such that $\rho r = \underline{r}$, we have $\underline{r \parallel s} = \underline{\rho(r \parallel s)} = \underline{\rho r \parallel s} = \underline{r} \parallel s$. Moreover, $\underline{r} \parallel s = \underline{r} * \operatorname{sat} \underline{s}$ so all terms in $\underline{r} \parallel s$ have support A, and since for all play t with $|t| = A$ we have $\operatorname{sat} \underline{t} = \operatorname{sat} t$, we get

$$\operatorname{sat} \underline{r \parallel s} = \operatorname{sat}(\underline{r} \parallel s) = \sum_{\sigma \in \mathrm{G}^A} \sigma(\underline{r} \parallel s) = \sum_{\sigma \in \mathrm{G}^A} \sigma \underline{r} \parallel s = \sum_{\sigma \in \mathrm{G}^A} \sum_{\tau \in \mathrm{G}^A} \sigma \underline{r} * \tau \underline{s} = \operatorname{sat} \underline{r} * \operatorname{sat} \underline{s}$$

which concludes the proof. ☐

The commutation property could actually be written $\Delta_X(u \parallel v) = \Delta_X(u) \parallel \Delta_X(v)$ since permuted and static synchronisations coincide in the static order algebra $\mathcal{A}(|X|)$, but we keep the notations distinct to stress the fact that the second is static. This establishes an injective morphism of $\mathbb{S}$-semialgebras, however this morphism does not preserve outcomes: for a play s, we have $\lfloor \Delta s \rfloor = \sharp(\mathrm{G}^{|s|}) \lfloor s \rfloor$; since this factor depends on $|s|$, the outcome of $\Delta(u)$ is not even proportional to that of u in general.

30 **Proposition.** *Let X be an arena. Assume $\mathbb{S}$ is rational. Then the $\mathbb{S}$-semialgebra $\mathcal{A}(X)$ has a unit element if and only if the web $|X|$ is finite.*

Proof. Suppose $|X|$ is finite, then the set of plays $\mathcal{S}(X)$ is finite, so we can apply Lemma 27 to the whole set $\mathcal{S}(X)$, which provides a vector e and a non-zero integer n such that $e \parallel s = ns$ for all $s \in \mathcal{S}(X)$; then e/n is a neutral element for synchronisation. Now suppose that $|X|$ is infinite. Let u be an arbitrary vector in $\mathcal{C}(X)$. Since u is a finite linear combination of plays with finite support, there is an integer n such that all non-zero components of u are plays with supports of cardinal strictly less than n. Let A be a subset of $|X|$ of cardinal n, then we must have $u \parallel e_A = 0 \neq e_A$. This implies that no finite linear combination of plays can be neutral. ☐

2.3 Bases

In this section, we describe the $\mathbb{S}$-semimodule $\mathcal{A}(X)$ by providing a subset of plays whose equivalence classes forms a basis. Linear independence does not have a unique definition for modules over arbitrary semirings [3], so we state the appropriate definition for our needs, which clearly extends the standard one for vector spaces:

31 **Definition.** Let $\mathbb{S}$ be a semiring and E a semimodule over $\mathbb{S}$. A family $(u_i)_{i \in I}$ in E is linearly independent if, for any two families $(\lambda_i)_{i \in I}$ and $(\mu_i)_{i \in I}$ in $\mathbb{S}$ with finite support, if $\sum_{i \in I} \lambda_i u_i = \sum_{i \in I} \mu_i u_i$ then for all i, $\lambda_i = \mu_i$. A basis of E is a linearly independent generating family.

We first concentrate on the case of static order algebras. The first thing we can remark about observational equivalence is that plays of different supports are always independent, since compatibility explicitly requires having the same support, so we have the following decomposition:

32 **Proposition.** *For a finite static arena X, let $\mathcal{C}^s(X)$ be the submodule of $\mathcal{C}(X)$ generated by plays of support X. Define the* strict order algebra *over X as the submodule $\mathcal{A}^s(X)$ of $\mathcal{A}(X)$ made of equivalence classes of elements of $\mathcal{C}^s(X)$. Then for all static arena X we have*

$$\mathcal{A}(X) = \bigoplus_{Y \in \mathcal{P}_f(X)} \mathcal{A}^s(Y).$$

Proof. Clearly $\mathcal{C}(X)$ is the direct sum of the $\mathcal{C}^s(Y)$, since this decomposition amounts to partitioning the basis $\mathcal{S}(X)$ according to the supports Y of its elements. As a consequence, $\mathcal{A}(X)$ is the sum of the $\mathcal{A}^s(Y)$, and we have to prove that this sum is direct. Consider two vectors $u = \sum_{Y \in \mathcal{P}_f(X)} u_Y$ and $v = \sum_{Y \in \mathcal{P}_f(X)} v_Y$ such that $u \approx v$ and for all $Y \in \mathcal{P}_f(X)$, $u_Y, v_Y \in \mathcal{C}^s(Y)$ (necessarily, only finitely many of the u_Y and v_Y are not 0). For each $Y \in \mathcal{P}_f(X)$, we have $u * e_Y = u_Y$ and $v * e_Y = v_Y$, so $u_Y \approx v_Y$ since $*$ is compatible with $\approx$. As a consequence, the decomposition of a vector in $\mathcal{A}(X)$ on the submodules $\mathcal{A}^s(Y)$ is unique. □

We can thus focus on the study of strict order algebras. These have the definite advantage of being finitely generated, since there are finitely many different binary relations over a given finite set. We will now provide explicit bases for them, depending on the structure of $\mathbb{S}$.

Let X be a finite static arena. Clearly, for all inconsistent plays r we have $r \approx 0$, so we can consider only consistent plays, i.e. plays r such that $\leqslant_r$ is an order relation. In the following statements, as a slight abuse of notations, a play r with $|r| = X$ is identified with its order relation $\leqslant_r$, and also with its equivalence class in $\mathcal{A}^s(X)$. Let $\mathcal{O}(X)$ be the set of all partial order relations over X.

The notations $<_r$, $\geqslant_r$, $>_r$ are defined as expected. We denote by $|_r$ the incomparability relation: $x \mid_r y$ if and only if neither $x \leqslant_r y$ nor $y \leqslant_r x$. We write $x \parallel_r y$ if $x = y$ or $x \mid_r y$. If there is no ambiguity, we may omit the subscript r in these notations. The notation $[a < b]_X$, for $a, b \in X$, represents the smallest partial order over X for which $a < b$, that is $\mathrm{id}_X \cup \{(a, b)\}$. The notation extends to more complicated formulas, for instance $[a < b, c < d]$ is the smallest partial order for which $a < b$ and $c < d$. We write $r \smile s$ to denote that two partial orders r and s are compatible.

33 **Proposition.** *Let $\mathcal{T}(X)$ be the set of total orders over X, then $\mathcal{T}(X)$ is a linearly independent family in $\mathcal{A}^s(X)$.*

Proof. We prove the equivalent statement that two observationally equivalent combinations of total orders are necessarily equal. Let $u = \sum_{t \in \mathcal{T}(X)} \lambda_t t$ and $v = \sum_{t \in \mathcal{T}(X)} \mu_t t$ be two combinations such that $u \approx v$. If r and s are two distinct total orders over X, there exists a pair $(a, b) \in X^2$ such that $a <_r b$ and $b <_s a$, hence r and s are not compatible, so $\lfloor r * s \rfloor = 0$. Besides, it always holds that $\lfloor r * r \rfloor = 1$, so for all $t \in \mathcal{T}(X)$, $\lfloor u * t \rfloor = \lambda_t$ and $\lfloor v * t \rfloor = \mu_t$, so $u \approx v$ implies $\lambda_t = \mu_t$ for all t, hence $u = v$. □

However, in general, $\mathcal{T}(X)$ is not a generating family for $\mathcal{A}^s(X)$. The simplest counterexample can be found if X has two points. Write $X = \{a, b\}$, then $\mathcal{O}(X)$ has three elements:

$$\mathcal{O}(\{a, b\}) = \big\{ [a \mid b], [a < b], [a > b] \big\}.$$

Then in the canonical basis $([a \mid b], [a < b], [a > b])$ of $\mathcal{C}^s(X)$, the matrix of $(u, v) \mapsto \lfloor u * v \rfloor$ is

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

If $\mathbb{S}$ is the field of reals, for instance, then this matrix is invertible, which means that the three orders are linearly independent, hence $\mathcal{A}^s(X)$ is isomorphic to $\mathbb{S}^{\mathcal{O}(X)}$ (this isomorphism holds if and only if the cardinal of X is at most 2, as we shall see below). There is one case where $[a < b]$ and $[b < a]$ do generate $\mathcal{A}^s(\{a, b\})$, namely when addition in $\mathbb{S}$ is idempotent, i.e. when $1+1 = 1$.

34 **Proposition.** *$\mathcal{T}(X)$ is a basis of $\mathcal{A}^s(X)$ for all X if and only if addition in $\mathbb{S}$ is idempotent.*

Proof. By Proposition 33, we know that $\mathcal{T}(X)$ is always a linearly independent family, so all we have to prove is that it generates $\mathcal{A}^s(X)$ if and only if $1 + 1 = 1$ in $\mathbb{S}$.

Firstly, assume that $\mathcal{T}(X)$ generates $\mathcal{A}^s(X)$ for all X. Then, for $X = \{a, b\}$, there are two scalars $\lambda, \mu \in \mathbb{S}$ such that $[a \mid b] \approx \lambda[a < b] + \mu[a > b]$. Then we have

$$\lfloor [a \mid b] * [a < b] \rfloor = \lambda \lfloor [a < b] * [a < b] \rfloor + \mu \lfloor [a > b] * [a < b] \rfloor = \lambda$$

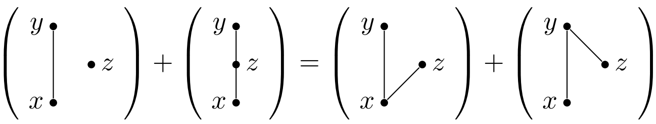
but by definition we have $\lfloor [a \mid b] * [a < b] \rfloor = 1$, so $\lambda = 1$. Similarly, we get $\mu = 1$, and so $[a \mid b] = [a < b] + [a > b]$. As a consequence, we have

$$1 = \lfloor [a \mid b] * [a \mid b] \rfloor = \lfloor [a < b] * [a \mid b] \rfloor + \lfloor [a > b] * [a \mid b] \rfloor = 1 + 1.$$

Reciprocally, assume $\mathbb{S}$ satisfies $1 + 1 = 1$. Let X be an arbitrary finite set and let $r \in \mathcal{O}(X)$. Let $u = \sum_{i=1}^{k} t_i$ be the sum of all total orders that are compatible with r. Consider an arbitrary order $s \in \mathcal{O}(X)$. Then we have $\lfloor u * s \rfloor = \sum_{i=1}^{k} \lfloor t_i * s \rfloor$ and each term of this sum is 0 or 1. If s is compatible with r, then there is a total order t that extends both r and s, so t is one of the t_i; since s and t are compatible, the sum contains at least one 1 so $\lfloor u * s \rfloor = 1 = \lfloor r * s \rfloor$. If s is incompatible with r, then it is incompatible with any order that contains r, and in particular it is incompatible with all the t_i, so $\lfloor u * s \rfloor = 0 = \lfloor r * s \rfloor$. As a consequence we have $r \approx u$, which proves that $\mathcal{T}(X)$ generates $\mathcal{A}^s(X)$. ☐

In the general case, without any hypothesis on the semiring $\mathbb{S}$, it happens that the family of all orders over X is not linearly independent, as soon as X has at least three points.

35 **Proposition.** *For all semiring $\mathbb{S}$, in $\mathcal{C}^s_{\mathbb{S}}(\{x, y, z\})$ we have*



Proof. We use the following notations: $a := [x < y]$, $b := [x < z < y]$, $c := [x < y, x < z]$, $d := [x < y, z < y]$, so that the equation we prove is $a + b = c + d$. Let s be a partial order over $\{x, y, z\}$. First remark that $s \smile a$ if and only if $s \smile c$ or $s \smile d$. Indeed, assume $s \smile a$, then there is a total order t that contains a and s. If $x <_t z$ then $[x < z] \subseteq t$ so $a * [x < z] \subseteq t$, hence $s \smile a * [x < z] = c$. Otherwise $z <_t x <_t y$ so $z <_t y$ then $s \smile d$. Reciprocally, if $s \smile c$ or $s \smile d$ then $s \smile a$ since a is included in c and d. Secondly, remark that $s \smile b$ if and only if $s \smile c$ and $s \smile d$. Indeed, assume that $s \smile c$ and $s \smile d$. Let $s' = s * c = s * a * [x < z]$. Suppose $s' \not\smile [z < y]$, then $y <_{s'} z$. By hypothesis we cannot have $y <_{a*s} z$, so (y, z) occurs in s' but not in $(s * a) \cup [x < z]$, which implies $y <_{s*a} x$. This contradicts the hypothesis $x <_a y$, hence $s' \smile [z < y]$, so $s \smile a * [x < z] * [z < y] = b$. The reciprocal implication is immediate since $a \subseteq b$ and $d \subseteq b$. As a consequence of the two remarks above, we have $\lfloor a * s \rfloor = 1$ if and only if $\lfloor c * s \rfloor = 1$ or $\lfloor d * s \rfloor = 1$, which is equivalent to $\lfloor (c + d) * s \rfloor \in \{1, 2\}$. Moreover, $\lfloor (c + d) * s \rfloor = 2$ if and only if $\lfloor c * s \rfloor = 1$ and $\lfloor d * s \rfloor = 1$, which is equivalent to $\lfloor b * s \rfloor = 1$. Therefore $\lfloor (a + b) * s \rfloor = \lfloor (c + d) * s \rfloor$. ☐

This proposition applies to orders on three points, but the exact same argument applies in any larger context, since the proof never uses the fact that there are no other points than x, y, z. So for any play r and points $x, y, z \in |r|$ such that $x <_r y$, $x |_r z$ and $y |_r z$ we have

$$r + (r * [x < z < y]) \approx (r * [x < z]) + (r * [z < y]).$$

This can also be deduced from Proposition 35 using the partial composition operators defined in Section 3.1. When $\mathbb{S}$ is a ring, it allows us to express each of the patterns of the equation in Proposition 35 as a linear combination of the others with coefficients 1 and -1. This implies that for each of these patterns, the set of all orders over X that do not contain the considered pattern generates $\mathcal{A}^s(X)$. In each case, the forbidden pattern defines a particular class of orders, respectively weak total orders (as of Proposition 36 below), orders of height at most 2 and forests with roots up or down.

36 **Proposition.** *Let $(X, \leqslant)$ be a partially ordered set. The following conditions are equivalent:*

- *For all $x, y, z \in X$, if $x < y$ then $x < z$ or $z < y$.*
- *The relation $\parallel$ is an equivalence.*
- *There is a totally ordered set $(Y, \leqslant)$ and a function $f : X \to Y$ such that, for all $x, y \in X$, $x < y$ if and only if $f(x) < f(y)$.*

Let $\mathcal{W}(X)$ be the set of orders that satisfy these conditions, called weak total orders *over X.*

Proof. Firstly, assume that for all $x, y, z \in X$, if $x < y$ then $x < z$ or $z < y$. It is clear that $\parallel$ is always reflexive and symmetric. Let $x, y, z \in X$ such that $x \parallel z$ and $z \parallel y$. If $x < y$, then by hypothesis we must have $x < z$ or $z < y$, which contradicts the hypothesis on x, y, z. Similarly we cannot have $y < x$, so $x \parallel y$. Therefore $\parallel$ is transitive and it is an equivalence relation.

Secondly, assume $\parallel$ is an equivalence relation. Let Y be the set of equivalence classes of $\parallel$. Define the relation $\sqsubseteq$ on Y as $A \sqsubseteq B$ if $a \leqslant b$ for some $a \in A$ and $b \in B$. The relation $\sqsubseteq$ is reflexive since for all $A \in Y$, for any $a \in A$ we have $a \leqslant a$ so $A \sqsubseteq A$. Assume $A \sqsubseteq B$ and $B \sqsubseteq A$ for some $A, B \in Y$, then there are $a, a' \in A$ and $b, b' \in B$ such that $a \leqslant b$ and $b' \leqslant a'$; if $a < b'$ then $a < a'$ which contradicts $a \parallel a'$, similarly if $b' < a$ then $b' < b$ which contradicts $b' \parallel b$, so $a \parallel b'$, which implies that A and B are the same class, therefore $\sqsubseteq$ is antisymmetric. Assume $A \sqsubseteq B$ and $B \sqsubseteq C$ for some $A, B, C \in Y$, then there are $a \in A$, $b, b' \in B$ and $c \in C$ such that $a \leqslant b$ and $b' \leqslant c$; if $a \parallel c$ then $A = C$ hence $A \sqsubseteq C$, otherwise we must have $a < c$ or $c < a$, but the second case implies $b' \leqslant c < a \leqslant b$ which contradicts $b \parallel b'$, so $a < c$ and $A \sqsubseteq C$, hence $\sqsubseteq$ is transitive. Totality is immediate: if A and B are two distinct classes, then every pair $(a, b) \in A \times B$ is comparable. Let f be the function that maps each element of X to its class. If $x < y$ then $f(x) \sqsubset f(y)$ by definition. Reciprocally, if $f(x) \sqsubset f(y)$, then x and y must be comparable (since they are in distinct classes), and $y < x$ would imply $f(y) \sqsubset f(x)$, so $x < y$.

Finally, assume there is $f : X \to Y$ where Y is totally ordered such that $x < y$ if and only if $f(x) < f(y)$. Let x, y, z be such that $x < y$, then $f(x) < f(y)$. Since the order on Y is total, we must have either $f(x) < f(z)$ or $f(z) < f(y)$ (or both), hence $x < z$ or $z < y$. ☐

In other words, a weak total order is a total order over sets of mutually incomparable points. Interestingly, this kind of order was considered long ago in scheduling theory [22] as the possibility to label events with time stamps in a possibly non-injective manner. It turns out that weak total orders form a basis.

37 **Definition.** Let $r \in \mathcal{O}(X)$. Two elements $a, b \in X$ are equivalent in r, written $a \sim_r b$, if for all $c \in X \setminus \{a, b\}$, $a <_r c$ if and only if $b <_r c$, and $c <_r a$ if and only if $c <_r b$. For a pair $a \sim_r b$ with $a \neq b$, let $r/(a \sim b)$ be the order $r \cap (X \setminus \{b\})^2$ over $X \setminus \{b\}$.

38 **Definition.** Let $a, b \in X$ with $a \neq b$. For each $r \in \mathcal{O}(X \setminus \{b\})$, define the relations

$$r_{a\sim b} := r \cup \{(x, b) \mid (x, a) \in r\} \cup \{(b, x) \mid (a, x) \in r\},$$
$$r_{a<b} := r_{a\sim b} \cup \{(a, b)\}, \qquad r_{a>b} := r_{a\sim b} \cup \{(b, a)\}.$$

Clearly, $r_{a\sim b}$, $r_{a<b}$ and $r_{a>b}$ are partial orders over X in which a and b are equivalent.

39 **Lemma.** *Let a, b be two distinct elements of X. For all $r \in \mathcal{W}(X)$ and $s \in \mathcal{O}(X \setminus \{b\})$,*

- *if $a <_r b$ then $r \smile s_{a\sim b}$ if and only if $r \smile s_{a<b}$, moreover $r \not\smile s_{a>b}$,*
- *if $a >_r b$ then $r \smile s_{a\sim b}$ if and only if $r \smile s_{a>b}$, moreover $r \not\smile s_{a<b}$,*
- *if $a \mid_r b$ then $r \smile s_{a\sim b}$ if and only if $r \smile s_{a<b}$ if and only if $r \smile s_{a>b}$.*

Proof. If $a <_r b$, we have $r \cup s_{a\sim b} = r \cup s_{a<b}$, since $s_{a\sim b}$ and $s_{a<b}$ only differ on (a, b), so the compatibility of the two pairs is equivalent to this union being acyclic. The same argument applies to the case $a >_r b$. For the case $a \mid_r b$, first assume $r \smile s_{a\sim b}$ and let $t = r * s_{a\sim b}$. By definition of weak orders, we have $a \sim_r b$. If $a <_t b$ then there exists a sequence $a = a_0, \ldots, a_n = b$ such that for each $i < n$, $a_i <_r a_{i+1}$ or $a_i <_{s_{a\sim b}} a_{i+1}$, but since a and b are equivalent in both r and $s_{a\sim b}$, we can replace b with a in this sequence, which leads to the contradiction $a <_t a$. By the same argument we cannot have $b <_t a$, so $a \mid_t b$. We thus have $t \smile [a < b]$ hence $s_{a<b} = s_{a\sim b} * [a < b] \smile r$, and similarly $r \smile s_{a>b}$. The reverse implications are immediate since $s_{a\sim b}$ is included in both $s_{a<b}$ and $s_{a>b}$. □

40 **Proposition.** *If $\mathbb{S}$ is a ring, then for all finite set X, $\mathcal{W}(X)$ is a basis of $\mathcal{A}^s(X)$.*

Proof. Let $Z(X)$ be the submodule of all the $u \in \mathcal{C}^s(X)$ such that for all order r over X, $\lfloor u * r \rfloor = 0$. We actually prove the fact that $\mathcal{C}^s(X)$ is isomorphic to the direct sum $\mathbb{S}^{\mathcal{W}(X)} \oplus Z(X)$, which is equivalent since by definition $\mathcal{A}^s(X)$ is $\mathcal{C}^s(X)/Z(X)$ when the semiring $\mathbb{S}$ is a ring.

We first prove that for all order r over X there is an $s \in \mathbb{S}^{\mathcal{W}(X)}$ such that $r - s \in Z(X)$. Let $N(r) = \{(a, b, c) \in X^3 \mid a <_r b, a \mid_r c, b \mid_r c\}$, we proceed by induction on $\sharp N(r)$. If $r = \emptyset$, then by Proposition 36 we have $r \in \mathcal{W}(X)$, so we can set $s = r$. Otherwise, consider a triple $(a, b, c) \in N(r)$. Define the orders $r_1 := r * [a < c]$, $r_2 := r * [c < b]$ and $r_3 := r * [a < c < b]$. By Proposition 35, we have $r_1 + r_2 - r_3 - r \in Z(X)$. Besides, for each $i \in \{1, 2, 3\}$, clearly $N(r_i) \subset N(r)$ and $(a, b, c) \in N(r) \setminus N(r_i)$, so $\sharp N(r_i) < \sharp N(r)$. We can then apply the induction hypothesis to get an $s_i \in \mathbb{S}^{\mathcal{W}(X)}$ such that $r_i - s_i \in Z(X)$. We can then conclude by setting $s := s_1 + s_2 - s_3$.

As a consequence we have $\mathcal{C}^s(X) = \mathbb{S}^{\mathcal{W}(X)} + Z(X)$, and we now prove that this sum is direct by proving $\mathbb{S}^{\mathcal{W}(X)} \cap Z(X) = \{0\}$. We proceed by recurrence on the size of X. If X has 0 or 1 element, then the only order over X is the trivial order t, and $\lfloor t * t \rfloor = 1 \neq 0$, so $Z(X) = \{0\}$ and the result trivially holds. Now let $n \geqslant 2$, suppose the result holds for all X with at most $n - 1$ points, and let $u \in \mathbb{S}^{\mathcal{W}(X)} \cap Z(X)$. We now prove that u is the zero function.

Let r be a weak total order that is not a total order, let $a, b \in X$ such that $a \mid_r b$. Let $X' = X \setminus \{b\}$. Define $u' \in \mathbb{S}^{\mathcal{W}(X')}$ by $u'(t) = u(t_{a\sim b})$ for all $t \in \mathcal{O}(X')$, so that $u(r) = u'(r/(a \sim b))$. For any orders $s \in \mathcal{W}(X)$ and $t \in \mathcal{O}(X')$, by Lemma 39 we have that $\lfloor s * (t_{a<b} + t_{a>b} - t_{a\sim b}) \rfloor$ is 0 if a and b are comparable in s, otherwise it is equal to $\lfloor s * t_{a\sim b} \rfloor$, which is itself equal to $\lfloor s/(a \sim b) * t \rfloor$ by restriction to X'. Let $s' = s/(a \sim b)$, we have

$$\lfloor u * (t_{a<b} + t_{a>b} - t_{a\sim b}) \rfloor = \sum_{s \in \mathcal{W}(X), a \mid_s b} u(s) \lfloor s * t_{a\sim b} \rfloor = \sum_{s \in \mathcal{W}(X), a \mid_s b} u'(s') \lfloor s' * t \rfloor$$

The mapping $s \mapsto s/(a \sim b)$ is a bijection from weak total orders over X such that $a \mid b$ to weak total orders over X', so the latter sum is equal to $\sum_{s' \in \mathcal{W}(X')} u'(s') \lfloor s' * t \rfloor = \lfloor u' * t \rfloor$. Besides, u

is in $Z(X)$ so $\lfloor u * (t_{a<b} + t_{a>b} - t_{a\sim b}) \rfloor = 0$, which implies $\lfloor u' * t \rfloor = 0$. This holds for all t, so $u' \in Z(X')$. By construction we have $u' \in \mathbb{S}^{\mathcal{W}(X')}$ so u' is in $\mathbb{S}^{\mathcal{W}(X')} \cap Z(X')$. By the induction hypothesis this is $\{0\}$, so $u' = 0$ and as a consequence we have $u(r) = u'(r/(a \sim b)) = 0$.

By the argument above, we thus know that $u(r) = 0$ as soon as r is not a total order. In other words, u is a linear combination of total orders. From Proposition 33 we know that total orders are linearly independent in $\mathcal{A}^s(X)$, so we can conclude that $u = 0$. □

As a consequence, weak total orders on subsets of $|X|$ form a basis of the static order algebra $\mathcal{A}(X)$. We can extend this property to arbitrary order algebras using the representation property.

41 **Theorem.** *Let X be an arena. Then $\mathcal{A}(X)$ has a basis $(b_i)_{i\in I}$ made of plays if*

- *$\mathbb{S}$ is idempotent, then the b_i are the orbits of totally ordered plays under G^X, or*
- *$\mathbb{S}$ is a regular ring, then the b_i are the orbits of weakly totally ordered plays under G^X.*

In both cases, if $\mathbb{S}$ is rational, then there exists a family of vectors $(b_i^*)_{i\in I}$ such that for all $i, j \in I$, $\lfloor b_i \parallel b_j^* \rfloor$ is 1 if $i = j$ and 0 otherwise.

Proof. Propositions 34 and 40 provide bases of the appropriate kinds for strict static order algebras. By Proposition 32, these yield bases for static order algebras. In each case, call the elements of these bases *base plays*. A permutation of a (weak) total order is always an order of the same kind, so from the fact that base plays generate $\mathcal{A}(|X|)$, we deduce that they also generate $\mathcal{A}(X)$. Now consider two linear combinations $u = \sum_{i\in I} \lambda_i r_i$ and $v = \sum_{i\in I} \mu_i r_i$, where the r_i are distinct base plays for $|X|$ and representants (as of Definition 22), and suppose $u \approx v$. By Proposition 29, we can deduce $\sum_{i\in I} \lambda_i \operatorname{sat} r_i \approx_{|X|} \sum_{i\in I} \mu_i \operatorname{sat} r_i$, and this equivalence is an equality since both sides are linear combinations of base plays. Now consider any $i \in I$. In $\sum_{i\in I} \lambda_i \operatorname{sat} r_i$, the coefficient of r_i is $\mu_X(r_i)$, so the equality above implies $\mu_X(r_i)\,\lambda_i = \mu_X(r_i)\,\mu_i$, and subsequently $\lambda_i = \mu_i$ since $\mu_X(r_i)$ is a non-zero integer and $\mathbb{S}$ is regular. Hence representants of base plays form a basis of $\mathcal{A}(X)$.

If $\mathbb{S}$ is an idempotent semiring, then by Proposition 34 the family $(b_i)_{i\in I}$ is made of total orders, so if we set $b_i^* = b_i$ for each i we have the expected property.

Now suppose $\mathbb{S}$ is a rational ring. Let A be a representant finite subset of $|X|$. Call $a_1, \dots, a_n$ the subset of the basis whose plays have support A, and let $M = (m_{ij})$ be the $n \times n$ matrix such that $m_{ij} = \lfloor a_i \parallel a_j \rfloor$. M has coefficients in natural numbers, and since the family (a_i) is linearly independent by hypothesis, M is invertible in $\mathbb{Q}$. Since $\mathbb{S}$ is a regular ring, it is an algebra over $\mathbb{Q}$, so M is also invertible in $\mathbb{S}$. Let $M^{-1} = (m'_{ij})$ and let $a_i^* := \sum_{j=1}^{n} m'_{ij} a_i$, then by construction $\lfloor a_i \parallel a_j^* \rfloor$ is 1 if $i = j$ and 0 otherwise. □

Observe that if $\mathbb{S}$ is a rational ring, then in particular it is an algebra over $\mathbb{Q}$, then $\mathcal{A}_{\mathbb{S}}(X) = \mathbb{S} \otimes \mathcal{A}_{\mathbb{Q}}(X)$ as $\mathbb{Q}$-algebras, since all plays decompose uniquely as linear combinations of base plays with integer coefficients. The outcome in $\mathcal{A}_{\mathbb{S}}(X)$ then appears as the tensor of the identity over $\mathbb{S}$ and the outcome over $\mathcal{A}_{\mathbb{Q}}(X)$. The algebra $\mathcal{A}_{\mathbb{Q}}(X)$ further decomposes into the direct sum of the strict order algebras $\mathcal{A}^s_{\mathbb{Q}}(Y)$ for all representant subset Y with the permutation group induced over it. This is particularly useful since the $\mathcal{A}^s_{\mathbb{Q}}(Y)$ are finite dimensional vector spaces over $\mathbb{Q}$.

On the other hand, if $\mathbb{S}$ is neither idempotent nor a regular ring, it is possible that there is no base. For instance, if $\mathbb{S} = \mathbb{N}$, then clearly a play s cannot be decomposed as a non-trivial sum of vectors, so any generating family must contain all plays, but then the equation of proposition 35 states that they are not linearly independent.

3 Logical structure

In this section, we describe constructions on order algebras. Although order algebras themselves have some interesting structure, the actual objects we are interested in are submodules of such algebras, hereafter called types, which enjoy better properties.

42 **Definition.** A *type* over an arena X is a submodule of $\mathcal{A}(X)$ generated by a family of plays in $\mathcal{S}(X)$. A type is *strict* if it does not contain the empty play. The notation $A : X$ is used to represent the fact that A is a type over X. A *morphism* between types $A : X$ and $B : Y$ is a linear map f from $\mathcal{A}(X)$ to $\mathcal{A}(Y)$ such that $f(A) \subset B$.

The requirement that types are generated by plays is justified by the idea that a type should be a constraint on the behaviours of processes, and that such a constraint should boil down to a constraint on the shape of plays that a process can exhibit. We could also define a type over X simply as a subset S of $\mathcal{S}(X)$, but the definition as submodules makes it clear that observationally equivalent vectors should belong to the same types, even if one is a combination of plays in S while the other is not (this can happen even if S is closed under permutations, because of the equation of Proposition 35).

43 *Example.* The intended meaning of order algebras is that vectors, that is linear combinations of plays, represent processes. Then types impose constraints on the possible behaviours of processes, based on the possible interactions scenarii they may exhibit. For instance, we can define the type of processes that perform three actions of label a, as the submodule generated by the plays that contain three points in the orbit a. Similarly, we could define the type of all plays that include as many a's as b's.

Typed may also used in particular to impose well-formedness conditions. For instance, when modelling a calculus like π that includes communication of bound names, one wants to impose that any play that contains an event on a bound name also contains the event that communicates this name.

44 *Example.* Note that the condition of being a submodule of $\mathcal{A}(X)$ imposes non-trivial conditions. For instance, consider the CCS algebra, as of Example 3, with $\mathbb{S}$ idempotent. We can define the type of processes in which all actions a are causally independent of all actions b, as generated by the plays where all occurrences of a are incomparable with all occurrences of b. This type does not contain the processes $a.b$ and $b.a$, obviously, but it does contain their sum $a.b + b.a$, which is observationally equivalent to the parallel composition $a \mid b$.

45 **Proposition.** *If $\mathbb{S}$ is a rational ring, then for all type $A : X$ there is a family of plays $(c_i)_{i\in I}$ and a family of vectors $(c_i^*)_{i\in I}$ in $\mathcal{A}(X)$ such that $(c_i)_{i\in I}$ is a basis of A and for all $i, j \in I$, $\lfloor c_i \parallel c_j^* \rfloor$ is 1 if $i = j$ and 0 otherwise.*

Proof. Recall that if $\mathbb{S}$ is a rational ring, then it is an algebra over $\mathbb{Q}$ and $\mathcal{A}_{\mathbb{S}}(X)$ can be seen as the $\mathbb{Q}$-algebra $\mathbb{S} \otimes \mathcal{A}_{\mathbb{Q}}(X)$. Since A is generated by plays, we can then decompose it as $\mathbb{S} \otimes A'$ for a type A' in $\mathcal{A}_{\mathbb{Q}}(X)$, so it is enough to prove the result in the case $\mathbb{S} = \mathbb{Q}$. In this case A is a subspace of the vector space $\mathcal{A}_{\mathbb{Q}}(X)$, so it is a standard result that from the generating family we can extract a basis.

Now assume that $(c_i)_{i\in I}$ is a basis of A, and consider a particular base play c_n. Set $J := \{i \in I \mid |c_i| = |c_n|\}$. Then $(c_i)_{i\in J}$ is a basis of the intersection of A and $\mathcal{A}^s(|c_n|, \mathrm{G}^{|c_n|})$, the strict order algebra over $|c_n|$ with the induced permutation group, which is a finite-dimensional $\mathbb{Q}$-vector space. Let f be a linear form over this algebra such that $f(c_n) = 1$ and for all $j \in J \setminus \{i\}$, $f(c_j) = 0$. Using the bases (b_n) and (b_n^*) from Theorem 41 we can define $c_n^* = \sum_{|b_k|=|c_n|} f(b_k) b_k^*$ and check that for all vector x with $|x| = |c_n|$ we have $f(x) = \lfloor c_n^* \parallel x \rfloor$. Then c_n^* satisfies the expected condition. ∎

3.1 Products and linear maps

When combining order algebras, we need a notion of combination of arenas. Disjoint union is the simplest way, and also the most sensible one:

46 **Definition.** Let $(X_i)_{i\in I}$ be a family of arenas with pairwise disjoint webs. Define the sum of the family (X_i) as

$$\sum_{i\in I} X_i := \Big(\biguplus_{i\in I} X_i, \prod_{i\in I} \mathrm{G}^{X_i}\Big) \quad \text{with} \quad \sigma \cdot x := \sigma_j \cdot x \quad \text{for all } \sigma \in \prod_{i\in I} \mathrm{G}^{X_i} \text{ and } x \in |X_j|.$$

We use equivalently the infix notation $X_1 + X_2 + \cdots + X_n$ for finite sums.

47 *Example.* Following on our first examples, if X_A and X_B are the arenas used for modelling CSP processes over alphabets A and B respectively (see Example 2), then assuming A and B are disjoint the webs $|X_A|$ and $|X_B|$ are disjoint too and $X_A + X_B$ is actually the arena for processes in the alphabet $A \uplus B$.

This sum can be seen as a coproduct in a suitable category of arenas. At the level of order algebras, however, this operation is not a Cartesian product or coproduct, and not even a tensor product in the sense of $\mathbb{S}$-algebras, because the algebra $\mathcal{A}(X+Y)$ contains more plays than those that appear as disjoint unions of a play in $|X|$ and one in $|Y|$. However, $\mathcal{A}(X+Y)$ contains products and tensors as submodules, hence our definition of types. In all statements below, unless explicitly stated, different arenas are always supposed to be disjoint.

48 **Proposition.** *For all types $A : X$ and $B : Y$, $A + B$ is a type over $X + Y$ that is isomorphic to the direct sum and Cartesian product of A and B.*

Proof. A is a submodule of $\mathcal{A}(X)$, which is itself obviously a submodule of $\mathcal{A}(X+Y)$. Similarly, B is a submodule of $\mathcal{A}(X+Y)$, and since X and Y are disjoint, so are A and B, since no permutation in $X+Y$ can map a point of $|X|$ to a point of $|Y|$. Hence the submodule generated by $A + B = \langle A \cup B\rangle$ in $\mathcal{A}(X+Y)$ is a direct sum of A and B. □

49 **Definition.** Let X be an arena, let Y be a subset of $|X|$ closed under permutations in G^X. Restriction to Y is the linear map res_Y over $\mathcal{C}(X)$ such that for all $r \in \mathcal{S}(X)$,

$$\mathrm{res}_Y\, r := \lfloor r \rfloor \cdot (|r| \cap Y, \leqslant_r \cap Y^2)$$

Restriction of a play r to a given subset $Y \subset |X|$ amounts to ignore the part of r that happens outside Y, considering that events in $|X| \setminus Y$ are private, hence unobservable. The fact that Y must be closed under permutations is in accordance with the intuition that two plays are indistinguishable when they are permutations of each other.

50 *Example.* Following on example 47, $\mathrm{res}_{X_A} : \mathcal{C}(X_A + X_B) \to \mathcal{C}(X_A)$ precisely represents CSP's restriction operator that maps a trace t to the restricted trace $t \restriction A$.

Hence, as we shall see (in detail in Section 4.3), restriction does *not* correspond to the hiding operator $(\boldsymbol{\nu})$ of the π-calculus and related languages. Indeed, the externally observable behaviours of $(\boldsymbol{\nu}u)P$ are those of P that do not involve an event on u, which to mapping to 0 all plays in P that contain an event on u, before actually restricting to the arena that does not contain events on u.

51 Note that in the definition we impose a coefficient $\lfloor r \rfloor$ on the restricted play. Since the outcome $\lfloor r \rfloor$ is 0 or 1 for any play r, this amounts to imposing that $\mathrm{res}_Y\, r$ be 0 if r is inconsistent. This condition is necessary because we want outcomes to be preserved by restriction: if a play is inconsistent, it means that it contains some deadlock, and hiding the place where this occurs

surely should not resolve the deadlock. For instance, an inconsistent play like ($a \bullet\!\!\rightleftarrows\!\!\bullet\, b$), when restricted to $\{a\}$, would yield the consistent play ($a\, \bullet$).

Incidentally, this implies that $\operatorname{res}_X : \mathcal{C}(X) \to \mathcal{C}(X)$ is not the identity, because it collapses all inconsistent plays to 0. However, up to observational equivalence, it is the identity.

52 **Proposition.** *Restriction is compatible with observational equivalence.*

Proof. First observe that, since Y is supposed to be closed by permutations, restriction commutes with permutations, hence $\Delta(\operatorname{res}_Y u) = \operatorname{res}_Y \Delta(u)$. Then by the representation property (Proposition 29), if $u \approx v$ then $\Delta(u) \approx_{|X|} \Delta(v)$, so it suffices to prove that restriction is compatible with observational equivalence in static arenas.

Let Z be a finite subset of $|X| \setminus Y$, define ext_Z as the linear map such that for all $t \in \mathcal{S}(X)$, $\operatorname{ext}_Z t$ is the play on $|t| \cup Z$, whose preorder relation is $\leqslant_t$ extended as the identity relation on Z. Let r be a play in $\mathcal{S}(X)$ such that $|r| \setminus Y = Z$. Suppose r is acyclic, then for all play s with $|s| \cup Z = |r|$ the relation $\leqslant_r \cup \leqslant_s$ is acyclic if and only if $(\leqslant_r \cap Y^2) \cup \leqslant_s$ is acyclic, so we have $\lfloor r * \operatorname{ext}_Z s \rfloor = \lfloor \operatorname{res}_Y r * s \rfloor$. If r is not acyclic, then the equality holds too since both sides are 0. The equality extends trivially to all plays s such that $|s| \subset Y$.

Consider a pair $u \approx_{|X|} v$ and a play $s \in \mathcal{S}(X)$ with $|s| \subset Y$. By Proposition 32 we can decompose u as $\sum_{C \in \mathcal{P}_f(|X|)} u_C$ with $u_C \in \mathcal{C}^s(C)$ and similarly for v so that for each C we have $u_C \approx_{|X|} v_C$. For a given C, let $Z = C \setminus Y$, then by linearity of the equation in the previous paragraph we get $\lfloor \operatorname{res}_Y u_Z * s \rfloor = \lfloor u_Z * \operatorname{ext}_Z s \rfloor$ for all play s with $|s| \subset Y$, and by the equivalence of u_Z and v_Z we get $\lfloor \operatorname{res}_Y u_Z * s \rfloor = \lfloor \operatorname{res}_Y v_Z * s \rfloor$. This trivially holds too if $|s| \not\subset Z$, so we get the equivalence $\operatorname{res}_Y u_Z \approx \operatorname{res}_Y v_Z$, and we deduce $\operatorname{res}_Y u \approx \operatorname{res}_Y v$ by linearity. □

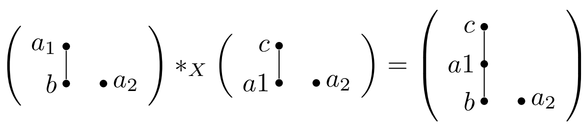
53 **Definition.** Let X, Y, Z be three arenas with pairwise disjoint supports. Define *partial static synchronisation* along X as the bilinear map $*_X$ from $\mathcal{C}(X + Y) \times \mathcal{C}(X + Z)$ to $\mathcal{C}(X + Y + Z)$ such that for all $r \in \mathcal{S}(X + Y)$ and $s \in \mathcal{S}(X + Z)$,

$$r *_X s := \begin{cases} \big(|r| \cup |s|, (\leqslant_r \cup \leqslant_s)^*\big) & \text{if } |r| \cap |X| = |s| \cap |X| \\ 0 & \text{otherwise} \end{cases}$$

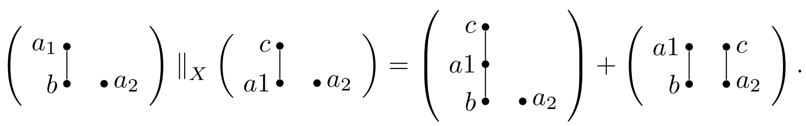
Deduce partial permuted synchronisation as

$$r \parallel_X s := \mu_X\,(\operatorname{res}_X s) \sum_{s' \in \mathrm{G}^X(s)} r *_X s'$$

54 *Example.* Consider an arena X containing at least two interchangeable actions labelled a_1, a_2 and arenas Y and Z containing events b and c respectively. Then we have the partial static synchronisation



and the partial permuted synchronisation



The factor $\mu_X\,(\operatorname{res}_X s)$ (which is 1 in this example) plays the same role as in full synchronisation (Definition 16), remarking that we only apply permutations on the X part.

55 **Proposition.** *Let X, Y, Z be three arenas with pairwise disjoint supports. Partial synchronisation along X is associative as*

$$(u \parallel_X v) \parallel_{X+Y+Z} w = u \parallel_{X+Y} (v \parallel_{X+Z} w)$$

for all $u \in \mathcal{C}(X+Y)$, $v \in \mathcal{C}(X+Z)$ and $w \in \mathcal{C}(X+Y+Z)$. It is compatible with observational equivalence and commutative up to equivalence.

Proof. Let X, Y, Z be three disjoint arenas. Consider three plays $r \in \mathcal{S}(X+Y)$, $s \in \mathcal{S}(X+Z)$ and $t \in \mathcal{S}(X+Y+Z)$. The partial synchronisations $(r *_X s) *_{X+Y+Z} t$ and $r *_{X+Y} (s *_{X+Z} t)$ are non-zero if and only if we can define

$$A = |r| \cap |X| = |s| \cap |X| = |t| \cap |X|, \quad B = |r| \cap |Y| = |t| \cap |Y|, \quad C = |s| \cap |Z| = |t| \cap |Z|,$$

and in this case the result is the play on $A \cup B \cup C$ whose preorder relation is $(\leqslant_r \cup \leqslant_s \cup \leqslant_t)^*$, so we have

$$(r *_X s) *_{X+Y+Z} t = r *_{X+Y} (s *_{X+Z} t).$$

Assume representants are chosen in each arena in such a way that for $D \subset |X|$ and $E \subset |Y|$, $\underline{D \cup E} = \underline{D} \cup \underline{E}$, and similarly for $X+Z$ and $X+Y+Z$. Choosing representants this way is always possible since permutations of X, Y and Z are independent in the sum arenas. Suppose r, s, t and A, B, C are representants. G^A is the same in all sums of arenas that involve X, and similarly for G^B and G^C. Moreover we have $\mathrm{G}^{A \cup B \cup C} = \mathrm{G}^A \times \mathrm{G}^B \times \mathrm{G}^C$, and similarly for other unions, so by similar considerations as for permuted synchronisation, we get

$$\begin{aligned}(r \parallel_X s) \parallel_{X+Y+Z} t &= \sum (r *_X \sigma_A s) *_{X+Y+Z} \sigma'_A \sigma_B \sigma_C t = \sum r *_{X+Y} (\sigma_A s *_{X+Z} \sigma'_A \sigma_B \sigma_C t) \\ &= \sum r *_{X+Y} \sigma_A \sigma_B (s *_{X+Z} \sigma'_A \sigma_C t) = r \parallel_{X+Y} (s \parallel_{X+Z} t)\end{aligned}$$

where the sums are indexed on $(\sigma_A, \sigma'_A, \sigma_B, \sigma_C) \in \mathrm{G}^A \times \mathrm{G}^A \times \mathrm{G}^B \times \mathrm{G}^C$. Partial synchronisation commutes with permutations so this equality extends to plays that are not representants, and by linearity is extends to arbitrary vectors.

Now consider $u, u' \in \mathcal{C}(X+Y)$, $v \in \mathcal{C}(X+Z)$ and $w \in \mathcal{C}(X+Y+Z)$, and suppose $u \approx u'$. By the same arguments as in the proof of Proposition 52, we get the equality $\lfloor u \parallel_{X+Y} (v \parallel_{X+Z} w) \rfloor = \lfloor u \parallel \mathrm{res}_{X+Y}(v \parallel_{X+Z} w) \rfloor$, then $u \approx u'$ implies that these are equal to $\lfloor u' \parallel_{X+Y} (v \parallel_{X+Z} w) \rfloor$, and applying associativity on this we can deduce $u \parallel_X v \approx u' \parallel_X v$. Commutativity of partial synchronisation is obvious, and it yields the compatibility with observational equivalence on the right. ☐

By similar arguments, we prove other "localized" associativities, the general case being

$$(u \parallel_{A+B} v) \parallel_{A+C+D} w = u \parallel_{A+B+C} (v \parallel_{A+D} w)$$

for $u \in \mathcal{A}(A+B+C+E)$, $v \in \mathcal{A}(A+B+D+F)$ and $w \in \mathcal{A}(A+C+D+G)$, where A, B, C, D, E, F, G are seven (!) pairwise disjoint arenas. Although this formulation is frighteningly heavy, the point is rather simple: when partially synchronising two vectors u and v, synchronise them along the arenas they have in common, and the result will be on the union of the arenas of u and v.

The simplest case of partial synchronisation is when "synchronising" two vectors $u \in \mathcal{A}(X)$ and $v \in \mathcal{A}(Y)$ along the empty arena, yielding $u \parallel_\emptyset v \in \mathcal{A}(X+Y)$. In this case, u and v are essentially kept independent, which in particular implies

$$\lfloor u \parallel_\emptyset v \rfloor = \lfloor u \rfloor \lfloor v \rfloor .$$

This is deduced by linearity from the case of plays, remarking that for $r \in \mathcal{S}(X)$ and $s \in \mathcal{S}(Y)$, $r \parallel_\emptyset s$ is the disjoint union of r and s, which is consistent if and only if r and s are consistent.

56 **Definition.** Let X and Y be two arenas. Define the bilinear map $\otimes$ from $\mathcal{A}(X) \times \mathcal{A}(Y)$ to $\mathcal{A}(X+Y)$ as $u \otimes v := u \parallel_\emptyset v$. For two types $A : X$ and $B : Y$, define $A \otimes B$ as the submodule of $\mathcal{A}(X+Y)$ generated by the image of $A \times B$ by $\otimes$.

Simply put, $A \otimes B$ is the $\mathbb{S}$-module consisting of processes that can be written as juxtapositions of a process in A and a process in B with no scheduling constraint between them, or as a sums of such things, *up to observational equivalence*. As illustrated in Example 44, this does not imply that any vector $\sum_{i\in I} \lambda_i r_i \in A \otimes B$ is syntactically a sum of $u_i \parallel_\emptyset v_i$ with $u_i \in A$ and $v_i \in B$.

57 **Proposition.** *If $\mathbb{S}$ is a rational ring, then for all types $A : X$ and $B : Y$, $A \otimes B$ is the tensor product of A and B in the sense of $\mathbb{S}$-algebras.*

Proof. By Proposition 45 the types A and B have bases $(b_i)_{i\in I}$ and $(c_j)_{j\in J}$, and there are families of vectors $(b_i^*)_{i\in I}$ and $(c_j^*)_{j\in J}$ such that each b_n^* identifies b_n among the elements of $(b_i)_{i\in I}$, and similarly for c_n^*. We prove that the vectors $b_i \parallel_\emptyset c_j$ are linearly independent. Consider a linear combination $u = \sum_{(i,j)\in I\times J} \lambda_{ij}(b_i \parallel c_j)$ in $\mathcal{A}(X+Y)$. For each $(m,n) \in I \times J$ we have

$$\begin{aligned}\left\lfloor u \parallel_{X+Y} (b_m^* \parallel_\emptyset c_n^*)\right\rfloor &= \sum_{(i,j)\in I\times J} \lambda_{mn} \left\lfloor (b_i \parallel_\emptyset c_j) \parallel_{X+Y} (b_m^* \parallel_\emptyset c_n^*)\right\rfloor \\ &= \sum_{(i,j)\in I\times J} \lambda_{mn} \left\lfloor (b_i \parallel_X b_m^*) \parallel_\emptyset (c_j \parallel_Y c_n^*)\right\rfloor = \sum_{(i,j)\in I\times J} \lambda_{mn} \lfloor b_i \parallel_X b_m^* \rfloor \lfloor c_j \parallel_Y c_n^* \rfloor\end{aligned}$$

using the associativity properties stated above. By definition of b_m^* and c_n^*, the only non-zero term in the final sum is for $(i,j) = (m,n)$, and this term is λ_{mn}. Applying this on every (m,n) implies the unicity of the decomposition of u on the $b_m \otimes c_n$. So the $b_m \otimes c_n$ form a linearly independent family, which proves that $A \otimes B$ is isomorphic to the tensor product of A and B, as $\mathbb{S}$-modules. The associativity property ensures that they are also isomorphic as $\mathbb{S}$-algebras. ☐

58 **Definition.** Let X, Y and Z be three arenas. Let $u \in \mathcal{A}(X+Y)$ and $v \in \mathcal{A}(Y+Z)$. Composition of u and v through Y is the vector $u \circ_Y v := \mathrm{res}_{X+Z}(u \parallel_Y v) \in \mathcal{A}(X+Z)$.

Let $A : X$ and $B : Y$ be two types. The type $A \multimap B : X + Y$ is the submodule of $\mathcal{A}(X+Y)$ generated by all plays r such that for all $u \in A$, $r \circ_X u \in B$.

By the remarks above, we get associativity of composition. In the special case where X is the empty arena, $\mathcal{A}(X+Y)$ is equal to $\mathcal{A}(Y)$ and $u \circ_Y v$ is a vector in Z, so v induces a linear map from $\mathcal{A}(Y)$ to $\mathcal{A}(Z)$. However, this mapping from vectors of $\mathcal{A}(X+Y)$ to linear maps from $\mathcal{A}(X)$ to $\mathcal{A}(Y)$ is neither injective nor surjective.

It is easy to check the standard adjunction $A \multimap (B \multimap C) = (A \otimes B) \multimap C$ for all types A, B, C of pairwise disjoint supports. Moreover, if we call $\mathbf{1}$ the non-trivial type over the empty arena, which is isomorphic to $\mathbb{S}$, we have for all type A that $A \otimes \mathbf{1} = \mathbf{1} \otimes A = \mathbf{1} \multimap A = A$.

3.2 Bialgebraic structure

59 **Definition.** Let X and Y be two arenas. Define the *indexing* of Y by X as the arena

$$X \triangleright Y := \left(|X| \times |Y|, \mathrm{G}^X \times (\mathrm{G}^Y)^{|X|}\right)$$

where permutations act as

$$(\sigma, \varphi)(x,y) := (\sigma x, \varphi(x)\, y)$$

We interpret indexing as follows: $|X \triangleright Y|$ consists of copies of Y indexed by points of X. A permutation in $X \triangleright Y$ consists in permuting each copy independently, using the function $\varphi : |X| \to \mathrm{G}^Y$ that provides a permutation for each copy, and then permuting the copies themselves using a permutation in X.

Note that we easily get the equality $(X+Y) \triangleright Z = (X \triangleright Z) + (Y \triangleright Z)$, however $X \triangleright (Y+Z)$ is not equal to $(X \triangleright Y) + (X \triangleright Z)$, since permutations of copies in the former operate the same way on the copies of X and those of Y, while in the latter they may not. There is also an isomorphism between $(X \triangleright Y) \triangleright Z$ and $X \triangleright (Y \triangleright Z)$, and these appear as $(|X| \times |Y| \times |Z|, \mathrm{G}^X \times (\mathrm{G}^Y)^{|X|} \times (\mathrm{G}^Z)^{|X| \times |Y|})$.

The structure of the indexing arena is used only for identifying and permuting copies, in particular we will not consider plays on this arena. The primary purpose of indexing is to build an arena in which the symmetric algebra over a given type will fit. It also generalises the direct sum when the indexing arena is static.

60 **Definition.** Let $\mathfrak{S}(\mathbb{N})$ be the arena with $|\mathfrak{S}(\mathbb{N})| = \mathbb{N}$, the set of natural numbers, and $\mathrm{G}^{\mathfrak{S}(\mathbb{N})} = \mathfrak{S}(\mathbb{N})$, the group of all permutations of $\mathbb{N}$. For all arena X, define $\sharp X := \mathfrak{S}(\mathbb{N}) \triangleright X$.

So the arena $\mathfrak{S}(\mathbb{N}) \triangleright X$ contains a countable number of interchangeable copies of X. Another useful construct is the following: identifying each integer n with the set $\{0, \ldots, n-1\}$, which is in turn identified with the static arena with this set as the web, the arena $n \triangleright X$ is isomorphic to the sum $X + \cdots + X$ with n independent copies of X. If $n = 0$, this yields the empty arena $\emptyset$. Then $n \triangleright \sharp X = n \triangleright \mathfrak{S}(\mathbb{N}) \triangleright X$ contains a countable set of copies of X, partitioned into n countable classes of interchangeable copies.

61 **Definition.** Let n be a strictly positive integer, let φ be a bijection from $n \times \mathbb{N}$ to $\mathbb{N}$. For all arena X, define the function $\gamma^n_\varphi : \mathcal{S}(n \triangleright \sharp X) \to \mathcal{S}(\sharp X)$ as

$$\left|\gamma^n_\varphi s\right| := \left\{(\varphi(i), x) \;\middle|\; (i,x) \in |s|\right\} \quad \text{and} \quad (\varphi(i), x) \leqslant_{\gamma^n_\varphi s} (\varphi(j), y) \text{ iff } (i,x) \leqslant_s (j,y).$$

Define the linear map $\delta^n : \mathcal{C}(\sharp X) \to \mathcal{C}(n \triangleright \sharp X)$ as

$$\delta^n s := \sum_{c \,:\, \pi_1(|s|) \to n} c \bullet s \quad \text{with} \quad \begin{array}{l} |c \bullet s| := \left\{((c(i), i), x) \;\middle|\; (i,x) \in |s|, i \in A\right\} \\ ((c(i), i), x) \leqslant_{c \bullet s} ((c(j), j), y) \text{ iff } (i,x) \leqslant_s (j,y) \end{array}$$

where π_1 is the first projection, so $\pi_1(|s|) = \{i \mid (i,x) \in |s|\}$.

The function γ^n_φ is a simple renaming of the copies of X using the function φ, which extends the bijection $\varphi : n \times \mathbb{N} \to \mathbb{N}$ to a bijection between $\mathcal{S}(n \triangleright \sharp X)$ and $\mathcal{S}(\sharp X)$. As explained below, this bijection is compatible with observational equivalence, but its quotient is not injective. Instead, it fuses the n independent copies of $\sharp X$ into one, which makes events from different copies interchangeable.

The linear map δ^n acts as a non-deterministic inverse operation. Given a play s in $\mathcal{S}(\sharp X)$, it enumerates all possible ways of partitioning the events of s into n identified subsets. The function c represents such a choice, and $c \bullet r$ applies this choice to the play r.

As we shall see, the operators γ^n_φ and δ^n are very similar to a multiplication and comultiplication in a bialgebra. They are analogous to concatenation and deconcatenation, which give a bialgebraic structure to tensor algebras [24].

62 **Proposition.** *Let X be an arena and let n be a strictly positive integer. The maps γ^n_φ and δ are compatible with observational equivalence and the quotient map of γ^n_φ is independent of φ. For all vectors $u \in \mathcal{A}(n \triangleright \sharp X)$ and $v \in \mathcal{A}(\sharp X)$ we have*

$$\gamma^n(u) \parallel_{\sharp X} v \approx_{\sharp X} u \parallel_{n \triangleright \sharp X} \delta^n(v).$$

Proof. Observe that for any permutation $\sigma \in \mathrm{G}^{n \triangleright \mathbb{N} \triangleright \sharp X}$ (i.e. a family of independent permutations on each copy of X in $n \triangleright \sharp X$) there is a permutation $\sigma' \in \mathrm{G}^{\mathbb{N} \triangleright \sharp X}$ such that $\gamma^n_\varphi \circ \sigma = \sigma' \circ \gamma^n_\varphi$, and the other way around for δ^n. As a consequence, by Proposition 29, we can deduce the expected result from the case where X is static. Then all considered permutations are in $\mathfrak{S}(n \times \mathbb{N})$ and $\mathfrak{S}(\mathbb{N})$.

The map γ^n_φ decomposes as the injection of $\mathcal{C}((n \triangleright \mathfrak{S}(\mathbb{N})) \triangleright X)$ into $\mathcal{C}(\mathfrak{S}(n \times \mathbb{N}) \triangleright X)$ and the renaming of $\mathfrak{S}(n \times \mathbb{N}) \triangleright X$ into $\mathfrak{S}(\mathbb{N}) \triangleright X$ through φ. The former consists in growing the permutation group on a fixed web and the latter is an isomorphism, so both are compatible with observational equivalence. For δ^n, given a permutation $\sigma \in \mathfrak{S}(\mathbb{N})$ and a play s, for all choice function c for σs we have $c \bullet \sigma s = \sigma'(c' \bullet s)$ with $c' = c \circ \sigma$ and $\sigma'(i,j) = (i, \sigma(j))$, which establishes a bijection between the choices of s and those of σs. From this we can conclude that δ^n is compatible with observational equivalence.

Let $r \in \mathcal{S}(n \triangleright \sharp X)$, let $s \in \mathcal{S}(\sharp X)$ and let φ be a bijection from $n \times \mathbb{N}$ to $\mathbb{N}$. Assume s is a representant. First suppose $\underline{\left|\gamma^n_\varphi(r)\right|} \neq |s|$, then $\gamma^n(r) \parallel s$ is zero. Suppose that there is a choice c such that $\overline{|c \bullet s|} = |r|$, then we get a permutation $\sigma \in \mathrm{G}^{n \triangleright \sharp X}$ that induces a bijection from $|c \bullet s|$ to $|r|$. By definition σ is a bijection between the pairs $(c(i), i)$ and $\pi_1(|r|)$, which can be extended into a bijection ψ from $n \times \mathbb{N}$ to $\mathbb{N}$, such that $\underline{\left|\gamma^n_\varphi(r)\right|} = |s|$. This contradicts the hypothesis $\underline{\left|\gamma^n_\varphi(r)\right|} \neq |s|$ since $\gamma^n_\varphi(r)$ and $\gamma^n_\psi(r)$ are necessarily permutations of each other, from the remarks above. Hence for all c we have $\underline{|c \bullet s|} \neq |r|$, so $r \parallel \delta^n(s) = 0$, and the equality holds.

Now suppose $\underline{\left|\gamma^n_\varphi(r)\right|} = |s|$. Applying a suitable permutation to r and choosing φ appropriately (we know from the above remarks that these operations are allowed) we can assume that $\gamma^n_\varphi(r)$ is a representant, so $\left|\gamma^n_\varphi(r)\right| = |s|$, and $\gamma^n_\varphi(r) \parallel s = \sum_{\sigma \in G} \varphi r * \sigma s \approx \sum_{\sigma \in G} r * \varphi^{-1} \sigma s$, where G is the group of permutations of $|s|$ induced by $\mathrm{G}^{\sharp X}$, that is the symmetric group of $\pi_1(|s|)$. For each $\sigma \in G$, the function $\pi_1 \varphi^{-1} \sigma$ is a choice function c_σ over $\pi_1(|s|)$, and $\sigma^*(i,j) := (i, \sigma^{-1}\varphi(i,j))$ is a permutation in $\mathrm{G}^{n \triangleright \mathbb{N}}$ such that $\sigma^* \varphi^{-1} \sigma(i) = (c_\tau(i), i)$, hence $\sigma^* \varphi^{-1} \sigma(s) = c_\sigma \bullet s$. By partitioning the sum for $\tau \in G$ according to choice functions, we get $\gamma^n_\varphi(r) \parallel s \approx \sum_c \sum_{\sigma \in G, c_\sigma = c} r * \sigma^{*-1}(c_\sigma \bullet s)$. By construction, for a fixed c, we have $\left\{\sigma'^{*-1} \;\middle|\; \sigma' \in G, c_{\sigma'} = c\right\} = \mathrm{G}^{n \triangleright \mathbb{N}} \sigma^{*-1}$, so we get $\gamma^n_\varphi(r) \parallel s \approx_{\sharp X} \sum_c r \parallel_{n \triangleright \sharp X} (c \bullet s) = r \parallel_{n \triangleright \sharp X} \delta^n(s)$, from which we conclude by linearity. ☐

As a consequence, $\mathcal{A}(\sharp X)$ has the structure of a commutative algebra with γ^2 as the multiplication and the empty play as the unit. The δ^2 does not make it a bialgebra in general, because for an arbitrary $u \in \mathcal{A}(\sharp X)$, $\delta^2(u) \in \mathcal{A}(\sharp X + \sharp X)$ has no reason to be in the tensor product $\mathcal{A}(\sharp X) \otimes \mathcal{A}(\sharp X)$. The reason is that a given play in $\mathcal{S}(\sharp X)$, the components of $\delta^2(r)$ are not disjoint unions of plays on the two copies of $\sharp X$, but they may contain scheduling constraints that involve both copies. We do get a bialgebra if we restrict to the case of plays in which all copies stay independent.

63 **Definition.** Let X be an arena. For all integer n and play $r \in \mathcal{S}(X)$, define the play $n \bullet r \in \mathcal{S}(\sharp X)$ as in Definition 61 for the constant function n. This obviously induces an isomorphism between $\mathcal{A}(X)$ and $\mathcal{A}(\{n\} \triangleright X)$, which maps each type $A : X$ to an isomorphic type $n \bullet A : \{n\} \triangleright X$. However, the $\{n\} \triangleright X$ for distinct n are disjoint.

The arena $\{n\} \triangleright X$ is included in $\sharp X$, let $\varepsilon_n : \mathcal{C}(X) \to \mathcal{C}(\sharp X)$ be the inclusion map. Clearly all the ε_n are compatible with observational equivalence and their quotients are all equal. Name $\varepsilon : \mathcal{A}(X) \to \mathcal{A}(\sharp X)$ the quotient map.

For all type $A : X$, define the type $!A : \sharp X$ as

$$!A := \sum_{n \in \mathbb{N}} \gamma^n(A^n) \quad \text{where} \quad (A^n : n \triangleright X) := \bigotimes_{i=0}^{n-1} (i \bullet \varepsilon(A))$$

Define the degree of a vector $u \in\, !A$ as the smallest integer $d(u)$ such that u is in the partial sum $\sum_{n \leqslant d(u)} \gamma^n(A^n)$.

For all type $A : X$, the type $!A : \sharp X$ is again a commutative algebra with γ^2 as the product and the empty play as the unit. The degree function makes it a graded algebra, intuitively the degree of a vector u is the maximum number of different copies of A that u uses. If the type A is *strict* (i.e. if it does not contain the empty play) and $\mathbb{S}$ is rational, then $!A$ is isomorphic to the symmetric algebra of A. The strictness condition means that each copy of A is actually used, without this hypothesis the isomorphism fails because all powers of the empty play are necessarily equal to the empty play in $!A$.

The linear map δ^2 also makes $!A$ a cocommutative coalgebra whose counit is the linear form that maps the empty play to 1 and non-empty plays to 0. It is routine to check that the algebra and coalgebra structure are compatible, making $!A$ a bialgebra. Interestingly, if A is the unique strict type on the singleton arena (which is isomorphic to $\mathbb{S}$), then $!A$ is isomorphic to the bialgebra of polynomials in one variable over $\mathbb{S}$.

3.3 Towards differential linear logic

The mapping $A \mapsto\, !A$ is a functor in the category of types and linear maps. The map ε from the definition above is a natural transformation from A to $!A$, and by choosing a bijection from $\mathbb{N} \times \mathbb{N}$ into $\mathbb{N}$ we get a natural transformation from $!!A$ to $!A$ which makes $!A$ into a monad (the choice of a particular bijection is unimportant, for the same reasons as in Proposition 62). The quotient of the linear map that sends each play $n \bullet r$ to r and all other plays to zero is a natural transformation from $!A$ to A, and using any bijection from $\mathbb{N}$ to $\mathbb{N} \times \mathbb{N}$ we get a natural transformation from $!A$ to $!!A$, which also makes A a comonad.

We can also check the isomorphism $!(A \oplus B) \simeq\, !A \otimes !B$ for any strict types $A : X$ and $B : Y$ over disjoint arenas. The first type is in the arena $\mathfrak{S}(\mathbb{N}) \triangleright (X + Y)$ and the second one is in $(\mathfrak{S}(\mathbb{N}) \triangleright X) + (\mathfrak{S}(\mathbb{N}) \triangleright Y)$; these arenas are not isomorphic but the types themselves are thanks to the definition of the direct sum.

All these considerations show that the structure of our types supports most constructs of differential linear logic [17], including additives, multiplicatives and exponentials with structural and costructural rules. However, the construction is not yet a model of differential logic, for several reasons:

- One crucial thing that lacks in our framework is the axioms. They do not fit in the present work because our objects are too finitary: all vectors are finite linear combinations of finite plays, hence there can be no vector in $A \multimap A$ that is neutral for composition as soon as A is not finite dimensional. The reason is similar to the case of units for synchronisation in Proposition 30, and solving this defect requires a radical extension of this work, as explained in the introduction.

- The proper notion of duality needed to interpret logic, or equivalently the definition of the type $\perp$, is not clear at first sight. This type must be defined on the empty arena, and our notion of type only leaves two choices: $\mathbf{1} = \mathcal{A}(\emptyset)$ and $\{0\}$. The first one is degenerate given our definition of $A \multimap B$, the second one yields orthogonality with respect to the bilinear form $(u, v) \mapsto \lfloor u \parallel v \rfloor$ (note however that this bilinear form is not a scalar product, because it is not positive). We will not explore this case here because it exceeds the scope of the present work.

- Of course, building a model of linear logic requires to prove that the interpretation of proofs is preserved by cut-elimination. Most tools are present for this, assuming we restrict to an ill-structured logical system without the axiom rule. Here again, we defer this task to further work, as the questions of axioms and duality obviously have to be answered first for this to be of interest.

4 Interpretation of process calculi

In this section, we detail how process calculi can be interpreted in order algebras. As a particular case to work on, we use the π-calculus with internal mobility [32], that is the fragment of the π-calculus where output actions can only send fresh names. Most development here could be carried out in other similar calculi. Had we used CCS, essentially everything would have been the same up to section 4.3, in which the definition of arenas would have been simpler because of the mostly trivial name structure of CCS. The full π-calculus, on the other hand, would have required the handling of equality tests between names, which is perfectly doable at the cost of trickier definitions; this exceeds the scope of the present work.

4.1 Quantitative testing

We consider the π-calculus with internal mobility, or πI-calculus, extended with *outcomes* from a commutative semiring $\mathbb{S}$. We consider the monadic variant of the calculus for simplicity, but using the polyadic form would not pose any significant problem. More importantly, we restrict to finite processes.

64 **Definition.** We assume a countable set $\boldsymbol{N}$ of names. Polarities are elements of $\boldsymbol{P} = \{\downarrow, \uparrow\}$. Terms are generated by the following grammar:

branchings	$S, T :=$	$u^\varepsilon_\iota(x).P$	action, with $u, x \in \boldsymbol{N}$, $\varepsilon \in \boldsymbol{P}$ and $\iota \in \mathbb{N}$
		$S + T$	external choice
processes	$P, Q :=$	λ	outcome, with $\lambda \in \mathbb{S}$
		S	branching
		$P \mid Q$	parallel composition
		$(\boldsymbol{\nu} x)P$	hiding, with $x \in \boldsymbol{N}$

In an action $u^\varepsilon_\iota(x).P$, ι is the *location*, u is the *subject*, x is the *object* and P is the *continuation*. The name x is bound in P by the action, independently of the polarity ε.

Terms are considered up to injective renaming of bound names and commutation of restrictions, i.e. $(\boldsymbol{\nu} x)(\boldsymbol{\nu} y)P = (\boldsymbol{\nu} y)(\boldsymbol{\nu} x)P$, with the standard convention that all bound names are distinct from all other names. We also impose that in a given term all locations are always distinct. The set of locations occurring in a term P is written $|P|$.

Actions (without continuations) will be ranged over by Greek letters α, β, so that we can write expressions like $\alpha.P$ or $\alpha.(\beta.Q \mid R)$. By convention, an action $u^\downarrow(x)$ is called *positive* and is also written $u(x)$, an action $u^\uparrow(x)$ is called *negative* and is also written $\bar{u}(x)$. More generally, if α is an action, we write $\bar{\alpha}$ for the action with the same subject and the opposite polarity, in particular $\bar{u}^\varepsilon(x)$ is the action of the opposite polarity as $u^\varepsilon(x)$.

Locations are simply a way to give different identities to different occurrences of a given channel name in a term, so we can talk about "the action ι" in an unambiguous manner. Renamings of these locations are of course unobservable by the processes, so the distinctness condition is not a restrictions on the terms we can write. Terms with locations can be seen as decorations on standard terms of the πI-calculus.

We want to define an operational semantics in which commutation of independent transitions is allowed. To make this possible by only looking at transition labels, we enrich the labels

$$\frac{}{\alpha_\iota.P \xrightarrow{\alpha:\iota} P} \qquad \frac{P \xrightarrow{u^\varepsilon(x):\iota} P' \quad Q \xrightarrow{\bar{u}^\varepsilon(y):\kappa} Q'}{P \mid Q \xrightarrow{\{\iota,\kappa\}} (\boldsymbol{\nu}x)(P' \mid Q'[x/y])} \qquad \frac{P \xrightarrow{a} P' \quad x \notin a}{(\boldsymbol{\nu}x)P \xrightarrow{a} (\boldsymbol{\nu}x)P'}$$

$$\frac{S \xrightarrow{a} S'}{S + T \xrightarrow{a} S'} \qquad \frac{S \xrightarrow{a} S'}{T + S \xrightarrow{a} S'} \qquad \frac{P \xrightarrow{a} P'}{P \mid Q \xrightarrow{a} P' \mid Q} \qquad \frac{P \xrightarrow{a} P'}{Q \mid P \xrightarrow{a} Q \mid P'}$$

Table 1: Decorated labelled transition system for the πI-calculus

using locations so that different occurrences of a given action are distinguishable at the level of operational semantics.

65 **Definition.** Transition labels can be of one of two kinds:

$$\begin{array}{rl} a, b := u^\varepsilon(x){:}\iota & \text{visible action} \\ \{\iota, \kappa\} & \text{internal transition} \end{array}$$

Transitions are derived by the rules of Table 1. The notation $x \notin a$ means that the name x does not occur (free or bound) in the label a.

An interaction is a finite sequence of transition labels. A path is a finite sequence of internal transition labels. An interaction $p = a_1 a_2 \dots a_n$ is valid for P, written $p \in P$, if there are valid transitions $P \xrightarrow{a_1} P_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} P_n$.

The use of decorations to define a parallel operational semantics was first proposed by Boudol and Castellani as “proved transitions” [9, 12], and the technique we use here can be seen as a simplification for our purpose. It is clear that for all term P and interaction p, there is at most one term P/p (exactly one if $p \in P$) such that there is a transition sequence $P \xrightarrow{p} P/p$ (up to renaming of revealed bound names). Note that by removing all locations from labels (replacing $\{\iota, \kappa\}$ by τ) one gets the standard labeled transition system for the πI-calculus. For this reason, we allow ourselves to keep locations implicit when they are not important.

66 **Definition.** Prefixing in a term P is the partial order $\leqslant_P$ over $|P|$ such that $\iota <_P \kappa$ when in P the action at location κ occurs in the continuation of the action at location ι. Two labels a and b are *independent*, written $a \mathbin{/\!\!/_P} b$, if all locations occurring in a or b are distinct and pairwise incomparable for prefixing. Homotopy in a term P is the smallest equivalence $\approx_P$ over interactions of P such that $pabq \approx_P pbaq$ when $a \mathbin{/\!\!/_P} b$.

Two execution paths of a given term are homotopic if it is possible to transform one into the other by exchanging consecutive transitions if they are independent. Prefixing generates local constraints which propagate to paths by this relation.

67 **Proposition.** *Let p and q be two interactions of a term P such that p and q are reorderings of each other, then $p \approx_P q$ and $P/p = P/q$.*

Proof. We first prove that for any interaction $a_1 \dots a_n b \in P$ such that $b \in P$ we have $a_1 \dots a_n b \approx_P b a_1 \dots a_n$ and $P/(a_1 \dots a_n b) = P/(b a_1 \dots a_n)$, by induction on n. The case $n = 0$ is trivial. For the case $n \geqslant 1$, remark that the hypothesis implies $a_1 \mathbin{/\!\!/} b$: if some location in a_1 was less than a location in b then b could only occur after a_1, which contradicts $b \in P$, and $a_1 \in P$ also implies that no location in b is less than a location in a_1. Therefore we have $b a_1 \in P$ and $b a_1 \approx_P a_1 b$. The equality $P/a_1 b = P/b a_1$ is a simple check on the transition rules. Applying the induction hypothesis on P/a_1 yields $b a_2 \dots a_n \approx_P a_2 \dots a_n b$ and $P/a_1 b a_2 \dots a_n = P/b a_1 a_2 \dots a_n$ from which we conclude. The case of arbitrary reorderings follows by recurrence on the length of p and q. ∎

68 **Definition.** A pre-trace is a homotopy class of interactions. A run is a homotopy class of maximal paths. The sets of pre-traces and runs of a term P are written $\mathcal{P}(P)$ and $\mathcal{R}(P)$ respectively. The unique reduct of a term P by a pre-trace ρ is written P/ρ.

Runs are the intended operational semantics: they are complete executions of a given system, forgetting unimportant interleaving of actions and remembering only actual ordering constraints. A pre-trace can be seen as a Mazurkiewicz trace [14] on the infinite language of transition labels, with the independence relation from Definition 66, except that, because of our transition rules, each label occurs at most once in any interaction.

We now define a form of interactive observation, in the style of testing equivalences, that takes this notion of homotopy in account. Standard testing leads to interleaving semantics, so we have to refine our notion of test, and that is what outcomes are for. The set $\mathbb{S}$ is a semiring in order to represent two ways of combining results: multiplication is parallel composition of independent results and addition is combination of results from distinct runs.

69 **Definition.** The state $s(P) \in \mathbb{S}$ of a term P is defined inductively as

$$s(\lambda) := \lambda, \qquad s(S) := 1, \qquad s((\boldsymbol{\nu}x)P) := s(P), \qquad s(P \mid Q) := s(P)\, s(Q).$$

The outcome of a term P is $\lfloor P \rfloor = \sum_{\rho \in \mathcal{R}(P)} s(P/\rho)$. Two terms P and Q are observationally equivalent, written $P \simeq Q$, if $\lfloor P \mid R \rfloor = \lfloor Q \mid R \rfloor$ for all R.

In other words, the outcome of testing P against Q is the sum of the final states of all different runs of $P|Q$. Note that this sum is always finite since we only consider terms without replication or recursion, hence all terms have finitely many runs. Classic forms of test intuitively correspond to the case where $\mathbb{S}$ is the set of booleans for the two outcomes success and failure, with operations defined appropriately. This particular case is detailed at the end of Section 4.4.

4.2 Decomposition of processes

In this section, we prove several properties of terms up to observational equivalence. The purpose is to decompose arbitrary terms into simpler terms from which we will be able to easily extract a semantics in order algebras.

70 **Definition.** Let P be a term and let $\rho \in \mathcal{P}(P)$ be a pre-trace of P. By Proposition 67, ρ is identified with the set of its labels.

- The causal order in ρ is the partial order $\leqslant_\rho$ on labels in ρ such that $a \leqslant_\rho b$ if $a = b$ or a occurs before b in all interactions in ρ.
- The outcome of a pre-trace ρ is defined as $\lfloor \rho \rfloor := s(P/\rho)$.

This presentation is much simpler to handle than explicit sets of runs, so this is the one we will mainly use. Interactions that constitute a given pre-trace are simply the topological orderings of this partially ordered set of transitions. Traces in our sense are a further quotient of pre-traces, defined and studied in Section 4.3.

71 **Proposition.** *Observational equivalence is a congruence.*

Proof. Consider a family of equivalent processes $(P_i \simeq Q_i)_{1 \leqslant i \leqslant n}$, and let $(\alpha_i)_{1 \leqslant i \leqslant n}$ be a family of actions on fresh locations κ_i. Let $P = \sum_{i=1}^{n} \alpha_i.P_i$ and $Q = \sum_{i=1}^{n} \alpha_i.Q_i$, we prove $P \simeq Q$. Let R be an arbitrary process. The set $\mathcal{R}(P \mid R)$ can be split into $n+1$ parts: the set $\mathcal{R}_0$ of runs where no action α_i is triggered and the sets $\mathcal{R}_i$ of runs in which α_i is triggered, for each i. Then for each run $\rho \in \mathcal{R}_i$, there is a position ι such that $\{\kappa_i, \iota\} \in \rho$. Let ρ_1 be the partial run $\{a \mid a \in \rho, a \leqslant_\rho \{\kappa_i, \iota\}\}$, that is the minimal run that triggers α_i; we have $(P|R)/\rho_1 = (\boldsymbol{\nu}x)(P_i|R')$

commutativity	$P \mid Q \simeq Q \mid P$	$S + T \simeq T + S$
associativity	$(P \mid Q) \mid R \simeq P \mid (Q \mid R)$	$(S + T) + U \simeq S + (T + U)$
neutrality	$P \mid 1 \simeq P$	
scope commutation	$(\boldsymbol{\nu}x)(\boldsymbol{\nu}y)P \simeq (\boldsymbol{\nu}y)(\boldsymbol{\nu}x)P$	
scope extrusion	$(\boldsymbol{\nu}x)(P \mid Q) \simeq P \mid (\boldsymbol{\nu}x)Q$	with $x \notin \mathrm{fn}(P)$
scope neutrality	$(\boldsymbol{\nu}x)\lambda \simeq \lambda$	
inaction	$(\boldsymbol{\nu}u)u^{\varepsilon}(x).P \simeq 1$	
non-interference	$(\boldsymbol{\nu}u)(u(x).P \mid \bar{u}(x).Q) \simeq (\boldsymbol{\nu}ux)(P \mid Q)$	

Table 2: Basic equivalences.

for some x and R'; let $\rho_2 = \rho \setminus \rho_1$, so that ρ_2 is a run of $P_i \mid R'$ and $(P \mid R)/\rho = (\boldsymbol{\nu}x)(P_i \mid R')/\rho_2$. Let $\mathcal{S}_i$ be the set of triples (ρ_1, R', ρ_2) for all $\rho \in \mathcal{R}_i$. Obviously $\mathcal{R}(P \mid R)$ is in bijection with $\mathcal{R}_0 \uplus \biguplus_{i=1}^{n} \mathcal{S}_i$ and

$$\lfloor P \mid R \rfloor = \sum_{\rho \in \mathcal{R}_0} s(R/\rho) + \sum_{i=1}^{n} \sum_{(\rho_1, R', \rho_2) \in \mathcal{S}_i} s((P_i \mid R')/\rho_2)$$

Now let $\mathcal{L}_i = \{(\rho_1, R') \mid \exists \rho_2, (\rho_1, R', \rho_2) \in \mathcal{S}_i\}$, and let $(\rho_1, R') \in \mathcal{L}_i$. Since $\mathcal{R}_i$ contains all the runs of $P \mid R$ that trigger α_i, it contains all the runs of $P_i \mid R'$ since $P_i \mid R'$ can be reached from $P \mid R$, so we have $\{\rho_2 \mid (\rho_1, R', \rho_2) \in \mathcal{S}_i\} = \mathcal{R}(P_i \mid R')$, hence

$$\sum_{(\rho_1, R', \rho_2) \in \mathcal{S}_i} s((P_i \mid R')/\rho_2) = \sum_{(\rho_1, R') \in \mathcal{L}_i} \sum_{\rho_2 \in \mathcal{R}(P_i \mid R')} s((P_i \mid R')/\rho) = \sum_{(\rho_1, R') \in \mathcal{L}_i} \lfloor P_i \mid R' \rfloor$$

By hypothesis, for all R' we have $\lfloor P_i \mid R' \rfloor = \lfloor Q_i \mid R' \rfloor$ so

$$\lfloor P \mid R \rfloor = \sum_{\rho \in \mathcal{R}_0} s(R/\rho) + \sum_{i=1}^{n} \sum_{(\rho_1, R') \in \mathcal{L}_i} \lfloor Q_i \mid R' \rfloor = \lfloor Q \mid R \rfloor$$

since the reasoning above equally applies to Q. Therefore we get $P \simeq Q$.

For parallel composition, let R and S be arbitrary terms. It is clear that $(P \mid R) \mid S$ and $P \mid (R \mid S)$ have the same runs and that their reducts by a given run are the same up to the same associativity, so for all run ρ we have $s(((P \mid R) \mid S)/\rho) = s((P \mid (R \mid S))/\rho)$ and therefore $\lfloor (P \mid R) \mid S \rfloor = \lfloor P \mid (R \mid S) \rfloor$. Similarly we get $\lfloor (Q \mid R) \mid S \rfloor = \lfloor Q \mid (R \mid S) \rfloor$, and by hypothesis we have $P \simeq Q$ so $\lfloor P \mid (R \mid S) \rfloor = \lfloor Q \mid (R \mid S) \rfloor$, from which we conclude.

The equality $\lfloor (\boldsymbol{\nu}x)P \mid R \rfloor = \lfloor (\boldsymbol{\nu}x)Q \mid R \rfloor$ is justified by the fact that $\lfloor (\boldsymbol{\nu}x)P \mid R \rfloor$ and $\lfloor P \mid R \rfloor$ are equal if the name x is fresh with respect to R. □

72 **Proposition.** *The equivalences of Table 2 hold.*

Proof. For every equation $A \simeq B$ in the list except non-interference, it is clear that for all term T we have $\mathcal{R}(A|T) = \mathcal{R}(B|T)$ and that the reducts by any run ρ differ in the same way. Since these rules preserve states, in each case we get $\lfloor A \mid T \rfloor = \lfloor A \mid T \rfloor$, hence the expected equivalence. For the non-interference rule, remark that all runs of $(\boldsymbol{\nu}u)(u_\iota(x).P \,|\, \bar{u}_\kappa(x).Q) \,|\, R$ contain the transition $\{\iota, \kappa\}$, because of maximality and the fact that R cannot provide actions on u. The reduct by this transition is $(\boldsymbol{\nu}ux)(P \mid Q) \mid R$, and its runs are those of the original term without $\{\iota, \kappa\}$, so it has the same outcome. □

commutative monoid:	$P \oplus Q \simeq Q \oplus P \qquad (P \oplus Q) \oplus R \simeq P \oplus (Q \oplus R) \qquad P \oplus 0 \simeq P$
scalar multiplication:	$0 \cdot P \simeq 0 \qquad 1 \cdot P \simeq P \qquad \lambda_1\lambda_2 \cdot P \simeq \lambda_1 \cdot (\lambda_2 \cdot P)$ $(\lambda_1 + \lambda_2) \cdot P \simeq (\lambda_1 \cdot P) \oplus (\lambda_2 \cdot P) \qquad \lambda \cdot (P \oplus Q) \simeq \lambda \cdot P \oplus \lambda \cdot Q$
linearity of operators:	$P \mid (Q \oplus R) \simeq (P \mid Q) \oplus (P \mid R) \qquad P \mid (\lambda \cdot Q) \simeq \lambda \cdot (P \mid Q)$ $(\boldsymbol{\nu}x)(P \oplus Q) \simeq (\boldsymbol{\nu}x)P \oplus (\boldsymbol{\nu}x)Q \qquad (\boldsymbol{\nu}x)(\lambda \cdot P) \simeq \lambda \cdot (\boldsymbol{\nu}x)P$

Table 3: Module laws over processes.

Thanks to these properties, when considering processes up to observational equivalence, we can consider parallel composition to be associative and commutative. In this case we use the notation $\prod_{i \in I} P_i$ to denote the parallel composition without interaction of the P_i in any order (assuming only that I is finite).

In order to study processes up to observational equivalence, we will now describe some of the structure of the space of equivalence classes. The first ingredient is to identify an additive structure that represents pure non-determinism.

73 **Proposition.** *Let $\Pi_{\mathbb{S}}$ be the set of equivalence classes of processes over the semiring of outcomes $\mathbb{S}$. For all terms P and Q and all outcome λ, define*

$$\begin{aligned} P \oplus Q &:= (\boldsymbol{\nu}u)((u.P \mid u.Q) \mid \bar{u}.1) \quad \textit{where } u \textit{ is a fresh name,} \\ \lambda \cdot P &:= \lambda \mid P \end{aligned}$$

Then $(\Pi_{\mathbb{S}}, \oplus, 0, \cdot)$ is a $\mathbb{S}$-module, parallel compositions are bilinear operators and hiding is linear, i.e. the equivalences of Table 3 hold.

Proof. We first show that, for all terms P, Q and R, $\lfloor (P \oplus Q) \mid R \rfloor = \lfloor P \mid R \rfloor + \lfloor Q \mid R \rfloor$. Consider $\mathcal{R}((P \oplus Q) \mid R) = \mathcal{R}((\boldsymbol{\nu}u)((u_{\iota_1}.P \mid u_{\iota_2}.Q) \mid \bar{u}_\kappa.1) \mid R)$. It is clear that any run contains an interaction of $\bar{u}.1$ with either $u.P$ or $u.Q$, since none of these may interact with anything else. We can thus write $\mathcal{R}((P \oplus Q) \mid R) = \mathcal{R}_1 \uplus \mathcal{R}_2$ where $\mathcal{R}_1$ is the set of runs that contain (ι_1, κ) and $\mathcal{R}_2$ is the set of runs that contain (ι_2, κ). The runs in $\mathcal{R}_1$ are the runs of $(\boldsymbol{\nu}u)(u_{\iota_1}.P \mid \bar{u}) \mid R$ and each of these runs has the same outcome in both terms, so

$$\sum_{\rho \in \mathcal{R}_1} s\big(((P \oplus Q) \mid R)/\rho\big) = \lfloor (\boldsymbol{\nu}u)(u_{\iota_1}.P \mid \bar{u}) \mid R \rfloor = \lfloor P \mid R \rfloor$$

by the non-interference rule of Table 2. By a similar argument, we get the same for $\mathcal{R}_2$ and $\lfloor Q \mid R \rfloor$, so we finally get $\lfloor (P \oplus Q) \mid R \rfloor = \lfloor P \mid R \rfloor + \lfloor Q \mid R \rfloor$. This equality and the fact that $(\mathbb{S}, +, 0)$ is a commutative monoid implies that $(\Pi_{\mathbb{S}}, \oplus, 0)$ is a commutative monoid (where 0 is the atomic term with outcome 0).

For any terms P and Q and any outcome λ, it is clear that $\lfloor (\lambda \mid P) \mid Q \rfloor = \lambda \lfloor P \mid Q \rfloor$, since the term λ has no transition and contributes λ multiplicatively to all outcomes of the term. This directly implies that the operation $\lambda \cdot P$ has all required properties.

For the bilinearity of compositions, using the equation $\lfloor (P \oplus Q) \mid R \rfloor = \lfloor P \mid R \rfloor + \lfloor Q \mid R \rfloor$ and associativity and commutativity of parallel composition we get that parallel composition distributes over $\oplus$, and the fact that 0 is absorbing is equivalent to the rule $0 \cdot P \simeq 0$. Linearity of hiding is immediate from the scoping rules and the fact that $\lfloor (\boldsymbol{\nu}x)P \rfloor = \lfloor P \rfloor$ always holds. □

Observe that all syntactic constructions induce linear constructions on equivalence classes, except for the action prefix, which is not linear but actually affine. Indeed, for an action α, the term $\alpha.0$ is not equivalent to 0: it will be neutral in executions that do not trigger α, and

Linearity:	$\hat{\alpha}.(P \oplus Q) \simeq \hat{\alpha}.P \oplus \hat{\alpha}.Q \quad \hat{\alpha}.(\lambda \cdot P) \simeq \lambda \cdot \hat{\alpha}.P \quad (\boldsymbol{\nu} u)\hat{u}^{\varepsilon}(x).P \simeq 0$	
Asynchrony of inactions:	$\hat{\alpha}.(\beta.0 \mid P) \simeq \beta.0 \mid \hat{\alpha}.P$	if the subject of β is not bound by α
Composition of inactions:	$\sum_{i\in I} \alpha_i.0 \mid \sum_{i\in J} \alpha_i.0 \simeq 0$ if $\alpha_i = \bar{\alpha}_j$ for some $i \in I, j \in J$; $\sum_{i\in I\cup J} \alpha_i.0$ otherwise	
	$\alpha.0 + \alpha.0 \simeq \alpha.0$	

Table 4: Laws of linear actions and inactions.

multiply the outcome by 0 (thus annihilating it) in runs that do. It can be understood as a statement "I could have performed α but I will not do it" so that any run that contradicts this statement has outcome 0. The purely linear part of actions is the opposite: the linear action $\hat{\alpha}.P$ will act as $\alpha.P$ if its environment actually triggers the action, but will turn to 0 if it is never activated.

74 **Definition.** For all action α and term P, the linear action of α on P is

$$\hat{\alpha}.P := (\boldsymbol{\nu} w)(\alpha.(P \mid w.1) \mid w.0 \mid \bar{w}.1) \quad \text{where } w \text{ is a fresh name.}$$

An interaction is said to trigger the linear action if it triggers the action $w.1$. Terms of the form $\alpha.0$ are called inactions.

This definition has the expected behaviour because of the maximality of runs. If $\hat{\alpha}.P$ is in active position, then any run that does not trigger α must instead trigger $w.0$, hence any such run has outcome 0. A run in which the term $\hat{\alpha}.P$ does not produce 0 must activate α, so that $w.1$ acts instead of $w.0$.

In this respect the action $\hat{\alpha}$ is *linear*, in the sense of a linear resource: it must be used exactly once, otherwise the process must evolve to 0, as stated by the third equation of Table 4. As proved below, it is also linear as an operator $P \mapsto \hat{\alpha}.P$. These two features are deeply related: internal choice and outcomes may commute with the action prefix only if we know for sure that the prefix will eventually be used.

75 **Proposition.** *For all families of actions $(\alpha_i)_{i\in I}$ and processes $(P_i)_{i\in I}$,*

$$\sum_{i\in I} \alpha_i.P_i \simeq \bigoplus_{i\in I} \hat{\alpha}_i.P_i \oplus \sum_{i\in I} \alpha_i.0.$$

The function $P \mapsto \hat{\alpha}.P$ is linear and the equivalences of Table 4 hold.

Proof. For linearity, we use the fact that $\lfloor \hat{\alpha}.P \mid Q \rfloor$ is the sum of the $s((\hat{\alpha}.P \mid Q)/\rho)$ for the runs ρ that actually trigger α (and the witness action $w.1$). If $P = \lambda \mid P'$ for some $\lambda \in \mathbb{S}$, these runs are the same in $\hat{\alpha}.(\lambda \mid P') \mid Q$ and $\hat{\alpha}.P' \mid Q$, but the outcomes are multiplied by λ in the first case, so $\lfloor \hat{\alpha}.(\lambda \mid P') \mid Q \rfloor = \lambda \cdot \lfloor \hat{\alpha}.P' \mid Q \rfloor$ and $\hat{\alpha}.(\lambda \mid P') \simeq \lambda \mid \hat{\alpha}.P'$. If $P = P_1 \oplus P_2$, the choice is eventually active in all relevant runs, so each of these runs triggers either P_1 or P_2. We can thus establish a bijection between $\mathcal{R}(\hat{\alpha}.(P_1 \oplus P_2) \mid Q)$ and the disjoint union of $\mathcal{R}(\hat{\alpha}.P_1 \mid Q)$ and $\mathcal{R}(\hat{\alpha}.P_2 \mid Q)$. Since outcomes are preserved by this bijection, we finally get $\lfloor \hat{\alpha}.(P_1 \oplus P_2) \mid Q \rfloor = \lfloor \hat{\alpha}.P_1 \mid Q \rfloor + \lfloor \hat{\alpha}.P_2 \mid Q \rfloor$ and $(P_1 \oplus P_2) \mid Q \simeq (P_1 \mid Q) \oplus (P_2 \mid Q)$.

The equivalence $(\boldsymbol{\nu} u)u^{\varepsilon}(x).P \simeq 0$ can be deduced from previous equations:

$$\begin{aligned}(\boldsymbol{\nu} u)\hat{u}^{\varepsilon}(x).P &= (\boldsymbol{\nu} uw)(u^{\varepsilon}(x).(P \mid w.1) \mid (w.0 \mid \bar{w}.1)) \\ &\simeq (\boldsymbol{\nu} w)((\boldsymbol{\nu} u)u^{\varepsilon}(x).(P \mid w.1) \mid (w.0 \mid \bar{w}.1)) \\ &\simeq (\boldsymbol{\nu} w)(1 \mid (w.0 \mid \bar{w}.1)) \simeq (\boldsymbol{\nu} w)(w.0 \mid \bar{w}.1) \simeq (\boldsymbol{\nu} w)(0 \mid 1) \simeq 0\end{aligned}$$

For the decomposition, let f and g be the functions from $\Pi_{\mathbb{S}}$ to $\mathbb{S}$ such that $f(Q) = \lfloor (\sum_{i\in I} \alpha_i.P_i) \mid Q \rfloor$ and $g(Q) = \lfloor (\bigoplus_{i\in I} \hat{\alpha}_i.P_i \oplus \sum_{i\in I} \alpha_i.0) \mid Q \rfloor = \sum_{i\in I} g_i(Q) + g_0(Q)$, we prove $f = g$. By previous remarks we have $g(Q) = \sum_{i\in I} \lfloor \hat{\alpha}_i.P_i \mid Q \rfloor + \lfloor \sum_{i\in I} \alpha_i.0 \mid Q \rfloor$. Given a term Q, $\mathcal{R}((\sum_{i\in I} \alpha_i.P_i) \mid Q)$ decomposes into $\mathcal{R}_0$ for the runs that trigger none of the α_i and a $\mathcal{R}_i$ for all runs that trigger α_i, for each i. Clearly, $\mathcal{R}_0$ contains the runs of $\sum_{i\in I} \alpha_i.0 \mid Q$ that do not trigger any α_i, and all other runs of this term have outcome 0, so the sum of the outcomes of runs in $\mathcal{R}_0$ is $g_0(Q)$. For each i, the runs of $\mathcal{R}_i$ are in bijection with runs of $\hat{\alpha}_i.P_i \mid Q$ that trigger α_i and they have the same outcomes, and all other runs of this term have outcome 0, so the sum out the outcomes of these runs is $g_i(Q)$. As a consequence, we get the expected decomposition $f = g_0 + \sum_{i\in I} g_i$.

For the equivalence $\hat{\alpha}.(\beta.0 \mid P) \simeq \beta.0 \mid \hat{\alpha}.P$, assuming the subject of β is not the bound name of action α, let Q be an arbitrary term and consider $\mathcal{R}(\hat{\alpha}.(\beta.0 \mid P) \mid Q)$. Any run that does not trigger $\hat{\alpha}$ or that triggers both $\hat{\alpha}$ and β has outcome 0, so the only relevant runs are those that trigger $\hat{\alpha}$ but not β. Clearly these runs are the same as the runs of $(\beta.0 \mid \hat{\alpha}.P) \mid Q$ that trigger $\hat{\alpha}$ and not β, and they have the same outcomes.

For the composition of inactions, the relevant runs of a term $(\sum_{i\in I} \alpha_i.0 \mid \sum_{i\in J} \alpha.0) \mid P$ are those that do not trigger any of the α_i, so the number of occurrences of each α_i does not matter, and the fact that they are in a branching or in parallel does not matter either, as long as the branchings cannot interact. The only special case is when there are $i \in I$ and $j \in J$ such that $\alpha_i = \bar{\alpha}_j$, then each run must trigger one branching or the other, if nothing else by letting α_i and α_j interact. As a consequence, all runs of this term have outcome 0, so the composition of the two branchings is indistinguishable from 0. ☐

76 **Definition.** A term is *simple* if it is generated by the following grammar

$$\begin{array}{ll} \text{simple term} & P, Q := 1,\ N,\ \hat{\alpha}.P,\ (P \mid Q),\ (\boldsymbol{\nu} x)P \\ \text{inaction set} & N := \sum_{i\in I} \alpha_i.0 \end{array}$$

A pre-trace $\rho \in \mathcal{P}(P)$ is exhaustive if it triggers all linear actions and no inaction, and no sub-term of P/ρ has the form $Q \mid R$ with Q containing some $\alpha.0$ and R containing $\bar{\alpha}.0$. The set of such pre-traces is written $\mathcal{P}_e(P)$.

Simple terms have the property that the outcome of any run is either 1 or 0. More precisely, it is easy to see that the outcome of a run is 1 if and only if it triggers all linear actions and no inaction. The notion of exhaustive pre-trace is the correct extension of this notion to pre-traces, indeed every run of a simple term $P \mid Q$ with outcome 1 is made of an exhaustive pre-trace of P and an exhaustive pre-trace of Q. The condition on P/ρ simply rules out interactions of P that lead to a term P' where there are dual inactions that may interact, since that would imply $P' \simeq 0$, as a generalization of the equation $\alpha.0 \mid \bar{\alpha}.0 \simeq 0$. Observe that, by the decomposition of Proposition 75 and the linearity of all constructions of simple terms, we immediately prove that every term is equivalent to a linear combination of simple terms. As a consequence, two terms P and Q are equivalent if and only if for all *simple* term R, $\lfloor P \mid R \rfloor = \lfloor Q \mid R \rfloor$.

4.3 An order algebraic model

Thanks to the decomposition into simple terms, we are now ready to describe our order algebraic semantics. Following the initial intuition, we define a web whose points are action occurrences, with a group action that permutes actions of the same name and polarity while making sure that bound names are properly updated. We need an extra bit of information to represent inactions, and these will be represented as extra actions (somehow "potential" actions) with particular treatment.

77 **Definition.** The set C of *abstract channels* is defined as $C := \boldsymbol{N} \times (\boldsymbol{P} \times \mathbb{N})^*$. We write $u \cdot \varepsilon_1 n_1 \cdots \varepsilon_k n_k$ instead of $(u, ((\varepsilon_1, n_1), \dots, (\varepsilon_k, n_k)))$.

The arena E for the πI-calculus is such that $|E| = C \times \boldsymbol{P} \times (\mathbb{N} \cup \{\bot, \top\})$ and G^E is generated by permutations of the form $(x, \varepsilon, \sigma) \in C \times \boldsymbol{P} \times \mathfrak{S}(\mathbb{N})$, acting as

$$(x, \varepsilon, \sigma)(y) = \begin{cases} x \cdot \varepsilon\sigma(n) \cdot z & \text{if } y = x \cdot \varepsilon n \cdot z \text{ for some } n \in \mathbb{N} \text{ and } z \in (\boldsymbol{P} \times \mathbb{N})^* \\ y & \text{otherwise} \end{cases}$$

Abstract channels represent names in a way that allows us to avoid any renamings. Intuitively, u (that is $(u, ()))$ represents the free name u itself, $u \cdot \varepsilon n$ represents the bound name x in the action $u_n^\varepsilon(x)$, then $u \cdot \varepsilon n \cdot \varepsilon' n'$ represents the bound name y in $x_{n'}^{\varepsilon'}(y)$, and so on. So an abstract channel is the path to find a given name, free or bound. In a sense, this notion is an analogous for names in πI-terms of De Bruijn indices.

We can assume, without loss of generality, that all names in processes we use respect this intuition, so that we can mention any name without ambiguity and with no need of renaming. Under this hypothesis, given a term P and a pre-trace $\rho \in \mathcal{P}(P)$, the term P/ρ is uniquely defined, *not* up to renaming. Note however that P/ρ does not respect the intuition on free names if bound names were revealed, i.e. if ρ contains a visible action. With this discipline on names, we can assume without loss of generality that the set of free names $\boldsymbol{N}$ is *finite*.

Points in the web $|E|$ are of two kinds. The first kind is $x \cdot \varepsilon n$ for the occurrence of polarity ε at location n of the name x. The second kind is $x \cdot \varepsilon\bot$ or $x \cdot \varepsilon\top$ for the inaction of polarity ε with name x; the use of $\bot$ and $\top$ is a tool used to encode the behaviour of inactions, with the convention that $x \cdot \varepsilon\bot < x \cdot \varepsilon\top$ if $x^\varepsilon.0$ is present, and the points are incomparable otherwise.

A permutation (x, ε, σ) permutes the locations of the actions of polarity ε of the name x according to $\sigma : \mathbb{N} \to \mathbb{N}$. By definition, the n-th occurrence of polarity ε of x, namely $x \cdot \varepsilon n$, is renamed into $x \cdot \varepsilon\sigma(n)$, the m-th occurrence of polarity η of the name bound by it, namely $x \cdot \varepsilon n \cdot \eta m$, gets renamed as $x \cdot \varepsilon\sigma(n) \cdot \eta m$, i.e. its location is unchanged but its name is changed to reflect the change of its binder, and so on for other bound names. The inactions at $x \cdot \varepsilon$ are unchanged since x is unchanged, but those on names bound by x are renamed accordingly. A more explicit (but equivalent) construction of the permutation group consists in setting $\mathrm{G}^E := \mathfrak{S}(\mathbb{N})^C$ and defining the action of $\sigma \in \mathrm{G}^E$ as

$$\sigma(u \cdot \varepsilon_1 n_1 \cdots \varepsilon_k n_k) := u \cdot \varepsilon_1 \sigma(u)(n_1) \cdot \varepsilon_2 \sigma(u \cdot \varepsilon_1 n_1)(n_2) \cdots \varepsilon_k \sigma(u \cdot \varepsilon_1 n_1 \cdots \varepsilon_{k-1} n_{k-1})(n_k)$$

except if $n_k \in \{\bot, \top\}$ in which case the last pair remains as $\varepsilon_k n_k$.

78 **Definition.** A trace is a play t on the web E such that

- for all $x \cdot \varepsilon n \cdot \varepsilon' n' \in |t|$, $x \cdot \varepsilon n \in |t|$ and $x \cdot \varepsilon n <_t x \cdot \varepsilon n \cdot \varepsilon' n'$,
- for all $x \in \boldsymbol{N}$ and all $x = y \cdot \varepsilon' n \in |t|$, $x \cdot \varepsilon\bot$ and $x \cdot \varepsilon\top$ are in $|t|$, and for all $y \in |t| \setminus \{x \cdot \varepsilon\bot, x \cdot \varepsilon\top\}$, $x \cdot \varepsilon\bot$ and $x \cdot \varepsilon\top$ are incomparable with y.

The first condition is a kind of "justification" condition in the style of game semantics [21]. It means that for an action $x \cdot \varepsilon n \in |t|$, if the subject x is a bound name, then its binder (the action also named x) is also in $|t|$ and it is inferior in the scheduling order, i.e. it was revealed earlier. The second condition means that inactions information must be present for each known name and that inactions are not involved in scheduling.

79 **Definition.** Let P be a simple term and let ρ be an exhaustive pre-trace of P. The trace induced by ρ is the trace ρ^* such that

- $|\rho^*| = \{x \cdot \varepsilon n \mid x^\varepsilon{:}n \in \rho\} \cup N_\rho$, where N_ρ contains $x \cdot \varepsilon\bot$ and $x \cdot \varepsilon\top$ for all polarity ε and all name x such that $x \in \boldsymbol{N}$ or $x = y \cdot \varepsilon n$ for some $y^\varepsilon{:}n \in \rho$,

- $\leqslant$ is the causal order (as of Definition 70) restricted to visible transitions, augmented with $x \cdot \varepsilon\bot < x \cdot \varepsilon\top$ for each $x^\varepsilon.0$ that occurs in P/ρ.

Note that the justification condition is satisfied by ρ^*, because in the πI-calculus the action prefixes are synchronous: in an action $u(x).P$, the action $u(x)$ that binds x is automatically a prefix of all actions on x. However, synchrony is not necessary for this property to hold, the fact that the name is bound is the important point: even if internal transitions can occur on a bound name, visible transitions are possible only after the name has been revealed by the action it is bound to.

80 **Proposition.** *To each simple term P, associate the function $[\![P]\!] : \mathcal{S}(E) \to \mathbb{S}$ such that for all $t \in \mathcal{S}(E)$, $[\![P]\!](t) := \sharp\{\rho \in \mathcal{P}_e(P) \mid \rho^* = t\}$. This function clearly has finite support, so $[\![P]\!] \in \mathcal{C}(E)$. Let $u \mapsto \bar{u}$ be the linear map over $\mathcal{C}(E)$ that inverts polarities and exchanges $\bot$ and $\top$. Then for all simple terms P, Q, $\lfloor P \mid Q \rfloor = \left\lfloor [\![P]\!] \parallel \overline{[\![Q]\!]} \right\rfloor$.*

Proof. By construction, if P and Q are simple terms, then so is $P \mid Q$, so all its runs have outcome 0 or 1, thus $\lfloor P \mid Q \rfloor$ is the number of non-zero runs of $P \mid Q$. Every run $\rho \in \mathcal{R}(P \mid Q)$ can be uniquely decomposed as a pre-trace $\rho_1 \in \mathcal{P}(P)$ and a pre-trace $\rho_2 \in \mathcal{P}(Q)$. Moreover, by definition of exhaustive pre-traces, if the outcome of ρ is 1 then ρ_1 and ρ_2 are exhaustive pre-traces.

Now let ρ_1 and ρ_2 be any exhaustive pre-traces of P, we want to compute how many runs with outcome 1 they generate. A run $\rho \in \mathcal{R}(P \mid Q)$ projects to ρ_1 and ρ_2 if and only if it establishes a bijection from visible actions of ρ_1 to dual visible actions of ρ_2, such that scheduling constraints are respected and no opposite inactions exist between ρ_1 and ρ_2. Formulated in traces, this means a bijection $\varphi : |\rho_1^*| \to |\rho_2^*|$ such that:

- For all $a = x \cdot \varepsilon n \in |\rho_1^*|$, $\varphi(a) = y \cdot \neg\varepsilon m$ for some y and m (i.e. actions of opposite polarities are matched), and if $x \in |\rho_1^*|$ then $y \in |\rho_2^*|$ and $\varphi(x) = y$. This means that an action on a bound name must be matched with an action on another bound name and that these names are revealed by actions that were matched together (this is a typical property of the πI-calculus).

- The union of the orders $\varphi(\leqslant_{\rho_1^*})$ and $\leqslant_{\rho_2^*}$ is acyclic, which means that φ respects prefixing constraints so that we get an actual execution path.

Such a bijection φ establishes an identification between names revealed in the interactions ρ_1 and ρ_2, and the last thing to check is that under this bijection, there are no dual inactions between ρ_1 and ρ_2. By construction, that there are such inactions if and only if for some name x in $\boldsymbol{N}$ or $|\rho_1^*|$ and polarity ε we have $x \cdot \varepsilon\bot <_{\rho_1^*} x \cdot \varepsilon\top$ and $\varphi(x) \cdot \neg\varepsilon\top <_{\rho_2^*} x \cdot \neg\varepsilon\bot$, which exactly corresponds to a cycle in the union $\varphi(\leqslant_{\rho_1^*}) \cup \overline{\leqslant_{\rho_2^*}}$.

It is routine to check that bijections that satisfy the above conditions are exactly the bijections induced by elements of G^E such that $\varphi(\rho_1^*) * \overline{\rho_2^*} = 1$: the structure of G^E is made to ensure that the justification condition is satisfied, and the rest is ordering conditions. As a consequence, the number we seek is exactly $\rho_1^* \parallel \overline{\rho_2^*}$. By summing this on all pairs of exhaustive pre-traces of P and Q, we finally get $\lfloor P \mid Q \rfloor = \left\lfloor [\![P]\!] \parallel \overline{[\![Q]\!]} \right\rfloor$. □

The translation function $P \mapsto [\![P]\!]$ defined above applies to simple terms, but using the results of Section 4.2 we can extend it to all terms by linear combinations. The decomposition of terms as linear combinations of simple terms is not unique syntactically, however all decompositions are observationally equivalent by definition, and it is easy to check that the traces induced by all possible translations of a given term are the same, so the translation is actually a function from terms to vectors in $\mathcal{C}(E)$. The space of linear combinations of plays $\mathcal{C}(E)$ is larger than the set of translations of terms, so by Proposition 80 if translations of two terms P and Q are observationally equivalent in the sense of order algebras then these terms are equivalent in the sense of quantitative testing. Hence our final theorem:

81 **Theorem.** *Two terms of the πI-calculus are observationally equivalent for quantitative testing in a semiring $\mathbb{S}$ if and only if their translations in $\mathcal{A}_{\mathbb{S}}(E)$ are equal.*

4.4 Consequences

The first consequence of this model is that Theorem 41 provides a basis for the set of processes in two particular cases:

- If $\mathbb{S}$ is idempotent, then each term is equivalent to a linear combination of totally ordered traces. It is the case when $\mathbb{S}$ represents standard may or must testing. Then we lose the "quantitative" aspect since multiplicities are ignored, and we fall back to standard semantics as a special case. We get full abstraction in this case by showing that any base play can be implemented as a term of the calculus [5].
- If $\mathbb{S}$ is a regular ring, terms are combinations of weakly totally ordered traces. We can get full abstraction again if we slightly extend the calculus to allow parallel composition without interaction [6], this is needed only for the case of traces that contain concurrent dual actions. Actually the only needed feature is a multiple prefix $\{\alpha_1, \dots, \alpha_k\}.P$, which is enough to represent weakly ordered traces as terms. Simpler modifications of the calculus could lead to full abstraction, for instance by imposing a more structured naming discipline.

Although we will not write the proofs here in full detail, the interpretation of processes is compositional, and we can use the constructs of Section 3 to represent syntactic constructs as operators on order algebras. Define the arena Ch of channel ends as $|Ch| = (\mathbb{N} \times \boldsymbol{P})^* \times (\mathbb{N} \cup I)$ with $I = \{\bot, \top\}$, with permutations of the same kind as in E, then the definition of E from Definition 77 reformulates as

$$E = (\boldsymbol{N} \times \boldsymbol{P}) \rhd Ch \qquad \text{and} \qquad Ch = I + \sharp(\{*\} + \boldsymbol{P} \rhd Ch)$$

up to a simple isomorphism. These equations mean that a process appears as a family of channel ends indexed by free names and polarities, and that a channel end contains inaction information (the I part) and an arbitrary number of interchangeable occurrences (the $*$) each associated with a new channel end per polarity (the $\boldsymbol{P} \rhd Ch$).

The explicit mention of the $\sharp$ operator for the action occurrences allows us to use the γ and δ operators from Definition 61 as a systematic way of treating the inherent non-determinism in the multiple occurrences of each name. We can thus define parallel composition of vectors $p \mid q$ in the order algebra as follows:

- For each channel end $x \cdot \varepsilon$ in P, apply $\delta^2 : \mathcal{A}(\sharp Oc) \to \mathcal{A}(\sharp Oc + \sharp Oc)$, where $Oc = \{*\} + \boldsymbol{P} \rhd Ch$ is the arena for an action occurrence. This splits the occurrences of $x \cdot \varepsilon$ into those that will interact with Q and those that will not. Extend this to Ch by keeping the inaction part unchanged, and apply the same operator independently to each channel end, giving an operator $\delta' : \mathcal{A}(E) \to \mathcal{A}(E + (\boldsymbol{N} \times \boldsymbol{P}) \rhd \sharp Oc)$. The $\sharp Oc$ part contains actions that will *not* interact.

may and must			
$\cdot$	0	1	ω
0	0	0	0
1	0	1	ω
ω	0	ω	ω

may testing			
$+$	0	1	ω
0	0	1	ω
1	1	1	ω
ω	ω	ω	ω

must testing			
$+$	0	1	ω
0	0	1	ω
1	1	1	1
ω	ω	1	ω

Table 5: Observation semirings for may and must testing.

- Do the same for Q, and compose the result with the involution $u \mapsto \bar{u}$ from Proposition 80.
- Partially synchronize $\delta'(p)$ and $\overline{\delta'(q)}$ on the E part, to represent the actual interaction for the occurrences that must interact, which yields a vector $u \in \mathcal{A}(E + (\boldsymbol{N} \times \boldsymbol{P}) \triangleright (\sharp Oc + \sharp Oc))$. This partial synchronisation handles the conditions on inactions the same way as in Proposition 80.
- In the result, for each channel end $x \cdot \varepsilon$ in the E part, forget the actions on $x \cdot \varepsilon$ since they have interacted, then normalise the inaction part by mapping any $y \cdot \varepsilon\top < y \cdot \varepsilon\bot$ to the reverse order (this is a linear operator since it acts on plays) and inverting again the remaining part of Q to get back the original polarities on visible actions. Call $n : \mathcal{A}(E + (\boldsymbol{N} \times \boldsymbol{P}) \triangleright (\sharp Oc + \sharp Oc)) \to \mathcal{A}((\boldsymbol{N} \times \boldsymbol{P}) \triangleright (I + \sharp Oc + \sharp Oc))$ this operator.
- Finally, contract the action occurrences on each channel end in the result with the operator $\gamma^2 : \mathcal{A}(\sharp Oc + \sharp Oc) \to \mathcal{A}(\sharp Oc)$ applied on each channel end in $\boldsymbol{N} \times \boldsymbol{P}$, which defines an operator $\gamma' : \mathcal{A}((\boldsymbol{N} \times \boldsymbol{P}) \triangleright (I + \sharp Oc + \sharp Oc)) \to \mathcal{A}(E)$.

With this definitions, we finally get $p \mid q := \gamma'(n(\delta'(p) \parallel_E \overline{\delta'(q)}))$.

The other operators are easy to define. An outcome λ is translated as $\lambda.\emptyset$, where $\emptyset$ is the empty run. Branchings are decomposed as in Proposition 75, and the linear action is a linear operator that consists in introducing in each play an extra point for the new action, minimal for the scheduling order. Hiding a name x consists in mapping to 0 all plays that contain an action on x and forgetting the inaction information on x.

By choosing appropriate structures for $\mathbb{S}$, we can recover the standard may and must testing [13]. In both cases we have $\mathbb{S} = \{0, 1, \omega\}$, where ω represents success. Table 5 shows the rules for addition and multiplication for may and must. Using this definition it is clear that P and Q are equivalent for may or must testing if and only if, for all R, $\lfloor P \mid R \rfloor = \omega$ if and only if $\lfloor Q \mid R \rfloor = \omega$. Taking for $\mathbb{S}$ the minimal semiring $\{0, 1\}$ with $1 + 1 = 1$ gives the framework studied by the author in a previous work [5], which also leads to must testing semantics. In these semirings, all elements are idempotent for addition, so by Theorem 41 the model we get is actually interleaving.

References

[1] Samy Abbes and Albert Benveniste. *True-concurrency probabilistic models: branching cells and distributed probabilities for event structures.* Information and Computation, 204(2):231–274, 2006.

[2] Samson Abramsky, Radha Jagadeesan and Pasquale Malacaria. *Full abstraction for PCF.* In International Symposium on Theoretical Aspects of Computer Science (TACS), pages 1–15, 1994.

[3] Marianne Akian, Stéphane Gaubert and Alexander Guterman. *Linear independence over tropical semirings and beyond.* In Grigorii Lazarevich Litvinov and Sergei N. Sergeev, editors, Proceedings of the International Conference on Tropical and Idempotent Mathematics, Contemporary Mathematics, volume 495, pages 1–38. AMS, 2009.

[4] Patrick Baillot, Vincent Danos, Thomas Ehrhard and Laurent Regnier. *Believe it or not, AJM's games model is a model of classical linear logic.* In LICS, pages 68–75, 1997.

[5] Emmanuel Beffara. *An algebraic process calculus.* In Proceedings of the twenty-third annual IEEE symposium on logic in computer science (LICS), pages 130–141, 2008.

[6] Emmanuel Beffara. *Quantitative testing semantics for non-interleaving.* Technical report hal-00397551, Institut de Mathématiques de Luminy, April 2009. Available online at http://hal.archives-ouvertes.fr/hal-00397551/.

[7] Michele Boreale and Fabio Gadducci. *Denotational testing semantics in coinductive form.* In Branislav Rovan and Peter Vojtáš, editors, Mathematical Foundations of Computer Science 2003, Lecture Notes in Computer Science, volume 2747, pages 279–289. Springer, 2003.

[8] Michele Boreale and Fabio Gadducci. *Processes as formal power series: a coinductive approach to denotational semantics.* Theoretical Computer Science, 360:440–458, 2006.

[9] Gérard Boudol and Ilaria Castellani. *A non-interleaving semantics for CCS based on proved transitions.* Fundamenta Informaticae, XI:433–453, 1988.

[10] Vincenzo Ciancia and Ugo Montanari. *Symmetries, local names and dynamic (de)-allocation of names.* Information and Computation, 208(12):1349–1367, 2010.

[11] Silvia Crafa, Daniele Varacca and Nobuko Yoshida. *Compositional event structure semantics for the π-calculus.* In Proceedings of the 18th international conference on concurrency theory (CONCUR), Lecture Notes in Computer Science, volume 4703, pages 317–332. Springer, 2007.

[12] Philippe Darondeau and Pierpaolo Degano. *Causal trees.* In Giorgio Ausiello, Mariangiola Dezani-Ciancaglini and Simonetta Della Rocca, editors, International Conference Automata, Languages and Programming, Lecture Notes in Computer Science, volume 372, pages 234–248. Springer, 1989.

[13] Rocco de Nicola and Matthew Hennessy. *Testing equivalences for processes.* Theoretical Computer Science, 34:83–133, 1984.

[14] Volker Diekert and Grzegorz Rozenberg. *The Book of Traces.* World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1995.

[15] Thomas Ehrhard and Olivier Laurent. *Interpreting a finitary π-calculus in differential interaction nets.* In Luís Caires and Vasco T. Vasconcelos, editors, 18th International Conference on Concurrency Theory (Concur), LNCS, volume 4703, pages 333–348. Springer, September 2007.

[16] Thomas Ehrhard and Laurent Regnier. *The differential λ-calculus.* Theoretical Computer Science, 309(1):1–41, 2003.

[17] Thomas Ehrhard and Laurent Regnier. *Differential interaction nets.* Theoretical Computer Science, 364(2):166–195, 2006.

[18] Claudia Faggian and Mauro Piccolo. *Partial orders, event structures and linear strategies.* In Typed Lambda Calculi and Applications, Lecture Notes in Computer Science, volume 5608, pages 95–111. Springer, 2009.

[19] Murdoch J. Gabbay. *Foundations of nominal techniques: logic and semantics of variables in abstract syntax.* Bulletin of Symbolic Logic, 2011. In press.

[20] C. Anthony R. Hoare. *Communicating Sequential Processes.* Prentice Hall, 1985.

[21] J. Martin E. Hyland and Chih-Hao Luke Ong. *On full abstraction for pcf (parts i, ii and iii).* Information and Computation, 163(2):285–408, 2000.

[22] Leslie Lamport. *Time, clocks and the ordering of events in a distributed system.* Communications of the ACM, 21(7):558–565, July 1978.

[23] Serge Lang. *Algebra.* Addison-Wesley, 1965.

[24] Jean-Louis Loday. *Generalized bialgebras and triples of operads.* Astérisque, 320, 2008.

[25] Paul-André Melliès and Samuel Mimram. *From asynchronous games to concurrent games.* Submitted, September 2008.

[26] Robin Milner. *Communication and concurrency.* Prentice Hall, 1989.

[27] Robin Milner. *Communicating and Mobile Systems: the π-Calculus.* Cambridge University Press, may 1999.

[28] Ugo Montanari and Marco Pistore. *Structured coalgebras and minimal HD-automata for the π-calculus.* Theoretical Computer Science, 340(3):539–576, 2005.

[29] Ernst-Rüdiger Olderog and C. Anthony R. Hoare. *Specification-oriented semantics for communicating processes.* Acta Informatica, 23(1):9–66, 1986.

[30] Jean-Éric Pin. *Tropical semirings.* In Jeremy Gunawardena, editor, Idempotency (Bristol, 1994), Publications of the Newton Institute, volume 11, pages 50–69. Cambridge University Press, 1994.

[31] Jan J. M. M. Rutten. *Automata, power series, and coinduction: Taking input derivatives seriously.* In Jirí Wiedermann, Peter van Emde Boas and Mogens Nielsen, editors, Internatioanl Conference on Automata, Languages and Programming (ICALP), Lecture Notes in Computer Science, volume 1644, pages 707–707. Springer, 1999.

[32] Davide Sangiorgi. *π-calculus, internal mobility and agent-passing calculi.* Theoretical Computer Science, 167(2):235–274, 1996.

[33] Sam Staton and Glynn Winskel. *On the expressivity of symmetry in event structures.* In Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, pages 392–401. IEEE Computer Society, 2010.

[34] Daniele Varacca, Hagen Völzer and Glynn Winskel. *Probabilistic event structures and domains.* Theoretical Computer Science, 358(2–3):173–199, 2006.

[35] Glynn Winskel. *Event structure semantics for CCS and related languages.* In Proceedings of the 9th international colloquium on automata, languages and programming (ICALP), Lecture Notes in Computer Science, volume 140, pages 561–576. Springer, July 1982.

[36] Glynn WINSKEL. *Event structures.* In Advances in Petri nets: applications and relationships to other models of concurrency, pages 325–392. Springer Verlag, 1987.

[37] Glynn WINSKEL. *Event structures with symmetry.* Electronic Notes in Theoretical Computer Science, 172:611–652, 2007. Computation, Meaning, and Logic: Articles dedicated to Gordon Plotkin.