Mathematical Structures in Computer Science (2020), **30**, pp. 62–117
doi:10.1017/S0960129519000173

CAMBRIDGE
UNIVERSITY PRESS

PAPER

Monoidal-closed categories of tree automata

Colin Riba

Université de Lyon, EnsL, UCBL, LIP, F-69342, LYON Cedex 07, France
Email: colin.riba@ens-lyon.fr

(Received 7 May 2017; revised 11 March 2019; accepted 19 September 2019)

Abstract

This paper surveys a new perspective on tree automata and Monadic second-order logic (MSO) on infinite trees. We show that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Tree Theorem (the decidability of MSO) correspond to the connectives of *Intuitionistic Multiplicative Exponential Linear Logic* (IMELL). Namely, we equip a variant of usual alternating tree automata (that we call *uniform* tree automata) with a fibered monoidal-closed structure which in particular handles a linear complementation of alternating automata. Moreover, this monoidal structure is actually Cartesian on *non-deterministic* automata, and an adaptation of a usual construction for the simulation of alternating automata by non-deterministic ones satisfies the deduction rules of the $!(-)$ exponential modality of IMELL. (But this operation is unfortunately not a functor because it does not preserve composition.) Our model of IMLL consists in categories of games which are based on usual categories of two-player linear sequential games called *simple games*, and which generalize usual acceptance games of tree automata. This model provides a realizability semantics, along the lines of Curry–Howard proofs-as-programs correspondence, of a linear constructive deduction system for tree automata. This realizability semantics, which can be summarized with the slogan "automata as objects, strategies as morphisms," satisfies an expected property of witness extraction from proofs of existential statements. Moreover, it makes it possible to combine realizers produced as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata.

Keywords: Automata on infinite trees; monadic second-order logic; linear logic; curry-Howard correspondence; categorical logic

1. Introduction

Monadic second-order logic (MSO) on infinite trees is a rich system, which contains non-trivial mathematical theories (see, e.g., Börger et al. 1997; Rabin 1969), and which subsumes many logics, in particular modal logics (see, e.g., Blackburn et al. 2002) and logics for verification (see, e.g., Vardi and Wilke 2008). Rabin's Tree Theorem (1969), the decidability of MSO on infinite trees, is an "important and difficult decidability theorem for mathematical theories" (Börger et al. 1997, Section 1.3, p. 11).

The original proof of Rabin (1969) relied on an effective translation of formulae to finite-state automata running on infinite trees. Since then, there has been considerable work on Rabin's Tree Theorem, culminating in streamlined decidability proofs, as presented, for example, in Grädel et al. (2002), Perrin and Pin (2004), Thomas (1997). Most current approaches to MSO on infinite trees are based on translations of MSO-formulae to automata.[1]

In this paper, we show that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Tree Theorem correspond to the connectives of

$$(\textsc{Exchange})\ \frac{M\ ;\ \overline{\mathcal{A}},\,\mathcal{A},\,\mathcal{B},\,\overline{\mathcal{C}}\vdash\mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\,\mathcal{B},\,\mathcal{A},\,\overline{\mathcal{C}}\vdash\mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}}\vdash\mathcal{A}}{M\circ M'\ ;\ \overline{\mathcal{A}}\vdash\mathcal{A}}\ (\textsc{Subst})$$

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

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

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

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

$$(\textsc{Dereliction})\ \frac{M\ ;\ \overline{\mathcal{A}}\,,\,\mathcal{A}\,,\,\overline{\mathcal{B}}\vdash\mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\,,\,!\mathcal{A}\,,\,\overline{\mathcal{B}}\vdash\mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{N}}\vdash\mathcal{A}}{M\ ;\ \overline{\mathcal{N}}\vdash\,!\mathcal{A}}\ (\textsc{Promotion})$$

$$(\textsc{Weak}_{\textsc{ND}})\ \frac{M\ ;\ \overline{\mathcal{A}}\,,\,\overline{\mathcal{B}}\vdash\mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\,,\,\mathcal{N}\,,\,\overline{\mathcal{B}}\vdash\mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}}\,,\,\mathcal{N}\,,\,\mathcal{N}\,,\,\overline{\mathcal{B}}\vdash\mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\,,\,\mathcal{N}\,,\,\overline{\mathcal{B}}\vdash\mathcal{C}}\ (\textsc{Contr}_{\textsc{ND}})$$

$$(\textsc{Left}\ \exists)\ \frac{M\times\mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]},\,\mathcal{B}\vdash\mathcal{A}[\pi]}{M\ ;\ \overline{\mathcal{A}},\,\exists_\Gamma\mathcal{B}\vdash\mathcal{A}} \qquad \frac{M\times N\ ;\ \overline{\mathcal{A}}\vdash\mathcal{A}}{M\times N\ ;\ \overline{\mathcal{A}}\vdash(\exists_\Gamma\mathcal{A})[\pi]}\ (\textsc{Right}\ \exists)$$

$$(\textsc{Left}\ \forall)\ \frac{M\times N\ ;\ \overline{\mathcal{A}},\,\mathcal{B}\vdash\mathcal{A}}{M\times N\ ;\ \overline{\mathcal{A}},\,(\forall_\Gamma\mathcal{B})[\pi]\vdash\mathcal{A}} \qquad \frac{M\times\mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]}\vdash\mathcal{A}}{M\ ;\ \overline{\mathcal{A}}\vdash\forall_\Gamma A}\ (\textsc{Right}\ \forall)$$

Figure 1. Deduction rules on automata of Figure 15 (Section 5.3), Figure 16 (Section 6.3), and Figure 19 (Section 7.2.4), where M, M' are composable, $\mathcal{N}, \overline{\mathcal{N}}$ are non-deterministic, and where the weakening functor $(-)[\pi]$ takes automata over Σ to automata over $\Sigma \times \Gamma$.

Intuitionistic Multiplicative Exponential Linear Logic (IMELL) (Girard 1987). Namely, we equip a variant of usual alternating tree automata (that we call *uniform* tree automata) with a fibered monoidal-closed structure which in particular (via determinacy of ω-regular games) handles a linear complementation of alternating automata. Moreover, this monoidal structure is actually Cartesian on *non-deterministic* automata, and an adaptation of a usual construction for the simulation of alternating automata by non-deterministic ones satisfies the deduction rules of the $!(-)$ exponential modality of IMELL. (But this operation is unfortunately not a functor because it does not preserve composition.)

Our model of IMLL consists in categories of games which are based on usual categories of two-player linear sequential games called *simple games* (see, e.g., Abramsky 1997; Hyland 1997), and which generalize usual acceptance games of tree automata.[2] This model provides a realizability semantics, along the lines of the Curry–Howard proofs-as-programs correspondence (see, e.g., Girard et al. 1989; Sørensen and Urzyczyn 2006), of a linear constructive deduction system for tree automata (see Figure 1). This realizability semantics, which can be summarized with the slogan "automata as objects, strategies as morphisms," satisfies an expected property of witness extraction from proofs of existential statements. Moreover, it makes it possible to combine realizers obtained as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata.

Our motivation for this deduction system is that even if Rabin's Tree Theorem proves the existence of decision procedures for MSO on infinite trees, there is (as far as we know) no working implementation of such procedures. The reason is that all known translations of formulae to tree automata involve at some stage the determinization of automata on ω-words (McNaughton's Theorem 1966), which is believed not to be amenable to tractable implementation (see, e.g., Kupferman and Vardi 2005). We instead target semi-automatic approaches in which the user can interactively perform some proofs steps and can delegate sufficiently simple subgoals to automatic non-emptiness checkers (solving parity games). The partial proof tree built by the user is then translated to a combinator able to compose the witnessing strategies obtained from the algorithms.

This work builds on Riba (2015), 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 for a clearer connection of our model with IMELL. We follow the guidelines and axiomatizations provided by categorical logic and categorical approaches to the Curry–Howard correspondence, for which we refer to Jacobs (2001), Lambek and Scott (1986), and Amadio and Curien (1998). We moreover refer to Melliès (2009) for a comprehensive presentation of categorical axiomatizations of models of (subsystems of) linear logic. The present paper is a slightly shortened version of Riba (2018). In the remaining of this Introduction, we sketch some key points of Rabin's Tree Theorem (Sections 1.1 and 1.2) and then outline the main aspects of our decomposition of MSO in IMELL and the corresponding realizability interpretation (Sections 1.3–1.5).

1.1 MSO and (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 ϵ 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 Σ, Γ, etc.) are finite non-empty sets, and Σ-labeled $\mathfrak{D}$-ary trees are functions $T : \mathfrak{D}^* \to \Sigma$. Throughout the paper, we shall denote with overlines both vectors and finite words, so that, for example, $\overline{T}$ denotes a sequence $\overline{T} = T_1, \ldots, T_n$, while $\overline{\mathsf{a}} \in \Sigma^*$ denotes a word $\overline{\mathsf{a}} = \mathsf{a}_1. \cdots .\mathsf{a}_n$ where each a_i is a letter of Σ.

There are different expressively equivalent variants of MSO over infinite trees. The main idea is that we have a two-sorted logic, with a sort of *individuals* ranging over the positions of the full $\mathfrak{D}$-ary tree $\mathfrak{D}^*$ (i.e., over $\mathfrak{D}^*$ itself) and a sort of *monadic* second-order variables ranging over sets of positions (i.e., over $\mathcal{P}(\mathfrak{D}^*)$). When discussing translations to automata, it is actually customary and convenient (following, e.g., Thomas 1997), to only allow monadic variables, and to simulate quantifications over individuals via a (definable) singleton predicate. We shall moreover not be concerned with any particular choice of atomic predicates. We thus assume given a set At of atomic predicates. MSO-formulae are then given by

$$\varphi, \psi \quad ::= \quad \alpha \quad | \quad \bot \quad | \quad \top \quad | \quad \neg\varphi \quad | \quad \varphi \wedge \psi \quad | \quad (\exists X)\varphi \qquad\qquad (\text{where } \alpha \in \mathrm{At})$$

These formulae are interpreted in the full $\mathfrak{D}$-ary tree $\mathfrak{D}^*$ as expected, assuming an interpretation of the atomic predicates.

On the other hand, there are two families of tree automata involved in the interpretation of MSO-formulae: *non-deterministic* tree automata and *alternating* tree automata.[3] 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 of states, with a distinguished[4] initial state $q^\iota \in Q$, an acceptance condition given by an ω-regular set $\Omega \subseteq Q^\omega$, and a transition function ∂. A *non-deterministic* tree automaton $\mathcal{A}$ over Σ has a transition function of the form

$$\partial \quad : \quad Q \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathfrak{D} \longrightarrow Q)$$

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 Section 2. A *run tree* of $\mathcal{A}$ on $T : \mathfrak{D}^* \to \Sigma$ is a tree $R : \mathfrak{D}^* \to Q$ such that $R(\epsilon) = q^\imath$, and which respects the transitions of $\mathcal{A}$, in the sense that for each tree position $p \in \mathfrak{D}^*$, there exists a $\mathfrak{D}$-tuple $(q_d)_{d \in \mathfrak{D}} \in \partial(R(p), T(p))$ such that $R(p.d) = q_d$ for all $d \in \mathfrak{D}$. The run R is *accepting* if all its infinite paths belong to Ω. 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.

1.2 Games and alternating automata

The main difficulty when translating MSO-formulae to tree automata is the interplay between negation and (existential) quantification. Historically, Rabin (1969) translated MSO-formulae to *non-deterministic* tree automata. The major achievement of Rabin (1969) was to show that non-deterministic automata on infinite trees are closed under complement. This means that for every non-deterministic 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 (1969) of a complement $\sim\mathcal{A}$ from $\mathcal{A}$ has been considerably simplified by Gurevich and Harrington (1982), thanks to the notion of *acceptance game*. The idea is to model the evaluation of an automaton $\mathcal{A}$ on an input tree T as an infinite two-player game $\mathcal{G}(\mathcal{A}, T)$. In this game, the *Proponent* P (also called ∃loïse or Automaton) plays for acceptance while its *Opponent* O (also called ∀bélard or Pathfinder) plays for rejection, and $\mathcal{A}$ accepts T when P has a winning strategy. A typical (infinite) play χ in $\mathcal{G}(\mathcal{A}, T)$ has the form:

$$
\begin{array}{ccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} \\[1ex]
(q_{0,d})_{d\in\mathfrak{D}} & \cdot & d_0 & \cdot & (q_{1,d})_{d\in\mathfrak{D}} & \cdot & d_1 & \cdot \cdots \cdot & (q_{n+1,d})_{d\in\mathfrak{D}} & \cdot & d_{n+1} \cdot \cdots \\
\Cap & & \Cap & & \Cap & & \Cap & & \Cap & & \Cap \\
\partial(q^\imath, T(\epsilon)) & & \mathfrak{D} & & \partial(q_{0,d_0}, T(d_0)) & & \mathfrak{D} & & \partial(q_{n,d_n}, T(p)) & & \mathfrak{D}
\end{array}
$$

where $p = d_0 \cdot \ldots \cdot d_n$. Then χ is winning for P if the sequence of states $q^\imath, q_{0,d_0}, q_{1,d_1}, \ldots$ belongs to Ω; otherwise it is winning for O. Note that P chooses transitions $(q_d)_{d\in\mathfrak{D}}$ while O chooses tree directions $d \in \mathfrak{D}$. Hence, there is a bijection between accepting runs $R \in \mathcal{A}(T)$ and winning P-strategies in $\mathcal{G}(\mathcal{A}, T)$. Since acceptance games are determined, $\mathcal{A}$ does not accept T precisely when O has a winning strategy in $\mathcal{G}(\mathcal{A}, T)$. Gurevich and Harrington (1982) show that in acceptance games, winning strategies can be assumed to be finite-state w.r.t. game positions of the form $(p, q) \in \mathfrak{D}^* \times Q$, that is to only depend on a finite memory in addition to the game positions in $\mathfrak{D}^* \times Q$.[5] This makes it possible to devise an automaton $\sim\mathcal{A}$ which, using a usual projection operation, non-deterministically checks for the existence of winning O-strategies.

However, the construction of $\sim\mathcal{A}$ is still not trivial because the roles of P and 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 Emerson and Jutla (1991) and Muller and Schupp (1987, 1995) it is known that the construction of $\sim\mathcal{A}$ can be neatly decomposed using *alternating* automata. The original idea, as stated in, for example, Muller and Schupp (1987, 1995), is for an alternating automaton $\mathcal{A}$ with state set Q to have transitions with values in the free distributive lattice over $Q \times \mathfrak{D}$. But recall from, for example (Johnstone 1986, Lemma I.4.8) that free distributive lattices are given by irredundant disjunctive normal forms. Actually, following Walukiewicz (2002), we can give up irredundancy. We thus simply assume that transitions are of the form

$$\partial \quad : \quad Q \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathcal{P}(Q \times \mathfrak{D})) \tag{1}$$

and we read $\partial(q, \mathtt{a})$ as the disjunctive normal form

$$\bigvee_{\gamma \in \partial(q,\mathtt{a})} \; \bigwedge_{(q',d)\in\gamma} (q', d)$$

This results in acceptance games where intuitively P plays from disjunctions while O plays from conjunctions. A typical play in the acceptance game $\mathcal{G}(\mathcal{A}, T)$ with $\mathcal{A}$ alternating has the form

$$\begin{array}{cccccccc}
\mathsf{P} & \mathsf{O} & \mathsf{P} & \mathsf{O} & & \mathsf{P} & \\
\gamma_0 & \cdot\,(q_0, d_0)\,\cdot & \gamma_1 & \cdot\,(q_1, d_1)\,\cdot & \cdots\cdot & \gamma_{n+1} & \cdots \\
\Cap & \Cap & \Cap & \Cap & & \Cap & \\
\partial(q^\iota, T(\epsilon)) & \gamma_0 & \partial(q_0, T(d_0)) & \gamma_1 & & \partial(q_n, T(p)) &
\end{array}$$

Hence, P chooses relations $\gamma_k \in \mathcal{P}(Q \times \mathfrak{D})$ instead of tuples $(q_{k,d})_{d\in\mathfrak{D}}$ while O chooses pairs $(q_k, d_k) \in \gamma_k$ instead of just tree directions $d_k \in \mathfrak{D}$. The main consequence is that O may now be allowed to choose between pairs $(q'_k, d_k), (q''_k, d_k) \in \gamma_k$ with *different states* q'_k, q''_k for the same tree direction $d_k \in \mathfrak{D}$.

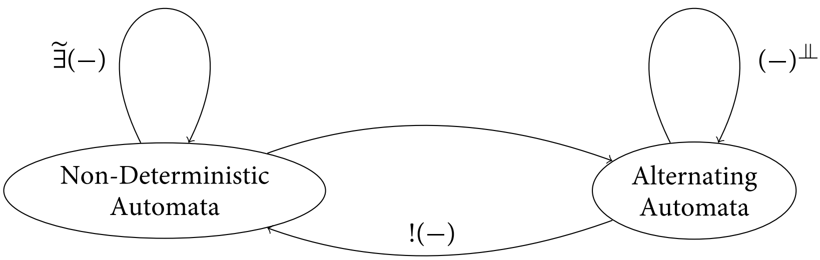
The extra possibility for O to choose states in addition to tree directions allows us to define a complement of $\mathcal{A}$ which essentially simulates $\mathcal{A}$ while reversing the roles of P and O. This can be implemented with an alternating automaton[6] $\mathcal{A}^{\perp\!\!\perp}$ having the same states as $\mathcal{A}$. The idea is that since the double powerset $\mathcal{P}(\mathcal{P}(Q \times \mathfrak{D}))$ in (1) represents disjunctive normal forms over $Q \times \mathfrak{D}$, the transition function of $\mathcal{A}^{\perp\!\!\perp}$ can just take $(q, \mathtt{a}) \in Q \times \Sigma$ to a disjunctive normal form representing the dual of $\partial(q, \mathtt{a})$. Then, if the acceptance condition of $\mathcal{A}^{\perp\!\!\perp}$ is the complement of Ω, it follows from game determinacy that $\mathcal{L}(\mathcal{A}^{\perp\!\!\perp})$ is the complement of $\mathcal{L}(\mathcal{A})$.

Every alternating automaton $\mathcal{A}$ can be simulated by a non-deterministic automaton $!\mathcal{A}$ of exponential size (this is the *Simulation Theorem*, Emerson and Jutla 1991; Muller and Schupp 1987, 1995, see also Section 7.2), while non-deterministic automata are linearly embedded into alternating automata via the obvious mapping

$$(q_d)_{d\in\mathfrak{D}} \in Q^{\mathfrak{D}} \quad\longmapsto\quad \{(q_d, d) \mid d \in \mathfrak{D}\} \in \mathcal{P}(Q \times \mathfrak{D})$$

On the other hand, non-deterministic automata are easily (and linearly in the number of states) closed under projections $\widetilde{\exists}_\Sigma(-)$ which implement the existential quantifications of MSO.

The situation can be pictured as follows:



(2)

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

$$\sim\mathcal{A} \quad=\quad !(\mathcal{A}^{\perp\!\!\perp}) \tag{3}$$

1.3 Toward linear logic

The model of Riba (2015) 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 Section 3), the extension of Riba (2015) proposed in this work shows that the decomposition depicted in (2) of the translation of MSO-formulae to non-deterministic tree automata via alternating automata corresponds to some extent to an IMELL-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 from $\mathcal{A}$ to $\mathcal{B}$ 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
$$\mathcal{A}^{\perp\!\!\!\perp} \quad \simeq \quad \mathcal{A} \multimap \perp\!\!\!\perp$$
(where $\perp\!\!\!\perp$ is a particular automaton accepting no tree), with as expected $T \in \mathcal{L}(\mathcal{A}^{\perp\!\!\!\perp})$ iff $T \notin \mathcal{L}(\mathcal{A})$.
- Second, we show that the simulation operation $!(-)$ satisfies the *deduction rules* of the usual modality $!(-)$ of IMELL. Moreover, the symmetric monoidal product $(-)\otimes(-)$ is Cartesian on non-deterministic automata, so that the picture (2) is similar to the usual linear-non-linear adjunctions in models of IMELL. (Unfortunately, in our models the operation $!(-)$ is not a functor.[7])

The connection between alternating automata and IMELL suggests that we may take variants of IMELL as intermediate steps between MSO and automata. In our setting, an IMELL-based language for MSO would consist of the following formulae:
$$\varphi, \psi \quad ::= \quad \alpha \quad | \quad \perp \quad | \quad \mathbf{I} \quad | \quad \varphi \otimes \psi \quad | \quad \varphi \multimap \psi \quad | \quad !\varphi \quad | \quad (\exists X)\varphi \quad | \quad (\forall X)\varphi$$
This language must be seen as a refinement of MSO with finer-grained connectives, which directly correspond to operations on automata (the primitive universal quantification is actually *non-standard*, see Section 6). Since the connectives of IMELL correspond to operations on automata, provided a (non-deterministic) automaton $\mathcal{A}(\alpha)$ is given for each atomic formula $\alpha \in \mathrm{At}$, one can (inductively) associate an automaton $\mathcal{A}(\varphi)$ to each IMELL formula φ.

It would have been natural to also consider the *additive* connectives & (conjunction) and $\oplus$ (disjunction) of linear logic, which do correspond to known constructions on alternating automata. However, the expected categorical properties of these connectives would require an extension of our setting that we leave for further work. Keeping this in mind, the translation of MSO to non-deterministic automata induced by (3) factors via the map $(-)^{\mathrm{nd}} : \mathsf{MSO} \to \mathsf{IMELL}$ given by
$$\alpha^{\mathrm{nd}} := \alpha \qquad \perp^{\mathrm{nd}} := \perp \qquad \top^{\mathrm{nd}} := \mathbf{I}$$
$$(\neg\varphi)^{\mathrm{nd}} := !(\varphi^{\mathrm{nd}} \multimap \perp)$$
$$(\varphi \wedge \psi)^{\mathrm{nd}} := \varphi^{\mathrm{nd}} \otimes \psi^{\mathrm{nd}}$$
$$((\exists X)\varphi)^{\mathrm{nd}} := (\exists X)\varphi^{\mathrm{nd}}$$
while the translation of MSO to alternating automata factors via the map $(-)^{\mathrm{alt}} : \mathsf{MSO} \to \mathsf{IMELL}$ given by
$$\alpha^{\mathrm{alt}} := \alpha \qquad \perp^{\mathrm{alt}} := \perp \qquad \top^{\mathrm{alt}} := \mathbf{I}$$
$$(\neg\varphi)^{\mathrm{alt}} := \varphi^{\mathrm{alt}} \multimap \perp$$
$$(\varphi \wedge \psi)^{\mathrm{alt}} := \varphi^{\mathrm{alt}} \otimes \psi^{\mathrm{alt}}$$
$$((\exists X)\varphi)^{\mathrm{alt}} := (\exists X)!\varphi^{\mathrm{alt}}$$
The factorizations of the translations of MSO to automata via IMELL are sound in the following sense.

Proposition 1.1. *Let $(-)^{\dagger}$ be either $(-)^{\mathrm{nd}}$ or $(-)^{\mathrm{alt}}$. A closed* MSO*-formulae φ is true in the full infinite $\mathfrak{D}$-ary tree if and only if $\mathcal{A}(\varphi^{\dagger})$ accepts the unique $\mathbf{1}$-labeled tree.*

1.4 Computational interpretation of proofs

In our view, proposing IMELL as an intermediate system between MSO and automata should rely on a suitable computational interpretation of proofs, along the lines of the Curry–Howard proofs-as-programs correspondence. We explain here our view that the relevant computational objects are runs of automata or P-strategies in acceptance games. This leads us to the slogan "automata as objects, strategies as morphisms," and implies that we consider a deduction system for *automata* rather than IMELL formulae.

Our deduction system manipulates sequents of the form

$$T\,;\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B} \tag{4}$$

where T is an infinite tree labeled over (say) the alphabet Σ, and $\mathcal{A}_1,\ldots,\mathcal{A}_n,\mathcal{B}$ are tree automata over Σ. We see these sequents with two different levels of interpretation. The first level interprets *provability*: if the sequent (4) is provable, then the automaton $\mathcal{B}$ accepts the tree T as soon as the automata $\mathcal{A}_1,\ldots,\mathcal{A}_n$ all accept T.

The second level is the *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 Γ defined as $\mathcal{A}$ but with transition function

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

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 Σ-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 $p\in\mathfrak{D}^*\longmapsto(q_d)_{q\in\mathfrak{D}}\in\partial_\mathcal{A}(R(p),T(p))$. It follows that given an accepting run R of $\widetilde{\exists}_\Sigma\mathcal{B}$ on $\mathbf{1}$, then from the induced function

$$p\in\mathfrak{D}^* \quad\longmapsto\quad (q_d)_{d\in\mathfrak{D}}\in\bigcup_{\mathtt{a}\in\Sigma}\partial_\mathcal{B}(R(p),\mathtt{a})$$

one can get a Σ-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. We will however rather rely on the more complex notions of acceptance games and strategies. There are two reasons for this choice. First, as discussed in Section 1.2 above, games give a smooth treatment of complementation of tree automata. The second reason, which we motivate with more details in Section 2, is that games and strategies are equipped with well-known categorical structures, which allow to easily define compositional interpretations of proofs.

Following the methodology of categorical logic, the categories proposed here and in Riba (2015) are *indexed* (or *fibered*) over a base category $\mathbf{T}$ of trees, whose objects are alphabets and whose morphisms from Σ to Γ induce functions from Σ-labeled trees to Γ-labeled trees (see Sections 2.2 and 4). In this setting, existential quantifications (in the categorical sense) are provided by a slight modification (denoted $\exists(-)$) of the usual projection $\widetilde{\exists}(-)$.

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 IMELL, is to provide realizability interpretations of MSO (in the spirit of, e.g., Kohlenbach 2008; Sørensen and Urzyczyn 2006). We think that the model presented here (consolidating Riba 2015) is a preliminary step toward this goal. Let us briefly describe our main results in this direction.

Generalizing (4), our deduction system also manipulates sequents of the form

$$M\,;\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B} \tag{5}$$

(see Section 2.2) where M is a $\mathbf{T}$-morphism, from say Σ to Γ and the automata $\mathcal{A}_1,\ldots,\mathcal{A}_n,\mathcal{B}$ have input alphabet Γ. In case M is the identity $\mathbf{T}$-map on Σ, the sequent (5) is written

$$\Sigma\,;\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B} \tag{6}$$

which in contrast with (4) and (5) does not mention any tree. The full system is presented in Figure 1, and we can state a second soundness result.

Proposition 1.2. *Given* IMELL*-formulae $\varphi_1,\ldots,\varphi_n,\varphi$, we have*

$$\varphi_1,\ \ldots,\ \varphi_n \vdash_{\mathsf{IMELL}} \varphi \qquad \Longrightarrow \qquad \mathbf{2}^p\,;\mathcal{A}(\varphi_1),\ \ldots,\ \mathcal{A}(\varphi_n) \vdash \mathcal{A}(\varphi)$$

(where $\varphi_1,\ldots,\varphi_n,\varphi$ have free variables among $X_1,\ldots,X_p$).

The symmetric monoidal-closed structure, together with the categorical quantifiers and the interpretation of simulation as an exponential modality $!(-)$, allows us to interpret proofs in the deduction system of Figure 1. From a proof $\mathscr{D}$ of a sequent (6), one can (compositionally w.r.t. the structure of $\mathscr{D}$) extract a winning finite-state strategy σ in an infinite game of the form $\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B}$. As indicated in Section 1.3, $\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n$ essentially evaluates the automata $\mathcal{A}_1,\ldots,\mathcal{A}_n$ in parallel, while the linear arrow $\multimap$ is a synchronous restriction of the usual linear arrow of simple games. When we have a strategy σ winning in $\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B}$, we say that σ is a *realizer* and write

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

In case (6) is of the form $\mathbf{1}\,; \vdash \exists_\Sigma \mathcal{N}$ (with $\mathcal{N}$ non-deterministic), we indeed obtain a computational interpretation of proofs in the sense of Section 1.4, since as shown in Section 7.1.2, we have

$$\sigma \Vdash \exists_\Sigma \mathcal{N} \qquad \Longleftrightarrow \qquad \sigma = \langle T, \tau\rangle \quad \text{where} \quad T : \mathfrak{D}^* \to \Sigma \quad \text{and} \quad \tau \Vdash \mathcal{N}(T)$$

Assume now that (6) is of the form

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

Then a Σ-labeled tree T induces a *substitution functor* $T^\star$, whose action on σ gives a function $T^\star(\sigma)$ taking any winning P-strategy τ on $\mathcal{A}(T)$ to a winning P-strategy $T^\star(\sigma) \circ \tau$ on $\mathcal{B}(T)$ (see Proposition 4.11). This gives the rule

$$\frac{\sigma \Vdash \mathcal{A} \multimap \mathcal{B}}{T^\star(\sigma) \Vdash \mathcal{A}(T) \multimap \mathcal{B}(T)}$$

and the function

$$\tau \Vdash \mathcal{A}(T) \qquad \longmapsto \qquad T^\star(\sigma) \circ \tau \Vdash \mathcal{B}(T)$$

In other words, realizers of sequents of the form (7) can be composed (via substitution) with strategies τ on $\mathcal{A}(T)$ obtained by any possible mean.

To summarize, we get the following soundness result.

Theorem 1.3. *Given automata $\overline{\mathcal{A}}, \mathcal{A}$ over Σ,*

$$\Sigma\,;\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{A} \quad\Longrightarrow\quad \Vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{A} \tag{8}$$

$$\Vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{A} \quad\Longrightarrow\quad \mathcal{L}(\mathcal{A}_1) \cap \cdots \cap \mathcal{L}(\mathcal{A}_n) \subseteq \mathcal{L}(\mathcal{A}) \tag{9}$$

More generally, the methodology behind 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. Our motivation is that even if Rabin's Tree Theorem proves the existence of decision procedures for MSO on infinite trees, there is (as far as we know) no working implementation of such procedures. The reason is that all known translations of formulae to tree automata involve at some stage the determinization of automata on ω-words (McNaughton's Theorem 1966), which is believed not to be amenable to tractable implementation (see, e.g., Kupferman and Vardi 2005). We instead target semi-automatic approaches in which the user can delegate sufficiently simple subgoals to automatic non-emptiness checkers (solving parity games). The partial proof tree built by the user is then translated to a combinator able to compose the strategies obtained by the algorithms.[8] To this end, some relevant properties of our framework are the following.

First, thanks to the (non-standard) primitive universal quantifications, games of the form $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ are equivalent to games of the form $\mathbf{1} \vdash \forall_\Sigma(\mathcal{A} \multimap \mathcal{B})$, which are themselves equivalent to ω-regular games on finite graphs. Thanks to the Büchi–Landweber Theorem (1969), one can thus decide if there is a strategy realizing the implication $\mathcal{A} \multimap \mathcal{B}$, and if such a strategy exists, then there exists a finite-state one, which is moreover effectively computable from $\mathcal{A}$ and $\mathcal{B}$ (see Corollary 6.5).

Second, following Example 6.10, our system can be extended with the rule

$$\frac{\mathcal{L}(\mathcal{A} : \mathbf{1}) \neq \emptyset}{\mathbf{1}\,;\ \vdash \mathcal{A}}$$

This rule has the following consequences:

(1) Assuming

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

(over say Σ) is realizable, following the same reasoning as for Corollary 6.5, we get (leaving implicit some structural and cut rules)

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{L}(\forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})) \neq \emptyset}{\mathbf{1}\,;\ \vdash \forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})}}{\Sigma\,;\ \vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B}}}{\Sigma\,;\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \vdash \mathcal{B}}}{\Sigma\,;\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B}}$$

This entails the rules of Example 5.9, and in particular allows us to derive the general (WEAK) rule

$$\frac{M\,;\overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\,;\overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}$$

(2) Given $\mathcal{A}, \mathcal{B} : \Sigma$ non-deterministic we have

$$\frac{\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset}{\Sigma\,;\mathcal{A} \vdash \mathcal{B}^{\perp}}$$

Indeed, from $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$ we can derive (again leaving implicit some structural and (CUT) rules)

$$\begin{array}{c}
\mathcal{L}\big(\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \multimap \bot\big) \neq \emptyset \\
\hline
\mathbf{1} \, ; \, \vdash \exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \multimap \bot \\
\hline
\mathbf{1} \, ; \exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \vdash \bot \\
\hline
\Sigma \, ; \mathcal{A} \otimes \mathcal{B} \vdash \bot \\
\hline
\Sigma \, ; \mathcal{A} \, , \, \mathcal{B} \vdash \bot \\
\hline
\Sigma \, ; \mathcal{A} \vdash \mathcal{B}^{\bot}
\end{array}$$

Moreover any (finite-state) O-strategy witnessing $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \emptyset$ can be lifted to a (finite-state) realizer of $\mathcal{A} \multimap \mathcal{B}^{\bot}$ (Proposition 7.7).

(3) In particular, for $\mathcal{A}, \mathcal{B} : \Sigma$ not-necessarily non-deterministic, we have

$$\frac{\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})}{\Sigma \, ; \, !\mathcal{A} \vdash ?\mathcal{B}}$$

where $?(-) := !((-) \multimap \bot) \multimap \bot$ (Proposition 7.17).

1.6 Outline

The paper is organized as follows. In Section 2, we expose some ingredients and methodology of our approach based on categorical logic, and we sketch the connection between IMELL and the interpretation of MSO in usual tree automata. We then turn in Section 3 to our notion of *uniform automata* (motivated by monoidal closure), and present basic material on zig-zag games required for our setting. Section 4 then deals with the fibered structure (which is essentially a refinement of Riba 2015), Section 5 presents the monoidal closure and the corresponding deduction rules, while Section 6 deals with quantifications. Finally, in Section 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 $!(-)$ IMELL-exponential modalities. Furthermore, Riba (2018, Appendix A) provides connections with usual game semantics, and additional examples showing that our setting can handle constructions of Colcombet and Löding (2008), Santocanale and Arnold (2005) are presented in Riba (2018, Appendix C).

2. Toward Categories of Games and Automata

The purpose of this section is twofold. First, in Sections 2.1–2.3, we expose some ingredients and methodology of our approach based on categorical logic. We state in Section 2.1 the minimal requirements imposed by categorical semantics of proofs, and Section 2.2 presents some basic ideas and motivations on indexed categories for modeling free variables and quantifications. Finally, in Section 2.3 we explain why it seems difficult to obtain a suitable categorical semantics of implications based on usual connectives on automata. Second, building on Riba (2015), in Sections 2.4 and 2.5 we sketch the connection between linear logic and the interpretation of MSO in tree automata mentioned in Section 1.3.

2.1 Compositionality and categorical semantics

The method of categorical semantics of proofs (see, e.g., Amadio and Curien 1998; Jacobs 2001; Lambek and Scott 1986; Melliès 2009) is to interpret *proofs* in a deduction system as *morphisms* in a category $\mathbb{C}$, such that $\mathbb{C}$ is equipped with some structure corresponding to the connectives and rules of the deduction system. For the moment, let us step back from acceptance games and consider run trees. Our task is thus to devise categories whose objects include all sets of the form

$\mathcal{A}(T)$, for an automaton $\mathcal{A}$ and a tree T, and such that the proofs of a sequent $T\,;\mathcal{A}\vdash\mathcal{B}$ can be interpreted as morphisms from $\mathcal{A}(T)$ to $\mathcal{B}(T)$.

The first requirement 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

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

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

$$f\circ\mathrm{id}_A \;=\; f \;=\; \mathrm{id}_B\circ f \qquad \text{for every } f\in\mathbb{C}[A,B] \tag{10}$$

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 (10) imply, for instance, that the three derivations below must be interpreted by the same morphism:

$$\frac{\dfrac{}{T\,;\mathcal{A}\vdash\mathcal{A}} \qquad \dfrac{\mathscr{D}}{T\,;\mathcal{A}\vdash\mathcal{B}}}{T\,;\mathcal{A}\vdash\mathcal{B}} \qquad \frac{\mathscr{D}}{T\,;\mathcal{A}\vdash\mathcal{B}} \qquad \frac{\dfrac{\mathscr{D}}{T\,;\mathcal{A}\vdash\mathcal{B}} \qquad \dfrac{}{T\,;\mathcal{B}\vdash\mathcal{B}}}{T\,;\mathcal{A}\vdash\mathcal{B}} \tag{11}$$

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., Jacobs 2001; Lambek and Scott 1986).

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

$$\frac{(\exists x)\varphi(x)\vdash\psi}{\varphi(x)\vdash\psi}\ \text{(double line)} \qquad (x \text{ not free in } \psi) \tag{12}$$

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 φ. As a consequence, in order to properly formulate (12) one should be able to interpret sequents of the form $\varphi(x)\vdash\psi$ with free variables. More generally, the formulae φ and ψ should be allowed to contain free variables distinct from x.

The idea underlying the general method (but see, e.g., Jacobs 2001 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 $(t_1,\ldots,t_n)$ of terms t_i of sort o_i whose free variables are among $x_{\iota_1},\ldots,x_{\iota_m}$ with x_{ι_j} of sort ι_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 $x_{\iota_1},\ldots,x_{\iota_n}$, and whose morphisms interpret proofs. Furthermore, $\mathbb{B}$-morphisms

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

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 φ with free variables among $y_{o_1},\ldots,y_{o_n}$ to (the interpretation of) the formula $\varphi[t_1/y_{o_1},\ldots,t_n/y_{o_n}]$ with free variables among $x_{\iota_1},\ldots,x_{\iota_m}$. Its action on the morphisms of $\mathbb{E}_{o_1\times\cdots\times o_n}$ allows us to interpret the *substitution rule*

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

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_ι (but with no actual occurrence of x_ι). Then the proper formulation of (12) is that existential quantification over x_ι is a functor

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

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

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

(where x_ι does not occur free in ψ since ψ 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* Σ *to* Γ*, denoted* $M, N, L, \ldots,$ *are functions of the form*

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

A $\mathbf{T}$-morphism $M \in \mathbf{T}[\Sigma, \Gamma]$ thus takes for each $n \in \mathbb{N}$ a sequence of input letters $\overline{\mathtt{a}} = \mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_n \in \Sigma^{n+1}$ and a sequence of tree directions $p = d_1 \cdot \ldots \cdot d_n \in \mathfrak{D}^n$ to an output letter $M(\overline{\mathtt{a}}, p) \in \Gamma$. In particular, we have $\mathbf{T}[\mathbf{1}, \Sigma] \simeq (\mathfrak{D}^* \to \Sigma)$, so each Σ-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 Σ to Γ.[9] $\mathbf{T}$-morphisms are composed in the expected way (see Section 4.3 for details).

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 Σ. Let us sketch the general idea with runs of non-deterministic automata. Given a non-deterministic automaton $\mathcal{A}$ over Γ and a morphism $M \in \mathbf{T}[\Sigma, \Gamma]$, a Σ-*run* of $\mathcal{A}$ on M is a tree

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

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

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

supplied with input letters $\mathtt{b} \in \Gamma$ computed by M from tree positions $p = d_1 \cdot \ldots \cdot d_n$ and sequences of input letters $\overline{\mathtt{a}} = \mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_n$ where $\mathtt{a}_k$ is given by the Σ-component of $R(d_1 \cdot \ldots \cdot d_k) \in \Sigma \times Q_{\mathcal{A}}$. (So $\mathtt{a}_0$ is given by $R(\epsilon)$ and $\mathtt{a}_n$ is given by $R(p)$.) Explicitly, R is a Σ-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 $\mathfrak{D}$-tuple $(q_d)_{d\in\mathfrak{D}} \in \partial_{\mathcal{A}}(q, \mathtt{b})$ 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 q_d.

Such a Σ-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 Σ-run trees of $\mathcal{A}$ on M, and simply write $\mathcal{A}(M)$ for $\Sigma \vdash \mathcal{A}(M)$ when Σ is clear from the context.

Roughly speaking, for each Σ, 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 an $\mathbb{E}_\Sigma$-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 (4). 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 Γ. Such sequents are to be thought about as our version of "*open* sequents" or "sequents with free variables" (here of sort Σ), with the usual implicit prenex universal quantification, and are to be interpreted as morphisms in the category $\mathbb{E}_\Sigma$ (the *fiber* over Σ). Substitution functors such as $L^\star : \mathbb{E}_\Sigma \to \mathbb{E}_\Delta$ above will act in the deduction system via a substitution rule

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

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 (4)), which are to be interpreted in the category $\mathbb{E}_\mathbf{1}$ (the fiber over $\mathbf{1}$), and are to be thought about as representing *closed* statements.

(b) Given a non-deterministic automaton $\mathcal{A}$ over Σ, 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

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

Consider now another automaton $\mathcal{B}$, also over Σ. 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 Colcombet and Löding 2008). Moreover, given a Σ-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

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

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 Σ is defined as $\mathcal{A}$ but with transition function:

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

In particular:

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

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

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

for the sequent

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

where

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

is the $\mathbf{T}$-tupling of $M_1, \ldots, M_n, L$ (see Corollary 4.6) and where the π_i's and π are suitable projections:

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

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 contributions of this work is that this implication can be internalized in automata. This will lead us outside of non-deterministic automata (see Section 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 Σ there is an automaton $(\mathcal{A} \multimap \mathcal{B})$ over Σ such that for each tree $T : \mathfrak{D}^* \to \Sigma$ there is a bijection

$$\mathbb{E}_{\mathbf{1}}[\mathcal{A}(\dot{T}),\ \mathcal{B}(\dot{T})] \quad \simeq \quad \mathbf{1} \vdash (\mathcal{A} \multimap \mathcal{B})(\dot{T})$$

that we informally write as

$$\dot{T} \,;\, \mathcal{A} \vdash \mathcal{B} \qquad \simeq \qquad \mathbf{1} \vdash (\mathcal{A} \multimap \mathcal{B})(\dot{T})$$

In other words, morphisms in the interpretation of $\dot{T}\,;\mathcal{A} \vdash \mathcal{B}$ will correspond to the runs of an automaton $(\mathcal{A} \multimap \mathcal{B})$ on T. This could suggest to interpret $\dot{T}\,;\mathcal{A} \vdash \mathcal{B}$ as the runs of an automaton of the form $\sim\mathcal{A} \vee \mathcal{B}$ over T, where $\sim\mathcal{A}$ is the complement of $\mathcal{A}$ (in the sense of Section 1.2) 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 Σ and with $\mathcal{A}_i = (Q_{\mathcal{A}_i}, q^\iota_{\mathcal{A}_i}, \partial_{\mathcal{A}_i}, \Omega_{\mathcal{A}_i})$, the non-deterministic automaton $\mathcal{A}_1 \oplus \mathcal{A}_2$ over Σ is

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

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

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

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

Note that in **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 $\mathbb{E}_{\mathbf{1}}[\mathcal{A}(\dot{T}),\ \mathcal{B}(\dot{T})]$ the set of runs of $(\sim\mathcal{A} \oplus \mathcal{B})$ on T, that is, the disjoint union $\sim\mathcal{A}(T) + \mathcal{B}(T)$. Then one faces the following difficulties.

- We have to devise identity morphisms, say

$$\mathrm{id}_{\mathcal{A}(\dot{T})} \ \in \ \sim\mathcal{A}(T) + \mathcal{A}(T)$$

  One may take for $\mathrm{id}_{\mathcal{A}(\dot{T})}$ either an accepting run of $\mathcal{A}$ on T or an accepting run of $\sim\mathcal{A}$ on T. But this raises two problems. First, it may be undecidable whether a possibly non-recursive tree is accepted or rejected by a given 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* Ong 2006), there seem to be no *canonical choice* of an actual accepting run $\mathrm{id}_{\mathcal{A}(\dot{T})} \in \sim\mathcal{A}(T) + \mathcal{A}(T)$.
- It is not clear how to define composition, say

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

  Given run trees, say

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

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

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

  may seem reasonable. But each of them breaks one of the equalities between the interpretations of the derivations depicted in (11).

The methodology of linear logic may suggest here to devise a linear implication of the form

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

where $\parr$ is a dual of the direct product $\otimes$ (see Sections 1.3 and 2.4 below), relying on a Cartesian product of states and evaluating its arguments in parallel, with acceptance given by a disjunction.

However, in contrast with ω-word automata (Pradic and Riba 2018), such a connective does not seem to exist on tree automata. The reason is that the universal quantification on paths (in the definition of acceptance) does not commute with disjunction.

2.4 The (synchronous) direct product of (non-deterministic) automata

The solution to obtain a suitable *closed* structure will be provided by uniform automata, to be defined in Section 3. On the other hand, part of the program announced up to now was already completed in Riba (2015). In that work, using the notions of *substituted acceptance games* and of *synchronous linear arrow games*, we obtained categories of (usual) alternating automata fulfilling the requirements of Sections 2.1 and 2.2. Although Riba (2015) does not explicitly mention any deduction system, it does devise categorical structures allowing for parts of the linear logic-based approach mentioned in Section 1.3.

We survey here the relevant connections between Riba (2015) and IMELL, as they underlie the role of linear logic in this paper. Returning to the general case of sequents of the form

$$M\,;\mathcal{A}_1,\dots,\mathcal{A}_n\vdash\mathcal{B} \tag{17}$$

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 $\mathcal{A}_i=(Q_{\mathcal{A}_i},q^\imath_{\mathcal{A}_i},\partial_{\mathcal{A}_i},\Omega_{\mathcal{A}_i})$, both over Σ, is the non-deterministic automaton over Σ

$$\mathcal{A}_1\otimes\mathcal{A}_2 \quad:=\quad \Big(Q_{\mathcal{A}_1}\times Q_{\mathcal{A}_2}\,,\,(q^\imath_{\mathcal{A}_1},q^\imath_{\mathcal{A}_2})\,,\,\partial_{\mathcal{A}_1\otimes\mathcal{A}_2}\,,\,\Omega_{\mathcal{A}_1\otimes\mathcal{A}_2}\Big)$$

with

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

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**

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

from which we immediately get

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

For similar reasons, the direct product $(-)\otimes(-)$ on non-deterministic automata is also Cartesian in the games of Riba (2015) provided one restricts to *total* automata.[11] This Cartesian structure on total non-deterministic automata implies that we can equip them with the deduction rules of a Cartesian product, such as the following (where **I** is a unit automaton similar to that of Example 3.2.(i)):

$$\text{(Left }\otimes)\ \frac{M\,;\overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{B}}\vdash\mathcal{C}}{M\,;\overline{\mathcal{A}},\ \mathcal{A}\otimes\mathcal{B},\ \overline{\mathcal{B}}\vdash\mathcal{C}}\qquad \frac{M\,;\overline{\mathcal{A}}\vdash\mathcal{A}\qquad M\,;\overline{\mathcal{B}}\vdash\mathcal{B}}{M\,;\overline{\mathcal{A}},\ \overline{\mathcal{B}}\vdash\mathcal{A}\otimes\mathcal{B}}\ \text{(Right }\otimes)$$

$$\text{(Left }\mathbf{I})\ \frac{M\,;\overline{\mathcal{A}},\ \overline{\mathcal{B}}\vdash\mathcal{C}}{M\,;\overline{\mathcal{A}},\ \mathbf{I},\ \overline{\mathcal{B}}\vdash\mathcal{C}}\qquad\qquad \frac{}{M\,;\ \vdash\mathbf{I}}\ \text{(Right }\mathbf{I}) \tag{18}$$

together with the structural *exchange rule*:

$$\text{(Exchange)}\ \frac{M\,;\overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{C}}\vdash\mathcal{C}}{M\,;\overline{\mathcal{A}},\ \mathcal{B},\ \mathcal{A},\ \overline{\mathcal{C}}\vdash\mathcal{C}} \tag{19}$$

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

$$(\textsc{Weak})\ \frac{M\,;\overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\,;\overline{\mathcal{A}},\ \mathcal{A},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\,;\overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{A},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\,;\overline{\mathcal{A}},\ \mathcal{A},\ \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}) \tag{20}$$

and the following general (multiplicative) cut rule:

$$(\textsc{Cut})\ \frac{M\,;\overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\,;\overline{\mathcal{B}},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\,;\overline{\mathcal{B}},\ \overline{\mathcal{A}},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \tag{21}$$

To summarize, with total non-deterministic automata, the left commas in sequents of the form (17) can be internalized as a product $(\otimes, \mathbf{I})$, whose deduction rules are induced by its structure in the computational interpretation.

2.5 Alternating automata and linear logic

With respect to the context of this paper, the basic insight of linear logic (Girard 1987) is that having an explicit control on the weakening and contraction structural rules depicted in (20) 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) λ-calculi (see, e.g., Amadio and Curien 1998).

In the case of conjunction, this can be phrased as follows. First, when suppressing the structural rules (Weak) and (Contr), the rules displayed in (18) and (19) do not specify anymore a Cartesian structure for the product $(\otimes, \mathbf{I})$, but merely a symmetric monoidal structure (see, e.g., Melliès 2009 for definitions). This implies that in contrast with intuitionistic sequents, the left commas in linear 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 linear sequents by rules of the form:[12]

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

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

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

$$\frac{A_1\,,\ \dots\ ,\ \dots\ ,\ A_n \vdash B}{A_1\,,\ \dots\ ,\ !A_i\,,\ \dots\ ,\ A_n \vdash B} \qquad \frac{A_1\,,\ \dots\ ,\ !A_i\,,\ !A_i\,,\ \dots A_n \vdash B}{A_1\,,\ \dots\ ,\ !A_i\,,\ \dots\ ,\ A_n \vdash B} \tag{22}$$

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

$$\frac{A_1\,,\ \dots\ ,\ A_i\,,\ \dots\ ,\ A_n \vdash B}{A_1\,,\ \dots\ ,\ !A_i\,,\ \dots\ ,\ A_n \vdash B} \qquad \frac{!A_1\,,\ \dots\ ,\ !A_n \vdash B}{!A_1\,,\ \dots\ ,\ !A_n \vdash !B} \tag{23}$$

Then (but see also Amadio and Curien 1998; Girard 1987; Melliès 2009 for details), the categorical interpretation of proofs gives an isomorphism

$$!A \otimes !B \quad \simeq \quad !(A \,\&\, B) \tag{24}$$

which implies that an intuitionistic sequent

$$A_1\,,\ \dots\ ,\ A_n \vdash B$$

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

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

where the left commas behave as a symmetric monoidal product $(-) \otimes (-)$.

The pertinence of intuitionistic linear logic in our context comes from the fact that the product $(-) \otimes (-)$ defined in Section 2.4 on non-deterministic automata extends to (total[13]) alternating automata, but induces a symmetric monoidal product which is not Cartesian.

Given alternating automata $\mathcal{A}$ and $\mathcal{B}$ over Σ, the automaton $\mathcal{A} \otimes \mathcal{B}$ over Σ has state set $Q_{\mathcal{A}} \times Q_{\mathcal{B}}$, and evaluates $\mathcal{A}$ and $\mathcal{B}$ along common paths $p \in \mathfrak{D}^*$ (see Riba 2015 for details). Now, recall that with alternating automata, O can choose states in addition to tree directions. Hence, given a P-strategy on $(\mathcal{A} \otimes \mathcal{B})(T)$ (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 P to make different choices according to previous O-moves. In particular, some choice of P in component $\mathcal{A}$ may depend on previous O-moves in $\mathcal{B}$. (Note that this was not possible with non-deterministic automata, since $p \in \mathfrak{D}^*$ determines uniquely the previous O-moves.) So a P-strategy on $(\mathcal{A} \otimes \mathcal{B})(T)$ may not uniquely determine a pair of strategies in $\mathcal{A}(T) \times \mathcal{B}(T)$.

On the other hand, in any model of intuitionistic linear logic, the isomorphism (24) implies that every object of the form $!A$ is a commutative comonoid w.r.t. $(\otimes, \mathbf{I})$ (see, e.g., Melliès 2009), 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*, Emerson and Jutla 1991; Muller and Schupp 1987, 1995) corresponds to an $!(-)$-modality of intuitionistic linear logic. In particular, all the $!(-)$-rules (22) and (23) 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.)

3. Uniform automata and zig-zag strategies

In view of Section 2.3, it seems that the categorical structure required for a Curry–Howard approach should involve some machinery not given by usual connectives on automata. In Riba (2015), we proposed categories of generalized acceptance games based on the technology of *game semantics*, and more precisely on *simple games* (see, e.g., Abramsky 1997; Hyland 1997), which stem from Berry and Curien's *sequential data structures* (see, e.g., Amadio and Curien 1998, Chapter 14, but also Melliès 2005). The model of Riba (2015) is based on usual alternating automata, which seem unfortunately not to induce categories equipped with a monoidal-*closed* structure while providing a computational interpretation of proofs in the sense of Section 1.4.

We present here the notion of automata (called *uniform* automata) on which this paper relies (Sections 3.1 and 3.2), as well as the adaption of the substituted acceptance games of Riba (2015) to this context (Section 3.3). Uniform automata are motivated by the extension of usual alternating automata with a monoidal-*closed* structure. Working with uniform automata instead of usual automata allows, w.r.t. Riba (2015), for a considerable simplification of the underlying technology of game semantics. We rely on a very simple category **DZ** of (total) *zig-zag* games presented in Section 3.4 (**DZ** stands for *Dialectica*-like *Zig-zag* games, see Remark 3.11), on top of which we build the counterpart for uniform automata of substituted acceptance games and synchronous arrows games (Section 3.5).

The proof that **DZ** is a category is given in Riba (2018, Appendix A), which also discusses the connections between our approach and usual simple games.

3.1 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 P and O-moves. The corresponding monoidal-closed structure is presented in Section 5.

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

$$\mathcal{A} \quad = \quad (Q_{\mathcal{A}}\,,\, q^{\iota}_{\mathcal{A}}\,,\, U\,,\, X\,,\, \partial_{\mathcal{A}}\,,\, \Omega_{\mathcal{A}}) \tag{25}$$

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

$$\partial_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad U \times X \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow Q_{\mathcal{A}}) \tag{26}$$

Following the usual terminology, an automaton $\mathcal{A}$ *as in (25) 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 Σ with state set $\mathbf{1}$ (with $\bullet$ initial) and acceptance condition $\mathbf{1}^{\omega}$. Explicitly,

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

where $\partial_{\mathbf{1}}$ is the unique function

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

We write $\mathbf{I}$ for $\mathbf{I}_{\Sigma}$ when Σ 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 P and O respect the transition function of $\mathcal{A}$, and switches to an accepting (resp. rejecting) state as soon as O (resp. P) plays a move not allowed by $\mathcal{A}$. Assuming

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

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

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

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

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

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

3.2 Full positive games and acceptance for uniform automata

The shape (26) of the transition functions of uniform automata allows for their acceptance games to be defined without imposing legality conditions on plays. This leads to a slightly simpler setting than for usual automata.

Definition 3.3 (Full Positive Games)**.**

- *A* full positive game *has the form* $A = (U, X)$ *where* U *and* X *are sets of resp.* P *and* O-moves. *We say that* $A = (U, X)$ *is* total *if* U *and* X *are both non-empty.*
- *A* full positive game with winning condition *is a full positive game* $A = (U, X)$ *together with a* winning condition $\mathcal{W}_A \subseteq (U \cdot X)^{\omega}$.

$$
\begin{array}{c|c|}
 & \mathcal{A}(T) \\
\hline
 & (\epsilon, q^{\imath}_{\mathcal{A}}) \\
\hline
 & \vdots \\
\hline
 & (p, q_{\mathcal{A}}) \\
\hline
\mathsf{P} & u \\
\mathsf{O} & (x, d) \\
\hline
 & (p.d, q'_{\mathcal{A}}) \\
\hline
\mathsf{P} & u' \\
\mathsf{O} & (x', d') \\
\hline
 & (p.d.d', q''_{\mathcal{A}}) \\
\hline
 & \vdots
\end{array}
$$

Figure 2. Acceptance game for uniform automata.

A typical (infinite) play χ in a full positive game A has the form

$$
\begin{array}{ccccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} \\
u_0 & \cdot & x_0 & \cdot & u_1 & \cdot & x_1 & \cdot & \cdots & \cdot & u_n & \cdot & x_n \; \cdot \; \cdots \\
\Cap & & \Cap & & \Cap & & \Cap & & & & \Cap & & \Cap \\
U & & X & & U & & X & & & & U & & X
\end{array}
$$

So P plays first (hence the term "*positive*") and all P-moves (resp. O-moves) are available to P (resp. O) when it has to play (hence the term "*full*"). Assuming A is equipped with the winning condition $\mathcal{W}_A$, a play χ as above is *winning* if $(u_k \cdot x_k)_k \in \mathcal{W}_A$.

Consider a uniform automaton $\mathcal{A} : \Sigma$ as in (25), and a Σ-labeled tree T. The *acceptance game* $\mathcal{A}(T)$ is the full positive game with P-moves U and O-moves $X \times \mathfrak{D}$. So a play in $\mathcal{A}(T)$ has the form

$$
\begin{array}{ccccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} \\
u_0 & \cdot & (x_0, d_0) & \cdot & u_1 & \cdot & (x_1, d_1) & \cdot & \cdots & \cdot & u_n & \cdot & (x_n, d_n) \; \cdot \; \cdots \\
\Cap & & \Cap & & \Cap & & \Cap & & & & \Cap & & \Cap \\
U & & X \times \mathfrak{D} & & U & & X \times \mathfrak{D} & & & & U & & X \times \mathfrak{D}
\end{array}
$$

Similarly as in acceptance games for a usual non-deterministic or alternating automaton (Section 1.2), O chooses tree directions. Note that if $\mathcal{A}$ is non-deterministic in the sense of Definition 3.1 (i.e., $X \simeq \mathbf{1}$), then O only chooses tree directions. Dually, if $\mathcal{A}$ is universal ($U \simeq \mathbf{1}$), then P has no choice. Finally if $\mathcal{A}$ is deterministic ($U \simeq X \simeq \mathbf{1}$), then the only choices available in the game $\mathcal{A}(T)$ are the O's choices of tree directions. Note also 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 $\mathcal{A}(T)$.

We now equip $\mathcal{A}(T)$ with a winning condition $\mathcal{W}_{\mathcal{A}(T)} \subseteq (U \cdot (X \times \mathfrak{D}))^{\omega}$. Each infinite play $\chi = (u_k \cdot (x_k, d_k))_k \in (U \cdot (X \times \mathfrak{D}))^{\omega}$ 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

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

Then χ is winning (i.e., $\chi \in \mathcal{W}_{\mathcal{A}(M)}$) iff $(q_k)_k$ is accepting, that is, iff $(q_k)_k \in \Omega_{\mathcal{A}}$. (See also Figure 2, where states and tree positions are explicitly represented.)

Strategies for P in full positive games are what one expects.

Definition 3.4 (Strategies in Full Positive Games)**.** *A* (P-)strategy *in a full positive game* $A = (U, X)$ *is a function*

$$\sigma \;:\; X^* \longrightarrow U$$

Assume now that A *is a game with winning condition* $\mathcal{W}_A$*. Given a strategy* $\sigma : X^* \to U$ *and a sequence* $(x_k)_k \in X^\omega$*, define the sequence* $(u_k)_k \in U^\omega$ *as*

$$u_n \quad := \quad \sigma(x_0 \cdot \ldots \cdot x_{n-1})$$

We then say that σ *is* winning *in* A *if* $(u_k \cdot x_k)_k \in \mathcal{W}_A$ *for all* $(x_k)_k \in X^\omega$*.*

Example 3.5. Continuing Example 3.2.(ii), given an alternating automaton $\mathcal{A}$ over Σ and a Σ-labeled tree T, P has a winning strategy in $\mathcal{A}(T)$ iff it has a winning strategy in $\widehat{\mathcal{A}}(T)$.

Definition 3.6. *Given a uniform automaton* $\mathcal{A} : \Sigma$ *and a* Σ*-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* Σ*-labeled trees which are accepted by* $\mathcal{A}$*. Moreover, a set* $\mathcal{L}$ *of* Σ*-labeled trees is* regular *if there is an automaton* $\mathcal{A} : \Sigma$ *such that* $\mathcal{L} = \mathcal{L}(\mathcal{A})$*.*

3.3 Substituted acceptance games

We now turn to substituted acceptance games, a simple but central notion of this paper, which allows us to obtain the indexed structure discussed in Section 2.2. Substituted acceptance games are simply the (essentially obvious) adaptation of the Σ-runs of Section 2.2 to the acceptance games of Section 3.2. A similar notion for usual alternating automata was introduced in Riba (2015).

Consider a uniform automaton $\mathcal{A} : \Gamma$ as in (25), 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

$$\begin{array}{ccccccccc} \mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} & \\ (\mathsf{a}_0, u_0) & \cdot & (x_0, d_0) & \cdot & (\mathsf{a}_1, u_1) & \cdot & (x_1, d_1) & \cdot & \cdots & \cdot & (\mathsf{a}_n, u_n) & \cdot & (x_n, d_n) & \cdot & \cdots \\ \Cap & & \Cap & & \Cap & & \Cap & & & & \Cap & & \Cap & \\ \Sigma \times U & & X \times \mathfrak{D} & & \Sigma \times U & & X \times \mathfrak{D} & & & & \Sigma \times U & & X \times \mathfrak{D} & \end{array}$$

Similarly as in a substituted acceptance game for a usual non-deterministic or alternating automaton (Riba 2015), P chooses input letters and O chooses tree directions. Similarly as in the acceptance games of Section 3.2, there is no maximal finite play in the game $\Sigma \vdash \mathcal{A}(M)$.

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 $\chi = ((\mathsf{a}_k, u_k) \cdot (x_k, d_k))_k \in ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$ generates an infinite sequence of states $(q_k)_k \in Q_{\mathcal{A}}^\omega$ as follows. We let $q_0 := q^\iota_{\mathcal{A}}$ and

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

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

Let us set some notations. When the input alphabet Σ is irrelevant or clear from the context, we omit it and write $\mathcal{A}(M)$ for $\Sigma \vdash \mathcal{A}(M)$. 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 Section 2.2 to uniform automata.

$$
\begin{array}{c|ccc|c}
 & A & \multicolumn{1}{c}{\multimap_{\mathbf{DZ}}} & B & \\
\hline
 & \vdots & & \vdots & \\
\hline
\mathsf{O} & u & & & \\
 & & & v & \mathsf{P} \\
 & & & y & \mathsf{O} \\
\mathsf{P} & x & & & \\
\hline
 & \vdots & & \vdots &
\end{array}
$$

Figure 3. A typical zig-zag play with full positive games $A = (U, X)$ and $B = (V, Y)$.

Definition 3.7. *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*

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

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

Example 3.8. Continuing Example 3.5, given a usual alternating automaton $\mathcal{A}$ over Γ and some $M \in \mathbf{T}[\Sigma, \Gamma]$, P has a winning strategy in $\mathcal{A}(M)$ (in the sense of Riba 2015) if and only if P has a winning strategy in $\widehat{\mathcal{A}}(M)$.

As expected, substituted acceptance games generalize usual acceptance games. Consider a uniform automaton $\mathcal{A} : \Sigma$ and a Σ-labeled tree T. Let $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$ be the $\mathbf{T}$-morphism corresponding to T (see Section 2.2). The game $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ is similar (actually isomorphic in the sense of Section 3.4) to the acceptance game $\mathcal{A}(T)$ defined in Section 3.2. A typical play of $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ has the form

$$
\begin{array}{ccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & \mathsf{P} & & \mathsf{O} & \\
(\bullet, u_0) & \cdot & (x_0, d_0) & \cdot & (\bullet, u_1) & \cdot & (x_1, d_1) & \cdot & \cdots \cdot & (\bullet, u_n) & \cdot & (x_n, d_n) & \cdot \ \cdots \\
\Cap & & \Cap & & \Cap & & \Cap & & & \Cap & & \Cap & \\
\mathbf{1} \times U & & X \times \mathfrak{D} & & \mathbf{1} \times U & & X \times \mathfrak{D} & & & \mathbf{1} \times U & & X \times \mathfrak{D} &
\end{array}
$$

In words, since $\mathbf{1} \simeq \{\bullet\}$ is a singleton, P has actually exactly the same choices in the game $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ as in the game $\mathcal{A}(T)$.

3.4 Zig-Zag strategies

Zig-zag strategies are at the core of the notion of morphism of our categories of automata. They stem from usual strategies in *simple games* (see, e.g., Abramsky 1997; Hyland 1997), by imposing a very strong restriction on the shape of plays, essentially corresponding to a form of simulation games. This lead to a very simple notion of strategy, which admits a very simple functional representation (at least compared to usual simple games, Amadio and Curien 1998; Bucciarelli and Ehrhard 1993).

Consider full positive games $A = (U, X)$ and $B = (V, Y)$. Intuitively, a total zig-zag strategy $\sigma : A \multimap_{\mathbf{DZ}} B$ amounts to a strategy for P in an infinite game which consists in countably many sequences of rounds. In a single round $n \in \mathbb{N}$, four moves occur in succession (see also Figure 3):

(1) O plays a move $u_n \in U$,
(2) P plays a move $v_n \in V$,
(3) O answers with a move $y_n \in Y$,
(4) P concludes with a move $x_n \in X$.

So, in a zig-zag strategy $\sigma : A \multimap_{\mathbf{DZ}} B$, each P-move v_n depends on the previous O-moves $u_0, \ldots, u_n$ and $y_0, \ldots, y_{n-1}$, while each P-move x_n depends on the previous O-moves $u_0, \ldots, u_n$ and $y_0, \ldots, y_{n-1}, y_n$. This leads to the following definition.

Definition 3.9. *Given full positive games $A = (U, X)$ and $B = (V, Y)$, a (total zig-zag)* strategy *$\sigma : A \multimap_{\mathbf{DZ}} B$ is a pair of functions $\sigma = (f, F)$ where*

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

Assume now that A and B are equipped with winning conditions $\mathcal{W}_A$ and $\mathcal{W}_B$. Given sequences $(u_n)_n \in U^\omega$ and $(y_n)_n \in Y^\omega$, a strategy σ induces sequences $(v_n)_n \in V^\omega$ and $(x_n)_n \in X^\omega$ defined as

$$\begin{array}{rrcl} & v_n & := & f(u_0 \cdots u_n\, ,\, y_0 \cdots y_{n-1}) \in V \\ \textit{and} & x_n & := & F(u_0 \cdots u_n\, ,\, y_0 \cdots y_{n-1} \cdot y_n) \in X \end{array}$$

Then σ is winning *if for all $(u_n)_n \in U^\omega$ and all $(y_n)_n \in Y^\omega$, we have $(v_n \cdot y_n)_n \in \mathcal{W}_B$ whenever $(u_n \cdot x_n)_n \in \mathcal{W}_A$.*

It is easy to see that (winning) total zig-zag strategies form a category. We give a proof of this fact in Riba (2018, Appendix A), which also contains further background on game semantics.

Proposition 3.10. *Full positive games (with winning) and (winning) total zig-zag strategies form a category* $\mathbf{DZ}^{(W)}$.

Remark 3.11. The functional representation of strategies of Amadio and Curien (1998), Bucciarelli and Ehrhard (1993) is (at least in spirit) inspired from approaches to Gödel's *Dialectica* interpretation (see, e.g., Avigad and Feferman 1998; Kohlenbach 2008) in categorical logic (de Paiva 1991) (see also, e.g., Hofstra 2011; Hyland 2002; Hyland and Schalk 2003 and Jacobs 2001, Exercise 1.10.11 for modern refinements and variations). Actually, the category **DZ** (for *Dialectica*-like *zig-zag* games) can be constructed (via a distributive law) from a category of *simple self-dualization* (Hyland and Schalk, 1999, 2003) (over the topos of trees, see, e.g., Birkedal et al. 2012), which can be seen as a skeleton of Dialectica-like categories, and our categories of automata (Section 4) have a shape similar to Dialectica fibrations. Besides, as we shall see in Example 6.4, there is an ∃∀-structure on automata which is reminiscent from Gödel's *Dialectica*.

The connection between the models presented in this paper and a linear variant of Gödel's Dialectica has been made precise in Pradic and Riba (2019) in the case of ω-words.

3.5 Toward uniform linear synchronous arrow games

We now prepare to introduce (categories of) uniform linear synchronous arrow games, the last simple but central notion of this paper. Similarly as with Section 3.3, the material of this section is essentially the adaptation to uniform automata of corresponding notions of Riba (2015). We shall however postpone the proper categorical treatment to Section 4, as it relies on more advanced material.

Consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$. Our goal is to devise a notion of morphism

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

with a behavior similar to the *linear synchronous arrow games* of Riba (2015). This would amount to devise a notion of strategy

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

$$
\begin{array}{c|ccc|c}
\Sigma & \mathcal{A}(M) & \multimap & \mathcal{B}(N) & \\
\hline
 & (\epsilon, \epsilon, q^{\imath}_{\mathcal{A}}) & & (\epsilon, \epsilon, q^{\imath}_{\mathcal{B}}) & \\
 & \vdots & & \vdots & \\
 & (p, \bar{\mathsf{a}}, q_{\mathcal{A}}) & & (p, \bar{\mathsf{a}}, q_{\mathcal{B}}) & \\
\hline
\mathsf{O} & (\mathsf{a}, u) & & & \\
 & & & (\mathsf{a}, v) & \mathsf{P} \\
 & & & (y, d) & \mathsf{O} \\
\mathsf{P} & (x, d) & & & \\
\hline
 & (p.d, \bar{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{A}}) & & (p.d, \bar{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{B}}) & \\
 & \vdots & & \vdots &
\end{array}
\qquad
\begin{array}{c|ccc|c}
\Sigma & A & \multimap & B & \\
\hline
 & \vdots & & \vdots & \\
\hline
\mathsf{O} & (\mathsf{a}, u) & & & \\
 & & & v & \mathsf{P} \\
 & & & (y, d) & \mathsf{O} \\
\mathsf{P} & x & & & \\
\hline
 & \vdots & & \vdots &
\end{array}
$$

Figure 4. Linear synchronous arrow games.

playing similarly as in Figure 4 (left). Such a strategy σ is therefore required to be a total zig-zag strategy $\mathcal{A}(M) \multimap_{\mathbf{DZ}} \mathcal{B}(N)$ in the sense of Definition 3.9. But in addition, we should as in Figure 4 also require the $\mathsf{a} \in \Sigma$ and $d \in \mathfrak{D}$ played by P to be the same as their immediate predecessors played by O. The approach we adopt in this paper is to simply remove these moves from the game. This leads to the following notion.

Definition 3.12 ($\mathsf{DialZ}(\Sigma)$-Games)**.** *Fix an alphabet* Σ.

- *A* $\mathsf{DialZ}(\Sigma)$*-object* A *is given by non-empty sets* U *and* X.
- *Given* $\mathsf{DialZ}(\Sigma)$*-objects* $A = (U, X)$, $B = (V, Y)$, *a* $\mathsf{DialZ}(\Sigma)$*-morphism* $\sigma : A \multimap_{\mathsf{DialZ}(\Sigma)} B$ *is a total zig-zag strategy (see Figure 4 (right))*

$$\sigma \quad : \quad (\Sigma \times U, X) \;\multimap_{\mathbf{DZ}}\; (V, Y \times \mathfrak{D})$$

Similarly as for $\mathbf{DZ}$, the name DialZ stands for *Dialectica*-like *zig-zag* games. But note the font change. The *sans-serif* categories $\mathsf{DialZ}(\Sigma)$ actually form an indexed category, postponed to Section 4. This will in particular allow us to equip these games with winning conditions, leading to the indexed category $\mathsf{DialAut}$ (for *Dialectica*-like *Automata*).

If we forget about winning, it is possible to see here why $\mathsf{DialZ}(\Sigma)$-games induce a generalization of the acceptance games of Section 3.2. First, a substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$ with $\mathcal{A}$ as in (25) induces the $\mathsf{DialZ}(\Sigma)$-game $A = (U, X)$. Hence substituted acceptance games $\Sigma \vdash \mathcal{A}(M), \mathcal{B}(N)$ induce a $\mathsf{DialZ}(\Sigma)$-game

$$\mathcal{A}(M) \;\multimap_{\mathsf{DialZ}(\Sigma)}\; \mathcal{B}(N)$$

in the obvious way. Consider now an automaton $\mathcal{A} : \Sigma$ as in (25) and a Σ-labeled tree T. As before, let $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$ be the $\mathbf{T}$-map corresponding to T. Recall the unit automaton $\mathbf{I} : \mathbf{1}$ of Example 3.2.(i). Then the moves allowed in $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ correspond exactly to those of the $\mathsf{DialZ}(\mathbf{1})$-game $\mathbf{I} \multimap_{\mathsf{DialZ}(\mathbf{1})} \mathcal{A}(\dot{T})$ (see Figure 5).

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 Section 2.2 and Jacobs 2001, Chapter 1). While substitution in Riba (2015) was defined directly at the level of synchronous arrow games (via a representation of strategies as relations), we devise here an indexed structure induced by a reformulation of synchronicity (in the sense of Riba 2015) based on Section 3.5. We use the techniques of monoid and comonoid indexing (Hyland and Schalk 1999, 2003) on zig-zag games, which allow for a smooth treatment of monoidal closure and universal quantifications. This will lead us to a proper treatment of the $\mathsf{DialZ}(\Sigma)$-games of Definition 3.12, and to the category $\mathsf{DialAut}$, fibered over $\mathbf{T}$.

$$
\begin{array}{c|lcr|c}
\mathbf{1} & \mathbf{I} & \multicolumn{1}{c}{\multimap_{\mathsf{DialZ}(\mathbf{1})}} & \mathcal{A}(\dot{T}) & \\
\hline
 & (\epsilon, \bullet) & & (\epsilon, q^{\imath}_{\mathcal{A}}) & \\
\cline{2-4}
 & \vdots & & \vdots & \\
\cline{2-4}
 & (p, \bullet) & & (p, q_{\mathcal{A}}) & \\
\hline
\mathsf{O} & (\bullet, \bullet) & & & \\
 & & & u & \mathsf{P} \\
 & & & (x, d) & \mathsf{O} \\
\mathsf{P} & \bullet & & & \\
\hline
 & (p.d, \bullet) & & (p.d, q'_{\mathcal{A}}) & \\
\cline{2-4}
 & \vdots & & \vdots & \\
\end{array}
$$

Figure 5. Acceptance games as DialZ(**1**)-games.

$$
\begin{array}{c|lcr|c}
 & A_i & \overset{\sigma_i}{\multimap}_{\mathbf{DZ}} & B_i & \\
\hline
 & \vdots & & \vdots & \\
\hline
\mathsf{O} & u_i & & & \\
 & & & v_i & \mathsf{P} \\
 & & & y_i & \mathsf{O} \\
\mathsf{P} & x_i & & & \\
\hline
 & \vdots & & \vdots & \\
\end{array}
\qquad
\begin{array}{c|lcr|c}
 & A_1 \otimes A_2 & \overset{\sigma_1 \otimes \sigma_2}{\multimap}_{\mathbf{DZ}} & B_1 \otimes B_2 & \\
\hline
 & \vdots & & \vdots & \\
\hline
\mathsf{O} & (u_1, u_2) & & & \\
 & & & (v_1, v_2) & \mathsf{P} \\
 & & & (y_1, y_2) & \mathsf{O} \\
\mathsf{P} & (x_1, x_2) & & & \\
\hline
 & \vdots & & \vdots & \\
\end{array}
$$

Figure 6. Action of $\otimes$ on $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i$.

The material of this section relies on the symmetric monoidal structure of **DZ**.

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 4.1. *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 in Figure 6. If the* $\sigma_i = (f_i, F_i)$ *where*

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

then $\sigma_1 \otimes \sigma_2 = (h, H)$ *where*

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

are defined as

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

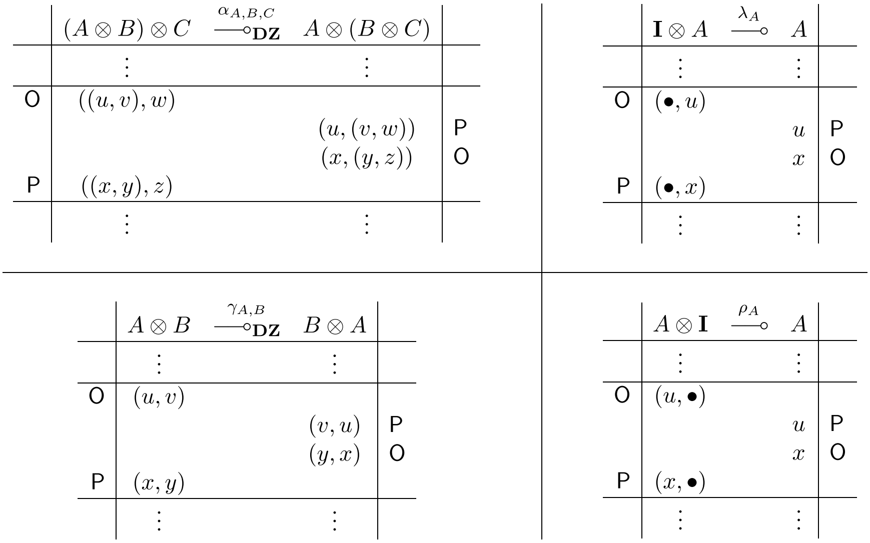


Figure 7. The structure maps of **DZ**, for $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$.

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

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

Figure 8. Coherence for a monoid (M, m, u) (where α, λ, ρ, and γ are symmetric monoidal structure maps).

The natural structure isomorphisms of **DZ** are depicted in Figure 7. 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 Σ. We are now going to see that the $\mathsf{DialZ}(\Sigma)$-games of Definition 3.12 form a category. Recall that given $\mathsf{DialZ}(\Sigma)$-objects $A = (U, X)$ and $B = (V, Y)$, a $\mathsf{DialZ}(\Sigma)$-morphism from A to B is a total zig-zag strategy

$$\sigma \ : \ (\Sigma \times U, X) \ \multimap_{\mathbf{DZ}} \ (V, Y \times \mathfrak{D})$$

We will use some algebraic structure. Objects of the form $(\mathbf{1}, M)$ (resp. $(K, \mathbf{1})$) are actually (commutative) monoids (resp. comonoids) in **DZ**. Recall from, for example, Melliès (2009) 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 the coherence conditions depicted in Figure 8. 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 $\mathbf{Comon}(\mathbb{C})$ 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:

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

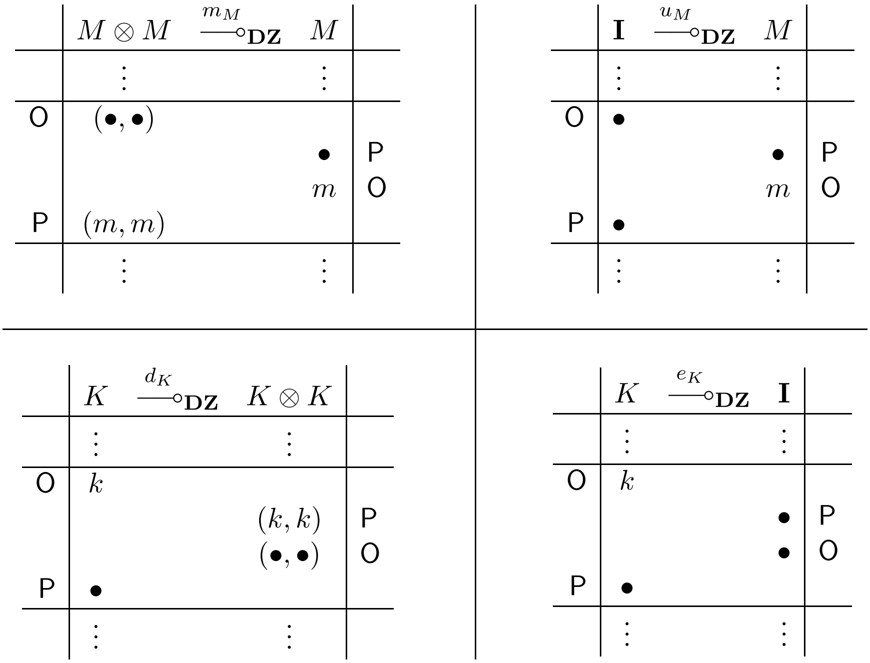


Figure 9. Structure maps for the monoid $M = (\mathbf{1}, M)$ and the comonoid $K = (K, \mathbf{1})$.

It is well known that the symmetric monoidal structure of $\mathbb{C}$ induces a Cartesian product on $\mathbf{Comon}(\mathbb{C})$ (see, e.g., Melliès 2009, Corollary 18, Section 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 4.2. *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 in Figure 9 (in the case of $I = \mathbf{1}$).*

From now on, we reason modulo the following **DZ**-isos (with the notations of Proposition 4.2):

$$(\Sigma \times U, X) \quad \simeq \quad \Sigma \otimes (U, X) \qquad \text{and} \qquad (V, Y \times \mathfrak{D}) \quad \simeq \quad (V, Y) \otimes \mathfrak{D}$$

It is well known (see, e.g., Hyland and Schalk 1999, 2003) 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 4.3. *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 μ and η 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 $(K \otimes (-), \delta, \varepsilon)$, where

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

The maps ρ, α, and λ 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 Λ of a comonad (G, δ, ε) over a monad (T, μ, η) on $\mathbb{C}$ is a natural map $\Lambda : G \circ T \Rightarrow T \circ G$ subject to some coherence conditions (see, e.g., Harmer et al. 2007), 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 Φ is the distributive law of the comonad of indexing with the comonoid Σ over the monad of indexing with the monoid $\mathfrak{D}$ in the category $\mathbf{DZ}$. The canonical 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 zig-zag strategy

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

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 σ plays as in Figure 4 (right) 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 Figure 10 (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 Figure 10 (bottom).

4.3 The indexed structure of $\mathsf{DialZ}(-)$ *and the base category* $\mathbb{T}$

We therefore have for each alphabet Σ a category $\mathsf{DialZ}(\Sigma)$. We now discuss an indexed structure on the categories $\mathsf{DialZ}(-)$, based on a 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., Jacobs 2001, Chapter 1 but also Hofstra 2011; Hyland 2002), and reminiscent from Maillard and Melliès (2015). 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 $f : I \times X \to Y$.

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

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

Figure 10. Decomposition of $\sigma^\uparrow$.

The functor $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ is the first projection, and the fiber over I is the Kleisli category of indexing with the comonoid I (see, e.g., Jacobs 2001, Example 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 fiber 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 $\mathbf{Comon}(\mathbb{C})^{\mathrm{op}} \to \mathbf{Cat}$ that we denote $\mathsf{CI}(\mathbb{C})$ (for *comonoid indexing* over $\mathbb{C}$). The corresponding Grothendieck construction $\int \mathsf{CI}(\mathbb{C})$ (see, e.g., Jacobs 2001, Chapter 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 $\int \mathsf{CI}(\mathbb{C})$ is fibered over $\mathbf{Comon}(\mathbb{C})$ via the first projection that we denote

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

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

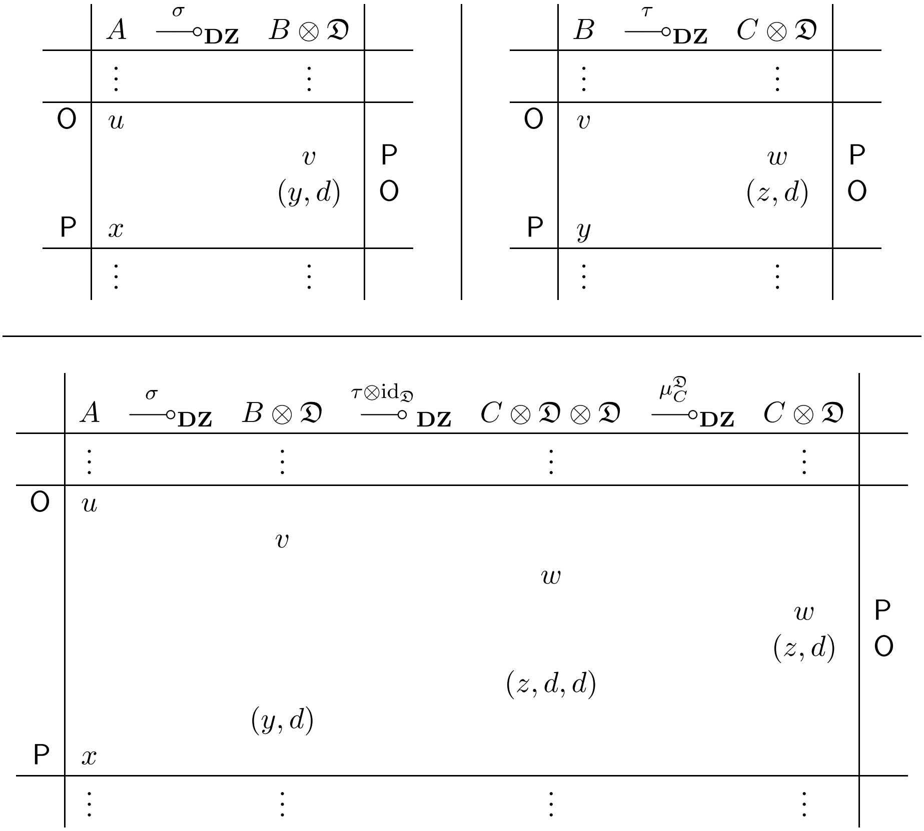


Figure 11. Composition in $\mathbf{DZ}_{\mathfrak{D}} = \mathbf{Kl}(\mathfrak{D})$.

Let us spell out composition in $\mathbf{DZ}_{\mathfrak{D}}$. First recall that for a monad (T, μ, η) 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 in Figure 11 (top), the composite $\tau \circ_{\mathbf{DZ}_{\mathfrak{D}}} \sigma$ is depicted (modulo associativity) in Figure 11 (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., Melliès 2009).

Proposition 4.4. *(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 in Figure 12. If the $\sigma_i = (f_i, F_i)$ where

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

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

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

Figure 12. Action of $\otimes_{\mathbf{DZ}_{\mathfrak{D}}}$ on $\sigma_i : A_i \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B_i$.

	Σ	$\overset{\tilde{d}_\Sigma}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\Sigma \otimes \Sigma$	
	$\vdots$		$\vdots$	
O	a			
			(a, a)	P
			d	O
P	$\bullet$			
	$\vdots$		$\vdots$	

	Σ	$\overset{\tilde{e}_\Sigma}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\mathbf{I}$	
	$\vdots$		$\vdots$	
O	a			
			$\bullet$	P
			d	O
P	$\bullet$			
	$\vdots$		$\vdots$	

Figure 13. Structure maps for the comonoid $\Sigma = (\Sigma, \mathbf{1})$.

then $\sigma_1 \otimes_{\mathbf{DZ}_{\mathfrak{D}}} \sigma_2 = (h, H)$ *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

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

Moreover, the $\mathbf{DZ}_{\mathfrak{D}}$*-structure maps* $\tilde{d}_\Sigma$ *and* $\tilde{e}_\Sigma$ *of the comonoid induced by* Σ *can be depicted as in Figure 13.*

It follows from Proposition 4.4 and the fact that Φ is a distributive law, that each category $\mathsf{DialZ}(\Sigma)$ is the Kleisli category of indexing with Σ 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 Definition 2.1), which is isomorphic to a full subcategory of $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$. First, it directly follows Definition 3.9 that $\mathbf{T}$-strategies from Σ to Γ in the sense of Definition 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 Σ to Γ. Hence $\mathbf{T}$ is isomorphic to a full subcategory of $\mathbf{DZ}_{\mathfrak{D}}$. Second, $\mathbf{T}$-maps induce comonoid maps.

Proposition 4.5. *The category* $\mathbf{T}$ *embeds in* $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$ *via the functor* $\mathsf{E}_{\mathbf{T}}$ *which takes an alphabet* Σ *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 Proposition 4.5 is given in Riba (2018). 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., Melliès 2009, Corollary 18, Section 6.5). By Proposition 4.5, this gives a Cartesian structure on $\mathbf{T}$.

Corollary 4.6. *The category* $\mathbf{T}$ *is Cartesian, with on objects the Cartesian product of alphabets, and with unit* $\mathbf{1}$.

4.4 The fibered category DialAut

We thus have a category DialZ indexed over $\mathbf{T}$, and whose fiber over Σ is the category $\mathsf{DialZ}(\Sigma)$. We will now define a fibration $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ (for *Dialectica*-like *Automata*) 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 applying the Grothendieck construction to an indexed category $(-)^\star : \mathbf{T}^{\mathrm{op}} \to \mathbf{Cat}$, which takes an alphabet Σ to a category $\mathsf{DialAut}_\Sigma$. The action of $(-)^\star$ on $\mathbf{T}$-maps is based on the indexed category DialZ.

Definition 4.7 (The Category $\mathsf{DialAut}_\Sigma$). *Fix an alphabet* Σ.

- *The objects of the category* $\mathsf{DialAut}_\Sigma$ *are 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* $\mathsf{DialZ}(\Sigma)$*-morphisms from* (U, X) *to* (V, Y)*, that is, total zig-zag strategies*

$$\sigma \quad : \quad \Sigma \otimes (U, X) \quad \multimap_{\mathbf{DZ}} \quad (V, Y) \otimes \mathfrak{D}$$

whose lift $\sigma^\uparrow$ *are winning strategies on*

$$(\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A) \quad \multimap_{\mathbf{DZ}^{\mathrm{W}}} \quad (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$$

Composition and identities of $\mathsf{DialAut}_\Sigma$ are induced by composition and identities of $\mathsf{DialZ}(\Sigma)$ (using the functoriality of $(-)^\uparrow$ for winning). 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 Section 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 Section 2.2).

Lemma 4.8. *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 Lemma 4.8 is given in Riba (2018). 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 id_A are taken in $\mathbf{DZ}_{\mathfrak{D}}$). It remains to check that $L^\star(\sigma)^\uparrow$ is winning whenever so is $\sigma^\uparrow$.

Proposition 4.9. *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 $\sigma^\uparrow$ is winning on*

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

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

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

A detailed proof of Proposition 4.9 is given in Riba (2018). We then obtain a strict 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 obtained by applying the Grothendieck construction to $(-)^\star$.

Definition 4.10 (The Fibered Category $\mathsf{DialAut}$)**.** *The objects of* $\mathsf{DialAut}$ *are pairs (Σ, A) where A is an object of* $\mathsf{DialAut}_\Sigma$. *Maps from (Σ, A) to (Γ, B) are pairs (L, σ) of a $\mathbf{T}$-map $L : \Sigma \to \Gamma$ and a* $\mathsf{DialAut}_\Sigma$*-map σ 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 $\mathsf{da}(\Sigma, A) := \Sigma$ *and* $\mathsf{da}(L, \sigma) := L$.

4.5 Substitution and language inclusion

We now check that $\mathsf{DialAut}_\Sigma$ is correct w.r.t. language inclusion. First, consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$ in the sense of Section 3.3. We thus obtain $\mathsf{DialAut}_\Sigma$ objects that we still write $\Sigma \vdash \mathcal{A}(M), \mathcal{B}(N)$. Now, it follows from Lemma 4.8 that given

$$\sigma \;:\; \mathcal{A}(M) \multimap \mathcal{B}(N) \qquad \text{and} \qquad L \in \mathbf{T}[\Gamma, \Sigma]$$

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

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

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

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

Assume given $\sigma : \mathcal{A} \multimap \mathcal{B}$. If $T \in \mathcal{L}(\mathcal{A})$, then there is some $\tau : \mathbf{I_1} \multimap \mathcal{A}(T)$. It follows that we obtain $\dot{T}^\star(\sigma) \circ \tau : \mathbf{I_1} \multimap \mathcal{B}(T)$, which implies $T \in \mathcal{L}(\mathcal{B})$. 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 4.11 (Theorem 1.3, (9))**.** *If* P *has a winning strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, then we have $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$.*

5. Symmetric monoidal-closed structure

We show here that the category **DZ** of full positive games and total zig-zag strategies is equipped with a monoidal-closed structure, and that this structure lifts to $\mathsf{DialZ}(\Sigma)$ and to the fibers of $\mathsf{DialAut}$. This, in particular, gives a (symmetric) monoidal-closed structure on uniform automata.

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

Recall from, for example, Melliès (2009) 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 (Mac Lane 1998, Theorem 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 $\mathbf{\Lambda}(f) : B \to C^A$ with

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

5.1 The symmetric monoidal closure of DZ

The monoidal-closed structure of **DZ** can actually be read off from the definition of zig-zag strategies given in Definition 3.9.

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 σ plays as in Figure 3. From each play s of σ, the responses $v \in V$ of σ to O-moves $u \in U$ define a function

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

and the responses $x \in X$ of σ to further O-moves $y \in Y$ define a function

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

This amounts to describe σ by a pair of maps

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

Proposition 5.1. *The category* **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 Proposition 5.1 is given in Riba (2018). The monoidal-closed structure of **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 strategies, but not *plays*.

5.2 The symmetric monoidal-closed structure of DialAut and tree automata

The symmetric monoidal-closed structure of $\mathsf{DialAut}$ and of tree automata is induced by the symmetric monoidal-closed structure of DialZ, which is itself lifted from **DZ**.

$$
\begin{array}{c|lcr|}
 & \Sigma \otimes A_i & \overset{\sigma_i}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}} & B_i \\
\hline
 & \vdots & & \vdots \\
\hline
\mathsf{O} & (\mathsf{a}, u_i) & & \\
\mathsf{P} & & & v_i \\
\mathsf{O} & & & (y_i, d) \\
\mathsf{P} & x_i & & \\
\hline
 & \vdots & & \vdots
\end{array}
\qquad
\begin{array}{c|lcr|c}
 & \Sigma \otimes (A_1 \otimes A_2) & \overset{\tau}{\multimap}_{\mathbf{DZ}_{\mathfrak{D}}} & B_1 \otimes B_2 & \\
\hline
 & \vdots & & \vdots & \\
\hline
\mathsf{O} & (\mathsf{a}, (u_1, u_2)) & & & \\
 & & & (v_1, v_2) & \mathsf{P} \\
 & & & ((y_1, y_2), d) & \mathsf{O} \\
\mathsf{P} & (x_1, x_2) & & & \\
\hline
 & \vdots & & \vdots &
\end{array}
$$

Figure 14. Action of $\otimes_{\mathsf{DialZ}(\Sigma)}$ on $\sigma_i : A_i \multimap_{\mathsf{DialZ}(\Sigma)} B_i$, where $\tau := \sigma_1 \otimes_{\mathsf{DialZ}(\Sigma)} \sigma_2$.

5.2.1 The symmetric monoidal structure of DialZ

We have seen in Proposition 4.4 that the symmetric monoidal structure of $\mathbf{DZ}$ 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 in Figure 14. If the $\sigma_i = (f_i, F_i)$ where

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

then $\sigma_1 \otimes_{\mathsf{DialZ}(\Sigma)} \sigma_2 = (h, H)$ where

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

are defined as

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

5.2.2 The symmetric monoidal closure of $\mathbf{DZ}_{\mathfrak{D}}$ *and* DialZ

The monoidal-closed structure of $\mathbf{DZ}$ 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. *The categories $\mathbf{DZ}_{\mathfrak{D}}$ 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 fiberwise symmetric monoidal-closed structure of DialAut (in the sense of Jacobs 2001, Section 1.8). The unit over Σ 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

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

and

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

where α and β are obtained by pointwise application:

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

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 fiberwise monoidal closed.*

5.2.4 *The symmetric monoidal-closed structure of uniform automata*

We now turn to uniform automata. Their symmetric monoidal-closed structure is inherited from $\mathsf{DialAut}_\Sigma$.

Definition 5.5 (Monoidal Product and Linear Implication on Uniform Automata)**.** *Assume $\mathcal{A}$ is as in (25) and*

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

so that

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

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

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

  with

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

  where

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

  and with $((q_n, q'_n))_n \in \Omega_{\mathcal{A} \otimes \mathcal{B}}$ iff $\left((q_n)_n \in \Omega_\mathcal{A}\right.$ and $\left.(q'_n)_n \in \Omega_\mathcal{B}\right)$.
- *We let $(\mathcal{A} \multimap \mathcal{B})$ be the automaton over Σ defined as*

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

with

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

where

$$q'_{\mathcal{A}} \;=\; \partial_{\mathcal{A}}(q_{\mathcal{A}} , \mathtt{a} , u , F(u, y, d) , d) \qquad \textit{and} \qquad q'_{\mathcal{B}} \;=\; \partial_{\mathcal{B}}(q_{\mathcal{B}} , \mathtt{a} , f(u) , y , d)$$

and with $((q_n, q'_n))_n \in \Omega_{\mathcal{A}\multimap\mathcal{B}}$ *iff* $\big((q_n)_n \in \Omega_{\mathcal{A}} \implies (q'_n)_n \in \Omega_{\mathcal{B}}\big)$.

Note that $\Omega_{\mathcal{A}\otimes\mathcal{B}}$ as well as $\Omega_{\mathcal{A}\multimap\mathcal{B}}$ are ω-regular since $\Omega_{\mathcal{A}}$ and $\Omega_{\mathcal{B}}$ are both assumed to be ω-regular. Note also 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,

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

5.3 Deduction, adequacy, and correctness

We now return to the deduction system for automata outlined in Section 1, and discuss the role of linearity in our setting following Sections 2.4 and 2.5. First, the monoidal structure of $\mathsf{DialAut}_\Sigma$ allows us to interpret sequents of the form

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

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

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

Second, the symmetric monoidal-closed structure allows us to interpret the deduction rules of IMLL. We gather them in Figure 15. Using the notations of Section 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 Figure 15, then there is a winning* P*-strategy σ in*

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

In particular, if $\mathcal{A} \vdash \mathcal{B}$ is derivable, then by combining Proposition 5.6 with Proposition 4.11, we obtain a strategy witnessing that $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$.

Note that the strategy σ is obtained from the derivation $\mathscr{D}$ in a purely compositional way. Moreover, all the rules of Figure 15 are compatible with cut-elimination.

Remark 5.7 (On Cut-Elimination). It follows from the fact that we have monoidal-closed categories (Proposition 5.4), that the interpretation of derivations as strategies for the rules of Figure 15 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., Melliès (2009, Section 3.3), then $\mathscr{D}$ and $\mathscr{D}'$ are interpreted by the same strategy. This in particular applies to the following two derivations:

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

$$\text{(Exchange)}\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{B},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}}$$

$$\text{(Cut)}\ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\ \overline{\mathcal{A}},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{}{M\ ;\ \mathcal{A} \vdash \mathcal{A}}\ \text{(Axiom)}$$

$$\text{(Left}\ \otimes)\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A} \otimes \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{A} \otimes \mathcal{B}}\ \text{(Right}\ \otimes)$$

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

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

Figure 15. Rules of IMLL for uniform automata.

Example 5.8. Proposition 5.6 yields a winning 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

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

Note that in Figure 15 we omitted the *weakening* and *contraction* rules (20):

$$\text{(Weak)}\ \frac{M ; \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M ; \overline{\mathcal{A}},\ \mathcal{A},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M ; \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{A},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M ; \overline{\mathcal{A}},\ \mathcal{A},\ \overline{\mathcal{B}} \vdash \mathcal{C}}\ \text{(Contr)}$$

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 providing winning P-strategies in the game

$$\mathcal{A} \quad \multimap \quad \mathcal{A} \otimes \mathcal{A} \tag{27}$$

If $\mathcal{A}$ is non-deterministic (and with P-moves U), then a winning P-strategy in (27) simply takes an 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

$$\mathcal{A} \otimes \mathcal{B} \quad \multimap \quad \mathcal{A} \tag{28}$$

Assuming $\mathcal{A}$ and $\mathcal{B}$ are as in Definition 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 IMELL in Section 7.

Example 5.9. Proposition 5.6 actually holds for any extension of the deduction system of Figure 15 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 (28):

$$\dfrac{}{\mathcal{A}_1\,,\,\dots\,,\,\mathcal{A}_n\vdash\mathcal{A}_i}$$

We thus get

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

So there is a winning P-strategy on

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

and by Proposition 4.11 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 (27):

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

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

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

Finally, we note that the monoidal structure together with (28) implies 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 Section 1.2 that usual alternating automata are equipped with a complementation construction $(-)^{\perp\!\!\!\perp}$ linear in the number of states (see, e.g., Muller and Schupp 1987). Thanks to the monoidal-closed structure, a similar construction can be obtained on uniform automata.

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

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

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

$$\partial_{\curlywedge}(\mathbb{b}\,,\ \_\,,\ d'\,,\ \bullet\,,\ d) \quad := \quad \begin{cases} \mathbb{f} \text{ if } \mathbb{b} = \mathbb{f} \text{ and } d = d' \\ \mathbb{t} \text{ otherwise} \end{cases}$$

Note that in the game $\Sigma \vdash \curlywedge$, O looses as soon as it does not play the same tree direction as proposed by P. On the other hand, $\curlywedge$ accepts no tree since in an acceptance game $\curlywedge(T)$, 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 $(\mathcal{A} \multimap \curlywedge)$ is isomorphic (via $X^{U\times\mathfrak{D}} \simeq X^{U\times\mathbf{1}\times\mathfrak{D}}$) to the automaton $\mathcal{A}^{\curlywedge}$ defined as

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

where

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

and where

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

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

Thanks to the determinacy of ω-regular games (see, e.g., Perrin and Pin 2004; Thomas 1997), we get:

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

A detailed proof of Proposition 5.12 is given in Riba (2018).

5.4.1 Deduction rules for $\curlywedge$ *and* $\mathcal{A}^{\curlywedge}$

Since the fiber categories $\mathsf{DialAut}_\Sigma$ are symmetric monoidal closed, they are in particular dialogue categories in the sense of Melliès (2013), with as exponentiating object any object of $\mathsf{DialAut}_\Sigma$. Hence, if as in Example 5.9 we extend the deduction system of Figure 15 with the realizable rules

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

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

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

6. Quantifications

We now discuss quantifications in the fibration $\mathsf{DialAut}$. We follow the categorical approach outlined in Section 2.2, according to which existential and universal quantifications (also called simple coproducts and products, Jacobs 2001, Chapter 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 families of operations $(\coprod_{I,J})_{I,J}$ and $(\prod_{I,J})_{I,J}$ are moreover required to satisfy some coherence conditions, called the *Beck–Chevalley* conditions, which insure that they are preserved by substitution.

Having both (categorical) existential and universal quantifications greatly simplifies some basic reasoning on games (see Corollary 6.5 and Example 6.10). Referring to Remark 3.11, this also allows for a clearer connection with Gödel's *Dialectica* interpretation (Example 6.4).

We first present quantifications in DialAut (Section 6.1), from which we then derive quantifications on automata (Section 6.2) and deduction rules for quantifications (Section 6.3).

6.1 Quantifications in DialAut

Quantifications in DialAut are induced by quantifications in DialZ, which are themselves based on quantifications in simple fibrations. It is well known (see, e.g., Jacobs 2001, Chapter 1) that the simple fibration $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ has always 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 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

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

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

The extension to DialAut just requires to handle winning and acceptance.

Proposition 6.2. *The fibration* 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 DialAut.

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

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

where

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

Quantifications on automata induce an $\exists\forall$-structure which is reminiscent from Gödel's *Dialectica* interpretation (see, e.g., Avigad and Feferman 1998; Kohlenbach 2008).

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

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

whose transition function

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

is obtained from $\partial_{\mathcal{A}}$ 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

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

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

$$\begin{array}{rcl} \Sigma \vdash (\exists_\Gamma \mathcal{A})(M) \multimap \mathcal{B}(N) & \simeq & \Sigma \times \Gamma \vdash \mathcal{A}(M \times \mathrm{Id}_\Gamma) \multimap \mathcal{B}(N \circ [\pi_\Sigma]) \\ \Sigma \vdash \mathcal{B}(N) \multimap (\forall_\Gamma \mathcal{A})(M) & \simeq & \Sigma \times \Gamma \vdash \mathcal{B}(N \circ [\pi_\Sigma]) \multimap \mathcal{A}(M \times \mathrm{Id}_\Gamma) \end{array} \tag{29}$$

It follows that P has winning strategies in

$$\Sigma \times \Gamma \vdash (\forall_\Gamma \mathcal{A})[\pi_\Sigma] \multimap \mathcal{A} \qquad \text{and} \qquad \Sigma \times \Gamma \vdash \mathcal{A} \multimap (\exists_\Gamma \mathcal{A})[\pi_\Sigma] \tag{30}$$

Thanks to the Büchi–Landweber Theorem (1969), we thus get the following corollary to Proposition 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 P-strategy on $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, and if there exists such a winning P-strategy, then there exists a finite-state one, which is moreover effectively computable from $\mathcal{A}$ and $\mathcal{B}$.*

Proof. By (29) and (30), 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 O can only play $\bullet$ in the component $\mathbf{I_1}$, similarly as in Section 3.5, 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 Thomas (1997, Example 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 the Büchi–Landweber Theorem (1969) (see also Thomas 1997, Theorem 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 (30) that existential quantifications are complete in the following sense:

$$(\textsc{Subst})\ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \circ M'\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}$$

$$(\textsc{Trans}_{\downarrow})\ \frac{[\mathtt{f}] \circ M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{B}[\mathtt{f}]} \qquad \frac{M\ ;\ \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{B}[\mathtt{f}]}{[\mathtt{f}] \circ M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B}}\ (\textsc{Trans}_{\uparrow})$$

$$(\textsc{Left}\ \exists)\ \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]}, \mathcal{B} \vdash \mathcal{A}[\pi]}{M\ ;\ \overline{\mathcal{A}}, \exists_\Gamma \mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \times N\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \times N\ ;\ \overline{\mathcal{A}} \vdash (\exists_\Gamma \mathcal{A})[\pi]}\ (\textsc{Right}\ \exists)$$

$$(\textsc{Left}\ \forall)\ \frac{M \times N\ ;\ \overline{\mathcal{A}}, \mathcal{B} \vdash \mathcal{A}}{M \times N\ ;\ \overline{\mathcal{A}}, (\forall_\Gamma \mathcal{B})[\pi] \vdash \mathcal{A}} \qquad \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]} \vdash \mathcal{A}}{M\ ;\ \overline{\mathcal{A}} \vdash \forall_\Gamma A}\ (\textsc{Right}\ \forall)$$

Figure 16. Substitution and quantification rules for uniform automata (where M, M' are composable, π is a suitable projection, and $\mathtt{f}$ is a function on alphabets).

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 Section 7. Dually, it follows from (30) that universal quantifications are correct (but they are complete only on *universal* automata, see Definition 3.1).

Corollary 6.7. *Given $\mathcal{A} : \Sigma \times \Gamma$, if $T \in \mathcal{L}(\forall_\Gamma \mathcal{A})$, then for all Γ-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 the isos (29) that we can extend the deduction system of Figure 15 with the rules of Figure 16 while preserving adequacy (Proposition 5.6), Example 5.9 and compatibility with cut-elimination (in the sense of Remark 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 Figures 15, 16 and of Example 5.9, then there is a winning* 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 Figure 16 involve internalized substitutions of the form $\mathcal{A}[\mathtt{f}]$ as defined in Definition 3.7. The transfer 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 transfer rule $(\textsc{Trans}_\downarrow)$, we can derive the following specific rules of substitution for $\mathbf{T}$-maps induced by functions $\mathtt{f} : \Sigma \to \Gamma$:

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

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

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

it follows that we can derive

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

Example 6.10. Continuing Example 5.9, we can extend the deduction system with the rule

$$\frac{\mathcal{L}(\mathcal{A} : \mathbf{1}) \neq \emptyset}{\vdash \mathcal{A}}$$

This rule actually subsumes Example 5.9. Indeed, following the same reasoning as for Corollary 6.5, assuming that

$$\Sigma ; \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \vdash \mathcal{B}$$

is realizable we get (leaving implicit some structural and cut rules)

$$\cfrac{\cfrac{\cfrac{\cfrac{\mathcal{L}(\forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})) \neq \emptyset}{\mathbf{1} ; \vdash \forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})}}{\Sigma ; \vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B}}}{\Sigma ; \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \vdash \mathcal{B}}}{\Sigma ; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}}$$

7. Non-deterministic automata

This final section focuses on structural properties of non-deterministic automata, on their role in Rabin's Tree Theorem (1969) (namely in the complementation of non-deterministic tree automata), and on their relationship with IMELL (Girard 1987) (see Sections 1.2, 1.3, and 2.5).

We first detail in Section 7.1 the Cartesian structure of non-deterministic automata announced in Section 1.3 (see also Sections 2.4 and 5.3). Technically, this Cartesian structure follows from the simple fact that non-deterministic automata generate comonoids in the fibers of DialAut (by a direct extension of Proposition 4.4, Section 4.3). As a consequence, we show that our model has the witnessing properties asked to computational interpretations of proofs (in the sense of Section 1.4), and moreover that it allows for combining strategies obtained from proofs with witnessing strategies computed by usual emptiness checking algorithms (see Section 1.5).

Second, we show that a powerset construction for the *Simulation Theorem* (Emerson and Jutla 1991; Muller and Schupp 1987, 1995) satisfies the usual deduction rules of the exponential modality ! of IMELL. This completes the picture sketched in Sections 1.3, 1.5, 2.4, and 2.5, and moreover allows us to obtain a deduction system which is complete w.r.t. intuitionistic and classical deduction (via usual translations). Furthermore, Riba (2018, Appendix C) details how two constructions from resp. Colcombet and Löding (2008) and Santocanale and Arnold (2005) can be reformulated in our setting.

The proofs of all statements of this section are given in Riba (2018).

7.1 The Cartesian structure of non-deterministic automata

Similarly as with usual (total) non-deterministic automata (see Section 2.4), the monoidal product of uniform automata is Cartesian on non-deterministic automata. Recall from Definition 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 Section 5.3, we thus get canonical realizers for

$$\mathcal{N}(L) \multimap \mathcal{N}(L) \otimes \mathcal{N}(L) \qquad \text{and} \qquad \mathcal{N}(L) \multimap \mathbf{I} \tag{31}$$

As we shall see now, these canonical realizer equip $\mathcal{N}(L)$ with the structure of a comonoid.[15] Thanks to well-known results (see, e.g., Melliès 2009, Corollary 18, Section 6.5), this implies that the monoidal structure of uniform automata is Cartesian on non-deterministic automata.

Recall from Proposition 4.2 that objects of the form (K, I) with $I \simeq \mathbf{1}$ are comonoids in $\mathbf{DZ}$, and from Proposition 4.4 that such objects are also comonoids in $\mathbf{DZ}_{\mathfrak{D}}$. On the other hand, we

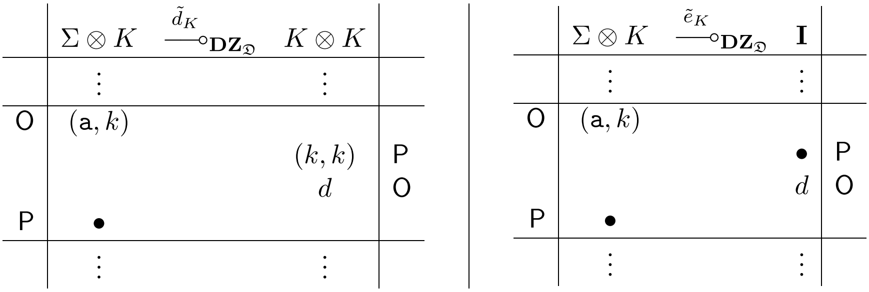


Figure 17. Structure maps in $\mathsf{DialZ}(\Sigma)$ for the comonoid $K = (K, \mathbf{1})$.

have seen that $\mathsf{DialZ}(\Sigma)$ is a Kleisli category of comonoid indexing in $\mathbf{DZ}_{\mathfrak{D}}$, whose symmetric monoidal structure is given by the extension of Proposition 4.4 to comonoid indexing given by Proposition 5.2. Actually, the lifting of comonoids given by Proposition 4.4 also extends to the case of comonoid indexing:

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 $(K,\ d \circ \varepsilon^C_K,\ e \circ \varepsilon^C_K)$ 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 in Figure 17 (where we omitted some $\bullet$-moves).*

The extension of Proposition 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 Σ, 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., Melliès 2009, Corollary 18, Section 6.5), we thus have the expected result that non-deterministic automata are equipped with a Cartesian structure.

Corollary 7.3. *For each alphabet Σ, 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 (see Section 2.4), Corollary 7.3 allows us to extend adequacy (Propositions 5.6 and 6.8) to the following weakening and contraction rules:

$$(\textsc{Weak}_{\mathrm{ND}})\ \frac{M\,;\overline{\mathcal{A}}\,,\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\,;\overline{\mathcal{A}}\,,\ \mathcal{N}\,,\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\,;\overline{\mathcal{A}}\,,\ \mathcal{N}\,,\ \mathcal{N}\,,\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\,;\overline{\mathcal{A}}\,,\ \mathcal{N}\,,\ \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}_{\mathrm{ND}}) \tag{32}$$

where $\mathcal{N}$ is required to be non-deterministic (while $\overline{\mathcal{A}}$, $\overline{\mathcal{B}}$, and $\mathcal{C}$ can be arbitrary). On the other hand, recall that the full weakening rule is actually derivable in the setting of Example 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., Jacobs 2001, Chapter 10).

Consider a non-deterministic automaton $\mathcal{N} : \Sigma \times \Gamma$ with set of P-moves U, and let T be a Σ-labeled tree (so that $T : \mathfrak{D}^* \to \Sigma$). It directly follows from Definition 3.9 that a winning P-strategy in $\mathbf{1} \vdash \mathbf{I} \multimap (\exists_\Gamma \mathcal{A})(\dot{T})$ is given by a function

$$\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Gamma \times U$$

or equivalently by a pair of functions

$$\left(\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \;\longrightarrow\; \Gamma \right) \times \left(\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \;\longrightarrow\; U \right)$$

which amount to a tree $T' : \mathfrak{D}^* \to \Gamma$ together with a winning P-strategy in $\mathbf{1} \vdash \mathbf{I} \multimap \mathcal{A}\langle \dot{T}, \dot{T}' \rangle$. We thus have shown

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 Σ-labeled tree and τ 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 Corollary 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 Proposition 7.4 that our computational interpretation makes it possible to effectively extract witnesses from (interpretations of) proofs, in the sense of Sections 1.4 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 Figures 15, 16, Example 5.9 and (32). Then adequacy (Propositions 5.6 and 6.8) gives a strategy

$$\sigma \;:\; \mathbf{I} \;\multimap\; \exists_\Sigma \mathcal{N}$$

(effectively computed by induction on $\mathscr{D}$), and which by Proposition 7.4 is of the form

$$\langle T, \tau \rangle \;:\; \textstyle\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \;\longrightarrow\; \Sigma \times U$$

$$\text{where} \qquad \tau \;:\; \mathbf{I} \;\multimap\; \mathcal{N}(T)$$

7.1.3 Application: effective realizers from witnesses of non-emptiness

Similarly as with usual non-deterministic automata (see, e.g., Thomas 1997), thanks to the Büchi–Landweber Theorem (1969), Corollary 7.5 implies the decidability of emptiness for non-deterministic automata as well as the *Rabin Basis Theorem* (1972), 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 Corollary 7.5 can be lifted to winning strategies in games of the form $\mathcal{A} \multimap \mathcal{C}$. Consider the case (mentioned in Section 1.5.(2)) 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 Proposition 5.12[16] 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* P*-strategies in $\mathcal{A} \otimes \mathcal{B} \multimap \bot$ and $\mathcal{A} \multimap \mathcal{B}^{\perp}$. Moreover, these* 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 Example 5.9.(ii), implies the following extension of Example 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 IMELL

Recall that similarly as in the usual setting, uniform automata have linear complements (Section 5.4), and that non-deterministic automata have correct existential quantifications (Section 7.5). On the other hand, we mentioned in Section 1.2 that in the usual setting, the *Simulation Theorem* (Emerson and Jutla 1991; Muller and Schupp 1987, 1995) 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 Walukiewicz (2002) gives a similar simulation operation $!(-)$, taking a uniform automaton $\mathcal{A} : \Sigma$ to a non-deterministic automaton $!\mathcal{A} : \Sigma$ with $\mathcal{L}(!\mathcal{A}) = \mathcal{L}(\mathcal{A})$, thus completing the picture (2) of Section 1.2 for our notion of uniform automata:

$$\begin{tikzcd} \text{Non-Deterministic Uniform Automata} \arrow[loop above, "\exists(-)", "(-)\otimes(-)"'] \arrow[rr, bend left] & & \text{Uniform Automata} \arrow[loop above, "(-)\otimes(-)", "(-)^{\perp}"'] \arrow[ll, bend left, "!(-)"] \end{tikzcd} \tag{33}$$

Moreover, we show that the operation $!(-)$ satisfies the *deduction* rules of the exponential modality $!(-)$ of IMELL:

$$\frac{M\,;\, !\overline{\mathcal{A}} \vdash \mathcal{A}}{M\,;\, !\overline{\mathcal{A}} \vdash !\mathcal{A}} \qquad \frac{M\,;\, \overline{\mathcal{A}},\ \mathcal{B} \vdash \mathcal{A}}{M\,;\, \overline{\mathcal{A}},\ !\mathcal{B} \vdash \mathcal{A}} \qquad \frac{M\,;\, \overline{\mathcal{A}},\ \vdash \mathcal{A}}{M\,;\, \overline{\mathcal{A}},\ !\mathcal{B} \vdash \mathcal{A}} \qquad \frac{M\,;\, \overline{\mathcal{A}},\ !\mathcal{B},\ !\mathcal{B} \vdash \mathcal{A}}{M\,;\, \overline{\mathcal{A}},\ !\mathcal{B} \vdash \mathcal{A}} \tag{34}$$

It follows that the exponential $!(-)$ makes it possible to define, using Girard's decomposition, an intuitionistic implication $(-) \to (-)$ as $\mathcal{A} \to \mathcal{B} := !\mathcal{A} \multimap \mathcal{B}$.

The rules (34) are an obvious adaptation to our context of the rules displayed in (22) and (23) of Section 2.5. The last two rules (weakening and contraction) actually follow from the rules ($\text{W}\small{\text{EAK}}_{\text{ND}}$) and ($\text{C}\small{\text{ONTR}}_{\text{ND}}$) displayed in (32). 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 (in the sense of Remark 5.7).

The difficulty with the (PROMOTION) rule can be explained as follows. We have seen in Section 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 (31). It follows that similarly as with usual IMELL-exponentials (see Section 2.5 but also Melliès 2009), 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 Section 5.3 that for a uniform automaton $\mathcal{A}$ with set of O-moves X, realizers of

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

may not exist because O can play two different $(x, x') \in X \times X$ in the right component $\mathcal{A} \otimes \mathcal{A}$, that P may not be able to merge into a single $x'' \in X$ in the left component $\mathcal{A}$.

Usual solutions to this merging problem for IMELL-exponentials (see, e.g., Amadio and Curien 1998; Melliès 2004, 2009) 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 (3) Section 1.2, the crux of Rabin's Tree Theorem 1969), such recipes cannot (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 ω-regular games equipped with *parity* acceptance conditions (see e.g. Grädel et al. 2002; Thomas 1997). In our case, we rely for the (PROMOTION) rule on the stronger fact that in an ω-regular game whose winning condition is given by a disjunction of parity conditions (also called a *Rabin* condition), winning P-strategies can always be assumed to be positional (Klarlund 1994; Klarlund and Kozen 1995; Jutla 1997; Zielonka 1998). Unfortunately, positionality is not preserved by composition, and the interpretation of the (PROMOTION) rule is not preserved by cut-elimination (in the sense of Remark 5.7).

Remark 7.9. In (33), we have only displayed existential quantifications $\exists$ for non-deterministic automata, because as in the usual setting, they are correct (in the sense of Corollary 7.5) only on non-deterministic automata. Similarly, we have not displayed universal quantifications because they are only complete on *universal* automata (see Definition 3.1).

Note that on the other hand, the categorical properties of quantifications (Proposition 6.2) and thus the deduction rules of Figure 16, 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 Proposition 4.11.

7.2.2 An exponential construction on uniform automata

Our exponential construction $!(-)$ is an adaptation of the one used in Walukiewicz (2002). Given a parity automaton $\mathcal{A} : \Sigma$ with set of P-moves U and set of O-moves X, we let

$$!\mathcal{A} \quad := \quad (Q_{!\mathcal{A}}\,,\ q^\imath_{!\mathcal{A}}\,,\ U^{Q_\mathcal{A}}\,,\ \mathbf{1}\,,\ \partial_{!\mathcal{A}}\,,\ \Omega_{!\mathcal{A}})$$

where $Q_{!\mathcal{A}} := \mathcal{P}(Q_{\mathcal{A}} \times Q_{\mathcal{A}})$, $q^{\iota}_{!\mathcal{A}} := \{(q^{\iota}_{\mathcal{A}}, q^{\iota}_{\mathcal{A}})\}$ and $\partial_{!\mathcal{A}}$ is defined as follows: Given $\mathsf{a} \in \Sigma$, $f \in U^{Q_{\mathcal{A}}}$, $d \in \mathfrak{D}$ and $S \in Q_{!\mathcal{A}}$ with $\pi_2(S) = \{q' \mid \exists q.\ (q, q') \in S\} = \{q_1, \dots, q_n\}$, let

$$\partial_{!\mathcal{A}}(S, \mathsf{a}, f, \bullet, d) \quad := \quad T_1 \cup \dots \cup T_n$$

where, for each $k \in \{1, \dots, n\}$,

$$T_k \quad := \quad \{(q_k, q) \mid \exists x \in X.\ q = \partial_{\mathcal{A}}(q_k, \mathsf{a}, f(q_k), x, d)\}$$

Let a *trace* in an infinite sequence $(S_n)_n \in Q^{\omega}_{!\mathcal{A}}$ 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 ω-regular since $\Omega_{\mathcal{A}}$ is ω-regular (see Walukiewicz 2002, Section 4).

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

$$\tilde{\partial}_{!\mathcal{A}} \quad : \quad Q_{!\mathcal{A}} \times \Sigma \quad \longrightarrow \quad U^{Q} \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow Q_{!\mathcal{A}})$$

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 Proposition 7.10.

It is easy to show the adequacy of the dereliction rule. This amounts to provide co-unit-like winning P-strategies

$$\varepsilon \quad : \quad !\mathcal{A}(M) \quad \multimap \quad \mathcal{A}(M)$$

Proposition 7.12. *Given $\mathcal{A} : \Sigma$, there is a winning* P*-strategy ε 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 the existence of winning positional 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 P-moves U (resp. V) and set of 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

$$A_{\mathsf{P}} := \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{A}} \qquad A_{\mathsf{O}} := \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{A}} \times U$$
$$B_{\mathsf{P}} := \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{B}} \qquad B_{\mathsf{O}} := \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{B}} \times V$$

and with edges depicted in Figure 18, where $q'_{\mathcal{A}} := \partial_{\mathcal{A}}(q_{\mathcal{A}}, M(\bar{\mathsf{a}}.\mathsf{a}, p), u, x, d)$ (for some $x \in X$) and $q'_{\mathcal{B}} := \partial_{\mathcal{B}}(q_{\mathcal{B}}, N(\bar{\mathsf{a}}.\mathsf{a}, p), v, y, d)$ (for some $y \in Y$). Write 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 seen as a graph with root $((\epsilon, \epsilon, q^{\iota}_{\mathcal{A}}), (\epsilon, \epsilon, q^{\iota}_{\mathcal{B}}))$. We say that a strategy σ is *positional* if it agrees on plays with the same position, that is, if $m = m'$ whenever $s.m, t.m' \in \sigma$ and $\mathsf{pos}(s) = \mathsf{pos}(t)$.

Consider now parity automata $\mathcal{A}_1, \dots, \mathcal{A}_n$ and $\mathcal{B}$. The winning condition of a game of the form $\mathcal{A}_1(M_1) \otimes \dots \otimes \mathcal{A}_n(M_n) \multimap \mathcal{B}(N)$ 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 Klarlund (1994), Klarlund and Kozen (1995), Jutla (1997), Zielonka (1998) that if P has a winning strategy σ in such a game, then it has a winning *positional* strategy (w.r.t. G), which according to Zielonka (1998) is recursive in σ. The existence of winning positional P-strategies

$$
\begin{array}{c|ccc|c}
 & ((p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{A}}) & , & (p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{B}})) & A_{\mathsf{P}} \times B_{\mathsf{P}} \\
\mathsf{O} & \downarrow & & & \\
 & ((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u) & , & (p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{B}})) & A_{\mathsf{O}} \times B_{\mathsf{P}} \\
\mathsf{P} & & & \downarrow & \\
 & ((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u) & , & (p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{B}}\,,\,v)) & A_{\mathsf{O}} \times B_{\mathsf{O}} \\
\mathsf{O} & & & \downarrow & \\
 & ((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u) & , & (p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{B}})) & A_{\mathsf{O}} \times B_{\mathsf{P}} \\
\mathsf{P} & \downarrow & & & \\
 & ((p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{A}}) & , & (p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{B}})) & A_{\mathsf{P}} \times B_{\mathsf{P}}
\end{array}
$$

Figure 18. The edges of the graph G for $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$.

allows us to show the adequacy of the (PROMOTION) rule. A detailed proof of Proposition 7.13 is given in Riba (2018, Appendix B).

Proposition 7.13. *Given $\mathcal{N}, \mathcal{A} : \Sigma$ with $\mathcal{N}$ non-deterministic, if there is a winning* P*-strategy in $\Sigma \vdash \mathcal{N}(L) \multimap \mathcal{A}(M)$, then there is a winning* P*-strategy in $\Sigma \vdash \mathcal{N}(L) \multimap\, !\mathcal{A}(M)$.*

Remark 7.14. Consider the case of $!\mathcal{A}$ for $\mathcal{A}$ non-deterministic. Given $S \in Q_{!\mathcal{A}}$ with $\pi_2(S) = \{q_1, \ldots, q_n\}$, we have

$$\partial_{!\mathcal{A}}(S, \mathsf{a}, f, \bullet, d) \quad = \quad \{(q_k, \partial_{\mathcal{A}}(q_k, \mathsf{a}, f(q_k), \bullet, d)) \mid k = 1, \ldots, n\}$$

One can easily see that $\mathcal{A}$ is a retract of $!\mathcal{A}$. Moreover, since the initial state of $!\mathcal{A}$ is the singleton $q^{\iota}_{!\mathcal{A}} = \{(q^{\iota}_{\mathcal{A}}, q^{\iota}_{\mathcal{A}})\}$, plays in games of the form $!\mathcal{A}(M)$ only reach singleton states $\{(q_{\mathcal{A}}, q'_{\mathcal{A}})\} \in Q_{!\mathcal{A}}$. In particular, each play on $!\mathcal{A}(M)$ determines a unique play on $\mathcal{A}(M)$, and it follows that $!(-)$ extends to a functor on *non-deterministic* automata.

7.2.4 Applications

This paragraph gathers consequences of Propositions 7.12 and 7.13, thus mirroring Sections 7.1.1–7.1.3 and completing the picture announced in Sections 1.3, 1.5, and 2.5.

First, Proposition 7.12 implies that $\mathcal{L}(!\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A})$, while Proposition 7.13 gives the converse inclusion $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(!\mathcal{A})$. We thus have, as expected:

Corollary 7.15. $\mathcal{L}(\mathcal{A}) = \mathcal{L}(!\mathcal{A})$.

Corollary 7.15 gives the extension of Corollary 7.6 to general uniform automata.

Corollary 7.16. *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* P*-strategy on $\mathbf{1} \vdash \mathbf{I} \multimap \mathcal{A}(\dot{T})$.*

We also obtain the lifting property of Section 1.5.(3), extending Proposition 7.7. Let $?\mathcal{A} := (!(\mathcal{A}^{\perp}))^{\perp}$.

Proposition 7.17 (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* P*-strategy in $\Sigma \vdash\, !\mathcal{A} \multimap\, ?\mathcal{B}$.*

$$\text{(DERELICTION)}\ \frac{M\ ;\ \overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, !\mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{N}} \vdash \mathcal{A}}{M\ ;\ \overline{\mathcal{N}} \vdash !\mathcal{A}}\ \text{(PROMOTION)}$$

$$\text{(WEAK}_{\text{ND}}\text{)}\ \frac{M\ ;\ \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}}, \mathcal{N}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}\ \text{(CONTR}_{\text{ND}}\text{)}$$

Figure 19. Exponential rules (where $\mathcal{N}$ and $\overline{\mathcal{N}}$ are non-deterministic).

Proof. By Proposition 5.12 and Corollary 7.15, if $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$, then $\mathcal{L}(!\mathcal{A}) \cap \mathcal{L}(!(\mathcal{B}^{\perp})) = \emptyset$, and we conclude by Proposition 7.7. □

Proposition 7.17 is a completeness result on realizability w.r.t. language inclusion. It is only a weak converse to the soundness of realizability w.r.t. language inclusion (Proposition 4.11, Section 4.5), because it imposes constraints on the *shape* of automata for the implication to be realizable (while it imposes no constraint on the *languages* involved as $\mathcal{L}(\mathcal{A}) = \mathcal{L}(!\mathcal{A})$ and $\mathcal{L}(\mathcal{B}) = \mathcal{L}(?\mathcal{B})$).

On the other hand, Propositions 7.12 and 7.13 give adequacy for the rules displayed in (34).

Proposition 7.18 (Adequacy (Theorem 1.3 (8)))**.** *If the sequent $M\ ; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$ is derivable using the rules of Figures 15, 16, 19 and of Example 5.9, then there is a winning* P*-strategy in the game*

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

As an example of use of the exponential rules, we mention a negative translation of the law of Peirce $((A \to B) \to A) \to A$. The law of Peirce gives full classical logic when added to intuitionistic logic. Recall that $\mathcal{A} \to \mathcal{B} := !\mathcal{A} \multimap \mathcal{B}$.

Example 7.19. The law of Peirce $((?\mathcal{A} \to ?\mathcal{B}) \to ?\mathcal{A}) \to ?\mathcal{A}$ can be derived, thanks to the exponential rules.

Finally, returning to MSO and IMELL (in the sense of Sections 1.1 and 1.3), we obtain Proposition 1.1 As in Section 1.3, we assume given an automaton $\mathcal{A}(\alpha)$ for each atomic formula $\alpha \in \mathrm{At}$.

Proposition 7.20 (Proposition 1.1)**.** *Consider a closed* MSO*-formulae φ as in Section 1.1, and let $(-)^{\dagger}$ be either $(-)^{\mathrm{nd}}$ or $(-)^{\mathrm{alt}}$. Then φ is true (in the standard model) if and only if $\mathcal{A}(\varphi^{\dagger})$ accepts the unique* **1***-labeled tree.*

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 Tree Theorem (1969) correspond to the connectives of IMELL (Girard 1987). Namely, we equipped a variant of usual alternating tree automata (that we called *uniform* tree automata, Section 3) with a fibered monoidal-closed structure (Sections 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 (Section 6). Moreover, we have shown in Section 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 (Emerson and Jutla 1991; Muller and Schupp 1987, 1995) satisfies the *deduction rules* of an $!(-)$ IMELL-exponential modality.

Second, our approach is based on a realizability semantics for a linear constructive deduction system on tree automata, in which, thanks to the monoidal-closed structure, realizers are winning strategies in a generalization of acceptance games. Our realizability semantics satisfies an expected property of witness extraction from proofs of existential statements. Moreover, this realizability semantics is compositional and makes it possible 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 Pradic have recently obtained preliminary results in this direction for MSO on ω-words (Pradic and Riba 2017, 2018).

Furthermore, as shown in Example 6.4 (see also Remark 3.11), our interpretation shares a formal similarity with Gödel's *Dialectica* interpretation (see, e.g., Avigad and Feferman 1998; Kohlenbach 2008). Actually, the category **DZ** can be constructed (via a distributive law) from a category of *simple self-dualization* (Hyland and Schalk 1999, 2003) (over the topos of trees, see, e.g., Birkedal et al. 2012), which can be seen as a skeleton of Dialectica-like categories (de Paiva 1991), and the category DialZ has a shape similar to Dialectica fibrations (see Hofstra 2011; Hyland 2002 but also Jacobs 2001, Example 1.10.11). This connection has been made precise in Pradic and Riba (2019) in the case of ω-words and provides realizers for linear variants of Markov and choice rules.[18]

Moreover, we show in Riba (2018, Appendix C) that our setting easily handles known constructions from Colcombet and Löding (2008) and Santocanale and Arnold (2005) for language reduction and separation.

8.1 Further works

We plan to continue the line of research initiated here and in Riba (2015) along different directions. A central point w.r.t. most of them concerns the (PROMOTION) rule.

The interpretation of simulation as an $!(-)$ IMELL-exponential modality in Section 7.2 is interesting because it shows that an IMELL-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 Section 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 (Klarlund 1994; Klarlund and Kozen 1995; Jutla 1997; Zielonka 1998). 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 ω-regular games (i.e., from the Büchi–Landweber Theorem (1969), see also, e.g., Thomas 1997, Theorem 6.16).
However, in both cases this amounts to apply a non-trivial external algorithm, and there seems 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 Remark 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 (Section 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 Section 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) Taking inspiration from Melliès et al. (2009), We plan to investigate the existence of free exponentials in suitable extensions of **DZ**.
(2) Moreover, there seems to be a natural exponential, in which P essentially plays strategies, but which in the context of automata would lead to infinite-state automata.
(3) We also plan to look at non-synchronous exponentials, such as the Curien–Lamarche exponential of *sequential data structures* (see, e.g., Amadio and Curien 1998, Chapter 14, but also Melliès 2005), 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 Arnold (1999), Arnold and Niwinski (2007), Facchini et al. (2013). 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 (2006) that the notion of *Innocence* (originally introduced by Hyland and Ong 2000 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–Landweber Theorem) from emptiness checking in the sense of Corollary 6.5, Section 7.1.3, Corollary 7.16, and Proposition 7.17. On the other hand, the notion of positionality used in Section 7.2.3 may be preserved by composition for *non-deterministic* innocent strategies, in the vein of Hirschowitz and Pous (2012), Tsukada and Ong (2015). We do not know yet how such notions of non-deterministic strategies behave w.r.t. the construction of positional winning P-strategies for Rabin games as in, for example, Zielonka (1998). Also, the present setting has still to be compared with Melliès's *Higher-Order Automata* (Melliès 2017).

Our main target is the construction of realizability models for MSO. In the case of ω-words (i.e., taking $\mathfrak{D} = \mathbf{1}$ in this paper), and in the context of Church's synthesis, the aforementioned results of Pradic and Riba (2017, 2018) suggest that, together with the results of this paper, it is possible and pertinent to devise refinements of MSO based on *Intuitionistic Linear Logic* (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 MSO allows for a fibered 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 MSO, and moreover that ILL can provide classes of formulae with improved translations to automata w.r.t. the known non-elementary lower bound (see, e.g., Grädel et al. 2002, Chapter 13).

Moreover, in devising realizability models for 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 (Möllerfeld 2002), it seems that (PROMOTION) may be seen as a form of reflection scheme. Similarly as in the complementation construction of Thomas (1997, Theorem 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 MSO.

Notes

1 But with the notable exception of Blumensath (2013).
2 However, the IMLL-structure underlying our model differs from the usual IMLL-structure of simple games.
3 Alternating automata are not always made explicit (see, e.g., Thomas 1997).
4 It is also customary (and equivalent in terms of expressiveness) to allow several initial states.
5 This is trivial for P-strategies but not for O-strategies.
6 $(-)^{\perp\!\!\!\perp}$ was noted $\sim(-)$ in Riba (2015).
7 It does not preserve composition, because of issues with positionality of strategies. Possible workarounds, left as future work, are discussed in Section 8.1.
8 The author and Pradic have recently obtained preliminary results in this direction for MSO on ω-words (Pradic and Riba 2017, 2018).
9 The morphisms from Σ to Γ of the base category of Riba (2015) are restricted to $(\Sigma \to \Gamma)$-labeled trees.
10 Because universal quantifications commute over conjunctions!
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.
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.
13 *Total* alternating automata were called *complete* in Riba (2015).
14 (WEAK) actually holds (in a non-canonical way) for total alternating automata (i.e., the ! is not strictly necessary in the conclusion).
15 Recall from Section 4.2 that in this paper, by (co)monoid we always mean *commutative* (co)monoid.
16 More precisely, this is direction ($\Leftarrow$) in Proposition 5.12.
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 Carayol and Löding 2007; Carayol et al. 2010; Gurevich and Shelah 1983) may be surprised by this suggestion. Actually, choice rules in constructive arithmetics turn $\forall\exists$-statements into $\exists\forall$ ones, but do not necessarily induce well-orderings.

References

Abramsky, S. (1997). Semantics of interaction. In: Pitts, A. M. and Dybjer, P. (eds.) *Semantics and Logics of Computation*, Publications of the Newton Institute, vol. 14, Cambridge, Cambridge University Press, 1.

Amadio, R. M. and Curien, P.-L. (1998). *Domains and Lambda-Calculi*, Cambridge Tracts in Theoretical Computer Science, Cambridge, Cambridge University Press.

Arnold, A. and Niwinski, D. (2007). Continuous Separation of Game Languages. *Fundamenta Informaticae* **81** (1–3) 19–28.

Arnold, A. (1999). The μ-calculus alternation-depth hierarchy is strict on binary trees. *ITA* **33** (4/5) 329–340.

Avigad, J. and Feferman, S. (1998). Gödel's functional ("Dialectica") interpretation. In: Buss, S. (ed.) *Handbook Proof Theory*, Studies in Logic and the Foundations of Mathematics, vol. 137, Amsterdam, Elsevier, 337–405.

Birkedal, L., Møgelberg, R. E., Schwinghammer, J. and Støvring, K. (2012). First steps in synthetic guarded domain theory: step-indexing in the topos of trees. *Logical Methods in Computer Science* **8** (4) 1–45.

Blackburn, P., de Rijke M. and Venema, Y. (2002). *Modal Logic*, Cambridge Tracts in Theoretical Computer Science, Cambridge University Press.

Blumensath, A. (2013). An algebraic proof of Rabin's Tree Theorem. *Theoretical Computer Science* **478** 1–21.

Börger, E., Grädel, E. and Gurevich, Y. (1997). *The Classical Decision Problem*, Perspectives in Mathematical Logic, Springer.

Bucciarelli, A. and Ehrhard, T. (1993). A Theory of Sequentiality. *Theoretical Computer Science* **113** (2) 273–291.

Büchi, J. R. and Landweber, L. H. (1969). Solving sequential conditions by finite-state strategies. *Transaction of the American Mathematical Society* **138** 367–378.

Carayol, A. and Löding, C. (2007). MSO on the infinite binary tree: choice and order. In *CSL*, Lecture Notes in Computer Science, vol. 4646, Berlin, Heidelberg, Springer, 161–176.

Carayol, A., Löding, C., Niwinski, D. and Walukiewicz, I. (2010) Choice functions and well-orderings over the infinite binary tree. *Central European Journal of Mathematics* **8** (4) 662–682.

Colcombet, T. and Löding, C. (2008). The non-deterministic Mostowski Hierarchy and distance-parity automata. In *ICALP 2008*, Lecture Notes in Computer Science, vol. 5126, Berlin, Heidelberg, Springer, 398–409.

de Paiva, V. (1991) The Dialectica categories. Technical Report 213, University of Cambridge Computer Laboratory.

Emerson, E. A. and Jutla, C. S. (1991) Tree automata, Mu-Calculus and determinacy (extended abstract). In *FOCS*, Washington, DC, IEEE Computer Society, 368–377.

Facchini, A., Murlak, F. and Skrzypczak, M. (2013) Rabin-Mostowski index problem: a step beyond deterministic automata. In *LICS*, Washington, DC, IEEE Computer Society, 499–508.

Girard, J.-Y. (1987). Linear logic. *Theoretical Computer Science* **50** 1–102.

Girard, J.-Y., Lafont, Y. and Taylor, P. (1989). *Proofs and Types,* Cambridge Tracts in Theoretical Computer Science, Cambridge, Cambridge University Press.
Grädel, E., Thomas, W. and Wilke, T. (eds.) (2002). *Automata, Logics, and Infinite Games: A Guide to Current Research*, Lecture Notes in Computer Science, vol. 2500, Berlin, Heidelberg, Springer-Verlag.
Gurevich, Y. and Harrington, L. (1982). Trees, automata, and games. In: *STOC*, ACM, 60–65.
Gurevich, Y. and Shelah, S. (1983). Rabin's Uniformization Problem. *Journal of Symbolic Logic* **48** (4) 1105–1119.
Harmer, R., Hyland, M. and Melliès, P.-A. (2007). Categorical combinatorics for innocent strategies. In *LICS 2007*, Washington, DC, IEEE Computer Society, 379–388.
Hirschowitz, T. and Pous, D. (2012). Innocent strategies as presheaves and interactive equivalences for CCS. *Scientific Annals of Computer Science* **22** (1) 147–199.
Hofstra, P. J. W. (2011) The dialectica monad and its cousins. In: Makkai, M. and Hart, B.T. (eds.) *Models, Logics, and Higher-dimensional Categories: A Tribute to the Work of Mihály Makkai*, CRM Proceedings & Lecture Notes, Providence, RI, American Mathematical Society.
Hyland, J. M. E. (1997). Game semantics. In: Pitts, A. M. and Dybjer, P. (eds.) *Semantics and Logics of Computation*, Publications of the Newton Institute, vol. 14, Cambridge, Cambridge University Press, 131.
Hyland, J. M. E. (2002). Proof theory in the abstract. *Annals of Pure and Applied Logic* **114** (1–3) 43–78.
Hyland, J. M. E. and Ong, C.-H. (2000). On full abstraction for PCF: I, II, and III. *Information and Computation* **163** (2) 285–408.
Hyland, J. M. E. and Schalk, A. (1999). Abstract games for linear logic. *Electronic Notes in Theoretical Computer Science* **29** 127–150.
Hyland, J. M. E. and Schalk, A. (2003). Glueing and orthogonality for models of linear logic. *Theoretical Computer Science* **294** (1/2) 183–231.
Jacobs, B. (2001). *Categorical Logic and Type Theory*, Studies in Logic and the Foundations of Mathematics. Amsterdam, Elsevier.
Johnstone, P. T. (1986). *Stone Spaces*, Cambridge Studies in Advanced Mathematics, Cambridge, Cambridge University Press.
Jutla, C. S. (1997). Determinization and memoryless winning strategies. *Information and Computation* **133** (2) 117–134.
Klarlund, N. (1994). Progress measures, immediate determinacy, and a subset construction for tree automata. *Annals of Pure and Applied Logic* **69** (2–3) 243–268.
Klarlund, N. and Kozen, D. (1995). Rabin measures. *Chicago Journal of Theoretical Computer Science*. http://cjtcs.cs.uchicago.edu/articles/1995/3/selfcitation.bib..
Kohlenbach, U. (2008). *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*, Springer Monographs in Mathematics, Berlin, Heidelberg, Springer-Verlag.
Kupferman, O. and Vardi, M. Y. (2005). Safraless decision procedures. In: *Proceedings of FOCS'05*, Washington, DC, USA, 2005, IEEE Computer Society, 531–542.
Lambek, J. and Scott, P. J. (1986). *Introduction to Higher Order Categorical Logic*, Cambridge, CUP.
Mac Lane, S. (1998). *Categories for the Working Mathematician*, 2nd edn., New York, Springer-Verlag.
Maillard, K. and Melliès, P.-A. (2015). A fibrational account of local states. In: *30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015*, Computer Society, 402–413.
McNaughton, R. (1966). Testing and generating infinite sequences by a finite automaton. *Information and Control* **9** (5) 521–530.
Melliès, P.-A. (2004). Comparing hierarchies of types in models of linear logic. *Information and Computation* **189** (2) 202–234.
Melliès, P.-A. (2005). Sequential algorithms and strongly stable functions. *Theoretical Computer Science* **343** (1–2) 237–281.
Melliès, P.-A. (2006). Asynchronous games 2: the true concurrency of innocence. *Theoretical Computer Science* **358** (2–3) 200–228.
Melliès, P.-A. (2009). Categorical semantics of linear logic. In: *Interactive Models of Computation and Program Behaviour*, Panoramas et Synthèses, vol. 27, SMF.
Melliès, P.-A. (2013). On dialogue games and coherent strategies. In: *CSL*, LIPIcs, vol. 23, Dagstuhl, Germany, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 540–562.
Melliès, P.-A. (2017). Higher-order parity automata. In: *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20–23, 2017*, Washington, DC, IEEE Computer Society, 1–12.
Melliès, P.-A., Tabareau, N. and Tasson, C. (2009). An explicit formula for the free exponential modality of linear logic. In: *Proceedings of ICALP'09*, Lecture Notes in Computer Science, vol. 5556, Berlin, Heidelberg, Springer, 247–260.
Möllerfeld, M. (2002). *Generalized Inductive Definitions - The μ-Calculus and Π^1_2-Comprehension*. PhD thesis, Westfälischen Wilhelms-Universität Münster. https://miami.uni-muenster.de/Record/9dfa74b6-186b-4e95-a51f-9965d7e1e508.
Muller, D. E. and Schupp, P. E. (1987). Alternating automata on infinite trees. *Theoretical Computer Science* **54** 267–276.
Muller, D. E. and Schupp, P. E. (1995). Simulating alternating tree automata by nondeterministic automata: new results and new proofs of the theorems of Rabin, McNaughton and Safra. *Theoretical Computer Science* **141** (1&2) 69–107.
Ong, C.-H. L. (2006). On model-checking trees generated by higher-order recursion schemes. In: *Proceedings of LICS'06*, Washington, DC, IEEE Computer Society, 81–90.

Perrin, D. and Pin, J.-É. (2004). *Infinite Words: Automata, Semigroups, Logic and Games*, Pure and Applied Mathematics, Amsterdam, Elsevier.

Pradic, P. and Riba, C. (2017). A Curry–Howard approach to Church's synthesis. In: *Proceedings of FSCD'17*, LIPIcs, vol. 84, Dagstuhl, Germany, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 30:1–30:16.

Pradic, P. and Riba, C. (2018). LMSO: a Curry–Howard approach to Church's synthesis via linear logic. In *LICS*, New York, NY, ACM, 849–858.

Pradic, P. and Riba, C. (2019). A Dialectica-like interpretation of a linear MSO on infinite words. In *FoSSaCS*, Cham, Springer.

Rabin, M. O. (1969). Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society* **141** 1–35.

Rabin, M. O. (1972). Automata on Infinite Objects and Church's Problem, American Mathematical Society.

Riba, C. (2015). Fibrations of tree automata. In: *TLCA*, LIPIcs, vol. 38, Dagstuhl, Germany, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 302–316.

Riba, C. (2018). Monoidal-Closed Categories of Tree Automata. Available on HAL (hal-01261183) https://hal.archives-ouvertes.fr/hal-01261183.

Santocanale, L. and Arnold, A. (2005). Ambiguous classes in mu-calculi hierarchies. *Theoretical Computer Science* **333**(1–2) 265–296.

Sørensen, M. H. and Urzyczyn, P. (2006). *Lectures on the Curry–Howard Isomorphism*, Studies in Logic and the Foundations of Mathematics, vol. 149, Amsterdam, Elsevier Science Inc.

Thomas, W. (1997). Languages, automata, and logic. In: Rozenberg, G. and Salomaa, A. (eds.) *Handbook of Formal Languages*, vol. III, Berlin, Heidelberg, Springer-Verlag, 389–455.

Tsukada, T. and Ong, C.-H. L. (2015). 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*, Washington, DC, IEEE Computer Society, 220–231.

Vardi, M. Y. and Wilke, T. (2008). Automata: from logics to algorithms. In: *Logic and Automata*, Texts in Logic and Games, vol. 2, Amsterdam, Amsterdam University Press, 629–736.

Walukiewicz, I. (2002). Monadic second-order logic on tree-like structures. *Theoretical Computer Science* **275**(1–2) 311–346.

Zielonka, Z. (1998). Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science* **200**(1–2) 135–183.

Cite this article: Riba C (2020). Monoidal-closed categories of tree automata. *Mathematical Structures in Computer Science* **30**, 62–117. https://doi.org/10.1017/S0960129519000173