# Monoidal-Closed Categories of Tree Automata

Colin Riba
ENS de Lyon, Université de Lyon, LIP[*]
colin.riba@ens-lyon.fr
http://perso.ens-lyon.fr/colin.riba/

This paper surveys a new perspective on Rabin's Tree Theorem, the decidability of Monadic Second-Order Logic (MSO) on infinite trees, one of the most important and difficult decidability theorems for mathematical theories. We show that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Tree Theorem can be organized in a deduction system based on the multiplicative fragment of intuitionistic linear logic (ILL). We propose a realizability semantics for automata on infinite trees, which interprets this deduction system and which is based on categories of games built on usual *simple games*, and generalizing usual acceptance games of tree automata. Our approach can be summarized with the slogan "*automata as objects, strategies as morphisms*".

Namely, we equip a variant of usual alternating tree automata (that we call *uniform* tree automata) with a fibred monoidal closed structure which in particular, *via* game determinacy handles a linear complementation of alternating automata, as well as deduction rules for existential and universal quantifications. This monoidal structure is actually Cartesian on *non-deterministic* automata. Moreover, an adaptation of a usual construction for the simulation of alternating automata by non-deterministic ones satisfies the deduction rules of the $!(-)$ ILL-exponential modality.

Our realizability semantics satisfies an expected property of witness extraction from proofs of existential statements. Moreover, it allows to combine realizers produced as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata, possibly obtained using external algorithms.

## 1 Introduction

Monadic Second-Order Logic (MSO) on infinite trees is a rich theory which contains non trivial mathematical theories (see e.g. [Rab69, BGG97]), and which subsumes many logics, in particular modal logics (see e.g. [BdRV02]) and modal logics for verification (see e.g. [VW08]). Rabin's Tree Theorem [Rab69], the decidability of MSO on infinite trees, "*is one of the most important and difficult decidability theorems for mathematical theories*" according to [BGG97, §1.3, p. 11].

The original proof of [Rab69] relied on an effective translation of formulae to finite state automata running on infinite trees. Since then, there have been considerable work on Rabin's

[*] UMR 5668 CNRS ENS Lyon UCBL INRIA

Tree Theorem, culminating in streamlined decidability proofs, as presented in e.g. [Tho97, GTW02, PP04]. Most current approaches to MSO on infinite trees (but with the notable exception of [Blu13]) are based on translations of MSO-formulae to automata.

We are interested in decomposing the translations of MSO-formulae to automata in a constructive (actually *linear*) deduction system for tree automata, and in a compositional computational interpretation of this deduction system along the lines of the Curry-Howard proofs-as-programs correspondence (see e.g. [GLT89, SU06]). We follow the guidelines and axiomatizations provided by categorical logic and categorical approaches to the Curry-Howard correspondence (see e.g. [Jac01, LS86] and [AC98]).

Our interpretation can be summarized with the slogan "*automata as objects, strategies as morphisms*". It consists in categories which are based on usual categories of two-player linear sequential games called *simple games* (see e.g. [Abr97, Hyl97]), which live in a denotational model of *Intuitionistic Linear Logic* (ILL) [Gir87][1], and which generalize usual acceptance games of tree automata. We refer to [Mel09] for a comprehensive presentation of categorical axiomatizations of models of (subsystems of) linear logic.

This work builds on [Rib15], which proposed monoidal fibrations of games and tree automata, and extends it with a monoidal *closed* structure, based on a variant of alternating automata (that we call *uniform automata*), and which allows a clearer connection of our model with ILL. The present paper is a slightly shortened version of [Rib17].

In the remaining of this Introduction, we survey the main aspects of our realizability approach (§1.1-1.2 and §1.5) and outline the key ingredients of the decomposition of MSO in ILL (§1.3-1.4).

**1.1 (Non-Deterministic) Tree Automata.** Let us set some concepts and notations. Concatenation of sequences $s, t$ is denoted either $s.t$ or $s \cdot t$, and $\varepsilon$ is the empty sequence. We fix throughout the paper a finite non-empty set $\mathfrak{D}$ of *tree directions*. We are interested in labelings of the full $\mathfrak{D}$-ary tree $\mathfrak{D}^*$ over different *alphabets*. Alphabets (denoted $\Sigma, \Gamma$, etc) are finite non-empty sets, and $\Sigma$-labeled $\mathfrak{D}$-ary trees are functions $T : \mathfrak{D}^* \to \Sigma$.

There are two families of automata involved in the interpretation of MSO-formulae: *non-deterministic* tree automata and *alternating* tree automata[2]. The simplest notion is that of non-deterministic automaton, and it is sufficient to introduce the basic motivations and methodology of this work.

A tree automaton $\mathcal{A}$ consists of a finite set $Q_{\mathcal{A}}$ of states, with a distinguished[3] initial state $q^{\imath}_{\mathcal{A}} \in Q_{\mathcal{A}}$, an acceptance condition given by an $\omega$-regular set $\Omega_{\mathcal{A}} \subseteq Q^{\omega}_{\mathcal{A}}$, and a transition function $\delta_{\mathcal{A}}$. A *non-deterministic* tree automaton $\mathcal{A}$ over $\Sigma$ has a transition function of the form

$$\delta_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathfrak{D} \to Q_{\mathcal{A}})$$

Acceptance for tree automata can equivalently be described by *games* or *run trees*. The notion of run tree is simpler and sufficient at various places in this Introduction and §2. A *run tree* of $\mathcal{A}$ on $T : \mathfrak{D}^* \to \Sigma$ is a tree $R : \mathfrak{D}^* \to Q_{\mathcal{A}}$ such that $R(\varepsilon) = q^{\imath}_{\mathcal{A}}$, and which respects the transitions of $\mathcal{A}$, in the sense that for each tree position $p \in \mathfrak{D}^*$, there exists a function $\mathsf{g} : \mathfrak{D} \to Q_{\mathcal{A}}$ in $\delta_{\mathcal{A}}(R(p), T(p))$ such that $R(p.d) = \mathsf{g}(d)$ for all $d \in \mathfrak{D}$. The run $R$ *accepting* if all its infinite paths belong to $\Omega_{\mathcal{A}}$. We say that $T$ is accepted by $\mathcal{A}$ if there exists an accepting run of $\mathcal{A}$ on $T$, and let $\mathcal{L}(\mathcal{A})$ be the set of trees accepted by $\mathcal{A}$. We moreover write $\mathcal{A}(T)$ for the set of accepting runs of $\mathcal{A}$ on $T$.

[1] However, the ILL-structure underlying our model differs from the usual ILL-structure of simple games.

[2] Alternating automata are not always explicited (see e.g. [Tho97]).

[3] It is customary (and equivalent in terms of expressivity) to allow possibly different initial states.

**1.2 Computational Interpretation of Proofs.** Our deduction system manipulates sequents of the form

$$T \ ; \ \mathcal{A}_1, \dots, \mathcal{A}_n \vdash \mathcal{B} \tag{1}$$

where $T$ is an infinite tree labeled over (say) the alphabet $\Sigma$, and $\mathcal{A}_1, \dots, \mathcal{A}_n, \mathcal{B}$ are tree automata over $\Sigma$. We see these sequents with two different levels of interpretation. The first level interprets *provability*: if the sequent (1) is provable, then the automaton $\mathcal{B}$ accepts the tree $T$ as soon as the automata $\mathcal{A}_1, \dots, \mathcal{A}_n$ all accept $T$. The second level is the traditional *computational* interpretation of *proofs* of the Curry-Howard correspondence. This is best exemplified with existential quantifications.

The existential quantifications of MSO are implemented by a *projection* operation on non-deterministic automata. Consider a non-deterministic automaton $\mathcal{A}$ over the alphabet $\Gamma \times \Sigma$. Its projection $\widetilde{\exists}_\Sigma \mathcal{A}$ is the non-deterministic automaton over $\Gamma$ defined as $\mathcal{A}$ but with the following transition function

$$\begin{array}{lccccc} \delta_{\widetilde{\exists}_\Sigma \mathcal{A}} & : & Q_\mathcal{A} \times \Gamma & \longrightarrow & \mathcal{P}(\mathfrak{D} \to Q_\mathcal{A}) \\ & & (q, \mathtt{b}) & \longmapsto & \bigcup_{\mathtt{a} \in \Sigma} \delta_\mathcal{A}(q, (\mathtt{b}, \mathtt{a})) \end{array}$$

As expected, $\widetilde{\exists}_\Sigma \mathcal{A}$ accepts $T : \mathfrak{D}^* \to \Gamma$ iff there exists $U : \mathfrak{D}^* \to \Sigma$ such that $\mathcal{A}$ accepts $\langle T, U \rangle : \mathfrak{D}^* \to \Gamma \times \Sigma$.

Consider now a non-deterministic automaton $\mathcal{B}$ over the alphabet $\Sigma \simeq \mathbf{1} \times \Sigma$, where $\mathbf{1} \simeq \{\bullet\}$ is a singleton set. By *computational interpretation of proofs*, we mean that from a formal proof of the sequent

$$\mathbf{1} \ ; \ \vdash \widetilde{\exists}_\Sigma \mathcal{B}$$

(where $\mathbf{1}$ stands for the unique $\mathbf{1}$-labeled tree) one should be able to extract a witness for the existential quantification $\widetilde{\exists}_\Sigma \mathcal{B}$, that is a $\Sigma$-labeled tree accepted by $\mathcal{B}$. Such witnesses can actually be extracted from the runs of $\widetilde{\exists}_\Sigma \mathcal{B}$ on $\mathbf{1}$. First note that a run $R$ of a non-deterministic automaton $\mathcal{A}$ on $T$ defines a function $p \in \mathfrak{D}^* \longmapsto \mathtt{g} \in \delta_\mathcal{A}(R(p), T(p))$. It follows that given an accepting run $R$ of $\widetilde{\exists}_\Sigma \mathcal{B}$ on $\mathbf{1}$, then from the induced function

$$p \in \mathfrak{D}^* \quad \longmapsto \quad \mathtt{g} \in \bigcup_{\mathtt{a} \in \Sigma} \delta_\mathcal{B}(R(p), \mathtt{a})$$

one can get a $\Sigma$-labeled tree $T$ such that $R$ is an accepting run of $\mathcal{B}$ on $T$. In other words, *runs* of automata convey the kind of information one is usually interested in with computational interpretations of proofs.

One could formulate the aim of this work as proposing a deduction system together with an interpretation of proofs as runs of automata. However, we will rather rely on the more complex notions of acceptance games and strategies. There are two reasons for this choice. First, as discussed in §1.3 below, games give a smooth treatment of complementation of tree automata. The second reason, which we explain in more details in §2, is that games and strategies are equipped with well-known categorical structures, which allow to easily define compositional interpretations of proofs.

**1.3 Games and Alternating Automata.** The main difficulty when translating MSO-formulae to tree automata is the interplay between negation and (existential) quantification. Historically, Rabin [Rab69] translated MSO-formulae to *non-deterministic* tree automata. The major achievement of Rabin [Rab69] was to show that non-deterministic automata on infinite trees are closed under complement. This means that for every alternating automaton $\mathcal{A}$ one can build a non-deterministic automaton $\sim\mathcal{A}$ which accepts exactly the trees rejected by $\mathcal{A}$.

Rabin's original construction [Rab69] of a complement $\sim\mathcal{A}$ from $\mathcal{A}$ has been considerably simplified by Gurevich and Harrington [GH82] thanks to the notion of *acceptance game*. The evaluation of an automaton $\mathcal{A}$ on an input tree $T$ can be modeled by an infinite acceptance game $\mathcal{G}(\mathcal{A}, T)$, played by two players $\mathsf{P}$ and $\mathsf{O}$, and such that $\mathcal{A}$ accepts $T$ when $\mathsf{P}$ has a winning strategy in $\mathcal{G}(\mathcal{A}, T)$. A typical (infinite) play $\chi$ in $\mathcal{G}(\mathcal{A}, T)$ has the form:

$$\cdot \xrightarrow{\mathsf{P}} \mathsf{g}_0 \xrightarrow{\mathsf{O}} (q_1, d_1) \xrightarrow{\mathsf{P}} \mathsf{g}_1 \xrightarrow{\mathsf{O}} \dots \xrightarrow{\mathsf{O}} (q_{n+1}, d_{n+1}) \xrightarrow{\mathsf{P}} \mathsf{g}_{n+1} \xrightarrow{\mathsf{O}} \dots$$

where $q_{k+1} = \mathsf{g}_k(d_{k+1})$, and $\mathsf{g}_k \in \delta_{\mathcal{A}}(q_k, T(d_1 \cdot \dots \cdot d_k))$ with $q_0 := q^{\imath}_{\mathcal{A}}$. Then $\chi$ is winning for $\mathsf{P}$ if the sequence of states $q^{\imath}_{\mathcal{A}}, q_1, \dots$ belongs to $\Omega_{\mathcal{A}}$, otherwise it is winning for $\mathsf{O}$. Note that $\mathsf{P}$ chooses transitions $\mathsf{g} : \mathfrak{D} \to Q_{\mathcal{A}}$ while $\mathsf{O}$ chooses tree directions $d \in \mathfrak{D}$. Hence, there is a bijection between accepting runs $R \in \mathcal{A}(T)$ and winning $\mathsf{P}$-strategies in $\mathcal{G}(\mathcal{A}, T)$. Since acceptance games are determined, $\mathcal{A}$ does not accept $T$ precisely when $\mathsf{O}$ has a winning strategy in $\mathcal{G}(\mathcal{A}, T)$. Gurevich and Harrington [GH82] show that in acceptance games, winning strategies can always be assumed to be finite state w.r.t. game positions of the form $(p, q) \in \mathfrak{D}^* \times Q_{\mathcal{A}}$, that is to only depend on a finite memory in addition to the game positions in $\mathfrak{D}^* \times Q_{\mathcal{A}}$[4]. This allows to devise an automaton $\sim\mathcal{A}$ which, using a usual projection operation, non-deterministically checks the existence of winning $\mathsf{O}$-strategies.

However, the construction of $\sim\mathcal{A}$ is still not trivial because the roles of $\mathsf{P}$ and $\mathsf{O}$ in acceptance games are not symmetric, so that dualizing the acceptance game of a non-deterministic automaton $\mathcal{A}$ does not directly give a *non-deterministic* automaton $\sim\mathcal{A}$. Since [MS87, EJ91, MS95] it is known that the construction of $\sim\mathcal{A}$ can be neatly decomposed using *alternating* automata. Alternating automata generalize non-deterministic automata with (following the presentation of [Wal02]), transition functions of the form

$$\delta_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D})) \tag{2}$$

A play in the acceptance game $\mathcal{G}(\mathcal{A}, T)$ with $\mathcal{A}$ alternating has the form

$$\cdot \xrightarrow{\mathsf{P}} \gamma_0 \xrightarrow{\mathsf{O}} (q_1, d_1) \xrightarrow{\mathsf{P}} \gamma_1 \xrightarrow{\mathsf{O}} \dots \xrightarrow{\mathsf{O}} (q_{n+1}, d_{n+1}) \xrightarrow{\mathsf{P}} \gamma_{n+1} \xrightarrow{\mathsf{O}} \dots$$

where $(q_{k+1}, d_{k+1}) \in \gamma_k$ and $\gamma_k \in \delta_{\mathcal{A}}(q_k, T(d_1. \cdots .d_k))$ with $q_0 := q^{\imath}_{\mathcal{A}}$. Hence, $\mathsf{P}$ chooses relations $\gamma_k \in \mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D})$ instead of functions $\mathsf{g}_n : \mathfrak{D} \to Q_{\mathcal{A}}$, while $\mathsf{O}$ chooses pairs $(q_{k+1}, d_{k+1}) \in \gamma_k$ instead of just tree directions $d_k \in \mathfrak{D}$. The main consequence is that $\mathsf{O}$ may now have to choose between pairs $(q'_{k+1}, d_{k+1}), (q''_{k+1}, d_{k+1}) \in \gamma_k$ with *different states* $q'_{k+1}, q''_{k+1}$ for the same tree direction $d_{k+1} \in \mathfrak{D}$.

The extra possibility for $\mathsf{O}$ to choose states in addition to tree directions allows to define a complement of $\mathcal{A}$ which (essentially) simulates $\mathcal{A}$ while reversing the roles of $\mathsf{P}$ and $\mathsf{O}$. This can be implemented with an alternating automaton[5] $\mathcal{A}^{\perp\!\!\!\perp}$ having the same states as $\mathcal{A}$. The idea is that the double powerset $\mathcal{P}(\mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}))$ in (2) represents disjunctive normal forms over $(Q_{\mathcal{A}} \times \mathfrak{D})$, so that the transition function $\delta_{\mathcal{A}^{\perp\!\!\!\perp}}$ of $\mathcal{A}^{\perp\!\!\!\perp}$ just takes $(q, \mathsf{a}) \in Q_{\mathcal{A}} \times \Sigma$ to a disjunctive normal form representing the dual of $\delta_{\mathcal{A}}(q, \mathsf{a})$. Then, if the acceptance condition of $\mathcal{A}^{\perp\!\!\!\perp}$ is the complement of $\Omega_{\mathcal{A}}$, it follows from game determinacy that $\mathcal{L}(\mathcal{A}^{\perp\!\!\!\perp})$ is the complement of $\mathcal{L}(\mathcal{A})$.

On the other hand, every alternating automaton $\mathcal{A}$ can be simulated by a non-deterministic automaton $!\mathcal{A}$ of exponential size (this is the *Simulation Theorem* [MS87, EJ91, MS95], see

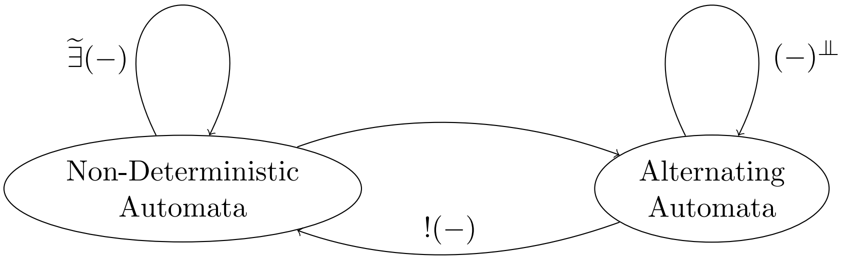
[4] This is trivial for $\mathsf{P}$-strategies but not for $\mathsf{O}$-strategies.

[5] $(-)^{\perp\!\!\!\perp}$ was noted $\sim(-)$ in [Rib15].

also §7.2), while non-deterministic automata are linearly embedded into alternating automata *via* the obvious mapping

$$\mathsf{g} : \mathfrak{D} \to Q_{\mathcal{A}} \quad \longmapsto \quad \{(\mathsf{g}(d), d) \mid d \in \mathfrak{D}\} \subseteq Q_{\mathcal{A}} \times \mathfrak{D} \tag{3}$$

The situation can be pictured as follows:



(4)

Accordingly, in most modern approaches to MSO on infinite trees, the complementation of non-deterministic tree automata can be decomposed as

$$\sim\!\mathcal{A} \quad = \quad !(\mathcal{A}^{\perp\!\!\!\perp}) \tag{5}$$

**1.4 Toward Linear Logic.** The model of [Rib15] consists in categories of two-player sequential games generalizing the usual acceptance games of tree automata. Using the notion of *uniform automata* (to be introduced in §3), the extension of [Rib15] proposed in this work shows that the decomposition depicted in (4) of the translation of MSO-formulae to non-deterministic tree automata *via* alternating automata corresponds to some extent to an ILL-structure:

- First, the usual direct synchronous product of alternating automata (which we denote $(-) \otimes (-)$) has a symmetric monoidal structure. Moreover, thanks to the monoidal-*closed* structure of $(-) \otimes (-)$ on uniform automata, the set of morphisms interpreting a sequent $T; \mathcal{A} \vdash \mathcal{B}$ is in bijection with the set of winning P-strategies in the acceptance game of an automaton $(\mathcal{A} \multimap \mathcal{B})$ over $T$. In particular, linear complements are obtained with

$$\mathcal{A}^{\rotatebox[origin=c]{180}{$\Upsilon$}} \quad \simeq \quad \mathcal{A} \multimap \rotatebox[origin=c]{180}{$\Upsilon$}$$

  (where $\rotatebox[origin=c]{180}{$\Upsilon$}$ is a particular automaton accepting no tree), with as expected $T \in \mathcal{L}(\mathcal{A}^{\rotatebox[origin=c]{180}{$\Upsilon$}})$ iff $T \notin \mathcal{L}(\mathcal{A})$.

- Second, we show that the simulation operation $!(-)$ satisfies the *deduction rules* of the usual modality $!(-)$ of ILL. Moreover, the symmetric monoidal product $(-) \otimes (-)$ is Cartesian on non-deterministic automata, so that the picture (4) is similar to the usual linear-non-linear adjunctions of models of ILL. Unfortunately, in our models the operation $!(-)$ is not a functor[6].

Furthermore, following the methodology of categorical logic, the categories proposed here and in [Rib15] are *indexed* (or *fibred*) over a base category $\mathbf{T}$ of trees, whose objects are alphabets and whose morphisms from $\Sigma$ to $\Gamma$ induce functions from $\Sigma$-labeled trees to $\Gamma$-labeled trees. In this setting, existential quantifications (in the categorical sense) are provided by a slight modification (denoted $\exists(-)$) of the usual projection $\widetilde{\exists}(-)$ mentioned in §1.2.

[6] It does not preserves composition, because of issues with positionality of strategies. Possible workarounds, leaved as future work, are discussed in §8.1.

**1.5 Toward Realizability Interpretations of MSO.** The ultimate motivation for the Curry-Howard approach to automata on infinite trees proposed in this paper, together with the underlying decomposition of the translation of MSO-formulae to tree automata *via* ILL, is to provide realizability interpretations of MSO (in the spirit of e.g. [SU06, Koh08]). We think that the model presented here (consolidating [Rib15]) is a preliminary step toward this goal. Let us briefly describe our main results in this direction.

Generalizing (1), our deduction system also manipulates sequents of the form

$$M \ ; \ \mathcal{A}_1, \dots, \mathcal{A}_n \vdash \mathcal{B} \tag{6}$$

(see §2.2) where $M$ is a $\mathbf{T}$-morphism, from say $\Sigma$ to $\Gamma$ and the automata $\mathcal{A}_1, \dots, \mathcal{A}_n, \mathcal{B}$ have input alphabet $\Gamma$. In the case $M$ is the identity $\mathbf{T}$-map on $\Sigma$, the sequent (6) is written

$$\Sigma \ ; \ \mathcal{A}_1, \cdots, \mathcal{A}_n \vdash \mathcal{B} \tag{7}$$

which in contrast with (1) and (6) does not mention any tree.

The symmetric monoidal closed structure, together with the categorical quantifiers and the interpretation of simulation as an exponential modality $!(-)$, allows to interpret proofs in the deduction system made of the rules depicted on Fig. 21, Fig. 22 and Fig. 25. From a proof $\mathscr{D}$ of a sequent (7), one can (compositionally w.r.t. the structure of $\mathscr{D}$) extract a finite-state strategy $\sigma$ in an infinite game of the form

$$\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \quad \multimap \quad \mathcal{B}$$

in which $\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n$ essentially evaluates the automata $\mathcal{A}_1, \dots, \mathcal{A}_n$ in parallel, while the *linear arrow* $\multimap$ is a synchronous restriction of the usual linear arrow of simple games.

We think that extraction of such realizers $\sigma$ from proofs can be interesting for instance in the following contexts.

- First, in case (7) is of the form
$$\mathbf{1} \ ; \ \vdash \exists_\Sigma \mathcal{N}$$
with $\mathcal{N}$ non-deterministic, then $\sigma$ is of the form $\langle T, \tau \rangle$, where $T$ is a $\Sigma$-labeled tree and $\tau$ is a winning strategy on $\mathcal{N}(T)$ (see §7.1.2), so that we indeed obtain a computational interpretation of proofs in the sense of §1.2.

- Assuming (7) is of the form
$$\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B} \tag{8}$$
then a $\Sigma$-labeled tree $T$ induces a *substitution functor* $T^\star$, whose action on $\sigma$ gives a function $T^\star(\sigma)$ taking any winning P-strategy $\tau$ on $\mathcal{A}(T)$ to a winning P-strategy $T^\star(\sigma) \circ \tau$ on $\mathcal{B}(T)$ (see Prop. 4.12).

  In other words, realizers of sequents of the form (8) can be composed (*via* substitution) with strategies $\tau$ on $\mathcal{A}(T)$ obtained by any possible mean.

More generally, the methodology of our deduction system and its realizability interpretation targets interactive proofs systems, allowing possible human simplifications or decompositions of the goals given to automatic tools, and moreover to *combine* the corresponding witnessing strategies. This principle may be interesting for instance in the following scenarios:

(a) If the automaton $\mathcal{A}$ in (8) is of the form $\exists_\Sigma \mathcal{C}$, then a finite-state strategy $\tau$ witnessing that a regular tree $T$ belongs to $\mathcal{L}(\mathcal{C})$[7] can be combined with $T^\star(\sigma)$.

(b) Thanks to the monoidal closed structure (§5.3), in case sequent (7) is of the form

$$\Sigma \ ; \ \mathcal{A}_1, \ldots, \mathcal{A}_n, (\mathcal{A} \multimap \mathcal{C}) \vdash \mathcal{B}$$

then the realizer $\sigma$ derived from $\mathscr{D}$ can be composed with a realizer $\tau : \mathcal{A} \multimap \mathcal{C}$ which can be obtained by any possible mean.

In particular, games of the form $\mathcal{A} \multimap \mathcal{C}$ are equivalent to $\omega$-regular games on finite graphs, so thanks to the Büchi-Landweber Theorem [BL69], one can decide if there is a strategy $\tau$ realizing the implication $\mathcal{A} \multimap \mathcal{C}$, and if such a strategy exists, then there exists a finite state one, which is moreover effectively computable from $\mathcal{A}$ and $\mathcal{C}$ (see Cor. 6.5).

Such finite state strategies $\tau : \mathcal{A} \multimap \mathcal{C}$, to be typically combined with realizers resulting from proofs, can also be obtained in the following ways.

(i) If $\mathcal{C} = \mathcal{N}^{\perp}$, and $\mathcal{A}$, $\mathcal{N}$ are non-deterministic such that $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{N}) = \emptyset$, then any (finite-state) $\mathsf{O}$-strategy witnessing $\mathcal{L}(\mathcal{A} \otimes \mathcal{N}) = \emptyset$ can be lifted to a (finite-state) realizer of $\mathcal{A} \multimap \mathcal{N}^{\perp}$ (Prop. 7.7).

(ii) In particular, if $\mathcal{L}(\tilde{\mathcal{A}}) \subseteq \mathcal{L}(\tilde{\mathcal{C}})$ for $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{C}}$ not-necessarily non-deterministic, then any (finite-state) $\mathsf{O}$-strategy witnessing $\mathcal{L}(!\tilde{\mathcal{A}} \otimes !(\tilde{\mathcal{C}}^{\perp})) = \emptyset$ can be lifted to a (finite-state) realizer of $!\tilde{\mathcal{A}} \multimap !(\tilde{\mathcal{C}}^{\perp})^{\perp}$ (Prop. 7.16).

**1.6 Outline.** The paper is organized as follows. We begin in §2 with a semi-formal overview of our approach and of [Rib15]. We then turn in §3 to our notion of *uniform automata* (motivated by monoidal closure), and give the formal material on game semantics required for that setting. Section 4 then deals with the fibred structure (which is essentially a refinement of [Rib15]), §5 formally presents the monoidal closure and the corresponding deduction rules, while §6 deals with quantifications. Finally, in §7 we concentrate on the Cartesian structure of non-deterministic automata and present the interpretation of the Simulation Theorem using the deduction rules of usual $!(-)$ ILL-exponential modalities. Further examples, showing that our setting can handle constructions of [CL08, SA05], are presented in [Rib17, App. C].

## 2 Categories of Games and Automata

The purpose of this long Section is twofold. First, in §2.1–2.5, we expose with slightly more details than in §1 the main ingredients and methodology of our approach, namely some simple and basic aspects of categorical logic and (simple) game semantics. Second, in §2.6–2.10 we briefly recall how this material is applied in [Rib15] in order to provide a partial fulfillment of the program announced here and in §1. Besides, this framework already allows to sketch in §2.8–2.9 the connection between the interpretation of MSO in tree automata and ILL mentioned in §1.3 and §1.4.

**2.1 Compositionality and Categorical Semantics.** The method of categorical semantics of proofs (see e.g. [LS86, AC98, Jac01, Mel09]) is to interpret *proofs* as *morphisms* of a category $\mathbb{C}$, such that $\mathbb{C}$ is equipped with some structure corresponding to the connectives and rules of

[7] With say $T$ and $\tau$ computed from $\mathcal{C}$ by Rabin's Tree Basis Theorem [Rab72], see e.g. [Tho97, Thm. 6.18].

the deduction system. For the moment, let us step back from acceptance games and rather consider run trees. Our task is thus to devise categories whose objects include all sets of the form $\mathcal{A}(T)$, for $\mathcal{A}$ an automaton and $T$ a tree, and such that the proofs of a sequent $T\ ;\ \mathcal{A} \vdash \mathcal{B}$ can be interpreted as morphisms from $\mathcal{A}(T)$ to $\mathcal{B}(T)$.

One characteristic of categorical semantics is that the very notion of category already imposes interpretations to be *compositional*. Recall that the sets of morphisms of a (locally small) category $\mathbb{C}$ come with associative *composition* operations

$$(-)\circ(-) \quad : \quad \mathbb{C}[B,C]\times\mathbb{C}[A,B] \quad \longrightarrow \quad \mathbb{C}[A,C] \qquad \text{for each } \mathbb{C}\text{-objects } A,B,C$$

and with identity morphisms $\mathrm{id}_A \in \mathbb{C}[A,A]$ which are neutral for composition:

$$f\circ \mathrm{id}_A \quad = \quad f \quad = \quad \mathrm{id}_B\circ f \qquad \text{for every } f\in\mathbb{C}[A,B] \tag{9}$$

Composition and identities provide the interpretations respectively of the following instances of the usual *cut* and *axiom* rules:

$$(\textsc{Cut}_0)\ \frac{T\ ;\ \mathcal{A}\vdash\mathcal{B} \qquad T\ ;\ \mathcal{B}\vdash\mathcal{C}}{T\ ;\ \mathcal{A}\vdash\mathcal{C}} \qquad\qquad \frac{}{T\ ;\ \mathcal{A}\vdash\mathcal{A}}\ (\textsc{Axiom})$$

The identity laws (9) imply for instance that the three derivations below must be interpreted by the same morphism:

$$\frac{\dfrac{}{T\ ;\ \mathcal{A}\vdash\mathcal{A}} \qquad \dfrac{\mathscr{D}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \qquad \frac{\mathscr{D}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \qquad \frac{\dfrac{\mathscr{D}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \qquad \dfrac{}{T\ ;\ \mathcal{B}\vdash\mathcal{B}}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \tag{10}$$

**2.2 Indexed Structure: Substitution and Quantification Rules.** Our categories actually involve a slight generalization of the usual notion of acceptance (either with run trees or games) of automata. This generalization is induced by the axiomatization of quantification and substitution in categorical logic (see e.g. [Jac01, LS86]).

Let us briefly discuss the usual setting of first-order logic over a multisorted individual language. The categorical semantics of existential quantifications is given by an adjunction which is usually represented as

$$\frac{\exists x\,\varphi(x)\vdash\psi}{\overline{\overline{\varphi(x)\vdash\psi}}} \qquad (x \text{ not free in } \psi) \tag{11}$$

This adjunction induces a bijection between (the interpretations of) proofs of the sequents $\varphi(x)\vdash\psi$ and $\exists x.\varphi(x)\vdash\psi$, that we informally denote

$$\varphi(x)\vdash\psi \qquad \simeq \qquad \exists x\,\varphi(x)\vdash\psi$$

Now, in general the variable $x$ will occur free in $\varphi$. As a consequence, in order to properly formulate (11) one should be able to interpret sequents of the form $\varphi(x)\vdash\psi$ with free variables. More generally, the formulae $\varphi$ and $\psi$ should be allowed to contain free variables distinct from $x$.

The idea underlying the general method (but see e.g. [Jac01] for details), is to first devise a base category $\mathbb{B}$ of individuals, whose objects interpret products of sorts of the individual language, and whose maps from say $\iota_1\times\cdots\times\iota_m$ to $o_1\times\cdots\times o_n$ represent $n$-tuples $(t_1,\dots,t_n)$ of terms $t_i$ of sort $o_i$ whose free variables are among $x_{\iota_1},\dots,x_{\iota_m}$ (with $x_{\iota_j}$ of sort $\iota_j$). Then, for each object $\iota=\iota_1\times\cdots\times\iota_n$ of $\mathbb{B}$, one devises a category $\mathbb{E}_\iota$ whose objects represent formulae with free variables among $x_{\iota_1}, \dots, x_{\iota_n}$, and whose morphisms interpret proofs. Furthermore, $\mathbb{B}$-morphisms

$$t = (t_1, \dots, t_n) \quad : \quad \iota_1 \times \cdots \times \iota_m \quad \longrightarrow \quad o_1 \times \cdots \times o_n$$

induce *substitution functors*

$$t^\star \quad : \quad \mathbb{E}_{o_1 \times \cdots \times o_n} \quad \longrightarrow \quad \mathbb{E}_{\iota_1 \times \cdots \times \iota_m}$$

The functor $t^\star$ takes (the interpretation of) a formula $\varphi$ with free variables among $y_{o_1}, \dots, y_{o_n}$ to (the interpretation of) the formula $\varphi[t_1/y_{o_1}, \dots, t_n/y_{o_n}]$ with free variables among $x_{\iota_1}, \dots, x_{\iota_m}$. Its action on the morphisms of $\mathbb{E}_{o_1 \times \cdots \times o_n}$ allows to interpret the *substitution rule*

$$\frac{\varphi \vdash \psi}{\varphi[t_1/y_{o_1}, \dots, t_n/y_{o_n}] \vdash \psi[t_1/y_{o_1}, \dots, t_n/y_{o_n}]}$$

In very good situations, the operation $(-)^\star$ is itself functorial. Among the morphisms of $\mathbb{B}$, one usually requires the existence of projections, say

$$\pi \quad : \quad o \times \iota \quad \longrightarrow \quad o$$

Projections induce substitution functors, called *weakening* functors

$$\pi^\star \quad : \quad \mathbb{E}_o \quad \longrightarrow \quad \mathbb{E}_{o \times \iota}$$

which simply allow to see formula $\psi(y_o)$ with free variable $y_o$ as a formula $\psi(y_o, x_\iota)$ with free variables among $y_o, x_\iota$ (but with no actual occurrence of $x_\iota$). Then the proper formulation of (11) is that existential quantification over $x_\iota$ is a functor

$$\exists x_\iota\, (-) \quad : \quad \mathbb{E}_{o \times \iota} \quad \longrightarrow \quad \mathbb{E}_o$$

which is left-adjoint to $\pi^\star$:

$$\frac{\exists x_\iota\, \varphi(x_\iota, y_o) \vdash \psi(y_o)}{\overline{\varphi(x_\iota, y_o) \vdash \pi^\star(\psi)(x_\iota, y_o)}} \tag{12}$$

(where $x_\iota$ does not occur free in $\psi$ since $\psi$ is assumed to be (interpreted as) an object of $\mathbb{E}_o$, thus replacing the usual side condition). Universal quantifications are dually axiomatized as right adjoints to weakening functors. In both cases, the adjunctions are subject to additional conditions (called the *Beck-Chevalley* conditions) which ensure that they are preserved by substitution.

Returning to automata and infinite trees, we will take as base category the following category $\mathbf{T}$ of trees.

**Definition 2.1** (The Base Category $\mathbf{T}$). *The objects of* $\mathbf{T}$ *are alphabets, and its morphisms from* $\Sigma$ *to* $\Gamma$*, denoted* $M, N, L, \dots$*, are functions of the form*

$$\bigcup_{n \in \mathbb{N}} \left(\Sigma^{n+1} \times \mathfrak{D}^n\right) \quad \longrightarrow \quad \Gamma$$

$\mathbf{T}$-morphisms are composed in the expected way (see §4.3 for details). Note that a map $M \in \mathbf{T}[\Sigma, \Gamma]$ takes for each $n \in \mathbb{N}$ a sequence of input characters $\bar{\mathsf{a}} = \mathsf{a}_0 \cdot \ldots \cdot \mathsf{a}_n \in \Sigma^{n+1}$ and a sequence of tree directions $p = d_1 \cdot \ldots \cdot d_n \in \mathfrak{D}^n$ to an output character $M(\bar{\mathsf{a}}, p) \in \Gamma$. In particular, we have $\mathbf{T}[\mathbf{1}, \Sigma] \simeq (\mathfrak{D}^* \to \Sigma)$, so each $\Sigma$-labeled $\mathfrak{D}$-ary tree $T$ corresponds to a morphism $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$. Moreover, $(\Sigma \to \Gamma)$-labeled trees $M : \mathfrak{D}^* \to (\Sigma \to \Gamma)$ induce $\mathbf{T}$-morphisms from $\Sigma$ to $\Gamma$.[8]

[8] The morphisms from $\Sigma$ to $\Gamma$ of the base category of [Rib15] are restricted to $(\Sigma \to \Gamma)$-labeled trees.

We will therefore not devise a single category $\mathbb{C}$, but a $\mathbf{T}$-indexed collection of categories $\mathbb{E}_\Sigma$, one for each alphabet $\Sigma$. Let us sketch the general idea with runs of non-deterministic automata. Given a non-deterministic automaton $\mathcal{A}$ over $\Gamma$ and a morphism $M \in \mathbf{T}[\Sigma, \Gamma]$, a $\Sigma$-*run* of $\mathcal{A}$ on $M$ is a tree

$$R \quad : \quad \mathfrak{D}^* \quad \longrightarrow \quad \Sigma \times Q_{\mathcal{A}}$$

such that $R(\varepsilon) = (\mathtt{a}_0, q^\imath_{\mathcal{A}})$ for some $\mathtt{a}_0 \in \Sigma$, and which respects the transition function

$$\delta_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Gamma \quad \longrightarrow \quad \mathcal{P}(\mathfrak{D} \to Q_{\mathcal{A}})$$

supplied with input characters $\mathtt{b} \in \Gamma$ computed by $M$ from tree positions $p = d_1 \cdot \ldots \cdot d_n$ and sequences of input characters $\overline{\mathtt{a}} = \mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_n$ where $\mathtt{a}_k$ is given by the $\Sigma$-component of $R(d_1 \cdot \ldots \cdot d_k) \in \Sigma \times Q_{\mathcal{A}}$. (So $\mathtt{a}_0$ is given by $R(\varepsilon)$ and $\mathtt{a}_n$ is given by $R(p)$.) Explicitly, $R$ is a $\Sigma$-run tree when for $p$ and $\overline{\mathtt{a}}$ as above, if $R(p)$ is labeled with state $q \in Q_{\mathcal{A}}$, then there exists a function $\mathtt{g} \in \delta_{\mathcal{A}}(q, \mathtt{b})$ with $\mathtt{b} = M(\overline{\mathtt{a}}, p)$ and such that for all $d \in \mathfrak{D}$, $R(p \cdot d)$ is labeled with state $\mathtt{g}(d)$. Such a $\Sigma$-run $R$ is *accepting* if the $Q_{\mathcal{A}}$-labeled tree

$$p \in \mathfrak{D}^* \quad \longmapsto \quad \pi(R(p)) \in Q_{\mathcal{A}}$$

is accepting in the usual sense (where $\pi : \Sigma \times Q_{\mathcal{A}} \to Q_{\mathcal{A}}$ is the second projection), that is if all its infinite paths belong to $\Omega_{\mathcal{A}}$. We let $\Sigma \vdash \mathcal{A}(M)$ be the set of accepting $\Sigma$-run trees of $\mathcal{A}$ on $M$, and simply write $\mathcal{A}(M)$ for $\Sigma \vdash \mathcal{A}(M)$ when $\Sigma$ is clear from the context.

Roughly speaking, for each $\Sigma$, the objects of the category $\mathbb{E}_\Sigma$ will include all sets of the form $\Sigma \vdash \mathcal{A}(M)$. Moreover, given $L \in \mathbf{T}[\Delta, \Sigma]$, the substitution functor

$$L^\star \quad : \quad \mathbb{E}_\Sigma \quad \longrightarrow \quad \mathbb{E}_\Delta$$

will take a $\mathbb{E}_\Sigma$-object $\Sigma \vdash \mathcal{A}(M)$ to the $\mathbb{E}_\Delta$-object $\Delta \vdash \mathcal{A}(M \circ L)$, where the $\mathbf{T}$-map $L \circ M \in \mathbf{T}[\Delta, \Gamma]$ is the $\mathbf{T}$-composition of $L$ and $M$ (assuming $M \in \mathbf{T}[\Sigma, \Gamma]$ as above).

This will induce sequents generalizing (1). For instance, given $M \in \mathbf{T}[\Sigma, \Gamma]$, we have sequents of the form

$$M \ ; \ \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B} \tag{13}$$

where $\mathcal{A}_1, \ldots, \mathcal{A}_n$ and $\mathcal{B}$ are automata over $\Gamma$. Such sequents are to be thought about as our version of "*open* sequents" or "sequents with free variables" (here of sort $\Sigma$), with the usual implicit prenex universal quantification, and are to be interpreted in the category $\mathbb{E}_\Sigma$ (the *fibre* over $\Sigma$). Substitution functors such as $L^\star : \mathbb{E}_\Sigma \to E_\Delta$ above will act in the deduction system *via* a substitution rule

$$(\textsc{Subst}) \ \frac{M \ ; \ \mathcal{A}_1, \ldots \mathcal{A}_n \vdash \mathcal{B}}{M \circ L \ ; \ \mathcal{A}_1, \ldots \mathcal{A}_n \vdash \mathcal{B}} \qquad \text{(where } M \in \mathbf{T}[\Sigma, \Gamma] \text{ and } L \in \mathbf{T}[\Delta, \Sigma]\text{)} \tag{14}$$

Let us briefly sketch the most important instances of this construction.

(a) Consider a $\mathbf{T}$-map $\dot{T} : \mathbf{T}[\mathbf{1}, \Sigma]$ representing a tree $T : \mathfrak{D}^* \to \Sigma$. Then the accepting runs of $\mathcal{A}$ on $T$ are in bijection with the accepting $\mathbf{1}$-run trees of $\mathcal{A}$ on $\dot{T}$:

$$(\mathbf{1} \vdash \mathcal{A}(\dot{T})) \quad \simeq \quad \mathcal{A}(T)$$

Sequents of the form (13) thus indeed generalize sequents of the form

$$T \ ; \ \mathcal{A}_1 \, , \ \ldots \ , \ \mathcal{A}_n \vdash \mathcal{B}$$

with $T : \mathfrak{D}^* \to \Sigma$ (as depicted in (1)), which are to be interpreted in the category $\mathbb{E}_{\mathbf{1}}$ (the fibre over $\mathbf{1}$), and are to be thought about as representing *closed* statements.

(b) Given a non-deterministic automaton $\mathcal{A}$ over $\Sigma$, we write $\Sigma \vdash \mathcal{A}$ (or even just $\mathcal{A}$ when no ambiguity arises) for $\Sigma \vdash \mathcal{A}(\mathrm{Id}_\Sigma)$ where the $\mathbf{T}$-identity $\mathrm{Id}_\Sigma \in \mathbf{T}[\Sigma, \Sigma]$ is given by

$$\mathrm{Id}_\Sigma(\bar{\mathtt{a}} \cdot \mathtt{a}, p) \quad := \quad \mathtt{a}$$

Consider now another automaton $\mathcal{B}$ also over $\Sigma$. Then we write

$$\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B} \tag{15}$$

(or even $\mathcal{A} \vdash \mathcal{B}$) for the sequent $\mathrm{Id}_\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B}$. The *provability interpretation* of (15) will be that if (15) is provable, then $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$. The *computational interpretation* of (15) will consist in a form of uniform simulation of $\mathcal{A}$ by $\mathcal{B}$ (generalizing the notion used with the *guidable automata* of [CL08]). Moreover, given a $\Sigma$-labeled tree $T$ seen as a morphism $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$, the interpretation of the substitution rule

$$\frac{\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B}}{\dot{T} \ ; \ \mathcal{A} \vdash \mathcal{B}}$$

will take a morphism $\sigma \in \mathbb{E}_\Sigma[\mathcal{A}, \mathcal{B}]$ to a function $\dot{T}^\star(\sigma) : \mathcal{A}(T) \to \mathcal{B}(T)$.

(c) Any ordinary function $\mathtt{f} : \Sigma \to \Gamma$ induces a morphism $[\mathtt{f}] \in \mathbf{T}[\Sigma, \Gamma]$ defined as

$$[\mathtt{f}] \quad : \quad (\bar{\mathtt{a}} \cdot \mathtt{a}, p) \quad \longmapsto \quad \mathtt{f}(\mathtt{a})$$

The action of the substitution functor $[\mathtt{f}]^\star : \mathbb{E}_\Gamma \to \mathbb{E}_\Sigma$ on $\mathbb{E}_\Gamma$-objects of the form $\Gamma \vdash \mathcal{A}$ can be internalized in automata. We indeed have

$$[\mathtt{f}]^\star(\Gamma \vdash \mathcal{A}) \quad = \quad \Sigma \vdash \mathcal{A}([\mathtt{f}]) \quad = \quad \Sigma \vdash \mathcal{A}[\mathtt{f}]$$

where the automaton $\mathcal{A}[\mathtt{f}]$ over $\Sigma$ is defined as $\mathcal{A}$ but with transition function:

$$\begin{array}{rccll} \delta_{\mathcal{A}[\mathtt{f}]} & : & Q_\mathcal{A} \times \Gamma & \longrightarrow & \mathcal{P}(\mathfrak{D} \to Q_\mathcal{A}) \\ & & (q, \mathtt{b}) & \longmapsto & \delta_\mathcal{A}(q, \mathtt{f}(\mathtt{b})) \end{array}$$

In particular:

(i) $\mathbf{T}$-maps from $\Sigma \times \Gamma$ to $\Sigma$ indeed include projections $[\pi] : \mathfrak{D}^* \to (\Sigma \times \Gamma \to \Sigma)$ induced by $\mathbf{Set}$-projections $\pi : \Sigma \times \Gamma \to \Sigma$.

(ii) Consider automata $\mathcal{A}_1, \dots, \mathcal{A}_n$ and $\mathcal{B}$, with $\mathcal{A}_i$ over $\Sigma_i$ and $\mathcal{B}$ over $\Gamma$. Consider furthermore $\mathbf{T}$-morphisms $M_i \in \mathbf{T}[\Delta, \Sigma_i]$ and $L \in \mathbf{T}[\Delta, \Gamma]$. Then we write

$$\Delta \ ; \ \mathcal{A}_1(M_1) \, , \, \dots \, , \, \mathcal{A}_n(M_n) \vdash \mathcal{B}(L)$$

for the sequent

$$\langle M_1, \dots, M_n, L \rangle \ ; \ \mathcal{A}_1[\pi_1] \, , \, \dots \, , \, \mathcal{A}_n[\pi_n] \vdash \mathcal{B}[\pi]$$

where

$$\langle M_1, \dots, M_n, L \rangle \ \in \ \mathbf{T}[\Delta, \, \Sigma_1 \times \dots \times \Sigma_n \times \Gamma]$$

is the $\mathbf{T}$-tupling of $M_1, \dots, M_n, L$ (see Cor. 4.7) and where the $\pi_i$'s and $\pi$ are suitable projections:

$$\begin{array}{lclcl} \pi_i & : & \Sigma_1 \times \dots \times \Sigma_n \times \Gamma & \longrightarrow & \Sigma_i \\ \pi & : & \Sigma_1 \times \dots \times \Sigma_n \times \Gamma & \longrightarrow & \Gamma \end{array}$$

Unless otherwise stated, all the sequents seen up to now must from now on be thought about as being of the more general form (15), that is a with a $\mathbf{T}$-map $M$ (of appropriate type) instead of the labeled tree $T$.

**2.3 Toward a Semantics for Implications.** The *provability interpretation* of sequents tells us that in sequents of the form

$$M \ ; \ \mathcal{A} \vdash \mathcal{B} \tag{16}$$

the symbol $\vdash$ is a form of implication. One of the main contribution of this work is that this implication can be internalized in automata. This will lead us outside of non-deterministic automata (see §3), but for the moment let us sketch some salient consequences this imposes to the interpretation of the symbol $\vdash$ in sequents of the form (16).

Assume that proofs of our deduction system are interpreted in categories $\mathbb{E}_{(-)}$ indexed over $\mathbf{T}$. Then, internalizing $\vdash$ in automata will imply that given automata $\mathcal{A}$ and $\mathcal{B}$ over $\Sigma$ there is an automaton $(\mathcal{A} \multimap \mathcal{B})$ over $\Sigma$ such that there is a bijection

$$\mathbb{E}_{\Sigma}[\mathcal{A}, \mathcal{B}] \quad \simeq \quad \Sigma \vdash (\mathcal{A} \multimap \mathcal{B})$$

that we informally write as

$$\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B} \qquad \simeq \qquad \Sigma \vdash (\mathcal{A} \multimap \mathcal{B})$$

In other words, morphisms in the interpretation of $\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B}$ will correspond to the $\Sigma$-runs of an automaton $(\mathcal{A} \multimap \mathcal{B})$ (on $\mathrm{Id}_{\Sigma}$). This could suggest to interpret $\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B}$ as the $\Sigma$-runs of an automaton of the form $\sim\!\mathcal{A} \vee \mathcal{B}$, where $\sim\!\mathcal{A}$ is the complement of $\mathcal{A}$ (in the sense of §1.1) and $(-) \vee (-)$ is a disjunction on automata. Let us rule out this possibility, at least for the natural implementation of $(-) \vee (-)$ with an *additive* disjunction $(-) \oplus (-)$. Given automata $\mathcal{A}_1$ and $\mathcal{A}_2$, both over $\Sigma$ and with $\mathcal{A}_i = (Q_{\mathcal{A}_i}, q^{\imath}_{\mathcal{A}_i}, \delta_{\mathcal{A}_i}, \Omega_{\mathcal{A}_i})$, the non-deterministic automaton $\mathcal{A}_1 \oplus \mathcal{A}_2$ over $\Sigma$ is

$$\mathcal{A}_1 \oplus \mathcal{A}_2 \quad := \quad (Q_{\mathcal{A}_1} + Q_{\mathcal{A}_2} + \mathbf{1} \, , \, \bullet \, , \, \delta_{\mathcal{A}_1 \oplus \mathcal{A}_2} \, , \, \Omega_{\mathcal{A}_1 \oplus \mathcal{A}_2})$$

where, *via* the embedding of $Q^{\mathfrak{D}}_{\mathcal{A}_1} + Q^{\mathfrak{D}}_{\mathcal{A}_2}$ into $(Q_{\mathcal{A}_1} + Q_{\mathcal{A}_2})^{\mathfrak{D}}$, we let

$$\delta_{\mathcal{A}_1 \oplus \mathcal{A}_2}(q, \mathtt{a}) \quad := \quad \begin{cases} \delta_{\mathcal{A}_1}(q^{\imath}_{\mathcal{A}_1}, \mathtt{a}) + \delta_{\mathcal{A}_2}(q^{\imath}_{\mathcal{A}_2}, \mathtt{a}) & \text{if } q = \bullet \in \mathbf{1} \\ \delta_{\mathcal{A}_i}(q, \mathtt{a}) & \text{if } q \in Q_{\mathcal{A}_i} \end{cases}$$

and where $\bullet, q_1, q_2, \ldots \in \Omega_{\mathcal{A}_1 \oplus \mathcal{A}_2}$ iff either $q^{\imath}_{\mathcal{A}_1}, q_1, q_2, \ldots \in \Omega_{\mathcal{A}_1}$ or $q^{\imath}_{\mathcal{A}_2}, q_1, q_2, \ldots \in \Omega_{\mathcal{A}_2}$.

Note that in $\mathbf{Set}$, for every $M : \mathfrak{D}^* \to (\Gamma \to \Sigma)$ we have

$$(\mathcal{A}_1 \oplus \mathcal{A}_2)(M) \quad \simeq \quad \mathcal{A}_1(M) + \mathcal{A}_2(M)$$

so in particular

$$\mathcal{L}(\mathcal{A}_1 \oplus \mathcal{A}_2) \quad = \quad \mathcal{L}(\mathcal{A}_1) \cup \mathcal{L}(\mathcal{A}_2)$$

Assume now that we take for $\mathbb{E}_{\Sigma}[\mathcal{A}(M), \mathcal{B}(N)]$ the set of $\Sigma$-runs of $(\sim\!\mathcal{A}[\pi_1] \oplus \mathcal{B}[\pi_2])$ on $\langle M, N \rangle$, that is the disjoint union $\sim\!\mathcal{A}(M) + \mathcal{B}(N)$. Then one faces the following difficulties.

- We have to devise identity morphisms, say[9]

  $$\mathrm{id}_{\mathcal{A}(M)} \quad \in \quad \sim\!\mathcal{A}(M) + \mathcal{A}(M)$$

  Assuming say $M \in \mathbf{T}[\Sigma, \Gamma]$, one may take for $\mathrm{id}_{\mathcal{A}(M)}$ either an accepting $\Sigma$-run of $\mathcal{A}$ on $M$ or an accepting $\Sigma$-run of $\sim\!\mathcal{A}$ on $M$. But this raises two problems. First, it may be undecidable whether a possibly non-recursive tree is accepted or rejected by a given

[9] Note that $\mathcal{A}[\pi_i](\langle M, M \rangle) = \mathcal{A}([\pi_i] \circ \langle M, M \rangle) = \mathcal{A}(M)$.

automaton. So this precludes any *general and effective* computational interpretation of the deduction system. Second, even if we restrict to trees $T$ for which acceptance is known to be decidable (e.g. trees generated by *higher-order recursion schemes* [Ong06]), there seem to be no *canonical choice* of an actual accepting run $\mathrm{id}_{\mathcal{A}(T)} \in (\sim\mathcal{A})(T) + \mathcal{A}(T)$.

- It is not clear how to define composition, say

$$(-) \circ (-) \quad : \quad (\sim\mathcal{B} + \mathcal{C}) \times (\sim\mathcal{A} + \mathcal{B}) \quad \longrightarrow \quad \sim\mathcal{A} + \mathcal{C}$$

  Given run trees, say

$$R_{\mathcal{C}} \ \in \ \mathcal{C} \ \subseteq \ \sim\mathcal{B} + \mathcal{C} \qquad \text{and} \qquad R_{\sim\mathcal{A}} \ \in \ \sim\mathcal{A} \ \subseteq \ \sim\mathcal{A} + \mathcal{B}$$

  there seems to be no obvious choice for $R_{\mathcal{C}} \circ R_{\sim\mathcal{A}} \in \sim\mathcal{A} + \mathcal{C}$. Both

$$R_{\mathcal{C}} \circ R_{\sim\mathcal{A}} \ := \ R_{\mathcal{C}} \qquad \text{and} \qquad R_{\mathcal{C}} \circ R_{\sim\mathcal{A}} \ := \ R_{\sim\mathcal{A}}$$

  may seem reasonable. But each of them breaks one of the equalities between the interpretations of the derivations depicted in (10).

**2.4 Simple Games.** It therefore seems that the categorical structure should involve some extra machinery. We use the technology of *game semantics* to devise categories generalizing the usual *acceptance games* of tree automata.

Game semantics provide models of typed $\lambda$-calculi, and can thus be used, *via* the Curry-Howard correspondence, to build compositional models of deduction systems. In game models, *types* (and, *via* Curry-Howard, formulae, or in our context automata instantiated with trees or $\mathbf{T}$-maps) are interpreted by two-player sequential games where the *Proponent* $\mathsf{P}$ ($\exists$loïse) and the *Opponent* $\mathsf{O}$ ($\forall$belard) play in turn moves, producing plays subject to specified rules. Originally, game models where proposed because they provide *fully complete* models of various $\lambda$-calculi, in the sense that finite $\mathsf{P}$-strategies are definable by $\lambda$-terms. On the other hand, the notion of strategy naturally encompasses infinite objects, and is thus well suited to deal with runs of automata on infinite trees.

There are different families of game models. We use *simple games* (see e.g. [Abr97, Hyl97]), which stem from Berry & Curien's *sequential data structures* (see e.g. [AC98, Chap. 14], but also [Mel05]).

**Definition 2.2** (Simple Games)**.**

- *A* simple game *$A$ has the form*

$$A \quad = \quad (A_{\mathsf{P}}\,,\, A_{\mathsf{O}}\,,\, L_A)$$

  *where $A_{\mathsf{P}}$ is the set of* $\mathsf{P}$-moves*, $A_{\mathsf{O}}$ is the set of* $\mathsf{O}$-moves *and $L_A \subseteq (A_{\mathsf{P}} + A_{\mathsf{O}})^*$ is a non-empty prefix-closed set of* legal plays.

  *We let $s, t, \ldots$ range of over plays and $m, n, \ldots$ range over moves.*

- *A* simple game with winning *is a simple game $A$ equipped with a* set of winning plays *(or* winning condition*) $\mathcal{W}_A \subseteq (A_{\mathsf{P}} + A_{\mathsf{O}})^\omega$.*

- *The sets* $\wp_A^{\mathsf{P}}$ *and* $\wp_A^{\mathsf{O}}$ *of resp.* positive *and* negative *plays on* $A$ *are*

$$\begin{array}{rcl} \wp_A^{\mathsf{P}} & := & (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* + (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* \cdot A_{\mathsf{P}} \\ \wp_A^{\mathsf{O}} & := & (A_{\mathsf{O}} \cdot A_{\mathsf{P}})^* + (A_{\mathsf{O}} \cdot A_{\mathsf{P}})^* \cdot A_{\mathsf{O}} \end{array}$$

*The game* $A$ *is* positive *(resp.* negative*) if all its legal plays are positive (resp. negative), that is if* $L_A \subseteq \wp_A^{\mathsf{P}}$ *(resp.* $L_A \subseteq \wp_A^{\mathsf{O}}$*). So* P *starts in a positive game and* O *starts in a negative one. A game is* polarized *if it is either positive or negative.*

- *A play for player* $\xi \in \{\mathsf{P}, \mathsf{O}\}$ *(also called a* $\xi$-play*) is either the empty play or a non-empty play in which* $\xi$ *played last (i.e. which ends with a* $\xi$*-move).*

In the case of a non-deterministic automaton $\mathcal{A}$ on a tree $T$, following the usual setting (see e.g. [Tho97, GTW02, PP04]), the acceptance game $\mathcal{G}(\mathcal{A}, T)$ is defined as the positive simple game with winning

$$\mathcal{G}(\mathcal{A}, T) \quad := \quad (\mathfrak{D} \to Q_{\mathcal{A}},\ Q_{\mathcal{A}} \times \mathfrak{D},\ L_{\mathcal{A}(T)},\ \mathcal{W}_{\mathcal{A}(M)})$$

whose legal plays $s \in L_{\mathcal{A}(T)}$ are sequences of the form

$$\begin{array}{rrcl} & s & = & \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \\ \text{or} & s & = & \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \mathsf{g}_n \end{array}$$

where $n \geq 0$, $q_{k+1} = \mathsf{g}_k(d_{k+1})$ and $\mathsf{g}_k \in \delta_{\mathcal{A}}(q_k, T(d_1 \cdot \ldots \cdot d_k))$ with $q_0 := q^\imath_{\mathcal{A}}$. Note that O only chooses the tree directions $d_1, \ldots, d_n \in \mathfrak{D}$, while P chooses from each O-play

$$\mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n)$$

a function $\mathsf{g}_n : \mathfrak{D} \to Q_{\mathcal{A}}$ available in $\delta_{\mathcal{A}}(q_n, T(d_1 \cdot \ldots \cdot d_n))$.

The winning plays $\chi \in \mathcal{W}_{\mathcal{A}(T)}$ are generated from the acceptance condition $\Omega_{\mathcal{A}}$ in the expected way. We let $\mathcal{W}_{\mathcal{A}(T)} \subseteq ((\mathfrak{D} \to Q_{\mathcal{A}}) \cdot (Q_{\mathcal{A}} \times \mathfrak{D}))^\omega$ consist of the infinite sequences

$$\chi \quad = \quad \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \ldots$$

such that $(q_k)_{k \in \mathbb{N}} \in \Omega_{\mathcal{A}}$ (where $q_0 := q^\imath_{\mathcal{A}}$).

We now come to the definition of strategies in simple games. A strategy for player P or O is what one expects. The formal definition of strategy below emphasizes P (strategies for O are defined by duality), because in categories of games, composition and identities are only defined for the strategies of the negative player P. Moreover, the manipulation of strategies as morphisms is more convenient when strategies are presented as sets of plays rather than as functions on plays.

**Definition 2.3** (Strategies)**.** *A* P-strategy *on* $A$ *is a non-empty set of legal* P*-plays* $\sigma \subseteq L_A$ *which is*

- P***-prefix-closed:*** *if* $s.t \in \sigma$ *and* $s$ *is a* P*-play then* $s \in \sigma$*, and*

- P***-deterministic:*** *if* $s.n \in \sigma$ *and* $s.m \in \sigma$ *then* $n = m$*.*

*Consider now a polarized game with winning* $A$*. Given a* P*-strategy* $\sigma$ *on* $A$ *and an* O*-play* $s \in L_A$*, we say that* $s$ *is an* O-interrogation of $\sigma$ *if either* $s = \varepsilon$ *and* $A$ *is positive, or if* $s = t.m$ *for some* $t \in \sigma$*. We say that* $\sigma$ *is* total *if for every* O*-interrogation* $s$ *of* $\sigma$*, we have* $s.n \in \sigma$ *for some* $n$*. A* winning *(*P*-)strategy on* $A$ *is a total strategy* $\sigma$ *s.t. for all* $\chi \in (A_{\mathsf{P}} + A_{\mathsf{O}})^\omega$*, we have* $\chi \in \mathcal{W}_A$ *whenever* $\exists^\infty k \in \mathbb{N}.\ \chi(0) \cdot \ldots \cdot \chi(k) \in \sigma$*.*

The notion of (total, winning) $\mathsf{O}$-strategy is defined by duality. Each game $A$ has a dual $\overline{A} = (A_{\mathsf{O}}, A_{\mathsf{P}}, L_A)$, where we moreover let $\mathcal{W}_{\overline{A}} := (A_{\mathsf{P}} + A_{\mathsf{O}})^\omega \setminus \mathcal{W}_A$ if $A$ is a game with winning. Note that $\overline{A}$ is polarized iff $A$ is polarized, and that $\overline{A}$ is positive (resp. negative) iff $A$ is negative (resp. positive). Then, we say that a (total, winning) $\mathsf{O}$-strategy on $A$ is a (total, winning) $\mathsf{P}$-strategy on $\overline{A}$.

In the case of the acceptance game $\mathcal{G}(\mathcal{A}, T)$ described above, a $\mathsf{P}$-strategy $\sigma$ is therefore a non-empty set of sequences of the form

$$s \quad = \quad \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_{n-1}, d_{n-1}) \cdot \mathsf{g}_{n-1}$$

such that

$$s \cdot (q_n, d_n) \cdot \mathsf{g}_n \;\in\; \sigma \quad\Longrightarrow\quad s \in \sigma$$

and

$$s \cdot (q_n, d_n) \cdot \mathsf{g}_n \;,\; s \cdot (q_n, d_n) \cdot \mathsf{g}'_n \;\in\; \sigma \quad\Longrightarrow\quad \mathsf{g}_n = \mathsf{g}'_n$$

Moreover, $\sigma$ is total iff for every

$$\mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \mathsf{g}_n \;\in\; \sigma$$

and for every $d_{n+1} \in \mathfrak{D}$, there is some $\mathsf{g}_{n+1}$ such that

$$\mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \mathsf{g}_n \cdot (\mathsf{g}_n(d_{n+1}), d_{n+1}) \cdot \mathsf{g}_{n+1} \;\in\; \sigma$$

It follows that a total $\mathsf{P}$-strategy $\sigma$ on $\mathcal{G}(\mathcal{A}, T)$ is uniquely determined by a run tree $R$ such that $R(\varepsilon) = q^\imath_{\mathcal{A}}$, and such that for every $d_1 \cdot \ldots \cdot d_n \cdot d_{n+1} \in \mathfrak{D}^{n+1}$,

$$R(d_1 \cdot \ldots \cdot d_n \cdot d_{n+1}) \quad = \quad \mathsf{g}_n(d_{n+1})$$

where, for $0 \leq k \leq n$, the $\mathsf{g}_k$ are unique such that

$$\mathsf{g}_0 \cdot (\mathsf{g}_0(d_1), d_1) \cdot \ldots \cdot (\mathsf{g}_{k-1}(d_k), d_k) \cdot \mathsf{g}_k \;\in\; \sigma$$

Hence (winning) total $\mathsf{P}$-strategies in $\mathcal{G}(\mathcal{A}, T)$ are in bijection with (accepting) runs of $\mathcal{A}$ on $T$. Moreover, the game $\mathcal{G}(\mathcal{A}, T)$ has the same winning strategies as the usual acceptance games (see e.g. [Tho97, GTW02, PP04]).

**Remark 2.4** (Tree Games v.s. Graph Games)**.** *The realm of game semantics was originally mostly developed with games on trees, (such as simple games [Abr97, Hyl97], but also traditional Hyland-Ong games [HO00]). However, more recent trends of game semantics based on graphs also emerged, in particular in the work of Melliès (see e.g. [Mel05, Mel06, Mel12]). Wrt. the context of this paper, one should in particular note the connection of [Mel06] between innocence and a notion of positionality (but for games equipped with an asynchronous notion of monoidal product).*

*On the other hand, the framework of [Rib15] was itself based on games on graphs, and games on graphs will be considered in this paper in relation with positionality in §7.2.3. However, that notion of positionality is not yet clearly connected with the notion of positionality of [Mel06], most notably because of the* synchronous *nature of our games, and in particular of their monoidal structure (see §2.8 and §4.1). Moreover, the main developments of this paper, based on the category* **DZ** *of simple zig-zag games (see §3) are technically easier with games on trees, and we adopt this setting for the categories of games considered here.*

Consider now the more general case of a non-deterministic $\mathcal{A}$ over $\Gamma$ instantiated with $M \in \mathbf{T}[\Sigma, \Gamma]$. The *substituted acceptance game* $\Sigma \vdash \mathcal{G}(\mathcal{A}, M)$ is the positive game with winning

$$\Sigma \vdash \mathcal{G}(\mathcal{A}, M) \quad := \quad (\Sigma \times (\mathfrak{D} \to Q_{\mathcal{A}}),\ Q_{\mathcal{A}} \times \mathfrak{D},\ L_{\mathcal{A}(M)},\ \mathcal{W}_{\mathcal{A}(M)})$$

whose legal plays $s \in L_{\mathcal{A}(M)}$ are sequences of the form

$$\begin{array}{rrcl} & s & = & (\mathtt{a}_0, \mathtt{g}_0) \cdot (q_1, d_1) \cdot (\mathtt{a}_1, \mathtt{g}_1) \cdot \ldots \cdot (q_n, d_n) \\ \text{or} & s & = & (\mathtt{a}_0, \mathtt{g}_0) \cdot (q_1, d_1) \cdot (\mathtt{a}_1, \mathtt{g}_1) \cdot \ldots \cdot (q_n, d_n) \cdot (\mathtt{a}_n, \mathtt{g}_n) \end{array}$$

where $n \geq 0$, $q_{k+1} = \mathtt{g}_k(d_k)$ and $\mathtt{g}_k \in \delta_{\mathcal{A}}(q_k, M(\mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_k, d_1 \cdot \ldots \cdot d_k)$ with $q_0 := q^{\imath}_{\mathcal{A}}$. We let $\mathcal{W}_{\mathcal{A}(M)} \subseteq ((\Sigma \times (\mathfrak{D} \to Q_{\mathcal{A}})) \cdot (Q_{\mathcal{A}} \times \mathfrak{D}))^{\omega}$ consist of the infinite sequences

$$(\mathtt{a}_0, \mathtt{g}_0) \cdot (q_1, d_1) \cdot (\mathtt{a}_1, \mathtt{g}_1) \cdot \ldots \cdot (q_n, d_n) \cdot \ldots$$

such that $(q_k)_{k \in \mathbb{N}} \in \Omega_{\mathcal{A}}$ (where $q_0 := q^{\imath}_{\mathcal{A}}$).

Now $\mathsf{P}$ chooses the input characters $\mathtt{a}_k \in \Sigma$ in addition to the functions $\mathtt{g}_k : \mathfrak{D} \to Q_{\mathcal{A}}$. In a $\mathsf{P}$-play

$$(\mathtt{a}_0, \mathtt{g}_0) \cdot (q_1, d_1) \cdot (\mathtt{a}_1, \mathtt{g}_1) \cdot \ldots \cdot (q_n, d_n) \cdot (\mathtt{a}_n, \mathtt{g}_n)$$

the sequence of input characters $\mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_n$ chosen by $\mathsf{P}$ and the sequence of tree directions $d_1 \cdot \ldots \cdot d_n$ chosen by $\mathsf{O}$ are given as input to $M$, which produces a character $\mathtt{b} \in \Gamma$ and the function $\mathtt{g}_n$ chosen by $\mathsf{P}$ must belong to $\delta_{\mathcal{A}}(q_n, \mathtt{b})$.

Games of the form $\Sigma \vdash \mathcal{G}(\mathcal{A}, M)$ generalize usual acceptance games of the form $\mathcal{A}(T)$. Given a $\mathbf{T}$-map $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$ corresponding to a $\Sigma$-labeled tree $T$, the plays of the game $\mathbf{1} \vdash \mathcal{G}(\mathcal{A}, \dot{T})$ are of the form

$$(\bullet, \mathtt{g}_0) \cdot (q_1, d_1) \cdot (\bullet, \mathtt{g}_1) \cdot \ldots \cdot (q_n, d_n) \cdot (\bullet, \mathtt{g}_n)$$

and are thus in bijection with the plays of $\mathcal{A}(T)$. More generally, total (winning) $\mathsf{P}$-strategies on $\Gamma \vdash \mathcal{G}(\mathcal{A}, M)$ are in bijection with (accepting) $\Gamma$-runs of $\mathcal{A}$ on $M$.

**Notation 2.5** ($\mathcal{A}(T)$ and $\Gamma \vdash \mathcal{A}(M)$ as games)**.** *From now on, we will write $\Gamma \vdash \mathcal{A}(M)$ (resp. $\mathcal{A}(T)$) to denote indifferently a game $\Gamma \vdash \mathcal{G}(\mathcal{A}, M)$ (resp. $\mathcal{G}(\mathcal{A}, T)$) or a set of run trees (in the sense of §2.2).*

**2.5 Game Semantics: Linear Arrow Games and Copy-Cat.** Recall from §2.3 that a sequent $M$ ; $\mathcal{A} \vdash \mathcal{B}$ should be thought as a form of implication, but that the runs of the automaton ${\sim}\mathcal{A} \oplus \mathcal{B}$ seemed not to convey the right information. The first encountered difficulty concerned the existence of *canonical* identities $\mathrm{id}_{\mathcal{A}(M)} \in \mathbb{E}_{\Sigma}[\mathcal{A}(M), \mathcal{A}(M)]$ if the homset $\mathbb{E}_{\Sigma}[\mathcal{A}(M), \mathcal{A}(M)]$ were to be the set of accepting runs or winning $\mathsf{P}$-strategies $({\sim}\mathcal{A})(M) + \mathcal{A}(M)$. The solution of *game semantics* is to devise, from component games $A$ and $B$, an implication game $A \multimap_{\mathbf{SG}} B$ in which:

(a) The set of moves of $A \multimap_{\mathbf{SG}} B$ is the disjoint union of the sets of moves of $A$ and $B$, and the components $A$ and $B$ can be interleaved in plays on $A \multimap_{\mathbf{SG}} B$.

(b) $\mathsf{O}$ plays first in $A \multimap_{\mathbf{SG}} B$, and then the plays in $A \multimap_{\mathbf{SG}} B$ alternate between $\mathsf{P}$ and $\mathsf{O}$.

(c) The role of $\mathsf{P}$ and $\mathsf{O}$ are reversed in component $A$ and are preserved in component $B$ (*i.e.* $\mathsf{P}$ in $A \multimap_{\mathbf{SG}} B$ plays as $\mathsf{O}$ in $A$ and as $\mathsf{P}$ in $B$).

(d) In the case of simple games (see e.g. [Abr97, Hyl97]), $\mathsf{P}$ can switch between components at any of its moves, but $\mathsf{O}$ must stay in the same component (this is the *switching condition*).

**Definition 2.6** (Linear Arrow Games)**.** *Given polarized simple games $A$ and $B$ of the same* polarity*, the* linear arrow game $A \multimap_{\mathbf{SG}} B$ *is the negative game*

$$A \multimap_{\mathbf{SG}} B \quad := \quad (A_{\mathsf{O}} + B_{\mathsf{P}},\ A_{\mathsf{P}} + B_{\mathsf{O}},\ L_{A \multimap_{\mathbf{SG}} B})$$

*where $L_{A \multimap_{\mathbf{SG}} B}$ consists of those negative plays $s$ such that $s_{\restriction A} \in L_A$ and $s_{\restriction B} \in L_B$, where $s_{\restriction A}$ is the restriction of $s$ to $A_{\mathsf{P}} + A_{\mathsf{O}}$, and similarly for $s_{\restriction B}$.*

Let us check that $A \multimap_{\mathbf{SG}} B$ satisfies the *switching condition* (d) above (the other conditions (a)-(c) are direct consequences of the definitions): given a legal $\mathsf{O}$-play $s = t \cdot n \cdot m$, either $n, m$ are both in component $A$, or they are both in component $B$. Indeed, note that since $A \multimap_{\mathbf{SG}} B$ is negative, its legal $\mathsf{O}$-plays are odd-length. So if $s$ is a legal $\mathsf{O}$-play, then the lengths of $s_{\restriction A}$ and $s_{\restriction B}$ cannot have the same parity. Assume now that $s = t \cdot n \cdot m$ with $n$ and $m$ in different components. Since $A$ and $B$ are assumed to be of the same polarity, the moves $n$ and $m$ are of different polarities w.r.t. $A$ and $B$, so they are of the same polarity as moves of $A \multimap_{\mathbf{SG}} B$, contradicting the legality of $s$.

Simple games and (winning) strategies form a category $\mathbf{SG}^{(\mathrm{W})}$, whose objects are simple games (with winning), and whose morphisms from $A$ to $B$ are (winning) $\mathsf{P}$-strategies $\sigma : A \multimap_{\mathbf{SG}} B$. We refer to [Abr97, Hyl97, AC98] for full treatments, and in particular to [Abr97, Hyl97] for totality and winning. The general notion of winning in games of the form $A \multimap_{\mathbf{SG}} B$ is a bit technical. In this paper, we only need to consider the case of infinite plays on $A \multimap_{\mathbf{SG}} B$ whose projections on $A$ and $B$ are both infinite. We say that such a play is winning for $\mathsf{P}$ in $A \multimap_{\mathbf{SG}} B$ iff its projection on $B$ is winning for $\mathsf{P}$ whenever its projection on $A$ is winning for $\mathsf{P}$ (with the original polarities of $A$).

Consider now the definition of the identity strategy $\mathrm{id}_{\mathcal{A}(M)}$ in $\Sigma \vdash \mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$. Since $\mathsf{O}$ must begin in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$, but it is $\mathsf{P}$ who begins in the right component $\mathcal{A}(M)$, it follows that $\mathsf{O}$ must begin in the left component $\mathcal{A}(M)$ (taking the role of $\mathsf{P}$ in that component). It is then easy to define an identity "*copy-cat*" strategy for $\mathsf{P}$, which always switches component and copies the previous $\mathsf{O}$-move from the other component. A play of this strategy is depicted in Fig. 1 (where plays grow from top to bottom). Note that $\mathsf{P}$ plays in particular the same input characters $\mathsf{a}_k \in \Sigma$ and the same tree directions $d_k \in \mathfrak{D}$ as proposed by $\mathsf{O}$. Formally, $\mathrm{id}_{\mathcal{A}(M)}$ is the unique strategy in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$ such that

$$\mathrm{id}_{\mathcal{A}(M)} \quad = \quad \{ s \in L_{\mathcal{A}(M)^0 \multimap_{\mathbf{SG}} \mathcal{A}(M)^1} \mid s_{\restriction \mathcal{A}(M)^0} = s_{\restriction \mathcal{A}(M)^1} \} \tag{17}$$

(where we have written $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$ as $\mathcal{A}(M)^0 \multimap_{\mathbf{SG}} \mathcal{A}(M)^1$ in order to distinguish the two copies of $\mathcal{A}(M)$).

In particular, by construction of $\mathrm{id}_{\mathcal{A}(M)}$, the same sequences of states are produced in both copies of $\mathcal{A}(M)$. But such sequences are either winning for $\mathsf{P}$ in $\mathcal{A}(M)$ or are winning for $\mathsf{O}$ in $\mathcal{A}(M)$. So they are winning for $\mathsf{P}$ in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$.

**2.6 Linear Synchronous Arrow Games.** The morphisms of our categories of automata are based on the linear *synchronous arrow games* of [Rib15] (adapted to the base category $\mathbf{T}$). Synchronous arrow games are restrictions of the linear arrow of simple games between substituted acceptance games, in which $\mathsf{P}$ has to play the same input characters $\mathsf{a}$ and the same tree directions $d$ as proposed by $\mathsf{O}$.

Let $\mathcal{A}$ and $\mathcal{B}$ be non-deterministic automata over resp. $\Gamma$ and $\Delta$, and consider $M \in \mathbf{T}[\Sigma, \Gamma]$ and $N \in \mathbf{T}[\Sigma, \Delta]$. We will define the *synchronous arrow game*

$$\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N) \tag{18}$$

| | $\mathcal{A}(M)$ | $\overset{\mathrm{id}_{\mathcal{A}(M)}}{\multimap}_{\mathbf{SG}}$ | $\mathcal{A}(M)$ | |
|---|---|---|---|---|
| | $(\varepsilon, \varepsilon, q^\imath_\mathcal{A})$ | | $(\varepsilon, \varepsilon, q^\imath_\mathcal{A})$ | |
| O | $(\mathsf{a}_0, \mathsf{g}_0)$ | | $\vdots$ | if $\mathsf{g}_0 \in \delta_\mathcal{A}(q^\imath_\mathcal{A}, M(\mathsf{a}_0, \varepsilon))$ |
| P | $\vdots$ | | $(\mathsf{a}_0, \mathsf{g}_0)$ | |
| O | $\vdots$ | | $(q_1, d_1)$ | if $q_1 = \mathsf{g}_0(d_1)$ |
| P | $(q_1, d_1)$ | | $\vdots$ | |
| | $(d_1, \mathsf{a}_0, q_1)$ | | $(d_1, \mathsf{a}_0, q_1)$ | |
| | $\vdots$ | | $\vdots$ | |
| | $(p, \bar{\mathsf{a}}, q_n)$ | | $(p, \bar{\mathsf{a}}, q_n)$ | where $p = d_1 \cdot \ldots \cdot d_n$ |
| O | $(\mathsf{a}_n, \mathsf{g}_n)$ | | $\vdots$ | if $\mathsf{g}_n \in \delta_\mathcal{A}(q_n, M(\bar{\mathsf{a}} \cdot \mathsf{a}_n, p))$ |
| P | $\vdots$ | | $(\mathsf{a}_n, \mathsf{g}_n)$ | |
| O | $\vdots$ | | $(q_{n+1}, d_{n+1})$ | if $q_{n+1} = \mathsf{g}_n(d_{n+1})$ |
| P | $(q_{n+1}, d_{n+1})$ | | $\vdots$ | |
| | $(p \cdot d_{n+1},\ \bar{\mathsf{a}} \cdot \mathsf{a}_n,\ q_{n+1})$ | | $(p \cdot d_{n+1},\ \bar{\mathsf{a}} \cdot \mathsf{a}_n,\ q_{n+1})$ | |
| | $\vdots$ | | $\vdots$ | |

Figure 1: A play of the *copy-cat* identity strategy $\mathrm{id}_{\mathcal{A}(M)}$

also noted $\mathcal{A}(M) \multimap \mathcal{B}(N)$ when $\Sigma$ is clear from the context.

The game $\mathcal{A}(M) \multimap \mathcal{B}(N)$ will be a subgame of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$. It can be seen as a restriction of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ to plays which are *synchronous*, in the sense that $\mathcal{A}$ and $\mathcal{B}$ are evaluated along the same path in $\mathfrak{D}^\omega$, while $M$ and $N$ read the same input characters from $\Sigma$. The synchronous plays of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ are defined using the following notion of *trace*. Let

$$\mathrm{Tr}_\Sigma \quad := \quad (\Sigma \cdot \mathfrak{D})^* + (\Sigma \cdot \mathfrak{D})^* \cdot \Sigma$$

and define the *trace function* $\mathrm{tr}_{\mathcal{A}(M)} : L_{\mathcal{A}(M)} \longrightarrow \mathrm{Tr}_\Sigma$ inductively as follows

$$\begin{array}{rcl} \mathrm{tr}_{\mathcal{A}(M)}(\varepsilon) & := & \varepsilon \\ \mathrm{tr}_{\mathcal{A}(M)}(s \cdot (\mathsf{a}, \mathsf{g})) & := & \mathrm{tr}_{\mathcal{A}(M)}(s) \cdot \mathsf{a} \\ \mathrm{tr}_{\mathcal{A}(M)}(s \cdot (q, d)) & := & \mathrm{tr}_{\mathcal{A}(M)}(s) \cdot d \end{array}$$

We let the *trace* of a play $s \in L_{\mathcal{A}(M)}$ be the sequence $\mathrm{tr}_{\mathcal{A}(M)}(s)$. The trace function $\mathrm{tr}_{\mathcal{B}(N)} : L_{\mathcal{B}(N)} \longrightarrow \mathrm{Tr}_\Sigma$ is defined similarly. Note that both $\mathrm{tr}_{\mathcal{A}(M)}$ and $\mathrm{tr}_{\mathcal{B}(N)}$ have the same codomain $\mathrm{Tr}_\Sigma$, which only depends on the input alphabet of $M$ and $N$. Consider now a legal play $s$ in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$. We say that $s$ is *synchronous* if

$$\mathrm{tr}_{\mathcal{A}(M)}(s_{\restriction \mathcal{A}(M)}) \quad = \quad \mathrm{tr}_{\mathcal{B}(N)}(s_{\restriction \mathcal{B}(N)})$$

Note that trace functions are length-preserving, so that the trace of a play $s$ always has the same length as $s$. Hence if $s$ is a synchronous play in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$, then $s_{\restriction A(M)}$ and $s_{\restriction B(N)}$ have the same length, so that $s$ is even length. It follows that the synchronous plays of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ must be $\mathsf{P}$-plays.

| $\Sigma$ | $\mathcal{A}(M)$ | $\multimap$ | $\mathcal{B}(N)$ | |
|---|---|---|---|---|
| | $(\varepsilon, \varepsilon, q^{\imath}_{\mathcal{A}})$ | | $(\varepsilon, \varepsilon, q^{\imath}_{\mathcal{B}})$ | |
| | $\vdots$ | | $\vdots$ | |
| | $(p, \bar{\mathsf{a}}, q_{\mathcal{A}})$ | | $(p, \bar{\mathsf{a}}, q_{\mathcal{B}})$ | |
| O | $(\mathsf{a}, \mathsf{g}_{\mathcal{A}})$ | | $\vdots$ | if $\mathsf{g}_{\mathcal{A}} \in \delta_{\mathcal{A}}(q_{\mathcal{A}}, M(\bar{\mathsf{a}} \cdot \mathsf{a}, p))$ |
| P | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{B}})$ | if $\mathsf{g}_{\mathcal{B}} \in \delta_{\mathcal{B}}(q_{\mathcal{B}}, N(\bar{\mathsf{a}} \cdot \mathsf{a}, p))$ |
| O | $\vdots$ | | $(q'_{\mathcal{B}}, d)$ | if $q'_{\mathcal{B}} = \mathsf{g}_{\mathcal{B}}(d)$ |
| P | $(q'_{\mathcal{A}}, d)$ | | $\vdots$ | if $q'_{\mathcal{A}} = \mathsf{g}_{\mathcal{A}}(d)$ |
| | $(p \cdot d,\ \bar{\mathsf{a}} \cdot \mathsf{a},\ q'_{\mathcal{A}})$ | | $(p \cdot d,\ \bar{\mathsf{a}} \cdot \mathsf{a},\ q'_{\mathcal{B}})$ | |
| | $\vdots$ | | $\vdots$ | |

Figure 2: A typical synchronous play over $\Sigma$

We define the game $\mathcal{A}(M) \multimap \mathcal{B}(N)$ as the game $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$, but with as legal plays the legal plays of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ which are prefixes of synchronous plays. It follows that a P-strategy $\sigma : \mathcal{A}(M) \multimap \mathcal{B}(N)$ is a P-strategy $\sigma : \mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ whose plays are all synchronous. We call such strategies *synchronous*. In particular, the identity copy-cat strategy $\mathrm{id}_{\mathcal{A}(M)} : \mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$ is a synchronous strategy.

A typical synchronous play in $\mathcal{A}(M) \multimap \mathcal{B}(N)$ is depicted in Fig. 2. Note that synchronous plays must have the same zig-zag shape as the copy-cat plays, and moreover that O actually chooses both the input characters $\mathsf{a} \in \Sigma$ and the tree directions $d \in \mathfrak{D}$. This follows from the fact that in the game $\mathcal{A}(M) \multimap \mathcal{B}(N)$, O must begin in the component $\mathcal{A}(M)$, choosing in particular some $\mathsf{a} \in \Sigma$. Then, by synchronicity, P must switch to component $\mathcal{B}(N)$ and play a move containing *the same* $\mathsf{a} \in \Sigma$. Since O cannot switch component, its next move must be in component $\mathcal{B}(N)$, and so in particular contain some $d \in \mathfrak{D}$. But then, again by synchronicity, P must switch to component $\mathcal{A}(M)$ and play a move containing the same $d \in \mathfrak{D}$.

Synchronous arrow games are equipped with the winning conditions mentioned in §2.5. Given an infinite play $\chi$ in $\mathcal{A}(M) \multimap \mathcal{B}(N)$ whose projections on $\mathcal{A}(M)$ and $\mathcal{B}(N)$ are both infinite, we say that $\chi$ is *winning for* P if its projection on $\mathcal{B}(N)$ is winning for P whenever its projection on $\mathcal{A}(M)$ is winning for P.

Synchronous arrow games generalize usual acceptance games. Given an automaton $\mathcal{A}$ over $\Sigma$ and a $\Sigma$-labeled tree $T$, we have seen that plays and total (winning) P-strategies in $\mathcal{A}(T)$ are in bijection with plays and total (winning) P-strategies in $\mathbf{1} \vdash \mathcal{A}(\dot{T})$. Plays and total (winning) P-strategies in $\mathbf{1} \vdash \mathcal{A}(T)$ are in turn in bijection with plays and total (winning) P-strategies in $\mathbf{1} \vdash \mathbf{I}(\dot{T}) \multimap \mathcal{A}(\dot{T})$. Here, $\mathbf{I}$ is a *unit automaton* over $\Sigma$, with state set $\mathbf{1} \simeq \{\bullet\}$ (and thus $\bullet$ initial), acceptance condition $\Omega_{\mathbf{I}} := \mathbf{1}^{\omega}$, and transition function of the form

$$\delta_{\mathbf{I}} \quad : \quad (\bullet, \mathsf{a}) \quad \longmapsto \quad \{(d \in \mathfrak{D} \mapsto \bullet)\}$$

A typical play of $\mathbf{1} \vdash \mathbf{I}(\dot{T}) \multimap \mathcal{A}(\dot{T})$ is depicted on Fig. 4 (left). Note that in component $\mathbf{I}(\dot{T})$, there is always exactly one possible O-move since $\delta_{\mathbf{I}}(\bullet, \_)$ is a singleton, and that by synchronicity, the P-move $(d, \bullet)$ in completely determined by the $d \in \mathfrak{D}$ chosen by O.

**2.7 Game Semantics: Composition.** We now briefly sketch the composition of synchronous strategies. We refer to e.g. [Abr97, Hyl97] for a description of composition in **SG**, and to [Rib15]

| $\Sigma$ | $\mathcal{A}(M)$ | $\overset{\sigma}{\multimap}$ | $\mathcal{B}(N)$ |
|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{A}})$ | | $(\varepsilon, q^\imath_{\mathcal{B}})$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, q_{\mathcal{A}})$ | | $(p, q_{\mathcal{B}})$ |
| O | $(\mathsf{a}, \mathsf{g}_{\mathcal{A}})$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{B}})$ |
| O | $\vdots$ | | $(q'_{\mathcal{B}}, d)$ |
| P | $(q'_{\mathcal{A}}, d)$ | | $\vdots$ |
| | $(p.d,\ q'_{\mathcal{A}})$ | | $(p.d,\ q'_{\mathcal{B}})$ |
| | $\vdots$ | | $\vdots$ |

| $\Sigma$ | $\mathcal{B}(N)$ | $\overset{\tau}{\multimap}$ | $\mathcal{C}(L)$ |
|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{B}})$ | | $(\varepsilon, q^\imath_{\mathcal{C}})$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, q_{\mathcal{B}})$ | | $(p, q_{\mathcal{C}})$ |
| O | $(\mathsf{a}, \mathsf{g}_{\mathcal{B}})$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{C}})$ |
| O | $\vdots$ | | $(q'_{\mathcal{C}}, d)$ |
| P | $(q'_{\mathcal{B}}, d)$ | | $\vdots$ |
| | $(p.d,\ q'_{\mathcal{B}})$ | | $(p.d,\ q'_{\mathcal{C}})$ |
| | $\vdots$ | | $\vdots$ |

| $\Sigma$ | $\mathcal{A}(M)$ | $\overset{\sigma}{\multimap}$ | $\mathcal{B}(N)$ | $\overset{\tau}{\multimap}$ | $\mathcal{C}(L)$ | |
|---|---|---|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{A}})$ | | $(\varepsilon, q^\imath_{\mathcal{B}})$ | | $(\varepsilon, q^\imath_{\mathcal{C}})$ | |
| | $\vdots$ | | $\vdots$ | | $\vdots$ | |
| | $(p, q_{\mathcal{A}})$ | | $(p, q_{\mathcal{B}})$ | | $(p, q_{\mathcal{C}})$ | |
| O | $(\mathsf{a}, \mathsf{g}_{\mathcal{A}})$ | | $\vdots$ | | $\vdots$ | O |
| | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{B}})$ | | $\vdots$ | |
| P | $\vdots$ | | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{C}})$ | P |
| O | $\vdots$ | | $\vdots$ | | $(q'_{\mathcal{C}}, d)$ | O |
| | $\vdots$ | | $(q'_{\mathcal{B}}, d)$ | | $\vdots$ | |
| P | $(q'_{\mathcal{A}}, d)$ | | $\vdots$ | | $\vdots$ | P |
| | $(p.d,\ q'_{\mathcal{A}})$ | | $(p.d,\ q'_{\mathcal{B}})$ | | $(p.d,\ q'_{\mathcal{C}})$ | |
| | $\vdots$ | | $\vdots$ | | $\vdots$ | |

Figure 3: Interaction of strategies on non-deterministic automata

for the particular case of synchronous strategies. Composition of strategies for the precise setting of this paper will be detailed in §3. Consider P-strategies $\sigma : \mathcal{A}(M) \multimap \mathcal{B}(N)$ and $\tau : \mathcal{B}(N) \multimap \mathcal{C}(L)$ as in Fig. 3 (top). Their composite

$$\tau \circ \sigma \quad : \quad \mathcal{A}(M) \quad \multimap \quad \mathcal{C}(L)$$

is obtained by making $\sigma$ and $\tau$ interact in their common component $\mathcal{B}(N)$, as depicted on Fig. 3 (bottom). The crucial observation is that in an interaction of $\sigma$ and $\tau$ in component $\mathcal{B}(N)$, all the P-moves are played by $\sigma$ and all the O-moves are played by $\tau$. It follows that the interactions of $\sigma$ and $\tau$ in component $\mathcal{B}(N)$ are completely determined by $\sigma$ and $\tau$ and the O-moves in $\mathcal{A}(M) \multimap \mathcal{C}(L)$. The composite strategy $\tau \circ \sigma$ is then obtained by hiding the interaction of $\sigma$ and $\tau$ in their common component $\mathcal{B}(N)$ (see Fig. 4, right). It well-known that $\tau \circ \sigma$ is winning if $\tau$ and $\sigma$ are both winning.

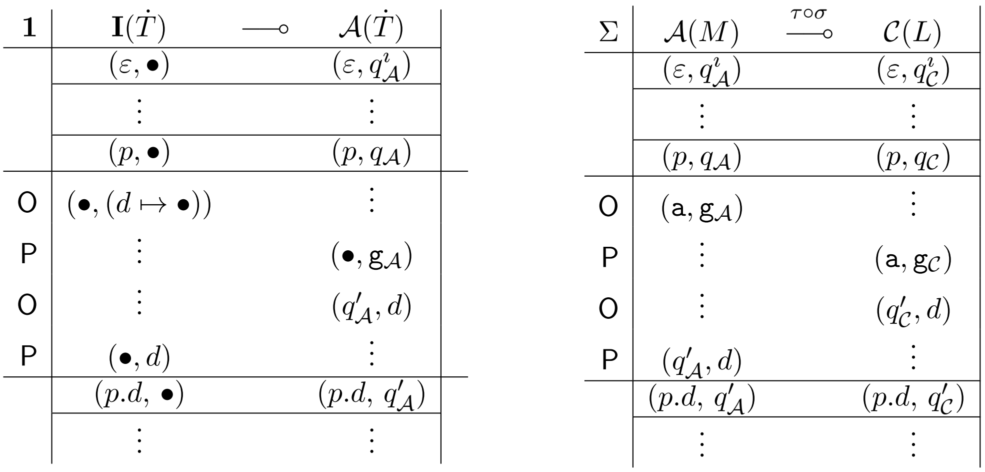

$\mathbf{1}$
$\mathbf{I}(\dot{T})$ $\multimap$ $\mathcal{A}(\dot{T})$
$(\varepsilon, \bullet)$ $(\varepsilon, q^{\imath}_{\mathcal{A}})$
$(p, \bullet)$ $(p, q_{\mathcal{A}})$
O $(\bullet, (d \mapsto \bullet))$
P $(\bullet, \mathsf{g}_{\mathcal{A}})$
O $(q'_{\mathcal{A}}, d)$
P $(\bullet, d)$
$(p.d, \bullet)$ $(p.d,\ q'_{\mathcal{A}})$
$\Sigma$
$\mathcal{A}(M)$ $\xrightarrow{\tau \circ \sigma}$ $\mathcal{C}(L)$
$(\varepsilon, q^{\imath}_{\mathcal{A}})$ $(\varepsilon, q^{\imath}_{\mathcal{C}})$
$(p, q_{\mathcal{A}})$ $(p, q_{\mathcal{C}})$
O $(\mathsf{a}, \mathsf{g}_{\mathcal{A}})$
P $(\mathsf{a}, \mathsf{g}_{\mathcal{C}})$
O $(q'_{\mathcal{C}}, d)$
P $(q'_{\mathcal{A}}, d)$
$(p.d,\ q'_{\mathcal{A}})$ $(p.d,\ q'_{\mathcal{C}})$

Figure 4: Lifting of acceptance games (left) and a composite strategy (right)

**2.8 The (Synchronous) Direct Product of (Non-Deterministic) Automata.** Returning to the general case of sequents of the form

$$M \ ; \ \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$$

the *provability interpretation* tells us that the left commas correspond to a form of conjunction. A conjunction on non-deterministic automata can be implemented with a direct (synchronous) product. The *direct product* $\mathcal{A}_1 \otimes \mathcal{A}_2$ of the non-deterministic automata $\mathcal{A}_i = (Q_{\mathcal{A}_i}, q^{\imath}_{\mathcal{A}_i}, \delta_{\mathcal{A}_i}, \Omega_{\mathcal{A}_i})$, both over $\Sigma$, is the non-deterministic automaton over $\Sigma$

$$\mathcal{A}_1 \otimes \mathcal{A}_2 \quad := \quad \left(Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2}\,,\, (q^{\imath}_{\mathcal{A}_1}, q^{\imath}_{\mathcal{A}_2})\,,\, \delta_{\mathcal{A}_1 \otimes \mathcal{A}_2}\,,\, \Omega_{\mathcal{A}_1 \otimes \mathcal{A}_2}\right)$$

with

$$\delta_{\mathcal{A}_1 \otimes \mathcal{A}_2}((q_1, q_2), \mathsf{a}) \quad := \quad \{\langle \mathsf{g}_1, \mathsf{g}_2 \rangle : \mathfrak{D} \to Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2} \mid \mathsf{g}_i \in \delta_{\mathcal{A}_i}(q_i, \mathsf{a}) \text{ for } i = 1, 2\}$$

and where $\Omega_{\mathcal{A}_1 \otimes \mathcal{A}_2}$ is $\Omega_{\mathcal{A}_1} \times \Omega_{\mathcal{A}_2}$ modulo $(Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2})^{\omega} \simeq Q^{\omega}_{\mathcal{A}_1} \times Q^{\omega}_{\mathcal{A}_2}$. For every tree $T$, the (accepting) runs of $\mathcal{A}_1 \otimes \mathcal{A}_2$ on $T$ are exactly[10] the pairs $\langle R_1, R_2 \rangle : \mathfrak{D}^* \to Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2}$ of (accepting) runs of $\mathcal{A}_1$ and $\mathcal{A}_2$ over $T$. We therefore have, in the category **Set**

$$(\mathcal{A}_1 \otimes \mathcal{A}_2)(T) \quad \simeq \quad \mathcal{A}_1(T) \times \mathcal{A}_2(T) \tag{19}$$

from which we immediately get

$$\mathcal{L}(\mathcal{A}_1 \otimes \mathcal{A}_2) \quad = \quad \mathcal{L}(\mathcal{A}_1) \cap \mathcal{L}(\mathcal{A}_2)$$

In games, the product $(-) \otimes (-)$ is actually Cartesian on *total* non-deterministic automata[11]. We say that a non-deterministic automaton $\mathcal{A}$ is *total* if the empty set is not in the range of its transition function $\delta_{\mathcal{A}}$, that is when

$$\delta_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathfrak{D} \to Q_{\mathcal{A}}) \setminus \{\emptyset\}$$

[10] Because universal quantifications commute over conjunctions!

[11] See e.g. [Shu08] for notions of monoidal indexed categories and fibrations.

| $\Sigma$ | $(\mathcal{A}_1[\pi_1] \otimes \mathcal{A}_2[\pi_2])\langle M_1, M_2\rangle$ | $\xrightarrow{\varpi_i}\!\!\circ$ | $\mathcal{A}_i(M_i)$ |
|---|---|---|---|
| | $(\varepsilon, (q^{\imath}_{\mathcal{A}_1}, q^{\imath}_{\mathcal{A}_2}))$ | | $(\varepsilon, q^{\imath}_{\mathcal{A}_i})$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, (q_1, q_2))$ | | $(p, q_i)$ |
| O | $(\mathsf{a}, (\mathsf{g}_1, \mathsf{g}_2))$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_i)$ |
| O | $\vdots$ | | $(\mathsf{g}_i(d), d)$ |
| P | $((\mathsf{g}_1(d), \mathsf{g}_2(d)), d)$ | | $\vdots$ |
| | $(p \cdot d, (q'_1, q'_2))$ | | $(p \cdot d, q'_i)$ |
| | $\vdots$ | | $\vdots$ |

where $q'_1 = \mathsf{g}_1(d)$ and $q'_2 = \mathsf{g}_2(d)$

Figure 5: The projection $\varpi_i$ for $\mathcal{A}_1$ and $\mathcal{A}_2$ *non-deterministic*

Note that any non-deterministic automaton can be translated to a total one at the cost of possibly adding one (rejecting) state. Consider now $\Sigma \vdash \mathcal{A}_1(M_1)$, $\Sigma \vdash \mathcal{A}_2(M_2)$ and $\Sigma \vdash \mathcal{C}(L)$, with $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{C}$ total non-deterministic. Then there are canonical total (and winning) *projection strategies*:

$$\varpi_i \quad : \quad (\mathcal{A}_1[\pi_1] \otimes \mathcal{A}_2[\pi_2])\langle M_1, M_2\rangle \quad \multimap \quad \mathcal{A}_i(M_i)$$

and total (winning) P-strategies

$$\theta \quad : \quad \mathcal{C}(L) \quad \multimap \quad (\mathcal{A}_1[\pi_1] \otimes \mathcal{A}_2[\pi_2])\langle M_1, M_2\rangle \tag{20}$$

are in bijection with pairs of total (winning) P-strategies

$$\sigma_1 \quad : \quad \mathcal{C}(L) \quad \multimap \quad \mathcal{A}_1(M_1) \qquad \text{and} \qquad \sigma_2 \quad : \quad \mathcal{C}(L) \quad \multimap \quad \mathcal{A}_2(M_2) \tag{21}$$

A typical play of $\varpi_i$ is depicted on Fig. 5. Note that since $\mathcal{A}_1$ and $\mathcal{A}_2$ are assumed to be non-deterministic, synchronicity completely determines the move $((\mathsf{g}_1(d), \mathsf{g}_2(d)), d)$ of $\varpi_i$ from the O-move $(\mathsf{g}_i(d), d)$ in component $\mathcal{A}_i(M_i)$. It follows that each strategy $\theta$ as in (20) determines strategies $\sigma_1$ and $\sigma_2$ as in (21) with $\sigma_i = \varpi_i \circ \theta$.

On the other hand, given fixed $\sigma_1$ and $\sigma_2$ as in (21), a strategy $\theta$ such that $\sigma_1 = \varpi_1 \circ \theta$ and $\sigma_2 = \varpi_2 \circ \theta$ is necessarily unique. As for existence of $\theta$, consider plays $s_1 \in \sigma_1$ and $s_2 \in \sigma_2$ with the same trace in $\mathrm{Tr}_\Sigma$, as in Fig. 6. Note that if O plays "the same way" on $\mathcal{C}(L)$ in $s_1$ and $s_2$, then $s_{1\restriction\mathcal{C}(L)} = s_{2\restriction\mathcal{C}(L)}$. Indeed, if $q_1 = q_2$ and $\mathsf{g}_1 = \mathsf{g}_2$, then, since $s_1$ and $s_2$ have the same trace, we have $q'_1 = \mathsf{g}_1(d) = \mathsf{g}_2(d) = q'_2$ for the same $d \in \mathfrak{D}$. It follows that $s_1$ and $s_2$ can be merged in the play depicted on Fig. 6. This defines a total (winning) $\theta$ such that $\varpi_i \circ \theta = \sigma_i$. Furthermore, the automaton $\mathbf{I}$ induces terminal objects: for every $\mathcal{C}(L)$ with $\mathcal{C}$ total non-deterministic, there is a unique total (and winning) $\mathbf{1}_{\mathcal{C}(L)} : \mathcal{C}(L) \multimap \mathbf{I}(L)$, see Fig. 7.

The Cartesian structure of $(\otimes, \mathbf{I})$ on total non-deterministic automata imply that we can equip total non-deterministic automata with deduction rules for a Cartesian product, such as the following rules (where $\overline{\mathcal{A}}$ denotes a sequence $\mathcal{A}_1, \dots, \mathcal{A}_n$):

$$\begin{gathered}(\textsc{Left}\ \otimes)\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A} \otimes \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{A} \otimes \mathcal{B}}\ (\textsc{Right}\ \otimes) \\ (\textsc{Left}\ \mathbf{I})\ \frac{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathbf{I},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{}{M\ ;\ \vdash \mathbf{I}}\ (\textsc{Right}\ \mathbf{I})\end{gathered} \tag{22}$$

| $\Sigma$ | $\mathcal{C}(L)$ | $\overset{s_1\in\sigma_1}{\multimap}$ | $\mathcal{A}_1(M_1)$ |
|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{C}})$ | | $(\varepsilon, q^\imath_{\mathcal{A}_1})$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, q_1)$ | | $(p, q_{\mathcal{A}_1})$ |
| O | $(\mathsf{a}, \mathsf{g}_1)$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{A}_1})$ |
| O | $\vdots$ | | $(q'_{\mathcal{A}_1}, d)$ |
| P | $(q'_1, d)$ | | $\vdots$ |
| | $(p.d,\ q'_1)$ | | $(p.d,\ q'_{\mathcal{A}_1})$ |
| | $\vdots$ | | $\vdots$ |

| $\Sigma$ | $\mathcal{C}(L)$ | $\overset{s_2\in\sigma_2}{\multimap}$ | $\mathcal{A}_2(M_2)$ |
|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{C}})$ | | $(\varepsilon, q^\imath_{\mathcal{A}_2})$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, q_2)$ | | $(p, q_{\mathcal{A}_2})$ |
| O | $(\mathsf{a}, \mathsf{g}_2)$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, \mathsf{g}_{\mathcal{A}_2})$ |
| O | $\vdots$ | | $(q'_{\mathcal{A}_2}, d)$ |
| P | $(q'_2, d)$ | | $\vdots$ |
| | $(p.d,\ q'_2)$ | | $(p.d,\ q'_{\mathcal{A}_2})$ |
| | $\vdots$ | | $\vdots$ |

| $\Sigma$ | $\mathcal{C}(L)$ | $\multimap$ | $(\mathcal{A}_1[\pi_1] \otimes \mathcal{A}_2[\pi_2])\langle M_1, M_2\rangle$ |
|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{C}})$ | | $(\varepsilon, (q^\imath_{\mathcal{A}_1}, q^\imath_{\mathcal{A}_2}))$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, q_1)$ | | $(p, (q_{\mathcal{A}_1}, q_{\mathcal{A}_2}))$ |
| O | $(\mathsf{a}, \mathsf{g}_1)$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, (\mathsf{g}_{\mathcal{A}_1}, \mathsf{g}_{\mathcal{A}_2}))$ |
| O | $\vdots$ | | $((q'_{\mathcal{A}_1}, q'_{\mathcal{A}_2}), d)$ |
| P | $(q'_1, d)$ | | $\vdots$ |
| | $(p.d,\ q'_1)$ | | $(p.d,\ (q'_{\mathcal{A}_1}, q'_{\mathcal{A}_2}))$ |
| | $\vdots$ | | $\vdots$ |

Figure 6: Pairing on non-deterministic automata

| $\Sigma$ | $\mathcal{C}(L)$ | $\overset{\mathbf{1}_{\mathcal{C}(L)}}{\multimap}$ | $\mathbf{I}(L)$ |
|---|---|---|---|
| | $(\varepsilon, q^\imath_{\mathcal{C}})$ | | $(\varepsilon, \bullet)$ |
| | $\vdots$ | | $\vdots$ |
| | $(p, q_{\mathcal{C}})$ | | $(p, \bullet)$ |
| O | $(\mathsf{a}, \mathsf{g}_{\mathcal{C}})$ | | $\vdots$ |
| P | $\vdots$ | | $(\mathsf{a}, (\_ \mapsto \bullet))$ |
| O | $\vdots$ | | $(\bullet, d)$ |
| P | $(\mathsf{g}_{\mathcal{C}}(d), d)$ | | $\vdots$ |
| | $(p.d,\ \mathsf{g}_{\mathcal{C}}(d))$ | | $(p.d,\ \bullet)$ |
| | $\vdots$ | | $\vdots$ |

Figure 7: The terminal object on non-deterministic automata

together with the structural *exchange rule*

$$(\textsc{Exchange})\quad \frac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{B},\, \mathcal{A},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \tag{23}$$

as well as the structural *weakening* and *contraction* rules:

$$(\textsc{Weak})\quad \frac{M\ ;\ \overline{\mathcal{A}}\,,\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\,,\, \mathcal{A}\,,\, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad\qquad \frac{M\ ;\ \overline{\mathcal{A}}\,,\, \mathcal{A}\,,\, \mathcal{A}\,,\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\,,\, \mathcal{A}\,,\, \overline{\mathcal{B}} \vdash \mathcal{C}}\quad (\textsc{Contr}) \tag{24}$$

and the following general (multiplicative) cut rule

$$(\textsc{Cut})\quad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\, \mathcal{A},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\, \overline{\mathcal{A}},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \tag{25}$$

To summarize, with total non-deterministic automata, the left commas in sequents of the form (1) can be internalized as a product $(\otimes, \mathbf{I})$, whose deduction rules are suggested by its structure in the computational interpretation.

**2.9 Alternating Automata and Linear Logic.** With respect to the context of this paper, the basic insight of linear logic [Gir87], is that having an explicit control on the weakening and contraction structural rules depicted in (24) gives rise to a decomposition of the usual intuitionistic connectives $\wedge, \to$ into more refined connectives (usually denoted $\otimes, \&, !, \multimap$), which in a lot of cases allow, thanks to the Curry-Howard correspondence, refined constructions of models of programming languages based on (typed) $\lambda$-calculi (see e.g. [AC98]).

In the case of conjunction, this can be phrased as follows. First, when suppressing the structural rules (Weak) and (Contr), the rules displayed in (22) and (23) do not specify anymore a Cartesian structure for the product $(\otimes, \mathbf{I})$, but merely a symmetric monoidal structure (see e.g. [Mel09] for definitions). This implies that in contrast with intuitionistic sequents, the left commas in ILL-sequents, which have the same structure as $(\otimes, \mathbf{I})$, do not anymore behave as a Cartesian product. Moreover, $(\otimes, \mathbf{I})$ is not anymore equivalent to the *additive* conjunction (usually denoted &, with unit $\top$), which, as a logical connective, would be defined in ILL-sequents by rules of the form[12]:

$$\frac{A_1\,,\,\dots\,,\,A_n \vdash B_1 \qquad A_1\,,\,\dots\,,\,A_n \vdash B_2}{A_1\,,\,\dots\,,\,A_n \vdash B_1 \,\&\, B_2} \qquad\qquad \frac{}{A_1\,,\,\dots\,,\,A_n \vdash \top}$$

$$\frac{A_1\,,\,\dots\,,\,A_i\,,\,\dots\,,\,A_n \vdash B}{A_1\,,\,\dots\,,\,A_i \,\&\, C\,,\,\dots\,,\,A_n \vdash B} \qquad \frac{A_1\,,\,\dots\,,\,A_i\,,\,\dots\,,\,A_n \vdash B}{A_1\,,\,\dots\,,\,C \,\&\, A_i\,,\,\dots\,,\,A_n \vdash B}$$

Second, the structural rules (Weak) and (Contr) are restored but for a specific *exponential modality* $!(-)$:

$$\frac{A_1\,,\,\dots\,,\,\dots\,,\,A_n \vdash B}{A_1\,,\,\dots\,,\,!A_i\,,\,\dots\,,\,A_n \vdash B} \qquad\qquad \frac{A_1\,,\,\dots\,,\,!A_i\,,\,!A_i\,,\,\dots A_n \vdash B}{A_1\,,\,\dots\,,\,!A_i\,,\,\dots\,,\,A_n \vdash B} \tag{26}$$

The modality $!(-)$ is itself subject to specific introduction rules, called *dereliction* and *promotion*:

$$\frac{A_1\,,\,\dots\,,\,A_i\,,\,\dots\,,\,A_n \vdash B}{A_1\,,\,\dots\,,\,!A_i\,,\,\dots\,,\,A_n \vdash B} \qquad\qquad \frac{!A_1\,,\,\dots\,,\,!A_n \vdash B}{!A_1\,,\,\dots\,,\,!A_n \vdash !B} \tag{27}$$

[12] We do not consider in this paper the usual additive conjunction on alternating automata, which would provide an implementation of &, because its categorical properties would require a slight extension of our setting.

Then (but see also [Gir87, AC98, Mel09] for details), the categorical interpretation of proofs gives an isomorphism

$$!A \otimes !B \quad \simeq \quad !(A \mathbin{\&} B) \tag{28}$$

which implies that an intuitionistic sequent

$$A_1 \,,\, \ldots \,,\, A_n \vdash B$$

where the left commas behave as a Cartesian product, corresponds to the linear sequent

$$!A_1 \,,\, \ldots \,,\, !A_n \vdash B$$

where the left commas behave as a symmetric monoidal product $(-) \otimes (-)$ which need not be Cartesian.

The pertinence of intuitionistic linear logic in our context comes from the fact that the product $(-) \otimes (-)$ defined in §2.8 on non-deterministic automata extends to (total[13]) alternating automata, but induces a symmetric monoidal product which is not Cartesian (there is no bijection between (20) and (21), and moreover $\mathbf{I}$ is not terminal). Recall from §1.3 that an alternating automaton $\mathcal{A}$ over $\Gamma$ has a transition function of the form

$$\delta_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Gamma \quad \longrightarrow \quad \mathcal{P}(\mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}))$$

Given $M \in \mathbf{T}[\Sigma, \Gamma]$, the substituted acceptance game $\Sigma \vdash \mathcal{G}(\mathcal{A}, M)$ is the positive game with winning

$$\Sigma \vdash \mathcal{G}(\mathcal{A}, M) \quad := \quad (\Sigma \times \mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}),\ Q_{\mathcal{A}} \times \mathfrak{D},\ L_{\mathcal{A}(M)},\ \mathcal{W}_{\mathcal{A}(M)})$$

whose legal plays $s \in L_{\mathcal{A}(M)}$ are sequences of the form

$$\begin{array}{rrcl} & s & = & (\mathsf{a}_0, \gamma_0) \cdot (q_1, d_1) \cdot (\mathsf{a}_1, \gamma_1) \cdot \ldots \cdot (q_n, d_n) \\ \text{or} & s & = & (\mathsf{a}_0, \gamma_0) \cdot (q_1, d_1) \cdot (\mathsf{a}_1, \gamma_1) \cdot \ldots \cdot (q_n, d_n) \cdot (\mathsf{a}_n, \gamma_n) \end{array}$$

where $n \geq 0$, $(q_{k+1}, d_{k+1}) \in \gamma_k$ and $\gamma_k \in \delta_{\mathcal{A}}(q_k, M(\mathsf{a}_0 \cdot \ldots \cdot \mathsf{a}_k, d_1 \cdot \ldots \cdot d_k))$ with $q_0 := q^\imath_{\mathcal{A}}$. The winning plays $\chi \in \mathcal{W}_{\mathcal{A}(M)}$ are generated from the acceptance condition $\Omega_{\mathcal{A}}$ in the expected way, and we say that $\mathcal{A}$ accepts $T$ (*i.e.* $T \in \mathcal{L}(\mathcal{A})$) when $\mathsf{P}$ has a winning strategy in $\mathbf{1} \vdash \mathcal{A}(\dot{T})$.

Alternating automata induce synchronous arrow games in the same way as non-deterministic automata (see [Rib15]). Trace functions extend to alternating automata (replacing $\mathsf{g}$'s by $\gamma$'s), and we can let $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$ be $\mathcal{G}(\mathcal{A}, M) \multimap_{\mathbf{SG}} \mathcal{G}(\mathcal{B}, N)$ restricted to (the prefixes of) its legal synchronous plays. A typical play is depicted on Fig. 8.

Given alternating automata $\mathcal{A}$ and $\mathcal{B}$ over $\Sigma$, the automaton $\mathcal{A} \otimes \mathcal{B}$ over $\Sigma$ has state set $Q_{\mathcal{A}} \times Q_{\mathcal{B}}$, and evaluates $\mathcal{A}$ and $\mathcal{B}$ along the same trace in $\mathrm{Tr}_\Sigma$ (see [Rib15] for details). Now, recall that with alternating automata, $\mathsf{O}$ can choose states in addition to tree directions. Hence, given a $\mathsf{P}$-strategy on $(\mathcal{A} \otimes \mathcal{B})(\dot{T})$ (for $T : \mathfrak{D}^* \to \Sigma$), and given a branch of this strategy following a given path $p \in \mathfrak{D}^*$, it is possible for $\mathsf{P}$ to make different choices according to previous $\mathsf{O}$-moves. In particular, some choice of $\mathsf{P}$ in component $\mathcal{A}$ may depend on previous $\mathsf{O}$-moves in $\mathcal{B}$. (Note that this was not possible with non-deterministic automata, since $p \in \mathfrak{D}^*$ determines uniquely the previous $\mathsf{O}$-moves.) So a $\mathsf{P}$-strategy on $(\mathcal{A} \otimes \mathcal{B})(\dot{T})$ may not uniquely determine a pair of strategies in $\mathcal{A}(\dot{T}) \times \mathcal{B}(\dot{T})$. It follows that in general there is no bijection between (20) and (21).

On the other hand, in any model of intuitionistic linear logic, the isomorphism (28) implies that every object of the form $!A$ is a commutative commonoid w.r.t. $(\otimes, \mathbf{I})$ (see e.g. [Mel09]),

[13] *Total* alternating automata were called *complete* in [Rib15].

$$\begin{array}{c|ccc|l}
\Sigma & \mathcal{A}(M) & \multimap & \mathcal{B}(N) & \\
\hline
 & (\varepsilon, q^{\imath}_{\mathcal{A}}) & & (\varepsilon, q^{\imath}_{\mathcal{B}}) & \\
\hline
 & \vdots & & \vdots & \\
\hline
 & (p, q_{\mathcal{A}}) & & (p, q_{\mathcal{B}}) & \\
\hline
\mathsf{O} & (\mathtt{a}, \gamma_{\mathcal{A}}) & & \vdots & \text{if } \gamma_{\mathcal{A}} \in \delta_{\mathcal{A}}(q_{\mathcal{A}}, M(p)(\mathtt{a})) \\
\mathsf{P} & \vdots & & (\mathtt{a}, \gamma_{\mathcal{B}}) & \text{if } \gamma_{\mathcal{B}} \in \delta_{\mathcal{B}}(q_{\mathcal{B}}, N(p)(\mathtt{a})) \\
\mathsf{O} & \vdots & & (q'_{\mathcal{B}}, d) & \text{if } (q'_{\mathcal{B}}, d) \in \gamma_{\mathcal{B}} \\
\mathsf{P} & (q'_{\mathcal{A}}, d) & & \vdots & \text{if } (q'_{\mathcal{A}}, d) \in \gamma_{\mathcal{A}} \\
\hline
 & (p.d,\, q'_{\mathcal{A}}) & & (p.d,\, q'_{\mathcal{B}}) & \\
\hline
 & \vdots & & \vdots &
\end{array}$$

Figure 8: A typical synchronous play with alternating automata

which essentially means that $(\otimes, \mathbf{I})$ has a Cartesian structure w.r.t. objects of the form $!A$. This indicates that non-deterministic automata behave as objects of the form $!A$, and it turns out that to some extent, the powerset construction translating an alternating automaton to an equivalent non-deterministic one (the *Simulation Theorem* [MS87, EJ91, MS95]), corresponds to an $!(-)$-modality of intuitionistic linear logic. In particular, all the $!(-)$-rules (26) and (27) can be interpreted in our categories[14]. But unfortunately, this interpretation is not compatible with usual cut-elimination, because the operation $!(-)$ fails to be a functor.

**2.10 Realizability and Compositionality in [Rib15].** As with have seen in this Section, a substantial part of the program announced in §1 was already completed in [Rib15].

The framework of [Rib15] consists in symmetric monoidal fibred categories of (usual) tree automata. Although [Rib15] does not explicitly mention any deduction system, it actually provides, for total alternating automata, a realizability interpretation (in the sense of §1.2 and §1.5) of the deduction system made of the rules displayed in (22), (23) and (25), as well as the (Axiom) rule of §2.1, the (Weak) rule displayed in (24), and the (Contr) rule displayed in (24) (with $\mathcal{A}$ total *non-deterministic*). The fibred structure of [Rib15] amounts to the fact that $\mathbf{T}$-maps $L \in \mathbf{T}[\Delta, \Sigma]$ induce substitution functors $L^\star$ taking a strategy

$$\Sigma \quad \vdash \quad \sigma \quad : \quad \mathcal{A}(M) \quad \multimap \quad \mathcal{B}(N)$$

to a strategy

$$\Delta \quad \vdash \quad L^\star(\sigma) \quad : \quad \mathcal{A}(M \circ L) \quad \multimap \quad \mathcal{B}(N \circ L) \tag{29}$$

(see Rem. 4.1, §4). This gives an interpretation of the substitution rule (14). Furthermore, this fibred structure is equipped with existential quantifications in the spirit of §2.2 and in the technical sense of *simple coproducts* (see e.g. [Jac01, §1.9]). Finally, a *simulation operation* $!(-)$ was already shown to satisfy the weak completeness property mentioned in 1.5.(bii) (with instead of $(-)^{\perp}$, a usual notion of linear complement $(-)^{\perp\!\!\!\perp}$, see §1.3).

Besides providing a detailed account of a fibred structure on tree automata, this paper extends [Rib15] with an explicit deduction system (whose rules are depicted on Fig. 21, Fig. 22

[14] (Weak) actually holds (in a non-canonical way) for total alternating automata (*i.e.* the ! is not strictly necessary in the conclusion).

and Fig. 25), and gives an explicit account of simulation as satisfying the (obvious adaptation to our context of the) *deduction rules* (26) of the usual exponential modality $!(-)$ of ILL (see also §1.4 and §7). Yet, the main contribution of this paper is to provide a monoidal-*closed* structure on automata. This involves variants of usual of alternating automata, that we call *uniform automata* (§3).

Usual alternating automata seem to not have enough structure to induce categories equipped with a monoidal-closed structure while providing a computational interpretation of proofs in the sense of §1.2. As usual (see [Mel13] and also §3.2 and §5.4), a monoidal closed structure $(\otimes, \mathbf{I}, \multimap)$ provides for each object $\mathtt{R}$ a contravariant functor $(-)^{\mathtt{R}} := (-) \multimap \mathtt{R}$, which automatically validates deduction rules of the form

$$\frac{A,\, B \vdash \mathtt{R}}{A \vdash B^{\mathtt{R}}} \qquad \frac{A \vdash B^{\mathtt{R}}}{A,\, B \vdash \mathtt{R}} \qquad \frac{A \vdash B^{\mathtt{R}}}{B \vdash A^{\mathtt{R}}} \qquad \frac{}{A \vdash A^{\mathtt{RR}}} \qquad \frac{A \vdash B}{B^{\mathtt{R}} \vdash A^{\mathtt{R}}} \qquad \frac{}{A^{\mathtt{RRR}} \vdash A^{\mathtt{R}}}$$

In the case of usual alternating automata, it is not difficult to have a linear complementation $(-)^{\perp\!\!\!\perp}$ which satisfies similar rules[15]. Given an alternating automaton $\mathcal{A}$, following [Wal02] it is natural to let $\mathcal{A}^{\perp\!\!\!\perp}$ have the same states as $\mathcal{A}$, and to take for $\delta_{\mathcal{A}^{\perp\!\!\!\perp}}(q, \mathtt{a})$ a dual of the disjunctive formula represented by $\delta_{\mathcal{A}}(q, \mathtt{a}) \in \mathcal{P}(\mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}))$, that is, to let $\delta_{\mathcal{A}^{\perp\!\!\!\perp}}(q, \mathtt{a})$ be the set of all $\gamma^{\perp\!\!\!\perp} \subseteq \bigcup \delta_{\mathcal{A}}(q, \mathtt{a})$ such that $\gamma^{\perp\!\!\!\perp} \cap \gamma \neq \emptyset$ for all $\gamma \in \delta_{\mathcal{A}}(q, \mathtt{a})$. Then, it is not difficult to validate a rule of the form

$$\frac{\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B}}{\Sigma \ ; \ \mathcal{B}^{\perp\!\!\!\perp} \vdash \mathcal{A}^{\perp\!\!\!\perp}} \tag{30}$$

However, as detailed in [Rib17, App. A], it is not obvious how to obtain an operation $(-)^{\perp\!\!\!\perp}$ with a functorial action of strategies, in particular which preserves composition.

## 3 Uniform Automata and Zig-Zag Games

This Section presents the notion of automata (called *uniform* automata) on which this paper relies (§3.2 and §3.3). Uniform automata are motivated by the extension of usual alternating automata with a monoidal *closed* structure (§3.1, see also §1.5 and §2.10). Working with uniform automata instead of usual automata allows, w.r.t. [Rib15], a considerable simplification of the underlying technology of game semantics. We rely on a very simple category **DZ** of (total) *zig-zag* games (§3.4 and §3.6), on top of which the counterpart on uniform automata of synchronous arrows games is built (§3.5 and §3.7).

**3.1 Toward Monoidal Closure.** The main contribution of this paper w.r.t. [Rib15] is that we propose a variant of usual alternating automata which is equipped with a closed structure w.r.t. the symmetric monoidal product $(\otimes, \mathbf{I})$. This means that from automata $\mathcal{B}$ and $\mathcal{C}$, we can define an automaton $(\mathcal{B} \multimap \mathcal{C})$ such that for every automaton $\mathcal{A}$, we have an isomorphism

$$\mathcal{A} \otimes \mathcal{B} \multimap \mathcal{C} \qquad \simeq \qquad \mathcal{A} \multimap (\mathcal{B} \multimap \mathcal{C})$$

(subject to some naturality condition). This implies that we can extend our deduction system with rules for the linear implication connective $\multimap$ of ILL:

$$(\textsc{Left} \multimap)\ \frac{M \ ; \ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M \ ; \ \overline{\mathcal{B}},\, \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M \ ; \ \overline{\mathcal{B}},\, \overline{\mathcal{A}},\, \mathcal{A} \multimap \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{M \ ; \ \overline{\mathcal{A}},\, \mathcal{B} \vdash \mathcal{C}}{M \ ; \ \overline{\mathcal{A}} \vdash \mathcal{B} \multimap \mathcal{C}}\ (\textsc{Right} \multimap) \tag{31}$$

[15] Modulo some burden caused by totality, see [Rib15].

and that these rules are compatible with cut-elimination (see e.g. [Mel09], but also Rem. 5.7), in the sense that the following two derivations are interpreted by the same strategy:

$$\dfrac{\dfrac{\dfrac{\mathscr{D}_1}{\mathcal{A}\vdash\mathcal{B}}}{\mathbf{I}\vdash\mathcal{A}\multimap\mathcal{B}} \qquad \dfrac{\dfrac{\mathscr{D}_2}{\mathbf{I}\vdash\mathcal{A}} \qquad \dfrac{}{\mathcal{B}\vdash\mathcal{B}}}{\mathcal{A}\multimap\mathcal{B}\vdash\mathcal{B}}}{\mathbf{I}\vdash\mathcal{B}} \qquad\qquad \dfrac{\begin{array}{c}\vdots\\ \mathscr{D}_1[\mathscr{D}_2/\mathcal{A}]\end{array}}{\mathbf{I}\vdash\mathcal{B}} \tag{32}$$

Assume given alternating automata $\mathcal{A}$ and $\mathcal{B}$ over $\Sigma$, and let us see how to construct an automaton $(\mathcal{A}\multimap\mathcal{B})$ over $\Sigma$ such that total (winning) $\mathsf{P}$-strategies in the game $\Sigma\vdash\mathbf{I}\longrightarrow\!\!\circ\;(\mathcal{A}\multimap\mathcal{B})$ are in bijection with total (winning) $\mathsf{P}$-strategies in the game $\Sigma\vdash\mathcal{A}\longrightarrow\!\!\circ\;\mathcal{B}$. Given a play of $\sigma$ leading to a position $((p,q_{\mathcal{A}}),(p,q_{\mathcal{B}}))$ in the game $\Sigma\vdash\mathcal{A}\longrightarrow\!\!\circ\;\mathcal{B}$, the next moves of $\sigma$ induce maps

$$\begin{array}{rcccl} & f & : & \gamma_{\mathcal{A}} & \longmapsto & \gamma_{\mathcal{B}} \\ F & : & (\gamma_{\mathcal{A}}\ ,\ (q'_{\mathcal{B}},d)\in f(\gamma_{\mathcal{A}})) & \longmapsto & (q'_{\mathcal{A}},d)\in\gamma_{\mathcal{A}} \end{array}$$

where

$$\begin{array}{c|ccc|l}
\Sigma & \mathcal{A} & \overset{\sigma}{\longrightarrow\!\!\circ} & \mathcal{B} & \\
\cline{2-4}
 & (p,q_{\mathcal{A}}) & & (p,q_{\mathcal{B}}) & \\
\hline
\mathsf{O} & (\mathsf{a},\gamma_{\mathcal{A}}) & & \vdots & \\
\mathsf{P} & \vdots & & (\mathsf{a},\gamma_{\mathcal{B}}) & f(\gamma_{\mathcal{A}})=\gamma_{\mathcal{B}} \\
\mathsf{O} & \vdots & & (q'_{\mathcal{B}},d) & \\
\mathsf{P} & (q'_{\mathcal{A}},d) & & \vdots & F(\gamma_{\mathcal{A}},\,(q'_{\mathcal{B}},d))=(q'_{\mathcal{A}},d)
\end{array}$$

The main idea toward the definition of the automaton $(\mathcal{A}\multimap\mathcal{B})$ is to let the $\mathsf{P}$-moves in its transitions be given by pairs of maps

$$f\ :\ \mathcal{P}(Q_{\mathcal{A}}\times\mathfrak{D})\ \longrightarrow\ \mathcal{P}(Q_{\mathcal{B}}\times\mathfrak{D}) \qquad F\ :\ \mathcal{P}(Q_{\mathcal{A}}\times\mathfrak{D})\times(Q_{\mathcal{B}}\times\mathfrak{D})\ \longrightarrow\ Q_{\mathcal{A}}\times\mathfrak{D} \tag{33}$$

Note that the functions $f$ and $F$ do not explicitly depend from the input characters $\mathsf{a}\in\Sigma$, since input characters are played in the component $\mathbf{I}$ of $\Sigma\vdash\mathbf{I}\longrightarrow\!\!\circ\;(\mathcal{A}\multimap\mathcal{B})$, prior to the choice of $f$ and $F$.

**3.2 Uniform Automata.** In order to obtain the required categorical properties of a monoidal closed structure, we devise a "uniform" variant of usual alternating automata, whose transitions are given by explicit arbitrary non-empty finite sets of $\mathsf{P}$ and $\mathsf{O}$-moves.

**Definition 3.1** (Uniform Tree Automata)**.** *A* uniform tree automaton $\mathcal{A}$ *over* $\Sigma$ *(notation* $\mathcal{A}:\Sigma$*) has the form*

$$\mathcal{A}\quad=\quad(Q_{\mathcal{A}}\,,\,q^{\imath}_{\mathcal{A}}\,,\,U\,,\,X\,,\,\delta_{\mathcal{A}}\,,\,\Omega_{\mathcal{A}}) \tag{34}$$

*where* $Q_{\mathcal{A}}$ *is the finite set of* states, $q^{\imath}_{\mathcal{A}}\in Q_{\mathcal{A}}$ *is the* initial state, $U$ *and* $X$ *are finite non-empty sets of resp.* $\mathsf{P}$ *and* $\mathsf{O}$*-moves, the* acceptance condition $\Omega_{\mathcal{A}}$ *is an* $\omega$*-regular subset of* $Q^{\omega}_{\mathcal{A}}$*, and the* transition function $\delta_{\mathcal{A}}$ *has the form*

$$\delta_{\mathcal{A}}\quad:\quad Q_{\mathcal{A}}\times\Sigma\quad\longrightarrow\quad U\times X\quad\longrightarrow\quad(\mathfrak{D}\longrightarrow Q_{\mathcal{A}}) \tag{35}$$

*Following the usual terminology, an automaton* $\mathcal{A}$ *as in (34) is* non-deterministic *if* $X\simeq\mathbf{1}$*,* universal *if* $U\simeq\mathbf{1}$*, and* deterministic *if* $U\simeq X\simeq\mathbf{1}$*.*

**Example 3.2.** *(i) The* unit automaton $\mathbf{I}_\Sigma : \Sigma$ *is the unique uniform deterministic automaton over* $\Sigma$ *with state set* $\mathbf{1}$ *(with* $\bullet$ *initial) and acceptance condition* $\mathbf{1}^\omega$*. Explicitly,*

$$\mathbf{I}_\Sigma \quad := \quad (\mathbf{1}, \bullet, \mathbf{1}, \mathbf{1}, \delta_{\mathbf{1}}, \mathbf{1}^\omega)$$

*where* $\delta_{\mathbf{1}}$ *is the unique function*

$$\delta_{\mathbf{1}} \quad : \quad \mathbf{1} \times \Sigma \quad \longrightarrow \quad \mathbf{1} \times \mathbf{1} \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow \mathbf{1})$$

*We write* $\mathbf{I}$ *for* $\mathbf{I}_\Sigma$ *when* $\Sigma$ *is clear from the context.*

*(ii) Each alternating automaton* $\mathcal{A}$ *can be translated to a uniform automaton* $\widehat{\mathcal{A}}$*. The automaton* $\widehat{\mathcal{A}}$ *simulates* $\mathcal{A}$ *as long as* $\mathsf{P}$ *and* $\mathsf{O}$ *respect the transition function of* $\mathcal{A}$*, and switches to an accepting (resp. rejecting) state as soon as* $\mathsf{O}$ *(resp.* $\mathsf{P}$*) plays a move not allowed by* $\mathcal{A}$*. Assuming*

$$\delta_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}))$$

*we let* $\widehat{\mathcal{A}}$ *be the uniform automaton*

$$(\widehat{\mathcal{A}} : \Sigma) \quad := \quad (Q_{\mathcal{A}} + \mathbb{B} \, , \, q^\imath_{\mathcal{A}} \, , \, \mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}) \, , \, Q_{\mathcal{A}} \, , \, \delta_{\widehat{\mathcal{A}}} \, , \, \Omega_{\widehat{\mathcal{A}}})$$

*where* $\mathbb{B} := \{\mathbb{t}, \mathbb{f}\}$*, with transitions given by* $\delta_{\widehat{\mathcal{A}}}(\mathbb{b}, \mathsf{a}, \_, \_, \_) := \mathbb{b}$ *if* $\mathbb{b} \in \mathbb{B}$ *and for* $q \in Q_{\mathcal{A}}$*:*

$$\delta_{\widehat{\mathcal{A}}}(q, \mathsf{a}, \gamma, q', d) := \begin{cases} q' & \textit{if } \gamma \in \delta_{\mathcal{A}}(q, \mathsf{a}) \textit{ and } (q', d) \in \gamma \\ \mathbb{t} & \textit{if } \gamma \in \delta_{\mathcal{A}}(q, \mathsf{a}) \textit{ and } (q', d) \notin \gamma \\ \mathbb{f} & \textit{if } \gamma \notin \delta_{\mathcal{A}}(q, \mathsf{a}) \end{cases}$$

*and with* $\Omega_{\widehat{\mathcal{A}}} \; := \; \Omega_{\mathcal{A}} + Q^*_{\mathcal{A}}.\mathbb{t}^\omega$*.*

Given two uniform automata $\mathcal{A}$ and $\mathcal{B}$ over $\Sigma$, it is easy to define a *linear implication automaton* $(\mathcal{A} \multimap \mathcal{B})$ following the idea of (33).

**Definition 3.3** (Linear Implication of Uniform Automata)**.** *Assume* $\mathcal{A}$ *is as in (34) and let*

$$\mathcal{B} \quad = \quad (Q_{\mathcal{B}} \, , \, q^\imath_{\mathcal{B}} \, , \, V \, , \, Y \, , \, \delta_{\mathcal{B}} \, , \, \Omega_{\mathcal{B}})$$

*so that*

$$\begin{array}{lcccccccc} & \delta_{\mathcal{A}} & : & Q_{\mathcal{A}} \times \Sigma & \longrightarrow & U \times X & \longrightarrow & (\mathfrak{D} \longrightarrow Q_{\mathcal{A}}) \\ \textit{and} & \delta_{\mathcal{B}} & : & Q_{\mathcal{B}} \times \Sigma & \longrightarrow & V \times Y & \longrightarrow & (\mathfrak{D} \longrightarrow Q_{\mathcal{B}}) \end{array}$$

*We let* $(\mathcal{A} \multimap \mathcal{B})$ *be the automaton over* $\Sigma$ *defined as*

$$(\mathcal{A} \multimap \mathcal{B}) \quad := \quad (Q_{\mathcal{A}} \times Q_{\mathcal{B}} \, , \, (q^\imath_{\mathcal{A}}, q^\imath_{\mathcal{B}}) \, , \, V^U \times X^{U \times Y \times \mathfrak{D}} \, , \, U \times Y \, , \, \delta_{\mathcal{A} \multimap \mathcal{B}} \, , \, \Omega_{\mathcal{A} \multimap \mathcal{B}})$$

*with*

$$\delta_{\mathcal{A} \multimap \mathcal{B}}((q_{\mathcal{A}}, q_{\mathcal{B}}) \, , \, \mathsf{a} \, , \, (f, F) \, , \, (u, y) \, , \, d) \quad := \quad (q'_{\mathcal{A}} \, , \, q'_{\mathcal{B}})$$

*where*

$$q'_{\mathcal{A}} \quad = \quad \delta_{\mathcal{A}}(q_{\mathcal{A}} \, , \, \mathsf{a} \, , \, u \, , \, F(u, y, d) \, , \, d) \qquad \textit{and} \qquad q'_{\mathcal{B}} \quad = \quad \delta_{\mathcal{B}}(q_{\mathcal{B}} \, , \, \mathsf{a} \, , \, f(v) \, , \, y \, , \, d)$$

*and with* $((q_n, q'_n))_n \in \Omega_{\mathcal{A} \multimap \mathcal{B}}$ *iff* $((q_n)_n \in \Omega_{\mathcal{A}} \implies (q'_n)_n \in \Omega_{\mathcal{B}})$*. Note that* $\Omega_{\mathcal{A} \multimap \mathcal{B}}$ *is* $\omega$*-regular since* $\Omega_{\mathcal{A}}$ *and* $\Omega_{\mathcal{B}}$ *are both assumed to be* $\omega$*-regular.*

Definition 3.3 will be justified in §5. We obtain a notion of linear complement with

$$\mathcal{A}^{\perp\!\!\!\perp} \quad := \quad \mathcal{A} \multimap \perp\!\!\!\perp$$

where $\perp\!\!\!\perp$ is a particular automaton accepting no tree (see §5.4), and it follows from monoidal closure that $(-)^{\perp\!\!\!\perp}$ is a contravariant functor taking a total (winning) $\mathsf{P}$-strategy $\sigma : \mathcal{A}(M) \multimap \mathcal{B}(N)$ to a total (winning) $\sigma^{\perp\!\!\!\perp} : \mathcal{B}^{\perp\!\!\!\perp}(N) \multimap \mathcal{A}^{\perp\!\!\!\perp}(M)$.

**3.3 Full Positive Games and Acceptance for Uniform Automata.** The shape (35) of the transition functions of uniform automata allows their (substituted) acceptance games to be defined without using the notion of legal play. This leads to a slightly simpler setting than for usual automata.

**Definition 3.4** (Full Postitive Game)**.** *A* full positive game *(with winning) is a positive game $A$ (with winning) such that all its positive plays are legal, that is such that*

$$L_A \quad = \quad \wp^{\mathsf{P}}_A \quad = \quad (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* + (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* \cdot A_{\mathsf{P}}$$

*Note that a full positive game $A$ is completely characterized by its set of* P *and* O*-moves. We write $A = (U, X)$ to denote the full positive game $A$ with* P*-moves $A_{\mathsf{P}} = U$ and* O*-moves $A_{\mathsf{O}} = X$. We say that $A = (U, X)$ is* total *if $U$ and $X$ are non-empty.*

Consider a uniform automaton $\mathcal{A} : \Gamma$ as in (34), and a morphism $M \in \mathbf{T}[\Sigma, \Gamma]$. The *uniform substituted acceptance game* $\Sigma \vdash \mathcal{A}(M)$ is the full positive game with P-moves $\Sigma \times U$ and O-moves $X \times \mathfrak{D}$. So a play in $\Sigma \vdash \mathcal{A}(M)$ has the form

$$\cdot \xrightarrow{\mathsf{P}} (\mathtt{a}_0, u_0) \xrightarrow{\mathsf{O}} (x_0, d_0) \xrightarrow{\mathsf{P}} (\mathtt{a}_1, u_1) \xrightarrow{\mathsf{O}} \dots \xrightarrow{\mathsf{O}} (x_n, d_n) \xrightarrow{\mathsf{P}} (\mathtt{a}_{n+1}, u_{n+1}) \xrightarrow{\mathsf{O}} \dots$$

Similarly as in a substituted acceptance game for a usual non-deterministic or alternating automaton, P chooses input characters and O chooses tree directions. Note that because the sets of P and O-moves of a uniform automaton are always assumed to be non-empty (in this sense uniform automata are always total), there is no maximal finite play in the game $\mathcal{A}(M)$.

We now equip $\Sigma \vdash \mathcal{A}(M)$ with a winning condition $\mathcal{W}_{\mathcal{A}(M)} \subseteq ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$. Each infinite play $\chi = ((\mathtt{a}_k, u_k) \cdot (x_k, d_k))_k \in ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$ generates an infinite sequence of states $(q_k)_k \in Q^\omega_{\mathcal{A}}$ as follows. We let $q_0 := q^\imath_{\mathcal{A}}$ and

$$\begin{array}{rlcl} & q_{k+1} & := & \delta_{\mathcal{A}}(q_k \,,\, \mathtt{b}_k \,,\, u_k \,,\, x_k \,,\, d_k) \\ \text{where} & \mathtt{b}_k & := & M(\mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_k \,,\, d_0 \cdot \ldots \cdot d_{k-1}) \end{array}$$

Then $\chi$ is winning (*i.e.* $\chi \in \mathcal{W}_{\mathcal{A}(M)}$) iff $(q_k)_k$ is accepting (*i.e.* $(q_k)_k \in \Omega_{\mathcal{A}}$).

Let us set some notations. We write $\Gamma \vdash \mathcal{A}$ (or simply $\mathcal{A}$) for the game $\Gamma \vdash \mathcal{A}(\mathrm{Id}_\Gamma)$. Moreover, we extend the notation $\mathcal{A}[\mathtt{f}]$ of §2.2 to uniform automata.

**Definition 3.5.** *Given an ordinary function $\mathtt{f} : \Sigma \to \Gamma$ and a uniform automaton $\mathcal{A} : \Gamma$, we let $\mathcal{A}[\mathtt{f}] : \Sigma$ be the uniform automaton defined as $\mathcal{A}$, but with*

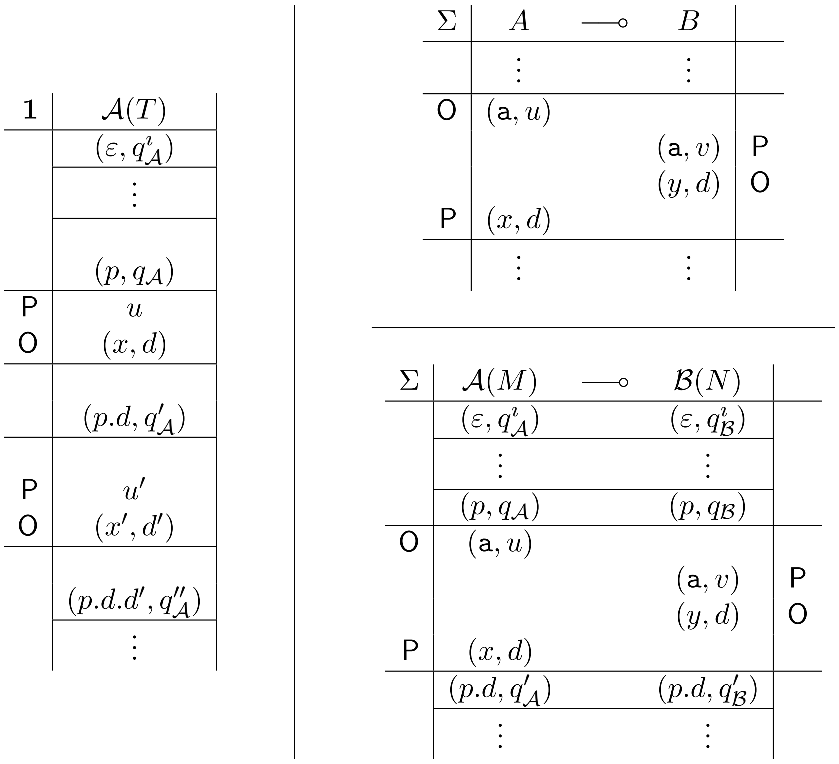
$$\delta_{\mathcal{A}[\mathtt{f}]}(q,\, \mathtt{a},\, u,\, x,\, d) \quad := \quad \delta_{\mathcal{A}}(q,\, \mathtt{f}(\mathtt{a}),\, u,\, x,\, d)$$

Similarly as in §2.2, the game $\Sigma \vdash \mathcal{A}([\mathtt{f}])$ is the same as the game $\Sigma \vdash \mathcal{A}[\mathtt{f}]$.

**Example 3.6.** *Continuing Ex. 3.2.(ii), given a usual alternating automaton $\mathcal{A}$ over $\Gamma$ and some $M \in \mathbf{T}[\Sigma, \Gamma]$,* P *has a winning strategy in $\mathcal{A}(M)$ if and only if* P *has a winning strategy in $\widehat{\mathcal{A}}(M)$.*

Consider a uniform automaton $\mathcal{A} : \Sigma$ and a $\Sigma$-labeled tree $T$. The game $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ (also written $\mathcal{A}(T)$) is similar to usual acceptance games (see §2.4). A typical play of $\mathcal{A}(T)$ is depicted on Fig. 9 (left). Note that the input alphabet of $\mathcal{A}(T)$ is $\mathbf{1}$, so that P only plays moves in $U$.

**Definition 3.7.** *Given a uniform automaton $\mathcal{A} : \Sigma$ and a $\Sigma$-labeled tree $T$, we say that $\mathcal{A}$* accepts *$T$ if* P *has a winning strategy in $\mathcal{A}(T)$, and we let $\mathcal{L}(\mathcal{A})$ be the set of $\Sigma$-labeled trees which are accepted by $\mathcal{A}$. Moreover, a set $\mathcal{L}$ of $\Sigma$-labeled trees is* regular *if there is an automaton $\mathcal{A} : \Sigma$ such that $\mathcal{L} = \mathcal{L}(\mathcal{A})$.*



$\mathbf{1}$ | $\mathcal{A}(T)$
$(\varepsilon, q^\imath_{\mathcal{A}})$
$(p, q_{\mathcal{A}})$
P $u$
O $(x, d)$
$(p.d, q'_{\mathcal{A}})$
P $u'$
O $(x', d')$
$(p.d.d', q''_{\mathcal{A}})$
$\Sigma$ | $A \multimap B$
O $(\mathsf{a}, u)$
$(\mathsf{a}, v)$ P
$(y, d)$ O
P $(x, d)$
$\Sigma$ | $\mathcal{A}(M) \multimap \mathcal{B}(N)$
$(\varepsilon, q^\imath_{\mathcal{A}})$ $(\varepsilon, q^\imath_{\mathcal{B}})$
$(p, q_{\mathcal{A}})$ $(p, q_{\mathcal{B}})$
O $(\mathsf{a}, u)$
$(\mathsf{a}, v)$ P
$(y, d)$ O
P $(x, d)$
$(p.d, q'_{\mathcal{A}})$ $(p.d, q'_{\mathcal{B}})$

Figure 9: Uniform automata and games

**3.4 Zig-Zag Games.** We will equip uniform automata with synchronous arrow games resulting from the obvious adaptation of [Rib15]. However, the presentation of the categorical structure will differ from [Rib15], and will rely on a very simple subcategory of the usual category **SG** of simple games consisting of (total) *zig-zag* strategies. As we shall see in §4 and §5, this will give us a decomposition of the indexed structure of synchronous arrow games allowing a smooth treatment of monoidal closure.

Consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$, where $\mathcal{A}$ and $\mathcal{B}$ are usual automata. Recall that the synchronicity constraint of [Rib15] presented in §2.6 imposes a legal P-play $s$ in $\mathcal{A}(M) \multimap \mathcal{B}(N)$ to satisfy

$$\mathrm{tr}_{\mathcal{A}(M)}(s_{\restriction \mathcal{A}(M)}) \quad = \quad \mathrm{tr}_{\mathcal{B}(N)}(s_{\restriction \mathcal{B}(N)})$$

Since the functions $\mathrm{tr}_{\mathcal{A}(M)}$ and $\mathrm{tr}_{\mathcal{B}(N)}$ are length-preserving, this imposes in particular $s_{\restriction \mathcal{A}(M)}$ and $s_{\restriction \mathcal{B}(N)}$ to have the same length. On the other hand, given simple games $A$ and $B$ of the same polarity, and a play $s$ in $A \multimap_{\mathbf{SG}} B$, if

$$\mathrm{length}(s_{\restriction A}) \quad = \quad \mathrm{length}(s_{\restriction B}) \tag{36}$$

then in $s$, each P-move must switch component w.r.t. the previous O-move. Let us discuss the case where (say) $A = (U, X)$ and $B = (V, Y)$ are full positive games. Recall that O begins in $A \multimap_{\mathbf{SG}} B$ and must play in component $A$ since $A$ and $B$ are positive. In order to maintain (36), P must then switch to component $B$. After the P-move in $B$, the switching condition imposes O to stay in $B$, and then P has to switch to $A$, again to maintain (36). It follows that $s$ must have the *zig-zag* shape depicted in Fig. 10.

$$\begin{array}{c|ccc|c} & A & \multimap_{\mathbf{DZ}} & B & \\ \hline & \vdots & & \vdots & \\ \hline \mathsf{O} & u & & & \\ & & & v & \mathsf{P} \\ & & & y & \mathsf{O} \\ \mathsf{P} & x & & & \\ \hline & \vdots & & \vdots & \end{array}$$

Figure 10: A typical zig-zag play with $A = (U, X)$ and $B = (V, Y)$ full positive games

**Definition 3.8** (Zig-Zag Plays and Strategies)**.** *Given simple games $A$ and $B$ of the same polarity, a play $s$ in $A \multimap_{\mathbf{SG}} B$ is a* zig-zag play *if*

$$\mathrm{length}(s_{\restriction A}) \quad = \quad \mathrm{length}(s_{\restriction B})$$

*A $\mathsf{P}$-strategy $\sigma : A \multimap_{\mathbf{SG}} B$ is a* zig-zag *strategy if all its plays are zig-zag plays.*

*We write $A \multimap_{\mathbf{DZ}} B$ for the game obtained by restricting $A \multimap_{\mathbf{SG}} B$ to (prefixes of) its legal zig-zag plays (so the $\mathsf{P}$-strategies on $A \multimap_{\mathbf{DZ}} B$ are exactly the zig-zag $\mathsf{P}$-strategies on $A \multimap_{\mathbf{SG}} B$).*

*Consider now games with winning $A$ and $B$. Note that if $\sigma : A \multimap_{\mathbf{DZ}} B$ is total, then for every $\chi \in ((A_{\mathsf{P}} + B_{\mathsf{O}}) \cdot (A_{\mathsf{O}} + B_{\mathsf{P}}))^{\omega}$, if $\chi$ has infinitely many finite prefixes in $\sigma$, then $\chi_{\restriction A}$ and $\chi_{\restriction B}$ are both infinite. We therefore let $\mathcal{W}_{A \multimap B} \subseteq ((A_{\mathsf{P}} + B_{\mathsf{O}}) \cdot (A_{\mathsf{O}} + B_{\mathsf{P}}))^{\omega}$ be the set of infinite sequences $\chi$ such that $(\chi_{\restriction A} \in \mathcal{W}_A \implies \chi_{\restriction B} \in \mathcal{W}_B)$.*

Full positive games (with winning) and total (winning) $\mathsf{P}$-strategies form a category $\mathbf{DZ}^{(\mathrm{W})}$ (to be defined in §3.6), which is the backbone of our categories of uniform automata. As we shall see in §4 and §5, with uniform automata, the synchronicity constraints of [Rib15] (see §2.6) have a decomposition in $\mathbf{DZ}$ which will turn out to be useful for the fibred monoidal-closed structure.

**3.5 Uniform Linear Synchronous Arrow Games.** The notion of linear synchronous arrow game of [Rib15] (see §2.6) readily extends to uniform automata. It is however convenient to work with a slight generalization.

Fix an alphabet $\Sigma$ and consider full positive games $A = (\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A)$ and $B = (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$. Note that $A$ and $B$ are not required to be substituted acceptance games. We define the *uniform linear synchronous arrow game*

$$\Sigma \vdash A \multimap B$$

(also noted $A \multimap B$ when $\Sigma$ is clear from the context) similarly as the synchronous arrow game of [Rib15], but with the obvious adaptation of the notion of trace. Given a full positive game $A = (\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A)$ as above, define

$$\mathrm{tr}_A \quad : \quad L_A \quad \longrightarrow \quad \mathrm{Tr}_{\Sigma}$$

inductively as follows:

$$\begin{array}{rcl} \mathrm{tr}_A(\varepsilon) & := & \varepsilon \\ \mathrm{tr}_A(s \cdot (\mathsf{a}, u)) & := & \mathrm{tr}_A(s) \cdot \mathsf{a} \\ \mathrm{tr}_A(s \cdot (x, d)) & := & \mathrm{tr}_A(s) \cdot d \end{array}$$

By directly adapting [Rib15], we say that a legal play $s$ in $A \multimap_{\mathbf{SG}} B$ is *synchronous* if

$$\mathrm{tr}_A(s_{\restriction A}) \quad = \quad \mathrm{tr}_B(s_{\restriction B})$$

Then, we let $A \multimap B$ be the restriction of $A \multimap_{\mathbf{SG}} B$ to the prefixes of its synchronous legal plays and call a $\mathsf{P}$-strategy on $A \multimap_{\mathbf{SG}} B$ *synchronous* if all its plays are synchronous. A typical play in $A \multimap B$ is depicted on Fig. 9 (top right). As expected, $\mathsf{P}$ has to play the same $\mathsf{a} \in \Sigma$ and tree directions $d \in \mathfrak{D}$ as chosen by $\mathsf{O}$. Moreover, similarly as with usual automata in §3.4, since the trace functions $\mathrm{tr}_A$ and $\mathrm{tr}_B$ are length-preserving, the (synchronous) plays in $A \multimap B$ are zig-zag, and $A \multimap B$ is a subgame of $A \multimap_{\mathbf{DZ}} B$. We therefore equip $A \multimap B$ with the winning condition $\mathcal{W}_{A \multimap B}$ of Def. 3.8, so that an infinite play of the form

$$\chi \quad = \quad ((\mathsf{a}_k, u_k) \cdot (\mathsf{a}_k, v_k) \cdot (y_k, d_k) \cdot (x_k, d_k))_k$$

is winning if and only if

$$((\mathsf{a}_k, u_k) \cdot (x_k, d_k))_k \;\in\; \mathcal{W}_A \quad \Longrightarrow \quad ((\mathsf{a}_k, v_k) \cdot (y_k, d_k))_k \;\in\; \mathcal{W}_B$$

**Notation 3.9.** *We write $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$ (or simply $\mathcal{A}(M) \multimap \mathcal{B}(N)$ when $\Sigma$ is clear from the context) for the uniform linear synchronous arrow game $\Sigma \vdash A \multimap B$ where the full positive games $A = (\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A)$ and $B = (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$ are induced by uniform substituted acceptance games as $A = (\Sigma \vdash \mathcal{A}(M))$ and $B = (\Sigma \vdash \mathcal{B}(N))$.*

A typical play of $\mathcal{A}(M) \multimap \mathcal{B}(N)$ is depicted on Fig. 9 (bottom right). We write $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ for the game $\mathcal{A}(\mathrm{Id}_\Sigma) \multimap \mathcal{B}(\mathrm{Id}_\Sigma)$ where $\mathcal{A}, \mathcal{B} : \Sigma$. Note that transitions do not depend anymore from the history of tree positions $p \in \mathfrak{D}^*$ since $q'_{\mathcal{A}} = \delta_{\mathcal{A}}(q_{\mathcal{A}}, \mathsf{a}, u, x, d)$ (and similarly for $q'_{\mathcal{B}}$).

**Example 3.10.** *Similarly as with usual automata (see §2.6), given a uniform $\mathcal{A} : \Sigma$ and a $\Sigma$-labeled tree $T$, $\mathsf{P}$ has a winning strategy in $\mathcal{A}(T)$ (i.e. $T \in \mathcal{L}(\mathcal{A})$) iff $\mathsf{P}$ has a winning strategy in $\mathbf{1} \vdash \mathbf{I_1} \multimap \mathcal{A}(\dot{T})$.*

**3.6 The Category $\mathbf{DZ}^{(\mathrm{W})}$ of Zig-Zag Games and Total (Winning) Strategies.** We now discuss composition and identities for total (winning) zig-zag strategies.

Let us step back to some well-known facts on simple games from [HS99]. There is a faithful functor $\mathrm{HS} : \mathbf{SG} \longrightarrow \mathbf{Rel}$ (the category of sets and relations) taking a simple game $A$ to its set of legal plays $L_A$, and a strategy $\sigma : A \multimap_{\mathbf{SG}} B$ to

$$\mathrm{HS}(\sigma) \quad := \quad \{(s_{\restriction A}, s_{\restriction B}) \mid s \in \sigma\} \quad \subseteq \quad L_A \times L_B$$

We can therefore faithfully represent strategies $\sigma : A \multimap B$ as spans

$$\begin{tikzcd} & \mathrm{HS}(\sigma) \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] & \\ L_A & & L_B \end{tikzcd} \tag{37}$$

and moreover, composition and identities of simple game can be seen as being induced from composition and identities on relations. Explicitly $\mathrm{HS}(\mathrm{id}_A)$ is identity relation on $L_A$ (as suggested by (17)), and given strategies $\sigma : A \multimap_{\mathbf{SG}} B$ and $\tau : B \multimap_{\mathbf{SG}} C$, the strategy $\tau \circ \sigma$ is the unique strategy such that $\mathrm{HS}(\tau \circ \sigma)$ is the relation $\mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$.

In other words, the category $\mathbf{SG}$ of simple games can be obtained from the category $\mathbf{Rel}$ thanks to the injectivity of HS seen as function from strategies to relations. In the case of total zig-strategies, composition and identities can be obtained along this scheme, but with much simpler combinatorics than with **SG**.

First, note that the map (that we still denote HS)

$$\mathrm{HS} \quad : \quad s \in L_{A \multimap_{\mathbf{SG}} B} \quad \longmapsto \quad (s_{\restriction A}, s_{\restriction B}) \in L_A \times L_B$$

is injective on *zig-zag* plays: given $(t, t') \in L_A \times L_B$, there is at most one *zig-zag* play $s$ such that $\mathrm{HS}(s) = (t, t')$. This immediately gives the injectivity of HS on *zig-zag* strategies.

**Lemma 3.11.** *(i) Given* zig-zag *plays* $s, t$ *in* $A \multimap_{\mathbf{DZ}} B$*, if* $\mathrm{HS}(s) = \mathrm{HS}(t)$ *then* $s = t$.

*(ii) The map* HS *is injective on* zig-zag *strategies:* $\mathrm{HS}(\sigma) = \mathrm{HS}(\tau)$ *implies* $\sigma = \tau$.

Second, *total* zig-zag strategies admit a very simple functional representation.

**Proposition 3.12.** *Consider full positive games* $A = (U, X)$ *and* $B = (V, Y)$*.* Total *zig-zag strategies* $\sigma : A \multimap_{\mathbf{DZ}} B$ *are in bijection with pairs of functions* $(f, F)$ *where*

$$\begin{array}{lclcl} f & : & \bigcup_{n \in \mathbb{N}} \left(U^{n+1} \times Y^n\right) & \longrightarrow & V \\ F & : & \bigcup_{n \in \mathbb{N}} \left(U^{n+1} \times Y^{n+1}\right) & \longrightarrow & X \end{array} \tag{38}$$

A detailed proof of Prop. 3.12 is given in [Rib17]. Given pairs of maps $(f, F)$ as in (38), for each $n > 0$, we write $f_n$ and $F_n$ for the induced maps

$$f_n \quad : \quad U^n \times Y^{n-1} \quad \longrightarrow \quad V^n \qquad \text{and} \qquad F_n \quad : \quad U^n \times Y^n \quad \longrightarrow \quad X^n$$

Note that for each full positive game $A = (U, X)$, there is a bijection (were $\wp_A^{\mathsf{even}}$ denotes the set of even-length plays of $A$):

$$\partial \quad = \quad \langle \partial_U, \partial_X \rangle \quad : \quad \wp_A^{\mathsf{even}} \quad \longrightarrow \quad \cup_{n \in \mathbb{N}} (U^n \times X^n)$$

defined as $\partial(\varepsilon) := (\varepsilon, \varepsilon)$ and $\partial(s.u.x) = (\partial_U(s).u, \partial_X(s).x)$.

The representation of strategies as pairs of maps $(f, F)$ of the form (38) provides an easy way to compose total zig-zag strategies. Given total zig-zag strategies $\sigma : A \multimap_{\mathbf{DZ}} B$ and $\tau : B \multimap_{\mathbf{DZ}} C$, we are looking for a composite $\tau \circ \sigma$. By injectivity of HS, it is sufficient to show that there exists a strategy $\theta$ such that $\mathrm{HS}(\theta) = \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$. But thanks to Prop. 3.12, given pairs of maps $(f, F)$ and $(g, G)$ representing resp. $\sigma$ and $\tau$, this amounts to provide a pair $(h, H)$ representing $\theta$. Write $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$. The relational composite $\mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$ is such that $(\partial^{-1}(\overline{u}, \overline{x}), \partial^{-1}(\overline{w}, \overline{z})) \in \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$ if and only if there are $(\overline{v}, \overline{y})$ such that

$$(\partial^{-1}(\overline{u}, \overline{x}) \, , \, \partial^{-1}(\overline{v}, \overline{y})) \in \mathrm{HS}(\sigma) \qquad \text{and} \qquad (\partial^{-1}(\overline{v}, \overline{y}) \, , \, \partial^{-1}(\overline{w}, \overline{z})) \in \mathrm{HS}(\tau)$$

But by Prop. 3.12 this is possible if and only if the following equations are satisfied:

$$\begin{array}{lcl} \overline{v} & = & f_n(\overline{u}, \, \blacktriangleright\overline{y}) \\ \overline{x} & = & F_n(\overline{u}, \, \overline{y}) \end{array} \qquad\qquad \begin{array}{lcl} \overline{w} & = & g_n(\overline{v}, \, \blacktriangleright\overline{z}) \\ \overline{y} & = & G_n(\overline{v}, \, \overline{z}) \end{array} \tag{39}$$

(where $\blacktriangleright\varepsilon := \varepsilon$ and $\blacktriangleright\overline{y}.y := \overline{y}$). The derived equation

$$\overline{y} \quad = \quad G_n(f_n(\overline{u}, \, \blacktriangleright\overline{y}) \, , \, \overline{z})$$

determines $\overline{y} = y(\overline{u},\overline{z}) = y_1.\dots.y_n$ uniquely from $\overline{u} = u_1.\dots.u_n$ and $\overline{z} = z_1.\dots.z_n$, as

$$y_k \quad = \quad G(f_k(u_1.\dots.u_k,\, y_1.\dots.y_{k-1})\,,\, z_1.\dots.z_k) \tag{40}$$

We can thus define a pair of maps

$$\begin{array}{lcllcl} h & : & \bigcup_{n\in\mathbb{N}}\left(U^{n+1}\times Z^n\right) & \longrightarrow & W \\ H & : & \bigcup_{n\in\mathbb{N}}\left(U^{n+1}\times Z^{n+1}\right) & \longrightarrow & X \end{array}$$

as follows

$$\begin{array}{rcl} h(\overline{u}u,\,\overline{z}) & := & g(f_{n+1}(\overline{u}u,\, y(\overline{u},\overline{z}))\,,\,\overline{z}) \\ H(\overline{u}u,\,\overline{z}z) & := & F(\overline{u}u,\, y(\overline{u}u,\overline{z}z)) \end{array}$$

Then, by construction of $(h,H)$, the total strategy $\theta : A \multimap_{\mathbf{DZ}} C$ it represents is such that $\mathrm{HS}(\theta) = \mathrm{HS}(\tau)\circ\mathrm{HS}(\sigma)$, so that we can let $\tau\circ\sigma := \theta$.

Note that the strategy $\tau\circ\sigma$ is total. Hence, totality is preserved by composition of *zig-zag* strategies, while on the other hand, it is well-known that totality is *not* preserved by composition of *arbitrary* **SG**-strategies (see e.g. [Abr97]).

In §2.7, we alluded to the usual method to compose **SG**-strategies $\sigma : A \multimap_{\mathbf{SG}} B$ and $\tau : B \multimap_{\mathbf{SG}} C$, which proceeds by letting $\sigma$ and $\tau$ interact in their common component $B$, and then hiding this interaction (see e.g. [Abr97, Hyl97] for details). This relies on the usual *zipping* property, stating that the interactions of $\sigma$ and $\tau$ in component $B$ are completely determined by the $\mathsf{O}$-moves in components $A$ and $C$ (with the polarities of $A \multimap_{\mathbf{SG}} C$). In our case, the zipping property follows from the definitions of $\overline{y}$ from $\overline{u}$ and $\overline{z}$ by (40), and of $\overline{v}$ and $\overline{x}$ from $\overline{u}$ and $\overline{y}$ in (39). We have in particular the following relational version of zipping (which actually holds for the full **SG** [HS99]):

**Lemma 3.13** (Relational Zipping)**.** *Given total zig-zag $\sigma : A \multimap_{\mathbf{DZ}} B$ and $\tau : B \multimap_{\mathbf{DZ}} C$, and given $(t_A, t_C) \in \mathrm{HS}(\tau)\circ\mathrm{HS}(\sigma)$, there is exactly one legal play $t_B \in L_B$ such that $(t_A, t_B) \in \mathrm{HS}(\sigma)$ and $(t_B, t_C) \in \mathrm{HS}(\tau)$.*

From the relational zipping Lemma 3.13 one gets the usual and expected fact that if $\sigma : A \multimap_{\mathbf{DZ}} B$ and $\tau : B \multimap_{\mathbf{DZ}} C$ are both total and winning, then $\tau\circ\sigma$ is total and winning. Indeed, given an infinite play $\chi \in ((A_{\mathsf{O}} + C_{\mathsf{P}})\cdot(A_{\mathsf{P}} + C_{\mathsf{O}}))^\omega$ of $\tau\circ\sigma$ (that is such that $\exists^\infty k \in \mathbb{N}.\ \chi(0)\cdot\ldots\cdot\chi(k) \in \tau\circ\sigma$), it follows from Lem. 3.13 that there are infinite plays $\chi_\sigma$ and $\chi_\tau$ of resp. $\sigma$ and $\tau$ such that

$$(\chi_\sigma)_{\restriction A} \;=\; \chi_{\restriction A} \qquad\text{and}\qquad (\chi_\sigma)_{\restriction B} \;=\; (\chi_\tau)_{\restriction B} \qquad\text{and}\qquad (\chi_\tau)_{\restriction C} \;=\; \chi_{\restriction C}$$

from which we get

$$(\chi_{\restriction A} \in \mathcal{W}_A) \quad\Longrightarrow\quad ((\chi_\sigma)_{\restriction B} \;=\; (\chi_\tau)_{\restriction B} \;\in\; \mathcal{W}_B) \quad\Longrightarrow\quad (\chi_{\restriction C} \in \mathcal{W}_C)$$

**Definition 3.14** (The Categories **DZ** and $\mathbf{DZ}^{\mathrm{W}}$)**.** *The category $\mathbf{DZ}^{(\mathrm{W})}$ has* total *full positive games (with winning) as objects (see Def. 3.4). Maps from $A$ to $B$ are total (winning) zig-zag strategies $\sigma : A \multimap_{\mathbf{DZ}} B$.*

Note that the identity and associativity laws for composition of strategies are lifted from the corresponding laws in **Rel** by the injectivity of HS. On the one hand, we have

$$\mathrm{id}_B\circ\sigma \quad=\quad \sigma \quad=\quad \sigma\circ\mathrm{id}_A$$

since $\mathrm{HS}(\mathrm{id}_A)$ (resp. $\mathrm{HS}(\mathrm{id}_B)$)) is the identity relation on $L_A$ (resp. $L_B$) and since

$$\mathrm{HS}(\mathrm{id}_B)\circ\mathrm{HS}(\sigma) \quad=\quad \mathrm{HS}(\sigma) \quad=\quad \mathrm{HS}(\sigma)\circ\mathrm{HS}(\mathrm{id}_A)$$

On the other hand, the associativity of composition (that is $\theta\circ(\tau\circ\sigma) = (\theta\circ\tau)\circ\sigma$) follows from the fact that $\mathrm{HS}(\theta)\circ(\mathrm{HS}(\tau)\circ\mathrm{HS}(\sigma)) = (\mathrm{HS}(\theta)\circ\mathrm{HS}(\tau))\circ\mathrm{HS}(\sigma)$.

**3.7 Categories of Uniform Synchronous Arrow Games.** We now define, for each alphabet $\Sigma$, categories $\mathbf{SAG}_{\Sigma}^{(\mathrm{W})}$ of uniform synchronous arrow games (with winning) over $\Sigma$. Composition and identities are based on $\mathbf{DZ}^{(\mathrm{W})}$, and on the representation of strategies as spans (37).

Consider a uniform synchronous arrow game $\Sigma \vdash A \multimap B$. Then, a $\mathsf{P}$-strategy $\sigma$ on $A \multimap_{\mathbf{SG}} B$ is synchronous if and only the following diagram commutes:

$$\begin{tikzcd}
 & \mathrm{HS}(\sigma) \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] & \\
L_A \arrow[dr, "\mathrm{tr}_A"'] & & L_B \arrow[dl, "\mathrm{tr}_B"] \\
 & \mathrm{Tr}_\Sigma &
\end{tikzcd} \tag{41}$$

Hence, if $\sigma$ is synchronous, then given a further $\mathsf{P}$-strategy $\tau$ on $\Sigma \vdash B \multimap C$, we have

$$\begin{tikzcd}
 & & \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma) \arrow[dl] \arrow[dr] & & \\
 & \mathrm{HS}(\sigma) \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] & & \mathrm{HS}(\tau) \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] & \\
L_A \arrow[drr, "\mathrm{tr}_A"'] & & L_B \arrow[d, "\mathrm{tr}_B"] & & L_C \arrow[dll, "\mathrm{tr}_C"] \\
 & & \mathrm{Tr}_\Sigma & &
\end{tikzcd}$$

(where by Lem. 3.13 top diamond is a pullback). Since $\mathrm{HS}(\tau \circ \sigma) = \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$, it follows that $\tau \circ \sigma$ is synchronous. We thus get the following categories of uniform synchronous arrow games (recall from §2.6 that identity strategies are synchronous).

**Definition 3.15** (The Categories $\mathbf{SAG}_{\Sigma}^{(\mathrm{W})}$)**.** *The objects of* $\mathbf{SAG}_{\Sigma}$ *are pairs* $(U, X)$ *of non-empty sets. The maps from* $(U, X)$ *to* $(V, Y)$ *are* total $\mathsf{P}$*-strategies on* $\Sigma \vdash (\Sigma \times U, X \times \mathfrak{D}) \multimap (\Sigma \times V, Y \times \mathfrak{D})$.

*The objects of* $\mathbf{SAG}_{\Sigma}^{\mathrm{W}}$ *have the form* $(U, X, \mathcal{W}_A)$*, where* $U, X$ *are non-empty sets and where* $\mathcal{W}_A \subseteq ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^{\omega}$. $\mathbf{SAG}_{\Sigma}^{\mathrm{W}}$*-maps from* $(U, X, \mathcal{W}_A)$ *to* $(V, Y, \mathcal{W}_B)$ winning $\mathsf{P}$*-strategies on* $\Sigma \vdash (\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A) \multimap (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$.

Given a uniform substituted acceptance game $A = (\Sigma \vdash \mathcal{A}(M))$ with $\mathcal{A}$ as in (34), write $A$ for the $\mathbf{SAG}_{\Sigma}^{(\mathrm{W})}$-object $(U, X)$ (resp. $(U, X, \mathcal{W}_A)$).

## 4 Fibrations of Tree Automata

In this Section we present an indexed structure for uniform synchronous linear arrow games, in which morphisms $L \in \mathbf{T}[\Delta, \Sigma]$ induce *substitution functors*, and such that the operation $(-)^{\star}$ is itself functorial (see §2.2 and [Jac01, Chap. 1]). While substitution in [Rib15] was defined directly at the level of synchronous arrow games (*via* the representation of strategies as relations), we devise here an indexed structure induced by a reformulation of synchronicity using monoid and comonoid indexing (and inspired from [HS99, HS03]) on zig-zag games, and which allows a smooth treatment of monoidal closure and universal quantifications. This will lead us to a category $\mathsf{DialAut}$, fibred over $\mathbf{T}$, and whose fibre over $\Sigma$ is isomorphic to $\mathbf{SAG}_{\Sigma}^{\mathrm{W}}$.

The material of this section relies on the symmetric monoidal structure of $\mathbf{DZ}$.

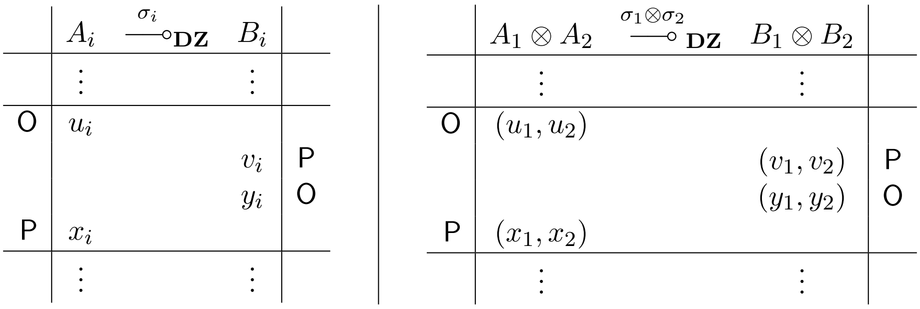

$A_i \xrightarrow{\sigma_i}_{\mathbf{DZ}} B_i$
O
$u_i$
$v_i$
P
$y_i$
O
P
$x_i$
$A_1 \otimes A_2 \xrightarrow{\sigma_1\otimes\sigma_2}_{\mathbf{DZ}} B_1 \otimes B_2$
O
$(u_1, u_2)$
$(v_1, v_2)$
P
$(y_1, y_2)$
O
P
$(x_1, x_2)$

Figure 11: Action of $\otimes$ on $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i$.

**Remark 4.1** (On Substitution in [Rib15])**.** *Looking at (41) (§3.7), synchronous strategies in* $\mathbf{SAG}_\Sigma^{(\mathrm{W})}$ *can be seen as relations in the slice category* $\mathbf{Set}/\mathrm{Tr}_\Sigma$*. Actually, in [Rib15], substitution functors (see (29), §2.10) are similar to substitution in the usual codomain fibrations (see e.g. [Jac01, Chap. 1]). We actually get a* strict *indexed structure since in contrast with the usual codomain fibrations, substitution in [Rib15] is given by* chosen *pullbacks.*

## 4.1 Symmetric Monoidal Structure of DZ

The category **DZ** has a particularly simple symmetric monoidal structure, but which differs from the usual ones in game semantics.

**Proposition 4.2.** *The category* **DZ** *is symmetric monoidal with unit* $\mathbf{I} := (\mathbf{1}, \mathbf{1})$ *and with* $A \otimes B := (U \times V, X \times Y)$ *for* $A = (U, X)$ *and* $B = (V, Y)$*.*

*The action of the tensor* $\otimes$ *on strategies* $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i$ *(for* $i = 1, 2$*,* $A_i = (U_i, X_i)$ *and* $B_i = (V_i, Y_i)$*) is depicted on Fig. 11. If the* $\sigma_i$ *are represented via Prop. 3.12 by pairs of functions* $(f_i, F_i)$ *where*

$$\begin{array}{rclcl} f_i & : & \bigcup_{n\in\mathbb{N}} \left(U_i^{n+1} \times Y_i^{n}\right) & \longrightarrow & V_i \\ F_i & : & \bigcup_{n\in\mathbb{N}} \left(U_i^{n+1} \times Y_i^{n+1}\right) & \longrightarrow & X_i \end{array}$$

*then* $\sigma_1 \otimes \sigma_2$ *is represented by* $(h, H)$ *where*

$$\begin{array}{rclcl} h & : & \bigcup_{n\in\mathbb{N}} \left((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n}\right) & \longrightarrow & V_1 \times V_2 \\ H & : & \bigcup_{n\in\mathbb{N}} \left((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n+1}\right) & \longrightarrow & X_1 \times X_2 \end{array}$$

*are defined as*

$$\begin{array}{rcl} h((\overline{u}_1, \overline{u}_2)\,,\,(\overline{y}_1, \overline{y}_2)) & := & (f_1(\overline{u}_1, \overline{y}_1)\,,\,f_2(\overline{u}_2, \overline{y}_2)) \\ H((\overline{u}_1, \overline{u}_2)\,,\,(\overline{y}_1, \overline{y}_2)) & := & (F_1(\overline{u}_1, \overline{y}_1)\,,\,F_2(\overline{u}_2, \overline{y}_2)) \end{array}$$

The natural structure isomorphisms of **DZ** are depicted on Fig. 12. This structure obviously lifts to $\mathbf{DZ}^{\mathrm{W}}$, but we shall not directly use this fact.

## 4.2 Monoid and Comonoid Indexing in DZ

Fix an alphabet $\Sigma$. For each $\mathbf{SAG}_\Sigma$-object $A = (U, X)$, there is a **DZ**-isomorphism

$$\iota_A \quad : \quad (\Sigma \times U,\ X \times \mathfrak{D}) \quad \overset{\simeq}{\multimap}_{\mathbf{DZ}} \quad (\Sigma, \mathbf{1}) \otimes ((U, X) \otimes (\mathbf{1}, \mathfrak{D})) \tag{42}$$

|  | $(A \otimes B) \otimes C$ | $\overset{\alpha_{A,B,C}}{\multimap_{\mathbf{DZ}}}$ | $A \otimes (B \otimes C)$ |  |
|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  |
| O | $((u,v),w)$ |  |  |  |
|  |  |  | $(u,(v,w))$ | P |
|  |  |  | $(x,(y,z))$ | O |
| P | $((x,y),z)$ |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  |

|  | $\mathbf{I} \otimes A$ | $\overset{\lambda_A}{\multimap}$ | $A$ |  |
|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  |
| O | $(\bullet, u)$ |  |  |  |
|  |  |  | $u$ | P |
|  |  |  | $x$ | O |
| P | $(\bullet, x)$ |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  |

|  | $A \otimes B$ | $\overset{\gamma_{A,B}}{\multimap_{\mathbf{DZ}}}$ | $B \otimes A$ |  |
|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  |
| O | $(u,v)$ |  |  |  |
|  |  |  | $(v,u)$ | P |
|  |  |  | $(y,x)$ | O |
| P | $(x,y)$ |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  |

|  | $A \otimes \mathbf{I}$ | $\overset{\rho_A}{\multimap}$ | $A$ |  |
|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  |
| O | $(u, \bullet)$ |  |  |  |
|  |  |  | $u$ | P |
|  |  |  | $x$ | O |
| P | $(x, \bullet)$ |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  |

Figure 12: The structure maps of $\mathbf{DZ}$, for $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$

|  | $\Sigma \otimes A$ | $\multimap_{\mathbf{DZ}}$ | $B \otimes \mathfrak{D}$ |  |
|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  |
| O | $(\mathsf{a}, u)$ |  |  |  |
|  |  |  | $v$ | P |
|  |  |  | $(y, d)$ | O |
| P | $x$ |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  |

Figure 13: A typical play in $\Sigma \otimes A \multimap_{\mathbf{DZ}} B \otimes \mathfrak{D}$

Let us write $\mathfrak{D}$ for the $\mathbf{DZ}$-object $(\mathbf{1}, \mathfrak{D})$ and $\Sigma$ for $(\Sigma, \mathbf{1})$, and consider the zig-zag game $\Sigma \otimes A \multimap_{\mathbf{DZ}} B \otimes \mathfrak{D}$, as depicted on Fig. 13. We shall now see that there is a category $\mathsf{DialZ}(\Sigma)$, whose objects are full positive games, whose maps from $A$ to $B$ are $\mathbf{DZ}$-maps

$$\sigma \quad : \quad \Sigma \otimes A \quad \multimap_{\mathbf{DZ}} \quad B \otimes \mathfrak{D}$$

and which is isomorphic to $\mathbf{SAG}_\Sigma$.

First, $\mathbf{SAG}_\Sigma$ is isomorphic to the category $\mathbf{SAG}^\bullet_\Sigma$ whose objects are full positive games $A = (U, X)$ and whose maps from $A$ to $B = (V, Y)$ are $\mathbf{DZ}$-maps $\sigma : \Sigma \otimes (A \otimes \mathfrak{D}) \multimap_{\mathbf{DZ}} \Sigma \otimes (B \otimes \mathfrak{D})$ such that the composite

$$\iota(\sigma) \quad := \quad \iota_B^{-1} \circ \sigma \circ \iota_A \quad : \quad (\Sigma \times U, X \times \mathfrak{D}) \quad \multimap_{\mathbf{DZ}} \quad (\Sigma \times V, Y \times \mathfrak{D})$$

is synchronous.

Second, the category $\mathbf{SAG}^\bullet_\Sigma$ is isomorphic to $\mathsf{DialZ}(\Sigma)$. This relies on some algebraic structure. Objects of the form $(\mathbf{1}, M)$ (resp. $(K, \mathbf{1})$) are actually (commutative) monoids (resp.

$$\begin{tikzcd}
(M\otimes M)\otimes M \arrow[r, "\alpha"] \arrow[d, "m\otimes \mathrm{id}_M"'] & M\otimes (M\otimes M) \arrow[r, "\mathrm{id}_M\otimes m"] & M\otimes M \arrow[d, "m"] \\
M\otimes M \arrow[rr, "m"'] & & M
\end{tikzcd}$$

$$\begin{tikzcd}
\mathbf{I}\otimes M \arrow[r, "u\otimes \mathrm{id}_M"] \arrow[dr, "\lambda"'] & M\otimes M \arrow[d, "m"] & M\otimes \mathbf{I} \arrow[l, "\mathrm{id}_M\otimes u"'] \arrow[dl, "\rho"] \\
& M &
\end{tikzcd}
\qquad
\begin{tikzcd}
M\otimes M \arrow[rr, "\gamma"] \arrow[dr, "m"] & & M\otimes M \arrow[dl, "m"'] \\
& M &
\end{tikzcd}$$

Figure 14: Coherence for a monoid $(M, m, u)$ (where $\alpha$, $\lambda$, $\rho$ and $\gamma$ are symmetric monoidal structure maps)

comonoids) in **DZ**. Recall from e.g. [Mel09] that a commutative monoid in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$ is an object $M$ equipped with structure maps $m : M \otimes M \to M$ and $u : \mathbf{I} \to M$ subject to coherence conditions depicted on Fig. 14. A (commutative) comonoid in $\mathbb{C}$ is a (commutative) monoid in $\mathbb{C}^{\mathrm{op}}$. In this paper, by (co)monoid we always mean *commutative* (co)monoid. Write **Comon**$(\mathbb{C})$ for the category of comonoids in $\mathbb{C}$. Maps from $(K, d, e)$ to $(K', d', e')$ are $\mathbb{C}$-maps $f : K \to K'$ which commute with the comonoid structure:

$$(f \otimes f) \circ d \;=\; d' \circ f \qquad \text{and} \qquad e \;=\; e' \circ f \tag{43}$$

It is well-known that the symmetric monoidal structure of $\mathbb{C}$ induces a Cartesian product on **Comon**$(\mathbb{C})$ (see e.g. [Mel09, Cor. 18, §6.5]), and conversely that if $(\mathbb{C}, \otimes, \mathbf{I})$ is Cartesian, then every $\mathbb{C}$-object has a canonical comonoid structure. Moreover, note that any set $I \simeq \mathbf{1}$ is a monoid in **Set**.

**Proposition 4.3.** *If $M, K$ are non-empty sets and $I \simeq \mathbf{1}$, then $M := (I, M)$ is a monoid and $K := (K, I)$ is a comonoid in* **DZ**. *Structure maps are depicted on Fig. 15 (in the case of $I = \mathbf{1}$).*

It is well-known (see e.g. [HS99, HS03]) that a monoid $M$ (resp. a comonoid $K$) in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$ induces a monad $(-) \otimes M$ of indexing with $M$ (resp. a comonad $K \otimes (-)$ of indexing with $K$).

**Proposition 4.4.** *Let $(\mathbb{C}, \otimes, \mathbf{I})$ be a symmetric monoidal category.*

*(a) A monoid $(M, m, u)$ in $\mathbb{C}$ induces a (lax symmetric monoidal) monad $((-) \otimes M, \mu, \eta)$. The functor $(-) \otimes M$ takes an object $A$ to $A \otimes M$ and a map $f : A \to B$ to $f \otimes \mathrm{id}_M : A \otimes M \to B \otimes M$. The natural maps $\mu$ and $\eta$ are given by*

$$\begin{array}{lclcrcl}
\mu_A & := & (\mathrm{id}_A \otimes m) \circ \alpha & : & (A \otimes M) \otimes M & \longrightarrow & A \otimes M \\
\eta_A & := & (\mathrm{id}_A \otimes u) \circ \rho^{-1} & : & A & \longrightarrow & A \otimes M
\end{array}$$

*(b) Dually, a comonoid $K = (K, d, e)$ in $\mathbb{C}$ induces an (oplax symmetric monoidal) comonad $(K \otimes (-), \delta, \epsilon)$, where*

$$\begin{array}{lclclcl}
\delta_A & := & \alpha \circ (d \otimes \mathrm{id}_A) & : & K \otimes A & \longrightarrow & K \otimes (K \otimes A) \\
\epsilon_A & := & \lambda \circ (e \otimes \mathrm{id}_A) & : & K \otimes A & \longrightarrow & A
\end{array}$$

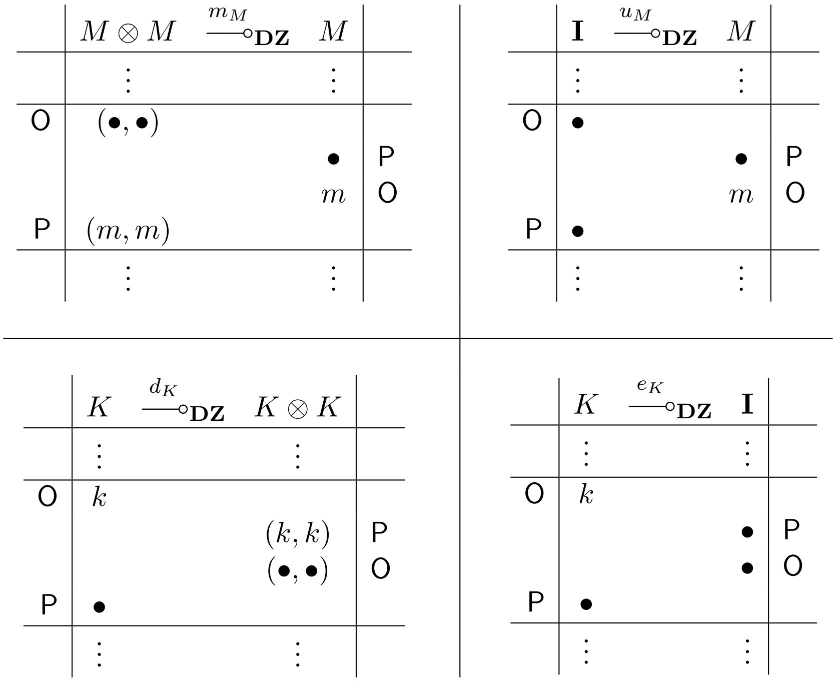

$M \otimes M \xrightarrow{m_M}_{\mathbf{DZ}} M$
O
$(\bullet,\bullet)$
$\bullet$
P
$m$
O
P
$(m,m)$
$\mathbf{I} \xrightarrow{u_M}_{\mathbf{DZ}} M$
O
$\bullet$
$\bullet$
P
$m$
O
P
$\bullet$
$K \xrightarrow{d_K}_{\mathbf{DZ}} K \otimes K$
O
$k$
$(k,k)$
P
$(\bullet,\bullet)$
O
P
$\bullet$
$K \xrightarrow{e_K}_{\mathbf{DZ}} \mathbf{I}$
O
$k$
$\bullet$
P
$\bullet$
O
P
$\bullet$

Figure 15: Structure maps for the monoid $M = (\mathbf{1}, M)$ and the comonoid $K = (K, \mathbf{1})$

*The maps $\rho$, $\alpha$ and $\lambda$ above are structural isomorphisms of* $(\mathbb{C}, \otimes, \mathbf{I})$.

Moreover, the comonad $K \otimes (-)$ is related to the monad $(-) \otimes M$ *via* a *distributive law*. A distributive law $\Lambda$ of a comonad $(G, \delta, \epsilon)$ over a monad $(T, \mu, \eta)$ on $\mathbb{C}$ is a natural map $\Lambda : G \circ T \Rightarrow T \circ G$ subject to some coherence conditions (see e.g. [HHM07]), which ensure that we have a category $\mathbf{Kl}(\Lambda)$ with the same objects as $\mathbb{C}$ and with homsets

$$\mathbf{Kl}(\Lambda)[A, B] \quad := \quad \mathbb{C}[GA, TB]$$

and that there is a lifting functor $(-)^{\uparrow} : \mathbf{Kl}(\Lambda) \to \mathbb{C}$ taking $f : GA \to TB$ to

$$f^{\uparrow} \quad := \quad G(\mu_B \circ Tf \circ \Lambda_A) \circ \delta_{TA} \quad : \quad GTA \quad \longrightarrow \quad GTB$$

In the case of comonoid and monoid indexing, a distributive law of $K \otimes (-)$ over $(-) \otimes M$ is given by the natural associativity maps:

$$\Phi_{(-)} \quad := \quad \alpha^{-1}_{K,(-),M} \quad : \quad K \otimes ((-) \otimes M) \quad \Longrightarrow \quad (K \otimes (-)) \otimes M$$

Returning to our case, we let

$$\mathsf{DialZ}(\Sigma) \quad := \quad \mathbf{Kl}(\Phi)$$

where $\Phi$ is the distributive law of the comonad of indexing with the comonoid $\Sigma$ over the monad of indexing with the monoid $\mathfrak{D}$ in the category $\mathbf{DZ}$. The lifting functor

$$(-)^{\uparrow} \quad : \quad \mathsf{DialZ}(\Sigma) \quad \longrightarrow \quad \mathbf{DZ}$$

takes a total zig-zag strategy

$$\sigma \quad : \quad \Sigma \otimes A \quad \multimap_{\mathbf{DZ}} \quad B \otimes \mathfrak{D}$$

to a total $\mathbf{SAG}^{\bullet}_{\Sigma}$-map from $A$ to $B$, that is to a total zig-zag strategy

$$\sigma^{\uparrow} \quad : \quad \Sigma \otimes (A \otimes \mathfrak{D}) \quad \multimap_{\mathbf{DZ}} \quad \Sigma \otimes (B \otimes \mathfrak{D})$$

such that composition with the isomorphisms (42) gives a synchronous strategy

$$\iota(\sigma^{\uparrow}) \quad : \quad (\Sigma \times U,\, X \times \mathfrak{D}) \quad \multimap \quad (\Sigma \times V,\, Y \times \mathfrak{D})$$

Modulo associativity, the strategy $\sigma^{\uparrow}$ is given by

$$(\mathrm{id}_{\Sigma} \otimes ((\mathrm{id}_B \otimes m_{\mathfrak{D}}) \circ (\sigma \otimes \mathrm{id}_{\mathfrak{D}}))) \circ (d_{\Sigma} \otimes \mathrm{id}_{A \otimes \mathfrak{D}}) \quad : \quad \Sigma \otimes A \otimes \mathfrak{D} \quad \multimap_{\mathbf{DZ}} \quad \Sigma \otimes B \otimes \mathfrak{D}$$

Note that if $\sigma$ plays as in Fig. 13, then the strategy

$$\dot{\sigma} \quad := \quad (\mathrm{id}_B \otimes m_{\mathfrak{D}}) \circ (\sigma \otimes \mathrm{id}_{\mathfrak{D}}) \quad : \quad \Sigma \otimes A \otimes \mathfrak{D} \quad \multimap_{\mathbf{DZ}} \quad B \otimes \mathfrak{D}$$

plays as in Fig. 16 (top). It follows that $\sigma^{\uparrow} = (\mathrm{id}_{\Sigma} \otimes \dot{\sigma}) \circ (d_{\Sigma} \otimes \mathrm{id}_{A \otimes \mathfrak{D}})$ plays as in Fig. 16 (bottom), so that the strategy $\iota(\sigma^{\uparrow})$ induced by (42) plays as in Fig. 9 (top right).

In the other direction, consider the map which takes a $\mathbf{SAG}^{\bullet}_{\Sigma}$-map $\sigma$ from $A$ to $B$ to to the composite

$$\begin{aligned} \mathring{\sigma} \quad := \quad \Sigma \otimes A \quad & \overset{\mathrm{id}_{\Sigma} \otimes \eta_A}{\multimap}_{\mathbf{DZ}} \quad \Sigma \otimes (A \otimes \mathfrak{D}) \quad \overset{\sigma}{\multimap}_{\mathbf{DZ}} \quad \Sigma \otimes (B \otimes \mathfrak{D}) \\ & \overset{\Phi_B}{\multimap}_{\mathbf{DZ}} \quad (\Sigma \otimes B) \otimes \mathfrak{D} \quad \overset{\epsilon_B \otimes \mathrm{id}_{\mathfrak{D}}}{\multimap}_{\mathbf{DZ}} \quad B \otimes \mathfrak{D} \end{aligned}$$

Note that we have $\sigma = (\mathring{\sigma})^{\uparrow}$. Since $(-)^{\uparrow}$ is injective, it follows that $(\mathring{-})$ is functorial.

## 4.3 The Indexed Structure of $\mathsf{DialZ}(-)$ and the Base Category $\mathbb{T}$

We therefore have for each alphabet $\Sigma$ a category $\mathsf{DialZ}(\Sigma)$ which is isomorphic to $\mathbf{SAG}_{\Sigma}$. We now discuss an indexed structure on the categories $\mathsf{DialZ}(-)$, based on pattern similar to the *simple fibration* $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ over a category $\mathbb{B}$ with finite products (see e.g. [Jac01, Chap. 1] but also [Hyl02, Hof11]). The objects of $\mathsf{s}(\mathbb{B})$ are pairs $(I, X)$ of $\mathbb{B}$-objects. The morphisms $(I, X) \to (J, Y)$ are pairs $(f_0, f)$ with $f_0 : I \to J$ and $f : I \times X \to Y$. The functor $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ is the first projection, and the fibre over $I$ is the Kleiseli category of indexing with the comonoid $I$ (see e.g. [Jac01, Ex. 1.3.4]).

A similar construction can be done if instead of a category $\mathbb{B}$ with finite products, one starts from a symmetric monoidal category $\mathbb{C}$, and take as base the category $\mathbf{Comon}(\mathbb{C})$. The fibre over the comonoid $K$ is the Kleisli category $\mathbf{Kl}(K)$ of indexing with $K$, and a comonoid morphism $u : K \to L$ induces a functor $u^{\star} : \mathbf{Kl}(L) \to \mathbf{Kl}(K)$ acting as the identity on objects and taking $f : L \otimes A \to B$ to $f \circ (u \otimes \mathrm{id}_A) : K \otimes A \to B$. It readily follows that $\mathrm{id}_K^{\star} = \mathrm{id}_{\mathbf{Kl}(K)}$ and that $(u \circ v)^{\star} = v^{\star} \circ u^{\star}$. In other words, we have a functor $\mathsf{Cl}(\mathbb{C}) : \mathbf{Comon}(\mathbb{C})^{\mathrm{op}} \to \mathbf{Cat}$. Its Grothendieck completion $\int \mathsf{Cl}(\mathbb{C})$ (see e.g. [Jac01, Chap. 1]) is the category whose objects are pairs $(K, A)$ of an object $K$ of $\mathbf{Comon}(\mathbb{C})$ and an object $A$ of $\mathbb{C}$, and whose morphisms from $(K, A)$ to $(L, B)$ are pairs $(u, f)$ where $u : K \to L$ is a comonoid morphism and $f : K \otimes A \to B$. The category $\int \mathsf{Cl}(\mathbb{C})$ is fibred over $\mathbf{Comon}(\mathbb{C})$ *via* the first projection, that we denote

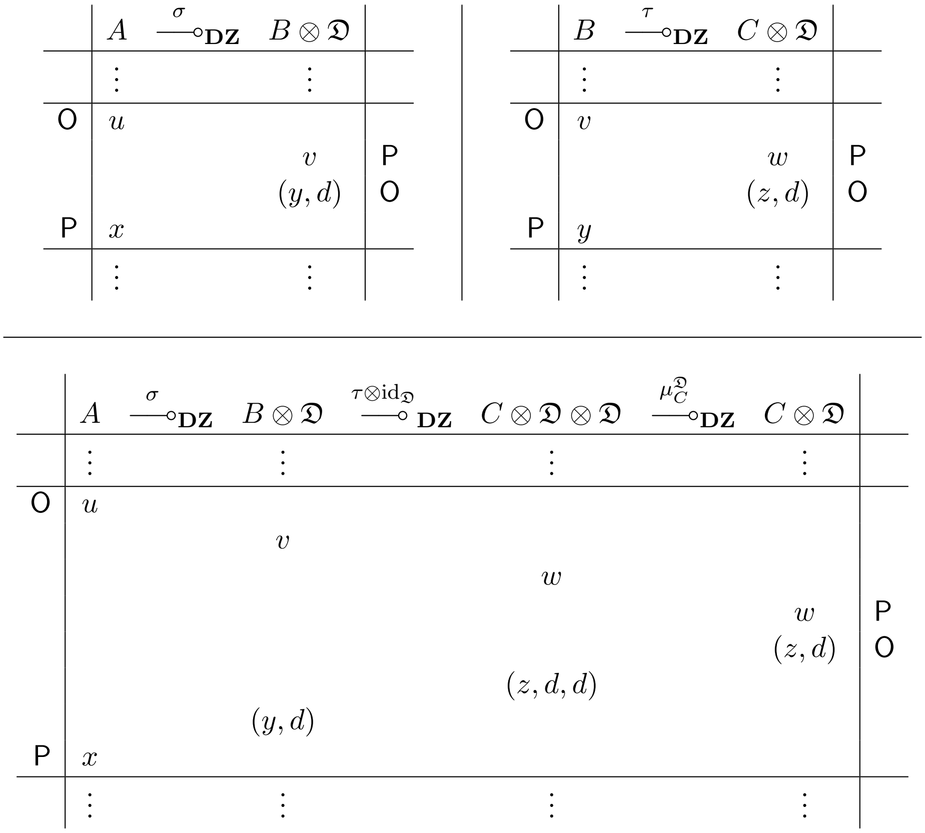
$$\mathsf{s}_{\mathsf{Cl}}(\mathbb{C}) \quad : \quad \int \mathsf{Cl}(\mathbb{C}) \quad \longrightarrow \quad \mathbf{Comon}(\mathbb{C})$$

Returning to our case, recall that $\mathsf{DialZ}(\Sigma) = \mathbf{Kl}(\Phi)$ where $\Phi$ is the distributive law of $\Sigma \otimes (-)$ over $(-) \otimes \mathfrak{D}$. The category $\mathsf{DialZ}(\Sigma)$ can alternatively be obtained as a Kleiseli category of

|  | $\Sigma \otimes A \otimes \mathfrak{D}$ | $\overset{\sigma \otimes \mathrm{id}_{\mathfrak{D}}}{\multimap}_{\mathbf{DZ}}$ | $B \otimes \mathfrak{D} \otimes \mathfrak{D}$ | $\overset{\mathrm{id}_B \otimes m_{\mathfrak{D}}}{\multimap}_{\mathbf{DZ}}$ | $B \otimes \mathfrak{D}$ |  |
|---|---|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  | $\vdots$ |  |
| O | $(\mathsf{a}, u)$ |  |  |  |  |  |
|  |  |  | $v$ |  |  |  |
|  |  |  |  |  | $v$ | P |
|  |  |  |  |  | $(y, d)$ | O |
|  |  |  | $(y, d, d)$ |  |  |  |
| P | $(x, d)$ |  |  |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  | $\vdots$ |  |

|  | $\Sigma \otimes A \otimes \mathfrak{D}$ | $\overset{d_\Sigma \otimes \mathrm{id}_{A \otimes \mathfrak{D}}}{\multimap}_{\mathbf{DZ}}$ | $\Sigma \otimes \Sigma \otimes A \otimes \mathfrak{D}$ | $\overset{\mathrm{id}_\Sigma \otimes \dot{\sigma}}{\multimap}_{\mathbf{DZ}}$ | $\Sigma \otimes B \otimes \mathfrak{D}$ |  |
|---|---|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  | $\vdots$ |  |
| O | $(\mathsf{a}, u)$ |  |  |  |  |  |
|  |  |  | $(\mathsf{a}, \mathsf{a}, u)$ |  |  |  |
|  |  |  |  |  | $(\mathsf{a}, v)$ | P |
|  |  |  |  |  | $(y, d)$ | O |
|  |  |  | $(x, d)$ |  |  |  |
| P | $(x, d)$ |  |  |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  | $\vdots$ |  |

Figure 16: Decomposition of $\sigma^\uparrow$



$A \xrightarrow{\sigma}_{\mathbf{DZ}} B \otimes \mathfrak{D}$
O $u$
$v$
$(y,d)$
P O
P $x$
$B \xrightarrow{\tau}_{\mathbf{DZ}} C \otimes \mathfrak{D}$
O $v$
$w$
$(z,d)$
P O
P $y$
$A \xrightarrow{\sigma}_{\mathbf{DZ}} B \otimes \mathfrak{D} \xrightarrow{\tau\otimes\mathrm{id}_{\mathfrak{D}}}_{\mathbf{DZ}} C \otimes \mathfrak{D} \otimes \mathfrak{D} \xrightarrow{\mu^{\mathfrak{D}}_C}_{\mathbf{DZ}} C \otimes \mathfrak{D}$
O $u$
$v$
$w$
$w$
$(z,d)$
P O
$(z,d,d)$
$(y,d)$
P $x$

Figure 17: Composition in $\mathbf{DZ}_{\mathfrak{D}} = \mathbf{Kl}(\mathfrak{D})$

indexing with comonoids over a symmetric monoidal category. Let $\mathbf{DZ}_{\mathfrak{D}}$ be the Kleisli category of indexing with the $\mathbf{DZ}$-monoid $\mathfrak{D}$. The objects of $\mathbf{DZ}_{\mathfrak{D}}$ are full positive games, and maps from $A$ to $B$ are $\mathbf{DZ}$-maps from $A$ to $B \otimes \mathfrak{D}$.

Let us spell out composition in $\mathbf{DZ}_{\mathfrak{D}}$. First recall that for a monad $(T, \mu, \eta)$ on a category $\mathbb{C}$, composition in the Kleisli category $\mathbf{Kl}(T)$ is given by

$$g \circ_{\mathbf{Kl}(T)} f \quad := \quad A \xrightarrow{f} TB \xrightarrow{Tg} TTC \xrightarrow{\mu_C} TC$$

for $f : A \to TB$ and $g : B \to TC$. In the case of $\mathbf{DZ}_{\mathfrak{D}}$-morphisms $\sigma : A \multimap_{\mathbf{DZ}} B \otimes \mathfrak{D}$ and $\tau : B \multimap_{\mathbf{DZ}} C \otimes \mathfrak{D}$ (where $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$) as depicted on Fig. 17 (top), their composite $\tau \circ_{\mathbf{DZ}_{\mathfrak{D}}} \sigma$ is depicted (modulo associativity) on Fig. 17 (bottom).

Since $\mathbf{DZ}_{\mathfrak{D}}$ is the Kleisli category of a lax symmetric monoidal monad on $\mathbf{DZ}$, it is symmetric monoidal with structure induced by that of $\mathbf{DZ}$ (see e.g. [Mel09]).

**Proposition 4.5.** *(a) Consider a monoid $M$ in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$. The Kleisli category $\mathbf{Kl}(M)$ is symmetric monoidal with $A \otimes_{\mathbf{Kl}(M)} B := A \otimes B$ on objects and unit $\mathbf{I}$.*

*Moreover, each comonoid $(K, d, e)$ in $\mathbb{C}$ induces a comonoid $(K, \eta^M_{K\otimes K} \circ d, \eta^M_{\mathbf{I}} \circ e)$ in $\mathbf{Kl}(M)$.*

*(b) In the case of $\mathbf{DZ}_{\mathfrak{D}} = \mathbf{Kl}(\mathfrak{D})$, the action of $\otimes_{\mathbf{DZ}_{\mathfrak{D}}}$ on maps $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i \otimes \mathfrak{D}$ (for $i = 1, 2$, $A_i = (U_i, X_i)$ and $B_i = (V_i, Y_i)$) is depicted on Fig. 18. If the $\sigma_i$ are represented*

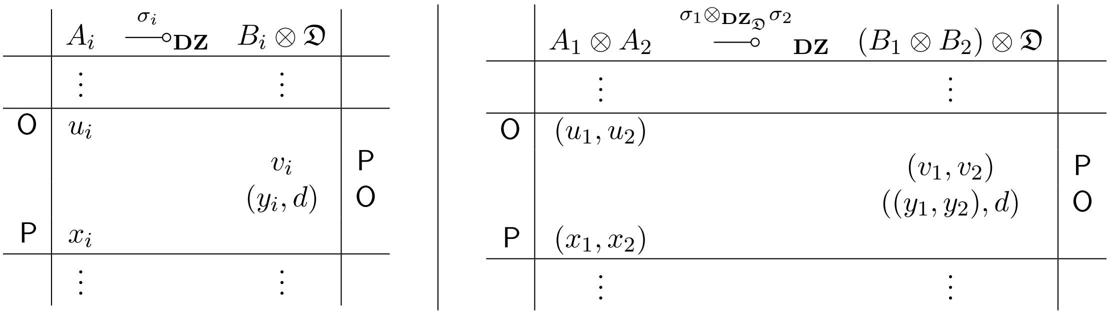

$A_i \overset{\sigma_i}{\multimap}_{\mathbf{DZ}} B_i \otimes \mathfrak{D}$
O
$u_i$
$v_i$
P
$(y_i, d)$
O
P
$x_i$
$A_1 \otimes A_2 \overset{\sigma_1 \otimes_{\mathbf{DZ}_{\mathfrak{D}}} \sigma_2}{\multimap}_{\mathbf{DZ}} (B_1 \otimes B_2) \otimes \mathfrak{D}$
O
$(u_1, u_2)$
$(v_1, v_2)$
P
$((y_1, y_2), d)$
O
P
$(x_1, x_2)$

Figure 18: Action of $\otimes_{\mathbf{DZ}_{\mathfrak{D}}}$ on $\sigma_i : A_i \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B_i$

*via Prop. 3.12 by pairs of functions $(f_i, F_i)$ where*

$$\begin{array}{lcl} f_i & : & \bigcup_{n\in\mathbb{N}} \left(U_i^{n+1} \times Y_i^n \times \mathfrak{D}^n\right) & \longrightarrow & V_i \\ F_i & : & \bigcup_{n\in\mathbb{N}} \left(U_i^{n+1} \times Y_i^{n+1} \times \mathfrak{D}^{n+1}\right) & \longrightarrow & X_i \end{array}$$

*then $\sigma_1 \otimes_{\mathbf{DZ}_{\mathfrak{D}}} \sigma_2$ is represented by $(h, H)$ where*

$$\begin{array}{lcl} h & : & \bigcup_{n\in\mathbb{N}} \left((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^n \times \mathfrak{D}^n\right) & \longrightarrow & V_1 \times V_2 \\ H & : & \bigcup_{n\in\mathbb{N}} \left((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n+1} \times \mathfrak{D}^{n+1}\right) & \longrightarrow & X_1 \times X_2 \end{array}$$

*are defined as*

$$\begin{array}{rcl} h((\overline{u}_1, \overline{u}_2)\,,\,(\overline{y}_1, \overline{y}_2)\,,\,p) & := & (f_1(\overline{u}_1, \overline{y}_1, p)\,,\,f_2(\overline{u}_2, \overline{y}_2, p)) \\ H((\overline{u}_1, \overline{u}_2)\,,\,(\overline{y}_1, \overline{y}_2)\,,\,p) & := & (F_1(\overline{u}_1, \overline{y}_1, p)\,,\,F_2(\overline{u}_2, \overline{y}_2, p)) \end{array}$$

*Moreover, the $\mathbf{DZ}_{\mathfrak{D}}$-structure maps $\tilde{d}_\Sigma$ and $\tilde{e}_\Sigma$ of the comonoid induced by $\Sigma$ can be depicted as in Fig. 19.*

It follows from Prop. 4.5 and the fact that $\Phi$ is a distributive law, that each category $\mathsf{DialZ}(\Sigma)$ is the Kleisli category of indexing with $\Sigma$ in $\mathbf{DZ}_{\mathfrak{D}}$. We can therefore index $\mathsf{DialZ}(-)$ with the comonoids of $\mathbf{DZ}_{\mathfrak{D}}$. We will actually index $\mathsf{DialZ}(-)$ over the base category $\mathbf{T}$ (of Def. 2.1), which is isomorphic to a subcategory of $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$. First, it directly follows from the representation of $\mathbf{DZ}$-maps given by Prop. 3.12 that $\mathbf{T}$-strategies from $\Sigma$ to $\Gamma$ in the sense of Def. 2.1 correspond exactly to total zig-zag strategies from $(\Sigma, \mathbf{1})$ to $(\Gamma, \mathfrak{D})$, that is to $\mathbf{DZ}_{\mathfrak{D}}$-maps from $\Sigma$ to $\Gamma$. Hence $\mathbf{T}$ is isomorphic to a subcategory of $\mathbf{DZ}_{\mathfrak{D}}$. Second, $\mathbf{T}$-maps induce comonoid maps.

**Proposition 4.6.** *The category $\mathbf{T}$ embeds in $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$ via the functor $\mathsf{E}_{\mathbf{T}}$ which takes an alphabet $\Sigma$ to the comonoid $(\Sigma, \tilde{d}_\Sigma, \tilde{e}_\Sigma)$ and a morphism $M : \mathbf{T}[\Gamma, \Sigma]$ to the $\mathbf{DZ}_{\mathfrak{D}}$-morphism*

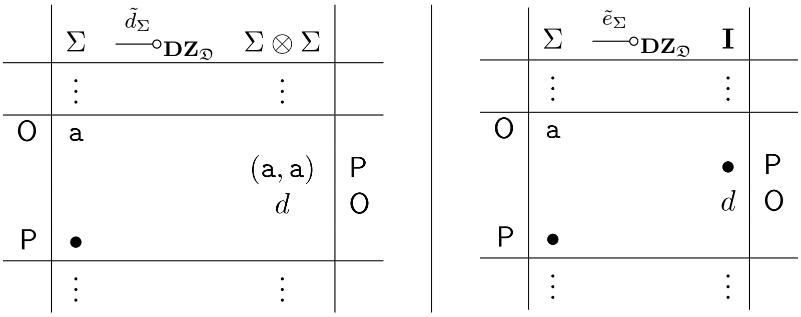
$$\tilde{M} \quad := \quad \jmath_\Sigma \circ M \quad : \quad (\Gamma, \mathbf{1}) \quad \multimap_{\mathbf{DZ}} \quad (\Sigma, \mathbf{1}) \otimes (\mathbf{1}, \mathfrak{D})$$

*induced by the $\mathbf{DZ}$-iso $\jmath_\Sigma : (\Sigma, \mathfrak{D}) \overset{\simeq}{\multimap}_{\mathbf{DZ}} (\Sigma, \mathbf{1}) \otimes (\mathbf{1}, \mathfrak{D})$.*

A detailed proof of Prop. 4.6 is given in [Rib17]. We thus get an indexed category

$$\mathsf{DialZ} \quad := \quad \mathsf{Cl}(\mathbf{DZ}_{\mathfrak{D}}) \circ \mathsf{E}_{\mathbf{T}} \quad : \quad \mathbf{T}^{\mathrm{op}} \quad \longrightarrow \quad \mathbf{Cat}$$

We already mentioned the well-known fact that the symmetric monoidal structure of a category induces a Cartesian structure on its category of comonoids (see e.g. [Mel09, Cor. 18, §6.5]). By Prop. 4.6, this gives a Cartesian structure on $\mathbf{T}$.



$\Sigma \overset{\tilde{d}_\Sigma}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}} \Sigma \otimes \Sigma$
O
a
(a, a)
P
$d$
O
P
●
$\Sigma \overset{\tilde{e}_\Sigma}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}} \mathbf{I}$
O
a
●
P
$d$
O
P
●

Figure 19: Structure maps for the comonoid $\Sigma = (\Sigma, \mathbf{1})$

**Corollary 4.7.** *The category* $\mathbf{T}$ *is Cartesian, with on objects the Cartesian product of alphabets, and with unit* $\mathbf{1}$.

## 4.4 The Fibred Category DialAut

We thus have a category DialZ indexed over $\mathbf{T}$, and whose fibre over $\Sigma$ is the category $\mathsf{DialZ}(\Sigma)$, which is isomorphic to $\mathbf{SAG}_\Sigma$. We will now define a fibration $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ of uniform substituted acceptance games, which essentially extends DialZ with winning (and acceptance). The fibration $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ is obtained by Grothendieck completion of an indexed category $(-)^\star : \mathbf{T}^{\mathrm{op}} \to \mathbf{Cat}$, which takes an alphabet $\Sigma$ to a category $\mathsf{DialAut}_\Sigma$ equivalent to $\mathbf{SAG}^{\mathrm{W}}_\Sigma$. The action of $(-)^\star$ on $\mathbf{T}$-maps is based on the indexed category DialZ.

**Definition 4.8** (The Category $\mathsf{DialAut}_\Sigma$)**.** *For each alphabet* $\Sigma$*, the category* $\mathsf{DialAut}_\Sigma$ *has the same objects as* $\mathbf{SAG}^{\mathrm{W}}_\Sigma$*, namely tuples* $(U, X, \mathcal{W}_A)$ *where* $U$ *and* $X$ *are non-empty sets and where* $\mathcal{W}_A \subseteq ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$.

*The* $\mathsf{DialAut}_\Sigma$*-morphisms from* $(U, X, \mathcal{W}_A)$ *to* $(V, Y, \mathcal{W}_B)$ *are total zig-zag strategies* $\sigma : \Sigma \otimes (U, X) \multimap_{\mathbf{DZ}} (V, Y) \otimes \mathfrak{D}$*, that is* $\mathsf{DialZ}(\Sigma)$*-morphisms from* $(U, X)$ *to* $(V, Y)$*, whose lift* $\iota(\sigma^\uparrow)$ *are* $\mathbf{SAG}^{\mathrm{W}}_\Sigma$*-strategies from* $(U, X, \mathcal{W}_A)$ *to* $(V, Y, \mathcal{W}_B)$*, that is winning synchronous strategies on* $(\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A) \multimap (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$.

Composition and identities of $\mathsf{DialAut}_\Sigma$ are induced by composition and identities of $\mathsf{DialZ}(\Sigma)$ (using the functoriality of $(-)^\uparrow$ for winning). Note that $\mathsf{DialAut}_\Sigma$ is isomorphic to $\mathbf{SAG}^{\mathrm{W}}_\Sigma$. In particular, given a uniform automaton $\mathcal{A} : \Delta$ and $M \in \mathbf{T}[\Sigma, \Delta]$, we still write $\Sigma \vdash \mathcal{A}(M)$ for the $\mathsf{DialAut}_\Sigma$-object induced by the uniform substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$ of §3.5.

We now turn to substitution and indexing. Morphisms $L \in \mathbf{T}[\Gamma, \Sigma]$ induce functors

$$L^\star \quad : \quad \mathsf{DialAut}_\Sigma \quad \longrightarrow \quad \mathsf{DialAut}_\Gamma$$

defined as follows. Given a $\mathsf{DialAut}_\Sigma$-object $A = (U, X, \mathcal{W}_A)$, we let $L^\star(A)$ be the $\mathsf{DialAut}_\Gamma$-object $(U, X, L^\star(\mathcal{W}_A))$, where

$$((\mathsf{b}_k, u_k) \cdot (x_k, d_k))_k \in L^\star(\mathcal{W}_A) \quad \text{iff} \quad ((L(\mathsf{b}_0. \cdots .\mathsf{b}_k,\, d_0. \cdots .d_{k-1}), u_k) \cdot (x_k, d_k))_k \in \mathcal{W}_A$$

When the $\mathsf{DialAut}_\Sigma$-object $A$ is induced by a uniform substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$, we have the expected result that $L^\star(A)$ is induced by the uniform substituted acceptance game $\Gamma \vdash \mathcal{A}(M \circ L)$ (see §2.2).

**Lemma 4.9.** *Given a uniform substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$ and $L \in \mathbf{T}[\Gamma, \Sigma]$, we have*

$$L^\star(\Sigma \vdash \mathcal{A}(M)) \quad = \quad \Gamma \vdash \mathcal{A}(M \circ L)$$

A detailed proof of Lem. 4.9 is given in [Rib17]. The action of $L^\star$ on maps is induced by $\mathsf{Cl}(\mathbf{DZ}_{\mathfrak{D}})(L) : \mathsf{DialZ}(\Sigma) \to \mathsf{DialZ}(\Gamma)$, so that for $\sigma \in \mathsf{DialAut}_\Sigma[A, B]$, we let

$$L^\star(\sigma) \quad := \quad \sigma \circ (L \otimes \mathrm{id}_A)$$

(where $\circ$, $\otimes$ and $\mathrm{id}_A$ are taken in $\mathbf{DZ}_{\mathfrak{D}}$).

**Proposition 4.10.** *Let $L \in \mathbf{T}[\Gamma, \Sigma]$ and consider $\mathsf{DialAut}_\Sigma$-objects $A = (U, X, \mathcal{W}_A)$ and $B = (V, Y, \mathcal{W}_B)$. Given a total strategy $\sigma : \Sigma \otimes (U, X) \multimap_{\mathbf{DZ}} (V, Y) \otimes \mathfrak{D}$, if the strategy $\iota(\sigma^\uparrow)$ is winning on*

$$(\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A) \quad \multimap \quad (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$$

*then the strategy $\iota(L^\star(\sigma)^\uparrow)$ is winning on*

$$(\Gamma \times U, X \times \mathfrak{D}, L^\star(\mathcal{W}_A)) \quad \multimap \quad (\Gamma \times V, Y \times \mathfrak{D}, L^\star(\mathcal{W}_B))$$

A detailed proof of Prop. 4.10 is given in [Rib17]. We thus obtain an indexed category $(-)^\star : \mathbf{T}^{\mathrm{op}} \to \mathbf{Cat}$ since $(-)^\star$ is itself functorial. We let $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ be its Grothendieck completion.

**Definition 4.11** (The Fibred Category $\mathsf{DialAut}$)**.** *The objects of $\mathsf{DialAut}$ are pairs $(\Sigma, A)$ where $A$ is an object of $\mathsf{DialAut}_\Sigma$. Maps from $(\Sigma, A)$ to $(\Gamma, B)$ are pairs $(L, \sigma)$ of a $\mathbf{T}$-map $L : \Sigma \to \Gamma$ and a $\mathsf{DialAut}_\Sigma$-map $\sigma$ from $A$ to $L^\star(B)$.*

*The fibration*

$$\mathsf{da} \quad : \quad \mathsf{DialAut} \quad \longrightarrow \quad \mathbf{T}$$

*is the first projection, so that $\mathsf{da}(\Sigma, A) := A$ and $\mathsf{da}(L, \sigma) := L$.*

## 4.5 Substitution and Language Inclusion

We now check that $\mathsf{DialAut}_\Sigma$ (and $\mathbf{SAG}^{\mathrm{W}}_\Sigma$) is correct w.r.t. language inclusion. First, it follows from Lem. 4.9 that given

$$\sigma \quad : \quad \mathcal{A}(M) \quad \multimap \quad \mathcal{B}(N)$$

and $L \in \mathbf{T}[\Gamma, \Sigma]$, we have

$$L^\star(\sigma) \quad : \quad \mathcal{A}(M \circ L) \quad \multimap \quad \mathcal{B}(N \circ L)$$

Hence, $\mathsf{DialAut}$ interprets all instances of the (Subst) rule (14) of the form

$$\frac{M \;;\; \mathcal{A} \vdash \mathcal{B}}{M \circ L \;;\; \mathcal{A} \vdash \mathcal{B}} \quad \text{(where } M \in \mathbf{T}[\Sigma, \Delta] \text{ and } L \in \mathbf{T}[\Gamma, \Sigma]\text{)}$$

In particular, given $\mathcal{A}, \mathcal{B} : \Sigma$, for all $\Sigma$-labeled tree $T$ (and using the notation of §2.2.(b)) we have

$$\frac{\Sigma \;;\; \mathcal{A} \vdash \mathcal{B}}{\dot{T} \;;\; \mathcal{A} \vdash \mathcal{B}}$$

Now, given $\sigma : \mathcal{A} \multimap \mathcal{B}$, if $T \in \mathcal{L}(\mathcal{A})$, then there is some $\tau : \mathbf{I_1} \multimap \mathcal{A}(T)$, so that $\dot{T}^\star(\sigma) \circ \tau : \mathbf{I_1} \multimap \mathcal{B}(T)$ and $T \in \mathcal{L}(\mathcal{B})$. In other words, $\sigma : \mathcal{A} \multimap \mathcal{B}$ and $T$ induce a function

$$\tau \;:\; \mathbf{I} \multimap \mathcal{A}(T) \qquad \longmapsto \qquad T^\star(\sigma) \circ \tau \;:\; \mathbf{I} \multimap \mathcal{B}(T)$$

and we have shown:

**Proposition 4.12.** *If $\mathsf{P}$ has a winning strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, then $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$.*

# 5 Symmetric Monoidal Closed Structure

We show here that the notion of linear implication of uniform automata presented in Def. 3.3 indeed corresponds to a (fibrewise, symmetric) monoidal closed structure in $\mathsf{DialAut}$. This monoidal closed structure is actually induced from a monoidal closed structure in the category $\mathbf{DZ}$ of full positive games and total zig-zag strategies.

We first discuss the closed structure of $\mathbf{DZ}$ (§5.1). We then show how the symmetric monoidal closed structure of $\mathbf{DZ}$ lifts to $\mathsf{DialAut}$ uniform tree automata (§5.2). This provides a realizability interpretation of a propositional linear (multiplicative) deduction system (§5.3). We finally show how the closed structure gives a (functorial) notion of linear complement (§5.4).

Recall from e.g. [Mel09] that a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$ is *closed* if for every object $A$, the functor $A \otimes (-)$ has a right adjoint $(-)^A$. According to [ML98, Thm. IV.1.2] it is sufficient to show that for every object $C$ there is an object $C^A$ and map

$$\mathsf{eval}_C \quad : \quad A \otimes C^A \quad \longrightarrow \quad C$$

such that for every $f : A \otimes B \to C$ there is a unique $\mathbf{\Lambda}(f) : B \to C^A$ such that

$$\begin{tikzcd}
A \otimes C^A \arrow[r, "\mathsf{eval}_C"] & C \\
A \otimes B \arrow[u, dotted, "\mathrm{id}_A \otimes \mathbf{\Lambda}(f)"] \arrow[ur, "f"'] &
\end{tikzcd}$$

## 5.1 The Symmetric Monoidal Closure of $\mathbf{DZ}$

The monoidal closed structure of $\mathbf{DZ}$ follows the pattern of §3.1 and Def. 3.3, and can actually be read off from the representation of $\mathbf{DZ}$-strategies as pairs of functions in Prop. 3.12.

Let us see how to define a linear exponent full positive game $B^A = (A \multimap_{\mathbf{DZ}} B)$ from full positive games $A = (U, X)$ and $B = (V, Y)$, such that a strategy $\sigma : A \multimap_{\mathbf{DZ}} B$ induces (modulo $A \simeq A \otimes \mathbf{I}$) a strategy $\mathbf{\Lambda}(\sigma) : \mathbf{I} \multimap_{\mathbf{DZ}} (A \multimap_{\mathbf{DZ}} B)$. Assume that $\sigma$ plays as in Fig. 10. From each play $s \in \sigma$, the responses $v \in V$ of $\sigma$ to $\mathsf{O}$-moves $u \in U$ define a function

$$f_s \quad : \quad U \quad \longrightarrow \quad V$$

and the responses $x \in X$ of $\sigma$ to further $\mathsf{O}$-moves $y \in Y$ define a function

$$F_s \quad : \quad U \times Y \quad \longrightarrow \quad X$$

In the context of Prop. 3.12, recalling that $\mathrm{HS}(s) = (\partial^{-1}(\overline{u}, \overline{x}), \partial^{-1}(\overline{v}, \overline{y}))$ where $\overline{x}$ and $\overline{v}$ are completely determined by $\sigma$ from $\overline{u}$ and $\overline{y}$, this amounts to describe $\sigma$ by a pair of maps

$$\begin{array}{lclcl} f & : & \bigcup_{n \in \mathbb{N}} (U^n \times Y^n) & \longrightarrow & (U \longrightarrow V) \\ F & : & \bigcup_{n \in \mathbb{N}} (U^n \times Y^n) & \longrightarrow & (U \times Y \longrightarrow X) \end{array} \tag{44}$$

**Proposition 5.1.** *The category* $\mathbf{DZ}$ *is symmetric monoidal closed. The linear exponent of* $A = (U, X)$ *and* $B = (V, Y)$ *is* $A \multimap_{\mathbf{DZ}} B \ := \ (V^U \times X^{U \times Y}, U \times Y)$.

A detailed proof of Prop. 5.1 is given in [Rib17]. The monoidal closed structure of $\mathbf{DZ}$ departs from traditional game semantics since the natural isomorphism $A \otimes B \multimap_{\mathbf{DZ}} C \simeq B \multimap_{\mathbf{DZ}} (A \multimap_{\mathbf{DZ}} C)$ relates only strategies, but not *plays*.

|  | $\Sigma \otimes A_i$ | $\overset{\sigma_i}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}}$ | $B_i$ |
|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |
| O | $(\mathsf{a}, u_i)$ |  |  |
| P |  |  | $v_i$ |
| O |  |  | $(y_i, d)$ |
| P | $x_i$ |  |  |
|  | $\vdots$ |  | $\vdots$ |

|  | $\Sigma \otimes (A_1 \otimes A_2)$ | $\overset{\tau}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}}$ | $B_1 \otimes B_2$ |  |
|---|---|---|---|---|
|  | $\vdots$ |  | $\vdots$ |  |
| O | $(\mathsf{a}, (u_1, u_2))$ |  |  |  |
|  |  |  | $(v_1, v_2)$ | P |
|  |  |  | $((y_1, y_2), d)$ | O |
| P | $(x_1, x_2)$ |  |  |  |
|  | $\vdots$ |  | $\vdots$ |  |

Figure 20: Action of $\otimes_{\mathsf{DialZ}(\Sigma)}$ on $\sigma_i : A_i \multimap_{\mathsf{DialZ}(\Sigma)} B_i$, where $\tau := \sigma_1 \otimes_{\mathsf{DialZ}(\Sigma)} \sigma_2$

## 5.2 The Symmetric Monoidal Closed Structure of DialAut and Tree Automata

The symmetric monoidal closed structure of DialAut and of tree automata is induced by the symmetric monoidal closed structure of DialZ, which is itself lifted from **DZ**.

**5.2.1 The Symmetric Monoidal Structure of DialZ.** We have seen in Prop. 4.5 that the symmetric monoidal structure of **DZ** lifts *via* monoid indexing to give a symmetric monoidal structure to $\mathbf{DZ}_{\mathfrak{D}}$. The same scheme actually applies to DialZ, which is symmetric monoidal with structure induced by comonoid indexing in $\mathbf{DZ}_{\mathfrak{D}}$.

**Proposition 5.2.** *(a) Consider a comonoid $K$ in a symmetric monoidal category $\mathbb{C}$. The Kleisli category $\mathbf{Kl}(K)$ is symmetric monoidal with $A \otimes_{\mathbf{Kl}(K)} B := A \otimes B$ on objects and unit $\mathbf{I}$.*

*(b) In the case of $\mathsf{DialZ}(\Sigma) = \mathbf{Kl}(\Sigma)$, the action of the tensor $\otimes_{\mathsf{DialZ}(\Sigma)}$ on strategies $\sigma_i : \Sigma \otimes A_i \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B_i$ (for $i = 1, 2$, $A_i = (U_i, X_i)$ and $B_i = (V_i, Y_i)$) is depicted on Fig. 20. If the $\sigma_i$ are represented via Prop. 3.12 by pairs of functions $(f_i, F_i)$ where*

$$\begin{array}{lclcl} f_i & : & \bigcup_{n \in \mathbb{N}} \left( \Sigma^{n+1} \times U_i^{n+1} \times Y_i^{n} \times \mathfrak{D}^{n} \right) & \longrightarrow & V_i \\ F_i & : & \bigcup_{n \in \mathbb{N}} \left( \Sigma^{n+1} \times U_i^{n+1} \times Y_i^{n+1} \times \mathfrak{D}^{n+1} \right) & \longrightarrow & X_i \end{array}$$

*then $\sigma_1 \otimes_{\mathsf{DialZ}(\Sigma)} \sigma_2$ is represented by $(h, H)$ where*

$$\begin{array}{lclcl} h & : & \bigcup_{n \in \mathbb{N}} \left( \Sigma^{n+1} \times (U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n} \times \mathfrak{D}^{n} \right) & \longrightarrow & V_1 \times V_2 \\ H & : & \bigcup_{n \in \mathbb{N}} \left( \Sigma^{n+1} \times (U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n+1} \times \mathfrak{D}^{n+1} \right) & \longrightarrow & X_1 \times X_2 \end{array}$$

*are defined as*

$$\begin{array}{rcl} h(\overline{\mathsf{a}}\,,\,(\overline{u}_1, \overline{u}_2)\,,\,(\overline{y}_1, \overline{y}_2)\,,\,p) & := & (f_1(\overline{\mathsf{a}}, \overline{u}_1, \overline{y}_1, p)\,,\,f_2(\overline{\mathsf{a}}, \overline{u}_2, \overline{y}_2, p)) \\ H(\overline{\mathsf{a}}\,,\,(\overline{u}_1, \overline{u}_2)\,,\,(\overline{y}_1, \overline{y}_2)\,,\,p) & := & (F_1(\overline{\mathsf{a}}, \overline{u}_1, \overline{y}_1, p)\,,\,F_2(\overline{\mathsf{a}}, \overline{u}_2, \overline{y}_2, p)) \end{array}$$

**5.2.2 The Symmetric Monoidal Closure of $\mathbf{DZ}_{\mathfrak{D}}$ and DialZ.** The monoidal closed structure of **DZ** lifts to $\mathbf{DZ}_{\mathfrak{D}}$ and to the fibers of DialZ. In the case of $\mathbf{DZ}_{\mathfrak{D}}$, since

$$\mathbf{DZ}_{\mathfrak{D}}[A \otimes B\ ,\ C] \quad = \quad \mathbf{DZ}[A \otimes B\ ,\ C \otimes \mathfrak{D}] \quad \simeq \quad \mathbf{DZ}[A\ ,\ (B \multimap_{\mathbf{DZ}} C \otimes \mathfrak{D})]$$

we should have $(A \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B) \otimes \mathfrak{D} \simeq (A \multimap_{\mathbf{DZ}} B \otimes \mathfrak{D})$. Given $A = (U, X)$ and $B = (V, Y)$ this leads to $(A \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B) = (W, Z)$ with

$$(W, Z \times \mathfrak{D}) \quad \simeq \quad (V^U \times X^{U \times Y \times \mathfrak{D}} \ , \ U \times Y \times \mathfrak{D})$$

We therefore let

$$(U, X) \multimap_{\mathbf{DZ}_{\mathfrak{D}}} (V, Y) \quad := \quad (V^U \times X^{U \times Y \times \mathfrak{D}} \ , \ U \times Y)$$

The closed structure of $\mathbf{DZ}_{\mathfrak{D}}$ directly lifts to $\mathsf{DialZ}(\Sigma)$ since

$$\mathsf{DialZ}(\Sigma)[A \otimes B \ , \ C] \quad = \quad \mathbf{DZ}_{\mathfrak{D}}[\Sigma \otimes (A \otimes B) \ , \ C] \quad \simeq \quad \mathbf{DZ}_{\mathfrak{D}}[\Sigma \otimes A \ , \ B \multimap_{\mathbf{DZ}_{\mathfrak{D}}} C]$$

**Proposition 5.3.** *$\mathbf{DZ}_{\mathfrak{D}}$ and $\mathsf{DialZ}(\Sigma)$ are symmetric monoidal closed.*

**5.2.3 The Symmetric Monoidal Closed Structure of DialAut.** The symmetric monoidal closed structure of DialZ gives the fibrewise symmetric monoidal closed structure of DialAut (in the sense of [Jac01, §1.8]). The unit over $\Sigma$ is $\mathbf{I}_{\Sigma} := (\mathbf{1}, \mathbf{1}, \mathbf{1}^{\omega})$. Given $\mathsf{DialAut}_{\Sigma}$-objects $A = (U, X, \mathcal{W}_A)$ and $B = (V, Y, \mathcal{W}_B)$, let

$$\begin{array}{rcl} A \otimes_{\mathsf{DA}} B & := & (U \times V \ , \ X \times Y \ , \ \mathcal{W}_A \sqcap \mathcal{W}_B) \\ A \multimap_{\mathsf{DA}} B & := & (V^U \times X^{U \times Y \times \mathfrak{D}} \ , \ U \times Y \ , \ \mathcal{W}_A \sqsupset \mathcal{W}_B) \end{array}$$

with

$$\varpi \in \mathcal{W}_A \sqcap \mathcal{W}_B \quad \text{iff} \quad \big(\varpi_{\restriction (\Sigma \times U) + (X \times \mathfrak{D})} \in \mathcal{W}_A \quad \text{and} \quad \varpi_{\restriction (\Sigma \times V) + (Y \times \mathfrak{D})} \in \mathcal{W}_B\big)$$

and

$$((\mathsf{a}_k \ , \ (f_k, F_k)) \cdot ((u_k, y_k) \ , \ d_k))_k \ \in \ \mathcal{W}_A \sqsupset \mathcal{W}_B \quad \text{iff} \quad (\alpha \in \mathcal{W}_A \quad \Longrightarrow \quad \beta \in \mathcal{W}_B)$$

where $\alpha$ and $\beta$ are obtained by pointwise application:

$$\begin{array}{rcl} \alpha & := & ((\mathsf{a}_k \ , \ u_k) \cdot (F_k(u_k, y_k, d_k) \ , \ d_k))_k \\ \beta & := & ((\mathsf{a}_k \ , \ f_k(u_k)) \cdot (y_k \ , \ d_k))_k \end{array}$$

In the notations $A \otimes_{\mathsf{DA}} B$ and $A \multimap_{\mathsf{DA}} B$ we omit the subscript DA and write $A \otimes B$ and $A \multimap B$ whenever possible.

**Proposition 5.4.** *The fibration* DialAut *is fibrewise monoidal closed.*

**5.2.4 The Symmetric Monoidal Closed Structure of Uniform Automata.** We now turn to uniform automata. The symmetric monoidal structure of $\mathsf{DialAut}_{\Sigma}$ gives a monoidal product on uniform automata. Moreover, linear implication automata in the sense of Def. 3.3 indeed correspond to the monoidal closure of $\mathsf{DialAut}_{\Sigma}$.

**Definition 5.5** (Monoidal Product of Uniform Automata)**.** *Assume $\mathcal{A}$ is as in (34) and*

$$\mathcal{B} \quad = \quad (Q_{\mathcal{B}} \ , \ q^{\imath}_{\mathcal{B}} \ , \ V \ , \ Y \ , \ \delta_{\mathcal{B}} \ , \ \Omega_{\mathcal{B}})$$

*so that*

$$\begin{array}{rcccccc} & \delta_{\mathcal{A}} & : & Q_{\mathcal{A}} \times \Sigma & \longrightarrow & U \times X & \longrightarrow & (\mathfrak{D} \longrightarrow Q_{\mathcal{A}}) \\ \textit{and} & \delta_{\mathcal{B}} & : & Q_{\mathcal{B}} \times \Sigma & \longrightarrow & V \times Y & \longrightarrow & (\mathfrak{D} \longrightarrow Q_{\mathcal{B}}) \end{array}$$

*We let $\mathcal{A} \otimes \mathcal{B}$ be the automaton over $\Sigma$ defined as*

$$\mathcal{A} \otimes \mathcal{B} \quad := \quad (Q_{\mathcal{A}} \times Q_{\mathcal{B}}\,,\, (q^{\imath}_{\mathcal{A}}, q^{\imath}_{\mathcal{B}})\,,\, U \times V\,,\, X \times Y\,,\, \delta_{\mathcal{A}\otimes\mathcal{B}}\,,\, \Omega_{\mathcal{A}\otimes\mathcal{B}})$$

*with*

$$\delta_{\mathcal{A}\otimes\mathcal{B}}((q_{\mathcal{A}}, q_{\mathcal{B}})\,,\, \mathtt{a}\,,\, (u, v)\,,\, (x, y)\,,\, d) \quad := \quad (q'_{\mathcal{A}}\,,\, q'_{\mathcal{B}})$$

*where*

$$q'_{\mathcal{A}} \;:=\; \delta_{\mathcal{A}}(q_{\mathcal{A}}\,,\, \mathtt{a}\,,\, u\,,\, x\,,\, d) \qquad \text{and} \qquad q'_{\mathcal{B}} \;:=\; \delta_{\mathcal{B}}(q_{\mathcal{B}}\,,\, \mathtt{a}\,,\, v\,,\, y\,,\, d)$$

*and with $((q_n, q'_n))_n \in \Omega_{\mathcal{A}\otimes\mathcal{B}}$ iff $((q_n)_n \in \Omega_{\mathcal{A}}$ and $(q'_n)_n \in \Omega_{\mathcal{B}})$. Note that $\Omega_{\mathcal{A}\otimes\mathcal{B}}$ is $\omega$-regular since $\Omega_{\mathcal{A}}$ and $\Omega_{\mathcal{B}}$ are both assumed to be $\omega$-regular.*

Note that $\mathcal{A} \otimes \mathcal{B}$ is non-deterministic (resp. universal, deterministic) if both $\mathcal{A}$ and $\mathcal{B}$ are non-deterministic (resp. universal, deterministic). Moreover, assuming $\mathcal{A}, \mathcal{B} : \Gamma$ and $M \in \mathbf{T}[\Sigma, \Gamma]$, we have, as $\mathsf{DialAut}_{\Sigma}$-objects,

$$\begin{array}{rrcl} & \Sigma \vdash (\mathcal{A}(M) \multimap_{\mathsf{DA}} \mathcal{B}(M)) & \simeq & \Sigma \vdash (\mathcal{A} \multimap \mathcal{B})(M) \\ \text{and} & \Sigma \vdash (\mathcal{A}(M) \otimes_{\mathsf{DA}} \mathcal{B}(M)) & \simeq & \Sigma \vdash (\mathcal{A} \otimes \mathcal{B})(M) \end{array}$$

### 5.3 Deduction, Adequacy and Correctness

Let us now return to the deduction system for automata outlined in §2. First, the monoidal structure of $\mathsf{DialAut}_{\Sigma}$ allows to interpret sequents of the form (13):

$$M \;;\; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B} \tag{13}$$

where $M \in \mathbf{T}[\Sigma, \Gamma]$ and $\mathcal{A}_1, \ldots, \mathcal{A}_n, \mathcal{B}$ are uniform automata over $\Gamma$. The sequent (13) is interpreted as the homset

$$\mathsf{DialAut}_{\Sigma}[\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M)\;,\; \mathcal{B}(M)]$$

Moreover, the monoidal closed structure implies that (13) can equivalently be interpreted as the set of winning $\mathsf{P}$-strategies in the uniform substituted acceptance game

$$\Sigma \vdash (\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})(M)$$

Second, the symmetric monoidal closed structure allows to interpret deduction rules for the multiplicative fragment of $\mathsf{ILL}$. Such rules were displayed in (22), (23), (25) and (31). We gather them on Fig. 21. Using the notations of §2.2, we write $\mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$ to denote the sequent $\mathrm{Id} \;;\; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$. Our model is sound w.r.t. this deduction system.

**Proposition 5.6** (Adequacy)**.** *If the sequent $M \;;\; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$ is derivable using the rules of Fig. 21, then there is a winning $\mathsf{P}$-strategy $\sigma$ in*

$$\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M) \quad \longrightarrow\!\!\circ \quad \mathcal{B}(M)$$

In particular, if $\mathcal{A} \vdash \mathcal{B}$ is derivable, then by combining Prop. 5.6 with Prop. 4.12, we obtain a strategy witnessing that $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$.

Note that the strategy $\sigma$ is obtained from the derivation $\mathscr{D}$ in a purely compositional way. Moreover, all the rules of Fig. 21 are compatible cut-elimination.

$$(\textsc{Exchange})\quad \dfrac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{B},\, \mathcal{A},\, \overline{\mathcal{C}} \vdash \mathcal{C}}$$

$$(\textsc{Cut})\quad \dfrac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\, \mathcal{A},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\, \overline{\mathcal{A}},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad\qquad \dfrac{}{M\ ;\ \mathcal{A} \vdash \mathcal{A}}\quad (\textsc{Axiom})$$

$$(\textsc{Left}\ \otimes)\quad \dfrac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{B},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A} \otimes \mathcal{B},\, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad\qquad \dfrac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}},\, \overline{\mathcal{B}} \vdash \mathcal{A} \otimes \mathcal{B}}\quad (\textsc{Right}\ \otimes)$$

$$(\textsc{Left}\ \mathbf{I})\quad \dfrac{M\ ;\ \overline{\mathcal{A}},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathbf{I},\, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad\qquad \dfrac{}{M\ ;\ \vdash \mathbf{I}}\quad (\textsc{Right}\ \mathbf{I})$$

$$(\textsc{Left}\ \multimap)\quad \dfrac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\, \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\, \overline{\mathcal{A}},\, \mathcal{A} \multimap \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad\qquad \dfrac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{B} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B} \multimap \mathcal{C}}\quad (\textsc{Right}\ \multimap)$$

Figure 21: Rules of the multiplicative fragment of $\mathsf{ILL}$ for uniform automata

**Remark 5.7** (On Cut-Elimination)**.** *It follows from the fact that we have monoidal closed categories (Prop. 5.4), that the interpretation of derivations as strategies for the rules of Fig. 21 is compatible with cut-elimination, in the sense that if a derivation $\mathscr{D}'$ is obtained from a derivation $\mathscr{D}$ by applying the proof transformation steps described in e.g. [Mel09, §3.3], then $\mathscr{D}$ and $\mathscr{D}'$ are interpreted by the same strategy. This in particular applies to the two derivations displayed in (32), §3.1.*

**Example 5.8.** *Proposition 5.6 yields a winning* $\mathsf{P}$*-strategy in*

$$\mathcal{B} \otimes \mathcal{B} \otimes (\mathcal{B} \multimap \mathcal{A}) \quad \multimap \quad \mathcal{A} \otimes \mathcal{B}$$

*obtained from the proof tree*

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{B} \vdash \mathcal{B}} \qquad \dfrac{}{\mathcal{A} \vdash \mathcal{A}}}{\mathcal{B}, \mathcal{B} \multimap \mathcal{A} \vdash \mathcal{A}} \qquad \dfrac{}{\mathcal{B} \vdash \mathcal{B}}}{\mathcal{B}, (\mathcal{B} \multimap \mathcal{A}), \mathcal{B} \vdash \mathcal{A} \otimes \mathcal{B}}}{\mathcal{B}, \mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}$$

Note that in Fig. 21 we omitted the *weakening* and *contraction* rules (24):

$$(\textsc{Weak})\quad \dfrac{M\ ;\ \overline{\mathcal{A}},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad\qquad \dfrac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{A},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \overline{\mathcal{B}} \vdash \mathcal{C}}\quad (\textsc{Contr})$$

Similarly as with usual automata, the contraction rule can be interpreted on *non-deterministic* uniform automata but not on general uniform automata. This rule amounts to provide winning $\mathsf{P}$-strategies for the game

$$\mathcal{A} \quad \multimap \quad \mathcal{A} \otimes \mathcal{A} \tag{45}$$

If $\mathcal{A}$ is non-deterministic (and with $\mathsf{P}$-moves $U$), then a winning $\mathsf{P}$-strategy in (45) simply takes an $\mathsf{O}$-move $u \in U$ in component $\mathcal{A}$ to the pair $(u, u) \in U \times U$ in component $\mathcal{A} \otimes \mathcal{A}$. Note that such strategy may not exist when $\mathcal{A}$ is a general uniform automaton, that is when it is equipped with a set of O-moves $X \not\simeq \mathbf{1}$, since O can play two different $(x, x') \in X \times X$ in the component $\mathcal{A} \otimes \mathcal{A}$, that P may not be able to merge into a single $x'' \in X$ in the left component $\mathcal{A}$.

On the other hand, the weakening rule, which asks for a winning P-strategy in

$$\mathcal{A} \quad \multimap \quad \mathbf{I}$$

can always be realized (since we required the set of P and O-moves to be always non-empty), but in a non-canonical way for general uniform automata. More generally, given $\mathcal{A}$ and $\mathcal{B}$ over the same input alphabet, there is always a winning P-strategy in

$$\mathcal{A} \otimes \mathcal{B} \quad \multimap \quad \mathcal{A} \tag{46}$$

Assuming $\mathcal{A}$ and $\mathcal{B}$ are as in Def. 5.5, such a strategy takes $(u, v) \in U \times V$ to $u \in U$ and takes $x \in X$ to $(x, y) \in X \times Y$, where $y$ is an arbitrarily chosen element of $Y$.

We shall come back on the connection between non-deterministic automata, the interpretation of the (Weak) and (Contr) rules and ILL in §7.

**Example 5.9.** *Proposition 5.6 actually holds for any extension of the deduction system of Fig. 21 with realizable rules, that is with rules*

$$\dfrac{}{\mathcal{A} \vdash \mathcal{B}}$$

*such that there is a winning* P*-strategy in $\mathcal{A} \multimap \mathcal{B}$. In particular:*

*(i) We can extend the system with the following generalization of (46):*

$$\dfrac{}{\mathcal{A}_1 \otimes \ldots \otimes \mathcal{A}_n \vdash \mathcal{A}_i}$$

*We thus get*

$$\dfrac{\dfrac{\dfrac{}{\mathcal{A} \vdash \mathcal{A}} \qquad \dfrac{}{\mathcal{B} \vdash \mathcal{B}}}{\mathcal{A}, \mathcal{B} \vdash \mathcal{A} \otimes B} \qquad \dfrac{}{\mathcal{A} \otimes \mathcal{B} \vdash \mathcal{A}}}{\dfrac{\mathcal{A}, \mathcal{B} \vdash \mathcal{A}}{\mathcal{A} \vdash \mathcal{B} \multimap \mathcal{A}}}$$

*So there is a winning* P*-strategy on*

$$\mathcal{A} \quad \multimap \quad (\mathcal{B} \multimap \mathcal{A})$$

*and by Prop. 4.12 we have*

$$\mathcal{L}(\mathcal{A}) \quad \subseteq \quad \mathcal{L}(\mathcal{B} \multimap \mathcal{A})$$

*(ii) For $\mathcal{B}$ non-deterministic, we can extend the system with the following generalizations of (45):*

$$\dfrac{}{\mathcal{B} \vdash \mathcal{B} \otimes \ldots \otimes \mathcal{B}}$$

*Continuing Ex. 5.8 with $\mathcal{B}$ non-deterministic, we thus have*

$$\dfrac{\dfrac{}{\mathcal{B} \vdash \mathcal{B} \otimes \mathcal{B}} \qquad \dfrac{\dfrac{\vdots}{\mathcal{B}, \mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}}{\mathcal{B} \otimes \mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}}{\mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}$$

The monoidal structure together with (46) imply that $\otimes$ indeed implements a conjunction on automata.

**Proposition 5.10.** *Given $\mathcal{A}, \mathcal{B} : \Sigma$, we have $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B})$.*

## 5.4 Falsity and Complementation

We have already seen in §1.3 and §2.10 that usual alternating automata are equipped with a complementation construction $(-)^{\perp\!\!\!\perp}$ linear in the number of states (see e.g. [MS87]). Using the monoidal closed structure, a similar construction can be done with uniform automata.

**Definition 5.11** (Falsity Uniform Automaton). *For each alphabet* $\Sigma$*, the* falsity uniform automaton $\bot$ *over* $\Sigma$ *is*

$$\bot \quad := \quad (\mathbb{B}\,,\, \mathbb{f}\,,\, \mathfrak{D}\,,\, \mathbf{1}\,,\, \delta_{\bot}\,,\, \Omega_{\bot})$$

*where* $\Omega_{\bot} := \mathbb{B}^* \cdot \mathbb{t}^\omega$ *and where*

$$\delta_{\bot}(\mathbb{b}\,,\, \_\,,\, d'\,,\, \bullet\,,\, d) \quad := \quad \begin{cases} \mathbb{f} & \textit{if } \mathbb{b} = \mathbb{f} \textit{ and } d = d' \\ \mathbb{t} & \textit{otherwise} \end{cases}$$

Note that in the game $\Sigma \vdash \bot$, O looses as soon as it does not play the same tree direction as proposed by P. On the other hand, $\bot$ accepts no tree since in an acceptance game $\bot(T)$, O can always play the same $d$ as P.

Consider a uniform automaton $\mathcal{A} : \Sigma$ with set of P-moves $U$ and set of O-moves $X$. The automaton $(\mathcal{A} \multimap \bot)$ is isomorphic (*via* $X^{U \times \mathfrak{D}} \simeq X^{U \times \mathbf{1} \times \mathfrak{D}}$) to the automaton $\mathcal{A}^{\bot}$ defined as

$$\mathcal{A}^{\bot} \quad := \quad (Q_{\mathcal{A}} \times \mathbb{B}\,,\, (q^{\imath}_{\mathcal{A}}, \mathbb{f})\,,\, \mathfrak{D}^U \times X^{U \times \mathfrak{D}}\,,\, U\,,\, \delta_{\mathcal{A}^{\bot}}\,,\, \Omega_{\mathcal{A}^{\bot}})$$

where

$$(q_k, \mathbb{b}_k)_k \in \Omega_{\mathcal{A}^{\bot}} \qquad \text{iff} \qquad ((q_k)_k \in \Omega_{\mathcal{A}} \quad \Longrightarrow \quad (\mathbb{b}_k)_k \in \mathbb{B}^* \cdot \mathbb{t}^\omega)$$

and where

$$\delta_{\mathcal{A}^{\bot}}(\mathtt{a}\,,\, (q_{\mathcal{A}}, \mathbb{b})\,,\, (f, F)\,,\, u\,,\, d) \quad := \quad \begin{cases} (q'_{\mathcal{A}}\,,\, \mathbb{f}) & \text{if } \mathbb{b} = \mathbb{f} \text{ and } d = f(u) \\ (q'_{\mathcal{A}}\,,\, \mathbb{t}) & \text{otherwise} \end{cases}$$

with $q'_{\mathcal{A}} := \delta_{\mathcal{A}}(\mathtt{a},\, q_{\mathcal{A}},\, u,\, F(u,d),\, d)$. Hence O looses as soon as it does not follow the direction proposed by P *via* $f$.

Thanks to the determinacy of $\omega$-regular games (see e.g. [Tho97, PP04]), we get:

**Proposition 5.12.** *Given* $\mathcal{A} : \Sigma$*, we have* $\mathcal{L}(\mathcal{A}^{\bot}) = \Sigma^{\mathfrak{D}^*} \setminus \mathcal{L}(\mathcal{A})$.

A detailed proof of Prop. 5.12 is given in [Rib17].

**5.4.1 Deduction Rules for $\bot$ and $\mathcal{A}^{\bot}$.** Since the fibre categories $\mathsf{DialAut}_\Sigma$ are symmetric monoidal closed, they are in particular dialogue categories in the sense of [Mel13], with as exponentiating object any object of $\mathsf{DialAut}_\Sigma$. Hence, if as in Ex. 5.9 we extend the deduction system of Fig. 5.9 with the realizable rules

$$\frac{}{\mathcal{A} \multimap \bot \vdash \mathcal{A}^{\bot}} \qquad \text{and} \qquad \frac{}{\mathcal{A}^{\bot} \vdash \mathcal{A} \multimap \bot}$$

then we can derive the following rules for $\bot$ and $\mathcal{A}^{\bot}$:

$$\frac{\mathcal{A}, \mathcal{B} \vdash \bot}{\mathcal{A} \vdash \mathcal{B}^{\bot}} \qquad \frac{\mathcal{A} \vdash \mathcal{B}^{\bot}}{\mathcal{A}, \mathcal{B} \vdash \bot} \qquad \frac{\mathcal{A} \vdash \mathcal{B}^{\bot}}{\mathcal{B} \vdash \mathcal{A}^{\bot}} \qquad \frac{}{\mathcal{A} \vdash \mathcal{A}^{\bot\bot}} \qquad \frac{\mathcal{A} \vdash \mathcal{B}}{\mathcal{B}^{\bot} \vdash \mathcal{A}^{\bot}} \qquad \frac{}{\mathcal{A}^{\bot\bot\bot} \vdash \mathcal{A}^{\bot}}$$

# 6 Quantifications

We now discuss quantifications in the fibration $\mathsf{DialAut}$. We follow the categorical approach outlined in §2.2, according to which existential and universal quantifications (also called simple coproducts and products [Jac01, Chap. 1]) in a fibration $\mathsf{p} : \mathbb{E} \to \mathbb{B}$ are given resp. by left adjoints $\coprod_{I,J} : \mathbb{E}_{I\times J} \to \mathbb{E}_I$ and right adjoints $\prod_{I,J} : \mathbb{E}_{I\times J} \to \mathbb{E}_I$ to the *weakening functors* $\pi^\star : \mathbb{E}_I \to \mathbb{E}_{I\times J}$ induced by $\mathbb{B}$-projections $\pi : I \times J \to I$. The adjunctions $\coprod_{I,J} \dashv \pi^\star \dashv \prod_{I,J}$ are moreover required to satisfy some coherence conditions, called the *Beck-Chevalley* conditions, which insure that they are preserved by substitution.

We first present quantifications in $\mathsf{DialAut}$ (§6.1), from which we then derive quantifications on automata (§6.2) and deduction rules for quantifications (§6.3).

## 6.1 Quantifications in DialAut

Quantifications in $\mathsf{DialAut}$ are induced by quantifications in $\mathsf{DialZ}$, which are themselves based on quantifications in simple fibrations. It is well-known (see e.g. [Jac01, Chap. 1]) that the simple fibration $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ always has simple coproducts, and has simple products iff $\mathbb{B}$ is Cartesian closed. They are given by

$$\coprod_{I,J}(I \times J, X) \quad := \quad (I, J \times X) \qquad \text{and} \qquad \prod_{I,J}(I \times J, X) \quad := \quad (I, X^J)$$

This directly extends to $\mathsf{DialZ}$.

**Proposition 6.1.** *The weakening functors* $[\pi]^\star : \mathsf{DialZ}(\Sigma) \to \mathsf{DialZ}(\Sigma\times\Gamma)$ *induced by projections* $\pi : \Sigma \times \Gamma \to \Sigma$ *have left and right adjoints given by*

$$\coprod_{\Sigma,\Gamma}(U, X) := (\Gamma \times U, X) \qquad \textit{and} \qquad \prod_{\Sigma,\Gamma}(U, X) := (U^\Gamma, \Gamma \times X) \simeq (\Gamma \multimap_{\mathbf{DZ}_{\mathfrak{D}}} (U, X))$$

The Beck-Chevalley conditions amount, for $L \in \mathbf{T}[\Delta, \Sigma]$, to the equalities

$$L^\star(\bigsqcup_{\Sigma,\Gamma}(U, X)) \quad = \quad \bigsqcup_{\Delta,\Gamma}(L \times \mathrm{Id}_\Gamma)^\star(U, X) \qquad \text{for } \bigsqcup \in \{\coprod, \prod\}$$

which follow from the fact that substitution functors are identities on objects.

The extension to $\mathsf{DialAut}$ just requires to handle winning and acceptance.

**Proposition 6.2.** *The fibration* $\mathsf{DialAut}$ *has existential and universal quantifications given by*

$$\coprod_{\Sigma,\Gamma}(U, X, \mathcal{W}_A) \;:=\; (\Gamma \times U, X, \coprod_{\Sigma,\Gamma} \mathcal{W}_A) \quad \textit{and} \quad \prod_{\Sigma,\Gamma}(U, X, \mathcal{W}_A) \;:=\; (U^\Gamma, \Gamma \times X, \prod_{\Sigma,\Gamma} \mathcal{W}_A)$$

*where* $\coprod \mathcal{W}_A$ *is defined from* $\mathcal{W}_A$ *via associativity and* $\prod \mathcal{W}_A$ *by pointwise function application as* $((\mathsf{a}_k, f_k) \cdot (\mathsf{b}_k, x_k, d_k))_k \in \prod_{\Sigma,\Gamma} \mathcal{W}_A$ *iff* $((\mathsf{a}_k, \mathsf{b}_k, f_k(\mathsf{b}_k)) \cdot (x_k, d_k))_k \in \mathcal{W}_A$.

## 6.2 Quantifications on Uniform Automata

Similarly as with the monoidal closed structure, the quantifications on automata and their deduction rules are obtained by direct adaptation of the quantifications of $\mathsf{DialAut}$.

**Definition 6.3.** *Given $\mathcal{A} : \Sigma \times \Gamma$ with set of $\mathsf{P}$-moves $U$ and set of $\mathsf{O}$-moves $X$, let*

$$\begin{array}{lcl} (\exists_\Gamma \mathcal{A} : \Sigma) & := & (Q_\mathcal{A}\,,\, q^\imath_\mathcal{A}\,,\, \Gamma \times U\,,\, X\,,\, \delta_{\exists_\Gamma \mathcal{A}}\,,\, \Omega_\mathcal{A}) \\ (\forall_\Gamma \mathcal{A} : \Sigma) & := & (Q_\mathcal{A}\,,\, q^\imath_\mathcal{A}\,,\, U^\Gamma\,,\, \Gamma \times X\,,\, \delta_{\forall_\Gamma \mathcal{A}}\,,\, \Omega_\mathcal{A}) \end{array}$$

*where*

$$\begin{array}{rlcl} & \delta_{\exists_\Gamma \mathcal{A}}(q,\, \mathtt{a},\, (\mathtt{b}, u),\, x,\, d) & := & \delta_\mathcal{A}(q,\, (\mathtt{a}, \mathtt{b}),\, u,\, x,\, d) \\ \textit{and} & \delta_{\forall_\Gamma \mathcal{A}}(q,\, \mathtt{a},\, f,\, (\mathtt{b}, x),\, d) & := & \delta_\mathcal{A}(q,\, (\mathtt{a}, \mathtt{b}),\, f(\mathtt{b}),\, x,\, d) \end{array}$$

Quantifications on automata induce an $\exists\forall$-structure which is reminiscent from Gödel's *Dialectica* interpretation (see e.g. [AF98, Koh08]).

**Example 6.4.** *Given $\mathcal{A} : \Sigma$ with set of $\mathsf{P}$-moves $U$ and set of $\mathsf{O}$-moves $X$, let $\mathcal{D}$ be the deterministic automaton*

$$(\mathcal{D} : \Sigma \times U \times X) \quad := \quad (Q_\mathcal{A},\, q^\imath_\mathcal{A},\, \mathbf{1},\, \mathbf{1},\, \delta_\mathcal{D},\, \Omega_\mathcal{A})$$

*whose transition function*

$$\delta_\mathcal{D} \quad : \quad Q_\mathcal{A} \times (\Sigma \times U \times X) \quad \longrightarrow \quad \mathfrak{D} \quad \longrightarrow \quad Q_\mathcal{A}$$

*is obtained from $\delta_\mathcal{A}$ in the obvious way. In $\mathsf{DialAut}_\Sigma$ we have $\mathcal{A} \simeq \exists_U \forall_X \mathcal{D}$.*

Let us now discuss the connection between quantifications on automata and in $\mathsf{DialAut}$. First, given $(\mathcal{A} : \Sigma \times \Gamma)$, we have, as $\mathsf{DialAut}_\Sigma$-objects,

$$(\Sigma \vdash \coprod_{\Sigma,\Gamma} \mathcal{A}) \quad = \quad (\Sigma \vdash \exists_\Sigma \mathcal{A}) \qquad \text{and} \qquad (\Sigma \vdash \prod_{\Sigma,\Gamma} \mathcal{A}) \quad = \quad (\Sigma \vdash \forall_\Sigma \mathcal{A})$$

It then follows that the Beck-Chevalley conditions in $\mathsf{DialAut}$ imply

$$\begin{array}{lclcl} \coprod_{\Sigma,\Gamma} \mathcal{A}(M \times \mathrm{Id}_\Gamma) & = & M^\star(\coprod_{\Delta,\Gamma} \mathcal{A}) & = & (\exists_\Gamma \mathcal{A})(M) \\ \prod_{\Sigma,\Gamma} \mathcal{A}(M \times \mathrm{Id}_\Gamma) & = & M^\star(\prod_{\Delta,\Gamma} \mathcal{A}) & = & (\forall_\Gamma \mathcal{A})(M) \end{array}$$

Thanks to the adjunctions $\coprod \dashv \pi^\star \dashv \prod$ in $\mathsf{DialAut}$, we then have

$$\begin{array}{lcl} \Sigma \vdash (\exists_\Gamma \mathcal{A})(M) \ \multimap\ \mathcal{B}(N) & \simeq & \Sigma \times \Gamma \vdash \mathcal{A}(M \times \mathrm{Id}_\Gamma) \ \multimap\ \mathcal{B}(N \circ [\pi_\Sigma]) \\ \Sigma \vdash \mathcal{B}(N) \ \multimap\ (\forall_\Gamma \mathcal{A})(M) & \simeq & \Sigma \times \Gamma \vdash \mathcal{B}(N \circ [\pi_\Sigma]) \ \multimap\ \mathcal{A}(M \times \mathrm{Id}_\Gamma) \end{array} \tag{47}$$

It follows that $\mathsf{P}$ has winning strategies in

$$\Sigma \times \Gamma \vdash (\forall_\Gamma \mathcal{A})[\pi_\Sigma] \ \longrightarrow\!\!\circ\ \mathcal{A} \qquad \text{and} \qquad \Sigma \times \Gamma \vdash \mathcal{A} \ \longrightarrow\!\!\circ\ (\exists_\Gamma \mathcal{A})[\pi_\Sigma] \tag{48}$$

We thus get the following corollary to Prop. 6.2.

**Corollary 6.5.** *Given uniform automata $\mathcal{A}, \mathcal{B} : \Sigma$, the game $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ is equivalent to a regular game on a finite graph. It is therefore decidable whether there exists a winning $\mathsf{P}$-strategy on $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, and if there exists such a winning $\mathsf{P}$-strategy, then there exists a finite-state one, which is moreover effectively computable from $\mathcal{A}$ and $\mathcal{B}$.*

*Proof.* By (47) and (48), $\mathsf{P}$ has a winning strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ iff it has a winning strategy in $\mathbf{1} \vdash \mathbf{I_1} \multimap \forall_\Sigma(\mathcal{A} \multimap \mathcal{B})$. But since in that game $\mathsf{O}$ can only play $\bullet$ in the component $\mathbf{I_1}$, similarly as in Ex. 3.10, it is equivalent to the acceptance game of the automaton $\forall_\Sigma(\mathcal{A} \multimap \mathcal{B}) : \mathbf{1}$ on the unique tree $\mathbf{1} : \mathfrak{D}^* \to \mathbf{1}$.

Reasoning as in [Tho97, Ex. 6.12], the game $\mathbf{1} \vdash \forall_\Sigma(\mathcal{A} \multimap \mathcal{B})$ is effectively equivalent to a regular game on a finite graph. Then, by Büchi-Landweber Theorem [BL69] (see also [Tho97, Thm. 6.18]), one can decide which player has a winning strategy, and the winner always has a finite-state winning strategy which is moreover effectively computable from the game graph. □

We also get from (48) that existential quantifications are complete in the following sense:

**Corollary 6.6.** *Given $\mathcal{A} : \Sigma \times \Gamma$, we have $\pi_\Gamma(\mathcal{L}(\mathcal{A})) \subseteq \mathcal{L}(\exists_\Gamma \mathcal{A})$.*

The converse inclusion (the correctness of existential quantifications) only holds for *non-deterministic* automata, and is detailed in §7. Dually, it follows from (48) that universal quantifications are correct (but they are complete only on *universal* automata, see Def. 3.1).

**Corollary 6.7.** *Given $\mathcal{A} : \Sigma \times \Gamma$, if $T \in \mathcal{L}(\forall_\Gamma \mathcal{A})$, then for all $\Gamma$-labeled tree $T'$ we have $\langle T, T' \rangle \in \mathcal{L}(\mathcal{A})$.*

## 6.3 Deduction Rules for Quantifications

We now turn to deduction rules for quantification. It follows from (47) that we can extend the deduction system of Fig. 21 with the rules of Fig. 22 while preserving adequacy (Prop. 5.6), Ex. 5.9 and compatibility with cut-elimination (in the sense of Rem. 5.7).

**Proposition 6.8** (Adequacy with Quantifications)**.** *If the sequent $M \; ; \; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$ is derivable using the rules of Fig. 21, Fig. 22 and of Ex. 5.9, then there is a winning $\mathsf{P}$-strategy in the game*

$$\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M) \quad \multimap \quad \mathcal{B}(M)$$

Note that the rules of Fig. 22 involve internalized substitutions of the form $\mathcal{A}[\mathtt{f}]$ as defined in Def. 3.5. The tranfert rules ($\textsc{Trans}_\uparrow$) and ($\textsc{Trans}_\downarrow$) allow to connect the internalized substitutions of the form $\mathcal{A}[\mathtt{f}]$ with the $\mathbf{T}$-substitution.

**Example 6.9.** *Using the transfert rule ($\textsc{Trans}_\downarrow$), we can derive the following specific rules of substitution for $\mathbf{T}$-maps induced by functions $\mathtt{f} : \Sigma \to \Gamma$:*

$$\frac{\mathrm{Id}_\Gamma \; ; \; \overline{\mathcal{A}} \vdash \mathcal{A}}{\mathrm{Id}_\Sigma \; ; \; \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{A}[\mathtt{f}]} \qquad\qquad \frac{M \times \mathrm{Id}_\Gamma \; ; \; \overline{\mathcal{A}} \vdash \mathcal{A}}{M \times \mathrm{Id}_\Sigma \; ; \; \overline{\mathcal{A}[\mathrm{id} \times \mathtt{f}]} \vdash \mathcal{A}[\mathrm{id} \times \mathtt{f}]}$$

*Indeed, since we have (as $\mathbf{T}$-morphisms)*

$$\mathrm{Id}_\Gamma \circ [\mathtt{f}] \quad = \quad [\mathtt{f}] \circ \mathrm{Id}_\Sigma \qquad \textit{and} \qquad (M \times \mathrm{Id}_\Gamma) \circ [\mathrm{id} \times \mathtt{f}] \quad = \quad (\mathrm{id} \times \mathtt{f}) \circ (M \times \mathrm{Id}_\Sigma)$$

*it follows that we can derive*

$$\frac{\dfrac{\mathrm{Id}_\Gamma \; ; \; \overline{\mathcal{A}} \vdash \mathcal{A}}{\mathrm{Id}_\Gamma \circ [\mathtt{f}] \; ; \; \overline{\mathcal{A}} \vdash \mathcal{A}}}{\mathrm{Id}_\Sigma \; ; \; \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{A}[\mathtt{f}]} \qquad \textit{and} \qquad \frac{\dfrac{M \times \mathrm{Id}_\Gamma \; ; \; \overline{\mathcal{A}} \vdash \mathcal{A}}{(M \times \mathrm{Id}_\Gamma) \circ [\mathrm{id} \times \mathtt{f}] \; ; \; \overline{\mathcal{A}} \vdash \mathcal{A}}}{M \times \mathrm{Id}_\Sigma \; ; \; \overline{\mathcal{A}[\mathrm{id} \times \mathtt{f}]} \vdash \mathcal{A}[\mathrm{id} \times \mathtt{f}]}$$

# 7 Non-Deterministic Automata

This final Section focuses on structural properties of non-deterministic automata, on their role in Rabin's Theorem [Rab69], namely in the complementation of non-deterministic tree automata, and on their relation with *Intuitionistic Linear Logic* (ILL) [Gir87] (see §1.3, §1.4 and §2.9).

We first detail in §7.1 the Cartesian structure of non-deterministic automata announced in §1.4 and §2.8. Technically, this Cartesian follows from the simple fact that non-deterministic automata generate comonoids in $\mathsf{DialAut}_{(-)}$ (by a direct extension of Prop. 4.5, §4.3). As a consequence, we show that our model has the witnessing properties asked to computational

$$(\textsc{Subst})\quad \frac{M \;;\; \overline{\mathcal{A}} \vdash \mathcal{A}}{M \circ N \;;\; \overline{\mathcal{A}} \vdash \mathcal{A}}$$

$$(\textsc{Trans}_{\downarrow})\quad \frac{[\mathtt{f}] \circ M \;;\; \overline{\mathcal{A}} \vdash \mathcal{B}}{M \;;\; \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{B}[\mathtt{f}]} \qquad\qquad \frac{M \;;\; \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{B}[\mathtt{f}]}{[\mathtt{f}] \circ M \;;\; \overline{\mathcal{A}} \vdash \mathcal{B}}\quad (\textsc{Trans}_{\uparrow})$$

$$(\textsc{Left}\ \exists)\quad \frac{M \times \mathrm{Id}_{\Gamma} \;;\; \overline{\mathcal{A}[\pi]}, \mathcal{B} \vdash \mathcal{A}[\pi]}{M \;;\; \overline{\mathcal{A}}, \exists_{\Gamma}\mathcal{B} \vdash \mathcal{A}} \qquad\qquad \frac{M \times N \;;\; \overline{\mathcal{A}} \vdash \mathcal{A}}{M \times N \;;\; \overline{\mathcal{A}} \vdash (\exists_{\Gamma}\mathcal{A})[\pi]}\quad (\textsc{Right}\ \exists)$$

$$(\textsc{Left}\ \forall)\quad \frac{M \times N \;;\; \overline{\mathcal{A}}, \mathcal{B} \vdash \mathcal{A}}{M \times N \;;\; \overline{\mathcal{A}}, (\forall_{\Gamma}\mathcal{B})[\pi] \vdash \mathcal{A}} \qquad\qquad \frac{M \times \mathrm{Id}_{\Gamma} \;;\; \overline{\mathcal{A}[\pi]} \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}} \vdash \forall_{\Gamma} A}\quad (\textsc{Right}\ \forall)$$

Figure 22: Substitution and quantification rules for uniform automata (where $M, N$ are composable, $\pi$ is a suitable projection and $\mathtt{f}$ is a function on alphabets)

interpretations of proofs (in the sense of §1.2), and moreover that it allows to combine strategies obtained from proofs with witnessing strategies computed by usual emptiness checking algorithms (see §1.5).

Second, we show that a powerset construction for the *Simulation Theorem* [MS87, EJ91, MS95] satisfies the usual deduction rules of the exponential modality ! of ILL. This completes the picture sketched in §1.4, §1.5 and §2.9, and moreover allows to obtain a deduction system which is complete w.r.t. intuitionistic and classical deduction (*via* usual translations). Furthermore, [Rib17, App. C] details how two constructions from resp. [CL08] and [SA05] can be reformulated in our setting.

The proofs of all statments of this Section are given in [Rib17].

## 7.1 The Cartesian Structure of Non-Deterministic Automata

Similarly as with usual (total) non-deterministic automata in §2.8, the monoidal product of uniform automata is Cartesian on non-deterministic automata. Recall from Def. 3.1 that a uniform automaton is non-deterministic if its set of O-moves is $\simeq \mathbf{1}$.

Consider a $\mathsf{DialAut}_{\Sigma}$-object $\mathcal{N}(L)$ with $\mathcal{N}$ non-deterministic and with set of P-moves $U$. Hence, the underlying $\mathsf{DialZ}(\Sigma)$-object of $\mathcal{N}(L)$ is of the form $(U, I)$ with $I \simeq \mathbf{1}$. As we have seen in §5.3, we thus get canonical realizers for

$$\mathcal{N}(L) \quad \multimap \quad \mathcal{N}(L) \otimes \mathcal{N}(L) \qquad \text{and} \qquad \mathcal{N}(L) \quad \multimap \quad \mathbf{I} \tag{49}$$

As we shall see now, these canonical realizer equip $\mathcal{N}(L)$ with the structure of a comonoid[16]. Thanks to well-known results (see e.g. [Mel09, Cor. 18, §6.5]), this implies that the monoidal structure of uniform automata is Cartesian on non-deterministic automata.

Recall from Prop. 4.3 that objects of the form $(K, I)$ with $I \simeq \mathbf{1}$ are comonoids in $\mathbf{DZ}$, and from Prop. 4.5 that such objects are also comonoids in $\mathbf{DZ}_{\mathfrak{D}}$. On the other hand, we have seen that $\mathsf{DialZ}(\Sigma)$ is a Kleisli category of comonoid indexing in $\mathbf{DZ}_{\mathfrak{D}}$, whose symmetric monoidal structure is given be the extension of Prop. 4.5 to comonoid indexing given by Prop 5.2. Actually, the lifting of comonoids given by Prop. 4.5 also extends to the case of comonoid indexing:

[16] Recall from §4.2 that in this paper, by (co)monoid we always mean *commutative* (co)monoid

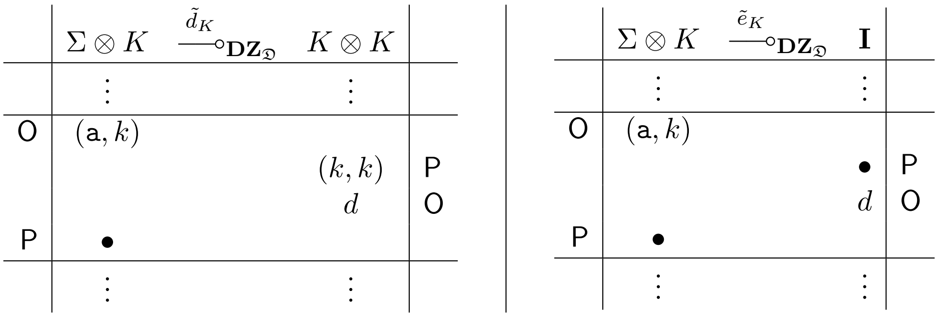


Figure 23: Structure maps in $\mathsf{DialZ}(\Sigma)$ for the comonoid $K = (K, \mathbf{1})$

**Proposition 7.1.** *Given a comonoid $C$ in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$, each comonoid $(K, d, e)$ in $\mathbb{C}$ induces a comonoid $(K,\, d \circ \epsilon^C_K,\, e \circ \epsilon^C_K)$ in the Kleisli category $\mathbf{Kl}(C)$ of indexing with $C$. In the case of $\mathsf{DialZ}(\Sigma)$, the structure maps $\tilde{d}_K$ and $\tilde{e}_K$ of the comonoid induced by $K = (K, \mathbf{1})$ can be depicted as on Fig. 23 (where we omitted some $\bullet$-moves).*

The extension of Prop. 7.1 to the $\mathsf{DialAut}_\Sigma$-objects induced by non-deterministic automata is direct. Moreover, $\mathsf{DialAut}_\Sigma$-morphisms between non-deterministic automata are comonoid morphisms.

**Proposition 7.2.** *For each alphabet $\Sigma$, objects of the form $\Sigma \vdash \mathcal{N}(L)$, where $\mathcal{N}$ is non-deterministic, are comonoids in $\mathsf{DialAut}_\Sigma$. Moreover, $\mathsf{DialAut}_\Sigma$-morphisms between such objects are comonoid morphisms.*

Since the category of comonoids of a symmetric monoidal category has finite products (see e.g. [Mel09, Cor. 18, §6.5]), we thus have the expected result that non-deterministic automata are equipped with a Cartesian structure.

**Corollary 7.3.** *For each alphabet $\Sigma$, the full subcategory $\mathsf{DialAut}^{\mathrm{ND}}_\Sigma$ of $\mathsf{DialAut}_\Sigma$, whose objects are of the form $(U, I, \mathcal{W})$ with $I \simeq \mathbf{1}$, is Cartesian.*

**7.1.1 Application: Deduction Rules for Non-Deterministic Automata.** Similarly as with usual (total) non-deterministic automata in §2.8, Cor. 7.3 allows to extend adequacy (Prop. 5.6 and Prop. 6.8) to the following restriction of the structural weakening and contraction rules:

$$(\textsc{Weak}_{\mathrm{ND}})\ \frac{M \ ;\ \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \ ;\ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M \ ;\ \overline{\mathcal{A}}, \mathcal{N}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \ ;\ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}_{\mathrm{ND}}) \tag{50}$$

where $\mathcal{N}$ is required to be non-deterministic (while $\overline{\mathcal{A}}$, $\overline{\mathcal{B}}$ and $\mathcal{C}$ can be arbitrary). Note that the full weakening rule is actually derivable in the setting of Ex. 5.9, but with non-canonical realizers of $\mathcal{A} \multimap \mathbf{I}$ when $\mathcal{A}$ is not non-deterministic.

**7.1.2 Application: Existential Quantifications and Extraction.** A nice consequence of the Cartesian structure of $\mathsf{DialAut}^{\mathrm{ND}}_{(-)}$ is the fact that existential quantifications behave similarly as the usual *sum types* of Type Theory (see e.g. [Jac01, Chap. 10]). Consider a non-deterministic automaton $\mathcal{N} : \Sigma \times \Gamma$ with set of P-moves $U$, and let $T$ be a $\Sigma$-labeled tree (so that $T : \mathfrak{D}^* \to \Sigma$).

It follows from the representation of **DZ**-strategies given by Prop. 3.12 that a winning P-strategy in $\mathbf{1} \vdash \mathbf{I} \multimap (\exists_\Gamma \mathcal{A})(\dot{T})$ is given by a function

$$\bigcup_{n\in\mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Gamma \times U$$

hence by a pair of functions

$$\left(\bigcup_{n\in\mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Gamma\right) \times \left(\bigcup_{n\in\mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad U\right)$$

and therefore by a tree $T' : \mathfrak{D}^* \to \Gamma$ and a winning P-strategy in $\mathbf{1} \vdash \mathbf{I} \multimap \mathcal{A}\langle \dot{T}, \dot{T}' \rangle$.

**Proposition 7.4.** *Given a non-deterministic automaton $\mathcal{N} : \Sigma \times \Gamma$, a winning* P*-strategy $\sigma : \mathbf{1} \multimap \exists_\Sigma \mathcal{N}$ is of the form $\sigma = \langle T, \tau \rangle$ where $T$ is a $\Sigma$-labeled tree and $\tau$ is a winning* P*-strategy in $\mathbf{1} \multimap \mathcal{N}(T)$ (so in particular $T \in \mathcal{L}(\mathcal{N})$).*

In particular, we get the following fact, which completes Cor. 6.6 and mirrors the well-known situation with usual non-deterministic automata.

**Corollary 7.5.** *If $\mathcal{N} : \Sigma \times \Gamma$ is* non-deterministic *then $\mathcal{L}(\exists_\Gamma \mathcal{N}) = \pi_\Gamma(\mathcal{L}(\mathcal{N}))$.*

Moreover, it follows from Prop. 7.4 that our computational interpretation allows to effectively extract witnesses from (interpretations of) proofs, in the sense of §1.2 and §1.5. Let $\mathcal{N} : \Sigma$ be non-deterministic with set of P-moves $U$, and consider a derivation $\mathscr{D}$ of the sequent

$$\mathbf{1}\ ;\ \vdash \exists_\Sigma \mathcal{N}$$

using the rules of Fig. 21, Fig. 22, Ex. 5.9 and (50). Then adequacy (Prop. 5.6 and Prop. 6.8) gives a strategy

$$\sigma \quad : \quad \mathbf{I} \quad \multimap \quad \exists_\Sigma \mathcal{N}$$

(effectively computed by induction on $\mathscr{D}$), and which by Prop. 7.4 is of the form

$$\begin{array}{lrcl} & \langle T, \tau \rangle & : & \bigcup_{n\in\mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Sigma \times U \\ \text{where} & \tau & : & \mathbf{I} \quad \multimap \quad \mathcal{N}(T) \end{array}$$

**7.1.3 Application: Effective Realizers from Witnesses of Non-Emptiness.** Similarly as with usual non-deterministic automata (see e.g. [Tho97]), thanks to the Büchi-Landweber Theorem [BL69], Cor. 7.5 implies the decidability of emptiness for non-deterministic automata as well as the *Rabin Basis Theorem* [Rab72], stating that if $\mathcal{L}(\mathcal{N}) \neq \emptyset$, then its contains a regular tree $T$ and a finite state winning P-strategy on $\mathcal{N}(T)$ (both effectively definable from $\mathcal{N}$).

**Corollary 7.6.** *Given a non-deterministic automaton $\mathcal{N} : \Sigma$, one can decide whether $\mathcal{L}(\mathcal{N})$ is empty. Moreover, if $\mathcal{L}(\mathcal{N}) \neq \emptyset$ then one can effectively build from $\mathcal{N}$ a regular tree $T \in \mathcal{L}(\mathcal{N})$ together with a finite state winning* P*-strategy on $\mathbf{I} \multimap \mathcal{N}(T)$.*

More generally, strategies witnessing (non-)emptiness obtained *via* Cor. 7.5 can be lifted to winning strategies in games of the form $\mathcal{A} \multimap \mathcal{C}$. Consider the case (mentioned in §1.5.(bi)) of $\mathcal{C} = \mathcal{B}^\perp$ and with $\mathcal{A}, \mathcal{B} : \Sigma$ non-deterministic. If $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, then an O-strategy witnessing $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \emptyset$, which corresponds *via* Prop. 5.12 to a P-strategy witnessing $\mathbf{1} \in \mathcal{L}((\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp)$, can be lifted to a winning P-strategy in $\mathcal{A} \multimap \mathcal{B}^\perp$.

**Proposition 7.7.** *Given non-deterministic $\mathcal{A}, \mathcal{B} : \Sigma$, if $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, then there are winning* $\mathsf{P}$*-strategies in $\mathcal{A} \otimes \mathcal{B} \multimap \perp$ and $\mathcal{A} \multimap \mathcal{B}^{\perp}$. Moreover, these* $\mathsf{P}$*-strategies can be assumed to be finite state and can be effectively obtained from $\mathcal{A}$ and $\mathcal{B}$.*

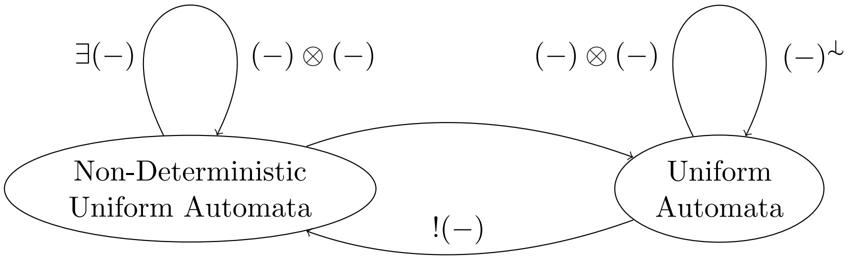
Proposition 7.7, together with Ex. 5.9.(ii), implies the following extension of Ex. 5.9.(i).

**Corollary 7.8.** *If $\mathcal{A}, \mathcal{B} : \Sigma$ are non-deterministic and such that $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, then $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B} \multimap \mathcal{A}) \subseteq \mathcal{L}(\mathcal{B}^{\perp})$.*

## 7.2 Simulation and the Exponential Modality of ILL

Recall that similarly as in the usual setting, uniform automata have linear complements (§5.4), and that non-deterministic automata have correct existential quantifications (§7.5). On the other hand, we mentioned in §1.3 that in the usual setting, the *Simulation Theorem* [MS87, EJ91, MS95] says that each alternating automaton $\mathcal{A}$ can be simulated by a non-deterministic automaton $!\mathcal{A}$ (of exponential size) with $\mathcal{L}(!\mathcal{A}) = \mathcal{L}(\mathcal{A})$.

We show here that in our setting, an easy adaptation of the construction used in [Wal02] gives a similar simulation operation $!(-)$, taking a uniform automaton $\mathcal{A} : \Sigma$ to an non-deterministic automaton $!\mathcal{A} : \Sigma$ with $\mathcal{L}(!\mathcal{A}) = \mathcal{L}(\mathcal{A})$, thus completing the picture (4) of §1.3 for our notion of uniform automata:



(51)

Moreover, we show that the operation $!(-)$ satisfies the *deduction* rules of the exponential modality ! of $\mathsf{ILL}$:

$$\frac{M \;;\; !\overline{\mathcal{A}} \vdash \mathcal{A}}{M \;;\; !\overline{\mathcal{A}} \vdash !\mathcal{A}} \qquad \frac{M \;;\; \overline{\mathcal{A}}, \mathcal{B} \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}}, !\mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \;;\; \overline{\mathcal{A}}, \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}}, !\mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \;;\; \overline{\mathcal{A}}, !\mathcal{B}, !\mathcal{B} \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}}, !\mathcal{B} \vdash \mathcal{A}} \tag{52}$$

It follows that the exponential ! allows to define, using Girard's decomposition, an intuitionistic implication $\to$ as $\mathcal{A} \to \mathcal{B} := !\mathcal{A} \multimap \mathcal{B}$.

The rules (52) are an obvious adaptation to our context of the rules displayed in (26) and (27) of §2.9. The last two rules (weakening and contraction) actually follow from the rules (Weak$_{\text{ND}}$) and (Contr$_{\text{ND}}$) displayed in (50). The second rule (Dereliction) will easily follow from the construction of $!\mathcal{A}$. The most difficult rule is the first one (Promotion), which is moreover not compatible with cut-elimination (see Rem. 5.7).

The difficulty with the (Promotion) rule can be explained as follows. We have seen in §7.1 above that the symmetric monoidal structure of $\mathsf{DialAut}_{\Sigma}$ is Cartesian on non-deterministic automata, in other words that non-deterministic automata have a canonical comonoid structure (49). It follows that similarly as with usual $\mathsf{ILL}$-exponentials (see §2.9 but also [Mel09]), the simulation operation $!(-)$ adds to an arbitrary automaton $\mathcal{A}$ the structure allowing $!\mathcal{A}$ to be equipped with canonical maps:

$$!\mathcal{A} \quad \multimap \quad !\mathcal{A} \otimes !\mathcal{A} \qquad \text{and} \qquad !\mathcal{A} \quad \multimap \quad \mathbf{I}$$

On the other hand, recall from §5.3 that for a uniform automaton $\mathcal{A}$ with set of $\mathsf{O}$-moves $X$, realizers of

$$\mathcal{A} \quad \multimap \quad \mathcal{A} \otimes \mathcal{A}$$

may not exist because $\mathsf{O}$ can play two different $(x, x') \in X \times X$ in the right component $\mathcal{A} \otimes \mathcal{A}$, that $\mathsf{P}$ may not be able to merge into a single $x'' \in X$ in the left component $\mathcal{A}$.

Usual solutions for $\mathsf{ILL}$-exponentials (see e.g. [Mel09, AC98, Mel04]) amount to equip objects of the form $!A$ with some duplication and memory abilities, essentially allowing $!A$ to run several copies of $A$ in parallel. However (and this is *via* (5) §1.3, the crux of Rabin's Theorem [Rab69]), such recipes can not (at least in an obvious way) be applied to automata on infinite trees, because $!\mathcal{A}$ must be a finite-state automaton, while plays in acceptance games (which are infinite) would require an infinite memory.

Phrased in modern terms, the solution is given by the existence of *positional* (*i.e.* memoryless) winning strategies in $\omega$-regular games equipped with *parity* acceptance conditions (see e.g. [Tho97, GTW02]). In our case, we rely for the (Promotion) rule on the stronger fact that in an $\omega$-regular game whose winning condition is given by a disjunction of parity conditions (also called a *Rabin* condition), winning $\mathsf{P}$-strategies can always be assumed to be positional [Kla94, KK95, Jut97, Zie98]. Unfortunately, positionality is not preserved by composition, and the interpretation of the (Promotion) rule is not preserved by cut-elimination (in the sense of Rem. 5.7).

**Remark 7.9.** *In (51), we have only displayed existential quantifications $\exists$ for non-deterministic automata, because as in the usual setting, they are correct (in the sense of Cor. 7.5) only on non-deterministic automata. Similarly, we did not displayed universal quantifications because they are only complete on* universal *automata (see Def. 3.1).*

*Note that on the other hand, the categorical properties of quantifications (Prop. 6.2) and thus the deduction rules of Fig. 22, hold on general uniform automata.*

### 7.2.1 Parity Automata.

Similarly as in the usual setting, we say that $\mathcal{A}$ is a *parity* automaton if $\Omega_{\mathcal{A}}$ is generated from a map $c_{\mathcal{A}} : Q_{\mathcal{A}} \to \mathbb{N}$ as the set of sequences $(q_k)_k$ such that the maximal number occurring infinitely often in $(c_{\mathcal{A}}(q_k))_k$ is even.

**Proposition 7.10.** *For every automaton $\mathcal{A} : \Sigma$, there is a parity automaton $\mathcal{A}^\dagger : \Sigma$ such that $\mathcal{A}^\dagger \simeq \mathcal{A}$ in $\mathsf{DialAut}_\Sigma$.*

Note that $\mathcal{A} \simeq \mathcal{A}^\dagger$ implies $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{A}^\dagger)$ by Prop. 4.12.

### 7.2.2 An Exponential Construction on Uniform Automata.

Our exponential construction $!(-)$ is an adaptation of the one used in [Wal02]. Given a parity automaton $\mathcal{A} : \Sigma$ with set of $\mathsf{P}$-moves $U$ and set of $\mathsf{O}$-moves $X$, we let

$$!\mathcal{A} \quad := \quad (Q_{!\mathcal{A}}\,,\, q^\imath_{!\mathcal{A}}\,,\, U^{Q_{\mathcal{A}}}\,,\, \mathbf{1}\,,\, \delta_{!\mathcal{A}}\,,\, \Omega_{!\mathcal{A}})$$

where $Q_{!\mathcal{A}} := \mathcal{P}(Q_{\mathcal{A}} \times Q_{\mathcal{A}})$, $q^\imath_{!\mathcal{A}} := \{(q^\imath_{\mathcal{A}}, q^\imath_{\mathcal{A}})\}$ and $\delta_{!\mathcal{A}}$ is defined as follows: Given $\mathsf{a} \in \Sigma$, $f \in U^{Q_{\mathcal{A}}}$, $d \in \mathfrak{D}$ and $\pi_2(S) = \{q' \mid \exists q.\ (q, q') \in S\} = \{q_1, \dots, q_n\}$, let

$$\delta_{!\mathcal{A}}(S, \mathsf{a}, f, \bullet, d) \quad := \quad T_1 \cup \dots \cup T_n$$

where, for each $k \in \{1, \dots, n\}$,

$$T_k \quad := \quad \{(q_k, q) \mid \exists x \in X.\ q = \delta_{\mathcal{A}}(q_k, \mathsf{a}, f(q_k), x, d)\}$$

Let a *trace* in an infinite sequence $(S_n)_n \in Q_{!\mathcal{A}}^{\omega}$ be a sequence $(q_n)_n$ such that for all $n$, $(q_n, q_{n+1}) \in S_{n+1}$. We let $\Omega_{!\mathcal{A}}$ be the set of sequences $(S_n)_n$ whose traces all belong to $\Omega_{\mathcal{A}}$. Note that $\Omega_{!\mathcal{A}}$ is $\omega$-regular since $\Omega_{\mathcal{A}}$ is $\omega$-regular (see e.g. [§4][Wal02]).

**Remark 7.11.** *Note that $Q_{!\mathcal{A}} = \mathcal{P}(Q \times Q)$ contains a "true" state $\emptyset \in Q_{!\mathcal{A}}$, so the map*

$$\tilde{\delta}_{!\mathcal{A}} \quad : \quad Q_{!\mathcal{A}} \times \Sigma \quad \longrightarrow \quad U^Q \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow Q_{!\mathcal{A}})$$

*is always total.*

For a uniform automaton $\mathcal{A}$ whose acceptance condition is not a parity condition, let $!\mathcal{A} := !(\mathcal{A}^{\dagger})$, where $\mathcal{A}^{\dagger}$ is obtained from Prop. 7.10.

It is easy to show the adequacy of the dereliction rule. This amounts to provide co-unit-like winning $\mathsf{P}$-strategies

$$\epsilon \quad : \quad !\mathcal{A}(M) \quad \multimap \quad \mathcal{A}(M)$$

**Proposition 7.12.** *Given $\mathcal{A} : \Sigma$, there is a winning $\mathsf{P}$-strategy $\epsilon$ in $\Sigma \vdash !\mathcal{A}(M) \multimap \mathcal{A}(M)$.*

### 7.2.3 Game Graphs and Positionality.

We now turn to the (Promotion) rule. Its adequacy relies on well-known but non-trivial results on the existence of winning positional $\mathsf{P}$-strategies for *Rabin* games, which are games whose winning conditions are disjunctions of parity conditions. The notion of *positional* strategy makes sense for games whose moves and winning condition are induced in an appropriate way by a given graph.

Consider uniform substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$, where $\mathcal{A}$ (resp. $\mathcal{B}$) has set of $\mathsf{P}$-moves $U$ (resp. $V$) and set of $\mathsf{O}$-moves $X$ (resp. $Y$). The *game graph* of $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$ is the graph $G$ with vertices:

$$(A_{\mathsf{P}} \times B_{\mathsf{P}}) + (A_{\mathsf{O}} \times B_{\mathsf{P}}) + (A_{\mathsf{O}} \times B_{\mathsf{O}})$$

where

$$\begin{array}{lcl@{\qquad\qquad}lcl} A_{\mathsf{P}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{A}} & A_{\mathsf{O}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{A}} \times U \\ B_{\mathsf{P}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{B}} & B_{\mathsf{O}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{B}} \times V \end{array}$$

and with edges depicted in Fig. 24, where $q'_{\mathcal{A}} := \delta_{\mathcal{A}}(q_{\mathcal{A}}, M(\bar{\mathsf{a}}.\mathsf{a}, p), u, x, d)$ (for some $x \in X$) and $q'_{\mathcal{B}} := \delta_{\mathcal{B}}(q_{\mathcal{B}}, N(\bar{\mathsf{a}}.\mathsf{a}, p), v, y, d)$ (for some $y \in Y$). Write $\mathsf{pos}$ for the graph morphism from the set of plays of $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$ (seen as a tree) to $G$. We say that a strategy $\sigma$ is *positional* if it agrees on plays with the same position, *i.e.* if $s.m \in \sigma$, $t.m' \in \sigma$ with $\mathsf{pos}(s) = \mathsf{pos}(t)$ implies $m = m'$.

Consider now parity automata $\mathcal{A}_1, \ldots, \mathcal{A}_n$ and $\mathcal{B}$. The winning condition of a game of the form $\mathcal{A}_1(M_1) \otimes \ldots \otimes \mathcal{A}_n(M_n) \multimap \mathcal{B}(N)$ is a disjunction of parity conditions, also called a *Rabin* condition, which is induced by colorings depending only on the vertices of its game graph $G$. It has been shown in [Kla94, KK95, Jut97, Zie98] that if $\mathsf{P}$ has a winning strategy $\sigma$ in such a game, then it has a winning *positional* strategy (w.r.t. $G$), which according to [Zie98] is recursive in $\sigma$.

The existence of winning positional $\mathsf{P}$-strategies allows us to show the adequacy of the (Promotion) rule. A detailed proof of Prop. 7.13 is given in [Rib17, App. B].

**Proposition 7.13.** *Given $\mathcal{N}, \mathcal{A} : \Sigma$ with $\mathcal{N}$ non-deterministic, if there is a winning $\mathsf{P}$-strategy in $\Sigma \vdash \mathcal{N}(L) \multimap \mathcal{A}(M)$, then there is a winning $\mathsf{P}$-strategy in the game $\Sigma \vdash \mathcal{N}(L) \multimap !\mathcal{A}(M)$.*

$$\begin{array}{c|ccc|c}
 & ((p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{A}}) & , & (p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{B}})) & A_{\mathsf{P}} \times B_{\mathsf{P}} \\
\mathsf{O} & \downarrow & & & \\
 & ((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u) & , & (p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{B}})) & A_{\mathsf{O}} \times B_{\mathsf{P}} \\
\mathsf{P} & & & \downarrow & \\
 & ((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u) & , & (p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{B}}\,,\,v)) & A_{\mathsf{O}} \times B_{\mathsf{O}} \\
\mathsf{O} & & & \downarrow & \\
 & ((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u) & , & (p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{B}})) & A_{\mathsf{O}} \times B_{\mathsf{P}} \\
\mathsf{P} & \downarrow & & & \\
 & ((p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{A}}) & , & (p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{B}})) & A_{\mathsf{P}} \times B_{\mathsf{P}}
\end{array}$$

Figure 24: The edges of the graph $G$ for $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$

**7.2.4 Applications.** This paragraph gathers consequences of Props. 7.12 and 7.13, thus mirroring §7.1.1-7.1.3 and completing the picture announced in §1.4, §1.5 and §2.9.

First, Prop. 7.12 implies that $\mathcal{L}(!\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A})$, while Prop. 7.13 gives the converse inclusion $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(!\mathcal{A})$. We thus have, as expected:

**Corollary 7.14.** *$\mathcal{L}(\mathcal{A}) = \mathcal{L}(!\mathcal{A})$.*

Corollary 7.14 gives the extension of Cor. 7.6 to general uniform automata.

**Corollary 7.15.** *Given a uniform automaton $\mathcal{A}$, one can decide whether $\mathcal{L}(\mathcal{A})$ is empty. Moreover, if $\mathcal{L}(\mathcal{A}) \neq \emptyset$ then one can effectively build from $\mathcal{A}$ a regular tree $T \in \mathcal{L}(\mathcal{A})$ together with a finite state winning $\mathsf{P}$-strategy on $\mathbf{1} \vdash \mathbf{I} \multimap \mathcal{A}(\dot{T})$.*

We also obtain the lifting property of §1.5.(bii), extending Prop. 7.7.

**Proposition 7.16** (Weak Completeness)**.** *Given automata $\mathcal{A}, \mathcal{B} : \Sigma$, if $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$ then there is an effective winning $\mathsf{P}$-strategy in $\Sigma \vdash !\mathcal{A} \multimap (!(\mathcal{B}^{\perp}))^{\perp}$.*

On the other hand, Props. 7.12 and 7.13 give adequacy for the rules displayed in (52).

**Proposition 7.17** (Adequacy)**.** *If the sequent $M \;;\; \mathcal{A}_1, \dots, \mathcal{A}_n \vdash \mathcal{B}$ is derivable using the rules of Fig. 21, Fig. 22 Fig. 25 and of Ex. 5.9, then there is a winning $\mathsf{P}$-strategy in the game*

$$\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M) \quad \multimap \quad \mathcal{B}(M)$$

As an example of use of the exponential rules, we mention a negative translation of the law of Peirce $((A \Rightarrow B) \Rightarrow A) \Rightarrow A$. The law of Peirce gives full classical logic when added to intuitionistic logic.

**Example 7.18.** *The law of Peirce $!((?\mathcal{A} \to ?\mathcal{B}) \to ?\mathcal{A}) \vdash ?\mathcal{A}$ (where $?\mathcal{A} := (!(\mathcal{A}^{\perp}))^{\perp}$ and where $\mathcal{A} \to \mathcal{B} := !\mathcal{A} \multimap \mathcal{B}$), can be derived using the exponential rules.*

# 8 Conclusion

We have presented preliminary results toward a Curry-Howard approach to automata on infinite trees. Our contributions concern mainly two related directions.

First, we have shown that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Theorem [Rab69] can be organized in a deduction

$$(\textsc{Dereliction})\ \frac{M \ ; \ \overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \ ; \ \overline{\mathcal{A}}, !\mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M \ ; \ \overline{\mathcal{N}} \vdash \mathcal{A}}{M \ ; \ \overline{\mathcal{N}} \vdash !\mathcal{A}}\ (\textsc{Promotion})$$

$$(\textsc{Weak}_{\textsc{ND}})\ \frac{M \ ; \ \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \ ; \ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M \ ; \ \overline{\mathcal{A}}, \mathcal{N}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \ ; \ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}_{\textsc{ND}})$$

Figure 25: Exponential rules (where $\mathcal{N}$ and $\overline{\mathcal{N}}$ are non-deterministic)

system based on intuitionistic linear logic (ILL) [Gir87]. Namely, we equipped a variant of usual alternating tree automata (that we called *uniform* tree automata, §3) with a fibred monoidal closed structure (§4 and §5), which in particular handles a conjunction and, *via* game determinacy, a linear complementation of alternating automata, as well as deduction rules for existential and universal quantifications (§6). Moreover, we have shown in §7 that this monoidal structure is Cartesian on non-deterministic automata, and in particular that (an adaptation of) a usual powerset construction for the Simulation Theorem [MS87, EJ91, MS95] satisfies the *deduction rules* of an !(−) ILL-exponential modality.

Second, our approach is based on a realizability semantics for our deduction system on tree automata, in which, thanks to the monoidal-closed structure, realizers are winning strategies in (almost usual) acceptance games. Our realizability semantics satisfies an expected property of witness extraction from proofs of existential statements. Moreover, this realizability semantics is compositional and allows to combine realizers produced as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata, possibly obtained using external algorithms.

We believe that this can provide a basis for semi-automatic approaches to MSO on infinite trees[17], in which, similarly as with interactive proof systems, decision algorithms can be combined with human-produced proofs or proof-search techniques. The author and P. Pradic have recently obtained preliminary results in this direction for MSO on $\omega$-words [PR17].

Furthermore, as shown in Ex. 6.4, our interpretation shares a formal similarity with Gödel's *Dialectica* interpretation (see e.g. [AF98, Koh08]). Actually, the category **DZ** can be constructed (*via* a distributive law) from a category of *simple self dualization* [HS99, HS03] (over the topos of trees, see e.g. [BMSS12]), which can be seen as a skeleton of Dialectica-like categories [dP91], and the category DialZ has a shape similar to Dialectica fibrations (see [Hyl02, Hof11] but also [Jac01, Ex. 1.10.11]). We do not know yet how far this connection can go, but it seems that it can provide, similarly as with the usual Dialectica interpretation, realizers for linear variants of Markov and choice rules[18].

Moreover, our setting easily handles known constructions from [CL08] and [SA05] for language reduction and separation.

**8.1 Further Works.** We plan to continue the line of research initiated here and in [Rib15] along different directions. A central point w.r.t. most of them concerns the (Promotion) rule.

The interpretation of Simulation as an !(−) ILL-exponential modality in §7.2 is interesting

[17] Even if there are numerous implementations of decision algorithms on *tree automata*, we are aware of no working implementation of decision procedures for the full language of MSO on infinite trees.

[18] The reader aware that choice is not expressible in the language of MSO on infinite trees (see e.g. [CL07]) may be surprised by this suggestion. Actually, choice rules in constructive arithmetics turn ∀∃-statements into ∃∀ ones, but do not necessarily induce wellorderings.

because it shows that an ILL-like exponential arises precisely where there is a semantic difficulty (positionality) together with a non-trivial exponential construction on automata. However, we find the interpretation of the (Promotion) rule in §7.2 not completely satisfactory for the following reasons.

(1) We have to rely on the *external* result that winning P-strategies can always be assumed to be positional in Rabin games [Kla94, KK95, Jut97, Zie98]. There seems to be essentially two ways to apply this result: (a) one could try to *extract* the positional strategy realizing the conclusion of (Promotion) from the realizer of the premise, or (b) one could obtain the strategy for the conclusion from an algorithm solving $\omega$-regular games (that is from the Büchi-Landweber Theorem [BL69], see also e.g. [Tho97, Thm. 6.16]).

    However, in both cases this amounts to apply a non-trivial external algorithm, and there seem to be no obvious structural relation between the realizer of the conclusion and the realizer of the premise.

(2) This interpretation of the (Promotion) rule is not compatible with cut-elimination (in the sense of Rem. 5.7), because the notion of positionality required for (Promotion) is not preserved by composition, so that $!(-)$ is not a functor.

    It is unclear to us whether this is a true drawback, because we can still compose realizers and extract witnesses for existentials (§7.1.2). The only point is that two derivations which are equal modulo cut-elimination may be interpreted by two different strategies. But still, the non-functoriality of $!(-)$ is somehow uncomfortable from a semantic perspective.

First, we plan to pursue some work on the category **DZ** of zig-zag games in order to get a better picture of its usual game semantics exponentials. According to the discussion of §7.2, such exponentials would involve some infinite memory, because plays are infinite in **DZ**. Moreover, it seems reasonable to target some relaxation of **DZ** with finite limits (typically by allowing games to be equipped with a notion of legal plays).

(1) Following the recipe of [MTT09], we plan the investigate the existence of free exponentials.

(2) Moreover, there seem to be a natural exponential, in which P essentially plays strategies, but which in the context of automata would lead to infinite state automata. Using iteration theorems such as e.g. Muchnik's Theorem (see e.g. [Wal02], but also [BCL08]) it may be possible to obtain a hierarchy (in the sense of the hierarchy of simple types) of possibly infinite state automata, but with decidable emptiness checking.

(3) We also plan to look at non-synchronous exponentials, such as the Curien-Lamarche exponential of *sequential data structures* (see e.g. [AC98, Chap. 14], but also [Mel05]), in particular because of its backtracking abilities. We suspect that this could allow to handle known results and constructions for reduction and separation properties, in the vein of [Arn99, AN07, FMS13]. However, we do not know yet if this can provide new results.

Second, an important direction of future work is to get a better semantic account of the notion of positionality used in the interpretation of the (Promotion) rule. In the realm of game semantics, it has been shown by Melliès [Mel06] that the notion of *Innocence* (originally introduced by [HO00] *via* a notion of pointers on moves), which characterizes a form of functional (state-free) behavior, corresponds to some notion of positionality. Innocence is actually a strong form of positionality, which is preserved by composition. It is possible to equip DialAut-games with an obvious notion of pointers, representing applications of the transition function of automata as unfoldings of fixpoints. This leads *via* innocence to a notion of positionality which seems to be equipped with a monoidal-closed structure (w.r.t. to the synchronous direct product of automata), but which seems too restrictive to handle strategies obtained (*via* Büchi-Lambweber Theorem) from emptiness checking in the sense of Cor. 6.5, §7.1.3, Cor. 7.15, and Prop. 7.16. On the other hand, the notion of positionality used in §7.2.3 may be preserved by composition for *non-deterministic* innocent strategies, in the vein of [HP12, TO15]. We do not know yet how such notions of non-deterministic strategies behave w.r.t. the construction of positional winning $\mathsf{P}$-strategies for Rabin games as in e.g. [Zie98].

Finally, our main target is the construction of realizability models for $\mathsf{MSO}$. In the case of $\omega$-words (that is taking $\mathfrak{D} = \mathbf{1}$ in this paper), and in the context of Church's synthesis, the aforementioned results of [PR17] suggest that, together with the results of this paper, it is possible and pertinent to devise refinements of $\mathsf{MSO}$ based on $\mathsf{ILL}$. We also already mentioned above the connection with Gödel's *Dialectica* interpretation, which suggests that it may be possible to realize linear variants of Markov and choice rules. Furthermore, this paper indicates that working in a linear deduction system for $\mathsf{MSO}$ allows to obtain a fibred monoidal closed structure, with in particular deduction rules for existential and universal quantifications. We think that this can provide a good basis to handle some axioms of $\mathsf{MSO}$, and moreover that $\mathsf{ILL}$ can provide classes of formulae with improved translations to automata w.r.t. the known non-elementary lower bound (see e.g. [GTW02, Chap. 13]).

Moreover, in devising realizability models for $\mathsf{MSO}$, and in particular following the approach of this paper which decomposes the translation of formulae to automata using linear logic, a crucial role is played by the logical interpretation of the (Promotion) rule. Following [Möl02], it seems that (Promotion) may be seen as a form of reflection scheme. Similarly as in the complementation construction of [Tho97, Thm. 6.9], such reflection scheme would simply say that, because they can be assumed to be positional, realizers can be seen as labeled $\mathfrak{D}$-ary trees. This would simply amount to the fact that predicates of the form $\exists\sigma(\sigma : \mathcal{A} \multimap \mathcal{B})$ are definable in $\mathsf{MSO}$.

## References

[Abr97] S. Abramsky. Semantics of Interaction. In A. M. Pitts and P. Dybjer, editors, *Semantics and Logics of Computation*, volume 14 of *Publications of the Newton Institute*, page 1. Cambridge University Press, 1997. 2, 13, 15, 16, 17, 19, 35

[AC98] R. M. Amadio and P.-L. Curien. *Domains and Lambda-Calculi.* Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1998. 2, 7, 13, 17, 24, 25, 61, 65

[AF98] J. Avigad and S. Feferman. Gödel's functional ("Dialectica") interpretation. In S. Buss, editor, *Handbook Proof Theory*, volume 137 of *Studies in Logic and the Foundations of Mathematics*, pages 337–405. Elsevier, 1998. 55, 64

[AN07] A. Arnold and D. Niwinski. Continuous Separation of Game Languages. *Fundam. Inform.*, 81(1-3):19–28, 2007. 65

[Arn99] A. Arnold. The $\mu$-calculus alternation-depth hierarchy is strict on binary trees. *ITA*, 33(4/5):329–340, 1999. 65

[BCL08] A. Blumensath, T. Colcombet, and C. Löding. Logical theories and compatible operations. In *Logic and Automata: History and Perspectives [in Honor of Wolfgang*

*Thomas]*., volume 2 of *Texts in Logic and Games*, pages 73–106. Amsterdam University Press, 2008. 65

[BdRV02] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2002. 1

[BGG97] E. Börger, E. Grädel, and Y. Gurevich. *The Classical Decision Problem*. Perspectives in Mathematical Logic. Springer, 1997. 1

[BL69] J. R. Büchi and L. H. Landweber. Solving Sequential Conditions by Finite-State Strategies. *Transation of the American Mathematical Society*, 138:367–378, 1969. 7, 55, 59, 65

[Blu13] A. Blumensath. An algebraic proof of rabin's tree theorem. *Theor. Comput. Sci.*, 478:1–21, 2013. 2

[BMSS12] L. Birkedal, R. E. Møgelberg, J. Schwinghammer, and K. Støvring. First steps in synthetic guarded domain theory: step-indexing in the topos of trees. *Logical Methods in Computer Science*, 8(4), 2012. 64

[CL07] A. Carayol and C. Löding. MSO on the Infinite Binary Tree: Choice and Order. In *CSL*, volume 4646 of *Lecture Notes in Computer Science*, pages 161–176. Springer, 2007. 64

[CL08] T. Colcombet and C. Löding. The Non-deterministic Mostowski Hierarchy and Distance-Parity Automata. In *ICALP 2008*, volume 5126 of *Lecture Notes in Computer Science*, pages 398–409. Springer, 2008. 7, 11, 57, 64

[dP91] V. de Paiva. The Dialectica categories. Technical Report 213, University of Cambridge Computer Laboratory, January 1991. 64

[EJ91] E. A. Emerson and C. S. Jutla. Tree Automata, Mu-Calculus and Determinacy (Extended Abstract). In *FOCS*, pages 368–377. IEEE Computer Society, 1991. 4, 26, 57, 60, 64

[FMS13] A. Facchini, F. Murlak, and M. Skrzypczak. Rabin-Mostowski Index Problem: A Step beyond Deterministic Automata. In *LICS*, pages 499–508. IEEE Computer Society, 2013. 65

[GH82] Y. Gurevich and L. Harrington. Trees, Automata, and Games. In *STOC*, pages 60–65. ACM, 1982. 4

[Gir87] J.-Y. Girard. Linear Logic. *Theoretical Computer Science*, 50:1–102, 1987. 2, 24, 25, 56, 64

[GLT89] J.-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1989. 2

[GTW02] E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, volume 2500 of *LNCS*. Springer, 2002. 2, 14, 15, 61, 66

[HHM07] R. Harmer, M Hyland, and P.-A. Melliès. Categorical combinatorics for innocent strategies. In *LICS 2007*, pages 379–388, 2007. 40

[HO00] J. M. E. Hyland and C.-H. Ong. On Full Abstraction for PCF: I, II, and III. *Information and Computation*, 163(2):285–408, 2000. 15, 65

[Hof11] P. J. W. Hofstra. The dialectica monad and its cousins. In M. Makkai and B.T. Hart, editors, *Models, Logics, and Higher-dimensional Categories: A Tribute to the Work of Mihály Makkai*, CRM proceedings & lecture notes. American Mathematical Society, 2011. 41, 64

[HP12] T. Hirschowitz and D. Pous. Innocent Strategies as Presheaves and Interactive Equivalences for CCS. *Sci. Ann. Comp. Sci.*, 22(1):147–199, 2012. 66

[HS99] J. M. E. Hyland and A. Schalk. Abstract Games for Linear Logic. *Electr. Notes Theor. Comput. Sci.*, 29:127–150, 1999. 33, 35, 36, 39, 64

[HS03] J. M. E. Hyland and A. Schalk. Glueing and orthogonality for models of linear logic. *Theoretical Computer Science*, 294(1/2):183–231, 2003. 36, 39, 64

[Hyl97] J. M. E. Hyland. Game Semantics. In A. M. Pitts and P. Dybjer, editors, *Semantics and Logics of Computation*, volume 14 of *Publications of the Newton Institute*, page 131. Cambridge University Press, 1997. 2, 13, 15, 16, 17, 19, 35

[Hyl02] J. M. E. Hyland. Proof theory in the abstract. *Ann. Pure Appl. Logic*, 114(1-3):43–78, 2002. 41, 64

[Jac01] B. Jacobs. *Categorical Logic and Type Theory.* Studies in logic and the foundations of mathematics. Elsevier, 2001. 2, 7, 8, 26, 36, 37, 41, 49, 54, 58, 64

[Jut97] C. S. Jutla. Determinization and Memoryless Winning Strategies. *Inf. Comput.*, 133(2):117–134, 1997. 61, 62, 65

[KK95] N. Klarlund and D. Kozen. Rabin Measures. *Chicago J. Theor. Comput. Sci.*, 1995, 1995. 61, 62, 65

[Kla94] N. Klarlund. Progress measures, immediate determinacy, and a subset construction for tree automata. *Annals of Pure and Applied Logic*, 69(2-3):243–268, 1994. 61, 62, 65

[Koh08] U. Kohlenbach. *Applied Proof Theory: Proof Interpretations and their Use in Mathematics.* Springer Monographs in Mathematics. Springer, 2008. 6, 55, 64

[LS86] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic.* CUP, 1986. 2, 7, 8

[Mel04] P.-A. Melliès. Comparing hierarchies of types in models of linear logic. *Inf. Comput.*, 189(2):202–234, 2004. 61

[Mel05] P.-A. Melliès. Sequential algorithms and strongly stable functions. *Theoretical Computer Science*, 343(1-2):237–281, 2005. 13, 15, 65

[Mel06] P.-A. Melliès. Asynchronous games 2: The true concurrency of innocence. *Theor. Comput. Sci.*, 358(2-3):200–228, 2006. 15, 65

[Mel09] P.-A. Melliès. Categorical semantics of linear logic. In *Interactive models of computation and program behaviour*, volume 27 of *Panoramas et Synthèses*. SMF, 2009. 2, 7, 24, 25, 28, 39, 43, 44, 47, 51, 57, 58, 60, 61

[Mel12] P.-A. Melliès. Game Semantics in String Diagrams. In *LICS*, pages 481–490. IEEE, 2012. 15

[Mel13] P.-A. Melliès. On dialogue games and coherent strategies. In *CSL*, volume 23 of *LIPIcs*, pages 540–562. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013. 27, 53

[ML98] S. Mac Lane. *Categories for the Working Mathematician*. Springer, 2nd edition, 1998. 47

[Möl02] M. Möllerfeld. *Generalized Inductive Definitions - The $\mu$-calculus and $\Pi^1_2$-comprehension*. PhD thesis, Westfälischen Wilhelms-Universität Münster, 2002. Available at https://miami.uni-muenster.de/Record/9dfa74b6-186b-4e95-a51f-9965d7e1e508. 66

[MS87] D. E. Muller and P. E. Schupp. Alternating Automata on Infinite Trees. *Theor. Comput. Sci.*, 54:267–276, 1987. 4, 26, 53, 57, 60, 64

[MS95] D. E. Muller and P. E Schupp. Simulating Alternating Tree Automata by Nondeterministic Automata: New Results and New Proofs of the Theorems of Rabin, McNaughton and Safra. *Theor. Comput. Sci.*, 141(1&2):69–107, 1995. 4, 26, 57, 60, 64

[MTT09] P.-A. Melliès, N. Tabareau, and C. Tasson. An Explicit Formula for the Free Exponential Modality of Linear Logic. In *Proceedings of ICALP'09*, volume 5556 of *Lecture Notes in Computer Science*, pages 247–260. Springer, 2009. 65

[Ong06] C.-H. L. Ong. On Model-Checking Trees Generated by Higher-Order Recursion Schemes. In *Proceedings of LICS'06*, pages 81–90. IEEE Computer Society, 2006. 13

[PP04] D. Perrin and J.-É. Pin. *Infinite Words: Automata, Semigroups, Logic and Games*. Pure and Applied Mathematics. Elsevier, 2004. 2, 14, 15, 53

[PR17] P. Pradic and C. Riba. A Curry-Howard Approach to Church's Synthesis. Submitted, 2017. 64, 66

[Rab69] M. O. Rabin. Decidability of Second-Order Theories and Automata on Infinite Trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969. 1, 3, 4, 56, 61, 63

[Rab72] M. O. Rabin. Automata on infinite objects and Church's Problem. *Amer. Math. Soc.*, 1972. 7, 59

[Rib15] C. Riba. Fibrations of tree automata. In *TLCA*, volume 38 of *LIPIcs*, pages 302–316. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015. 2, 4, 5, 6, 7, 9, 15, 17, 19, 25, 26, 27, 31, 32, 33, 36, 37, 64, 71

[Rib17] C. Riba. Monoidal-Closed Categories of Tree Automata. Available on HAL (hal-01261183) https://hal.archives-ouvertes.fr/hal-01261183, 2017. 2, 7, 27, 34, 44, 46, 47, 53, 57, 62

[SA05] L. Santocanale and A. Arnold. Ambiguous classes in mu-calculi hierarchies. *Theor. Comput. Sci.*, 333(1-2):265–296, 2005. 7, 57, 64

[Shu08] M. Shulman. Framed bicategories and monoidal fibrations. *Theory and Applications of Categories*, 20(18):650–738, 2008. 21

[SU06] M. H. Sørensen and P. Urzyczyn. *Lectures on the Curry-Howard Isomorphism*, volume 149 of *Studies in Logic and the Foundations of Mathematics*. Elsevier Science Inc., 2006. 2, 6

[Tho97] W. Thomas. Languages, Automata, and Logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume III, pages 389–455. Springer, 1997. 2, 7, 14, 15, 53, 55, 59, 61, 65, 66

[TO15] T. Tsukada and C.-H. L. Ong. Nondeterminism in Game Semantics via Sheaves. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015*, pages 220–231. IEEE Computer Society, 2015. 66

[VW08] M. Y. Vardi and T. Wilke. Automata: from logics to algorithms. In *Logic and Automata*, volume 2 of *Texts in Logic and Games*, pages 629–736. Amsterdam University Press, 2008. 1

[Wal02] I. Walukiewicz. Monadic second-order logic on tree-like structures. *Theor. Comput. Sci.*, 275(1-2):311–346, 2002. 4, 27, 60, 61, 62, 65

[Zie98] Z. Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science*, 200(1-2):135–183, 1998. 61, 62, 65, 66

# Contents