

A Compositional Approach to Parity Games^{*†}

Kazuki Watanabe

The Graduate University for Advanced Studies (SOKENDAI)
Hayama, Japan
National Institute of Informatics
Tokyo, Japan
kazukiwatanabe@nii.ac.jp

Clovis Eberhart

National Institute of Informatics
Tokyo, Japan
Japanese-French Laboratory of Informatics
IRL 3527, CNRS, Tokyo, Japan
eberhart@nii.ac.jp

Kazuyuki Asada

Tohoku University
Sendai, Japan
asada@riec.tohoku.ac.jp

Ichiro Hasuo

The Graduate University for Advanced Studies (SOKENDAI)
Hayama, Japan
National Institute of Informatics
Tokyo, Japan
Ritsumeikan University
Kusatsu, Japan
i.hasuo@acm.org

In this paper, we introduce open parity games, which is a compositional approach to parity games. This is achieved by adding open ends to the usual notion of parity games. We introduce the category of open parity games, which is defined using standard definitions for graph games. We also define a graphical language for open parity games as a prop, which have recently been used in many applications as graphical languages. We introduce a suitable semantic category inspired by the work by Grellois and Melliès on the semantics of higher-order model checking. Computing the set of winning positions in open parity games yields a functor to the semantic category. Finally, by interpreting the graphical language in the semantic category, we show that this computation can be carried out compositionally.

1 Introduction

Parity game is a major tool in theoretical computer science. Many formal verification problems such as model checking, satisfiability, etc.—can be reduced to solving parity games [34], where alternation of least and greatest fixed point operators in a specification is modeled by the parity winning condition. Efficient solutions of parity games, therefore, benefit many problems; recent algorithmic works include [9].

In this paper, we are interested in *compositionality* in formal verification in general, and in parity games in particular. It means that the property of a big system can be deduced from those of its constituent parts. One benefit is *efficiency*: compositionality can yield an efficient divide-and-conquer algorithm. Another is *maintainability*: compositional verification explicates an assumption that each subsystem must satisfy for the safety of the whole system; a subsystem can then be replaced freely as long as the local assumption is satisfied.

Compositional methods in model checking have been pursued in the literature, such as [8, 26]. Many of those methods require a user to provide interfaces between subsystems, either as systems [8] or as

*The full version of the present paper is available at <https://group-mmm.org/~kazuki/>.

†The authors are supported by ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603), JST Moonshot R&D (JPMJMS2033-04), and JSPS KAKENHI Grant No. 18K11156.

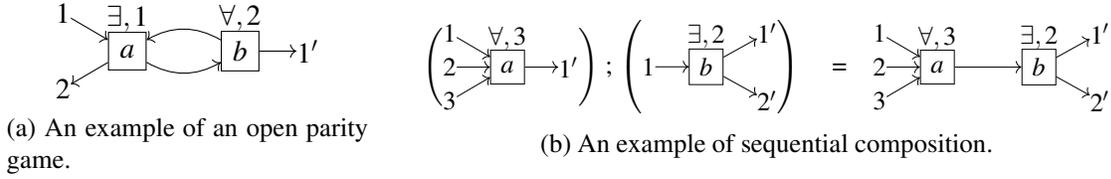


Figure 1: Examples of open parity games.

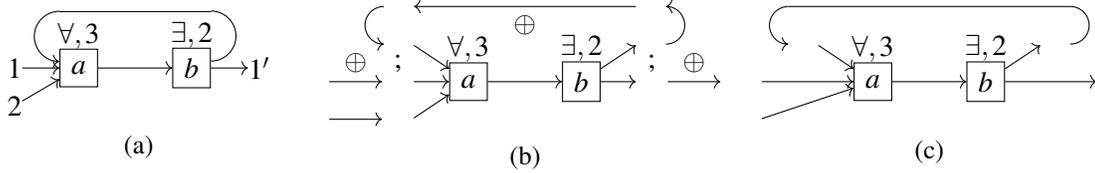


Figure 2: An example of a cycle and its decomposition using the compact closed structure and the traced monoidal structure.

specifications [26]. The role of compositionality is stressed in *higher-order model checking (HOMC)* [15, 32], too, where intermediate results are combined along typing rules.

In this paper, influenced by the semantical constructs from [15], we introduce a categorical framework in which parity games are both presented and solved in a compositional manner. The presentation is by a *prop* [28] (products and permutations category), a categorical notion of “monoidal” algebraic structure. This categorical presentation enables us to formulate compositionality as the preservation of suitable structures of certain functors. It also enables us to exploit general categorical structures (traced, compact closed, etc.) and properties (such as freeness). The use of props as graphical languages for various mathematical structures has been actively pursued recently (such as signal flow diagrams, matrices, and network games) [4, 5, 27]; the current work adds a new item to the list, namely parity games.

Contribution. The outline of our paper is Fig. 3. We extend parity games with so-called *open ends* so that we can compose them. The resulting notion (*open parity game*) is organized in a compact closed category denoted by \mathbf{OPG}_M . As a graphical language for open parity games, we use the prop $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ freely generated by a suitable monoidal (algebraic) theory $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$. The other category $\text{Int}(\mathbf{FinScottL}_M^{\text{op}})$ in Fig. 3 originates from [15]—it is our *semantic category* that tells which player is winning for (closed) parity games; for open parity games, it provides intermediate results of a suitable granularity to decide winners later.

Our main theorem (Thm. 5.10) is the commutativity of Fig. 3; it says that the semantics of parity games \mathcal{W}_M —defined as usual in terms of plays, strategies, and the parity acceptance condition—can be computed compositionally by a compact closed functor $\llbracket - \rrbracket_M$. The last compositional computation is illustrated in Ex. 5.11. After all, in the framework in Fig. 3, one writes down a parity game as a composition of smaller ones, in the graphical language of the prop $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$; when it comes to solving games, the winning positions for larger games are computed from those of the smaller ones, using that $\llbracket - \rrbracket_M$ preserves composition.

We illustrate our notion of *open parity games* that populates the category \mathbf{OPG}_M . Open parity games are parity games that come additionally with interfaces called *open ends*, along which they can be composed. An example is in Fig. 1(a). The domain interface consists of two *open ends*, 1 and 2, and the codomain interface simply of 1', while the internal positions are *a* and *b*, each equipped with a role and

$$\begin{array}{ccc}
 (\text{Int}(\mathbb{F}_{\text{tr}}(\Sigma_M^{\text{int}})) \simeq) \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) & \xrightarrow{\mathcal{R}_M} & \mathbf{OPG}_M (\simeq \text{Int}(\mathbf{OPG}_M^{\text{r}})) \\
 \swarrow \llbracket - \rrbracket_M & & \nwarrow \mathcal{W}_M \\
 & \text{Int}(\mathbf{FinScottL}_{!M}^{\text{op}}) &
 \end{array}$$

Figure 3: An outline. \mathcal{R}_M is the *realization functor* that maps a string diagram to an open parity game; \mathcal{W}_M is the *winning position functor* which extends the usual definition of winning positions in parity games; and $\llbracket - \rrbracket_M$ is the *interpretation functor*.

a priority, as usual in parity games. We give an example of sequential composition in Fig. 1(b). There, arrows are composed through the intermediate interfaces $1', 1$ between the two games. We also have a parallel composition \oplus , and Fig. 2(b) gives an example of how a cycle (in Fig. 2(a)) can be defined using sequential and parallel composition.

The technical key in Fig. 3 is the identification of *compact closed* structures. All the three categories are compact closed; moreover, we identify the prop $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ to be a *free* compact closed category in a suitable sense. The functors \mathcal{R}_M and $\llbracket - \rrbracket_M$ arise by the freeness; the commutativity is proved by the freeness, too.

In this paper, we find a new application of props as graphical languages in parity games. It allows one to solve parity games in a compositional manner (Ex. 5.11), thanks also to the identification of the right semantical domain (namely $\text{Int}(\mathbf{FinScottL}_{!M}^{\text{op}})$) that retains the right level of information in intermediate results. Such compositional solution has multiple potential applications. Firstly, the categorical structure we identify has a lot in common with those used for HOMC [15, 32]. Therefore we expect we can streamline known HOMC algorithms and reveal their categorical essences. Secondly, we will pursue algorithmic applications, such as efficient divide-and-conquer algorithms and those which accommodate blackbox components as part of a game.

Organization. In §2, we introduce open parity games. In §3, we define the graphical language $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ and the realization functor \mathcal{R}_M . In §4, we define the semantic category $\text{Int}(\mathbf{FinScottL}_{!M}^{\text{op}})$ and the interpretation functor $\llbracket - \rrbracket_M$. In §5, we define the winning position functor \mathcal{W}_M and establish the triangle in Fig. 3. We also exhibit an example of compositional solution of a parity game in Ex. 5.11. We conclude in §6.

Related Work. We use $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ as the graphical language for open parity games. The use of monoidal categories as graphical languages dates back to [29]. There have been numerous such languages; see [31] for a survey. Languages that compositionally describe graph-like structures are of particular interest to us: [10] describes the algebra of directed acyclic graphs but does not consider cyclic structures; [2] describes open Petri nets, and compositionality is achieved “externally” by the use of cospans.

In particular, props have been used extensively as graphical languages. They define graphical languages as models for some mathematical structures (signal flow diagrams [5], networks [1], Petri nets [3], automata [30], and the ZX-calculus [6, 7] respectively) and prove that the graphical language is equivalent to the category that they are studying. They can therefore transfer properties of the graphical language (for example, decidability of equivalence of diagrams) to the category they are studying. In our work, however, we use the graphical language for expressing open parity games compositionally, and we are not necessarily interested in equivalence between $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ and \mathbf{OPG}_M (see also Rem. 3.16).

Note that our work uses a 2-colored prop for modeling the two possible directions for edges in an open parity game, while [3, 5, 27] only have a single type of edges, which are undirected. In [30], the

authors use a colored prop to model different kinds of edges, and in particular, they use two colors to model directed edges.

Kissinger gives a general construction of the free traced symmetric monoidal categories $\mathbb{F}_{\text{tr}}(\Sigma)$, which are also props [24]. This is related to the free compact closed category $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ in the present paper, as explained in §3. Free traced monoidal categories are also given in [22] in the study of attribute grammars, where many-to-one signatures are treated while Kissinger's paper treats many-to-many signatures.

Another related work is [13], which introduces the concept of composing games. Their approach is mainly applied to economic models, and they use a symmetric monoidal category for compositional game theories. However, their framework is different from ours in the sense that the objects along which games are composed have different meanings: in our framework, games are composed along graph edges, while in theirs, games are composed along interfaces describing player choices, game utility, etc.

Notation. We use the following notations. (i) $[m] := \{1, \dots, m\}$. (ii) We write the unit and the multiplication of a free monoid C^* as ε and \cdot . (iii) For $w \in \{\mathbb{r}, \mathbb{l}\}^*$, \vec{w} is the number of \mathbb{r} in w , and \overleftarrow{w} is the number of \mathbb{l} in w . (iv) $\mathbb{N}_M := \{0, \dots, M\}$ where $M \in \mathbb{N}$. (v) $\mathbb{N}_{\geq 1} := \{i \in \mathbb{N} \mid i \geq 1\}$. (vi) $\mathbb{r}^\perp := \mathbb{l}$, $\mathbb{l}^\perp := \mathbb{r}$, and $(w_1 \cdots w_n)^\perp := w_n^\perp \cdots w_1^\perp$ where $w_1 \cdots w_n \in \{\mathbb{r}, \mathbb{l}\}^*$. (vii) $!_X^\emptyset$ is the unique function from \emptyset to X . (viii) For $f: A \rightarrow C$ and $g: B \rightarrow C$, we write $[f, g]: A + B \rightarrow C$ for the copairing function, i.e., $[f, g](a) = f(a)$ and $[f, g](b) = g(b)$. (ix) We often omit the injections $\text{in}_i: A_i \rightarrow A_1 + A_2$, i.e., for example, we may write $a \in A + B$ for $\text{in}_1(a) \in A + B$, if no confusion happens.

2 Categories of Open Parity Games

We introduce the notion of open parity games. It extends parity games by adding *open ends* to the game, which are used to define composition of parity games; specifically, we obtain a compact closed category \mathbf{OPG}_M of open parity games (Def. 2.11).

In this paper, we often encounter situations where the structure of interest can be organized both as a *traced symmetric monoidal category (TSMC)* or as a *compact closed category (CpCC)*. (Specifically, we have three such classes of structures, yielding three TSMCs and CpCCs. See Fig. 3.) While our applicational interests lie in the CpCC structures, we work mostly with the TSMC structures for technical convenience, and use the Int construction [21] to define the CpCC structures from them (\mathbf{OPG}_M is defined as $\text{Int}(\mathbf{OPG}_M^{\text{r}})$, and we show that $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is equivalent to $\text{Int}(\mathbb{F}_{\text{tr}}(\Sigma_M^{\text{int}}))$ for some signature Σ_M^{int}).

2.1 Open Parity Games

Recall that a *parity game* is a tuple $\mathcal{A} = (Q, E, \rho, \omega)$ where (Q, E) is a finite directed graph of *positions* and *edges*, $\rho: Q \rightarrow \{\exists, \forall\}$ is the *role function*, and $\omega: Q \rightarrow \mathbb{N}$ is the *priority function*. An *infinite play* on \mathcal{A} is an infinite sequence $q_0 q_1 \cdots \in Q^{\mathbb{N}}$ such that, for all $i \geq 0$, $(q_i, q_{i+1}) \in E$. A *finite play* is defined similarly. Let Play_{\exists} and Play_{\forall} be the sets of finite plays $q_0 \dots q_n$ such that $\rho(q_n) = \exists$ or $\rho(q_n) = \forall$, respectively. An infinite play $q_0 q_1 \dots$ is *winning* for \exists if the maximum priority appearing infinitely often in $\omega(q_0) \omega(q_1) \dots$ is even. A finite play $q_0 \dots q_n$ is winning for \exists if $\rho(q_n) = \forall$. A *strategy* of \exists is a partial function $\sigma_{\exists}: \text{Play}_{\exists} \rightarrow Q$ such that $(q_n, \sigma_{\exists}(q_0 \dots q_n)) \in E$ if $\sigma_{\exists}(q_0 \dots q_n)$ is defined. For any position q and strategies σ_{\exists} and σ_{\forall} , we denote by $\text{play}_q^{\sigma_{\exists}, \sigma_{\forall}}$ the unique play starting from q and consistent with both σ_{\exists} and σ_{\forall} . A strategy σ_{\exists} is *winning* for \exists from $q \in Q$ if for all strategies σ_{\forall} , $\text{play}_q^{\sigma_{\exists}, \sigma_{\forall}}$ is winning for \exists . A position $q \in Q$ is *winning* for \exists if there is a strategy σ_{\exists} winning for \exists from q . We define *open parity games* by extending parity games with open ends.

Definition 2.1 (open parity game). An *open parity game* from \bar{m} to \bar{n} is a tuple $(\bar{m}, \bar{n}, Q, E, \rho, M, \omega)$ such that the following conditions are satisfied:

1. $\bar{m} = (m_r, m_q)$ and $\bar{n} = (n_r, n_q)$ are pairs of natural numbers, where \bar{m} represents the domain interface of the game and \bar{n} the codomain interface.
2. Q is a finite set, whose elements are called *internal positions*.
3. E is a relation $E \subseteq ([m_r + n_q] + Q) \times ([n_r + m_q] + Q)$, whose element is called an *edge*. Moreover, for any $s \in [m_r + n_q]$, there is a unique $s' \in [n_r + m_q] + Q$ such that $(s, s') \in E$; and similarly for any $t \in [n_r + m_q]$, there is a unique $t' \in [m_r + n_q] + Q$ such that $(t', t) \in E$.
4. ρ is a function $\rho : Q \rightarrow \{\exists, \forall\}$, which assigns a *role* to each internal position.
5. $M \in \mathbb{N}$ is called the *maximal rank* and $\omega : Q \rightarrow \mathbb{N}_M$ is called the *priority function*.

We call an element of $([m_r + n_q] + [n_r + m_q]) + Q$ a *position*, one of $[m_r + n_q] + [n_r + m_q]$ an *open end*, one of $[m_r + n_q]$ an *entry position*, and one of $[n_r + m_q]$ an *exit position*.

We extend the priority function ω to $\omega : ([m_r + n_q] + [n_r + m_q]) + Q \rightarrow \mathbb{N}_M$ by $\omega(i) = 0$ for $i \in [m_r + n_q] + [n_r + m_q]$, i.e., we define the priority of each open end to be 0.

Example 2.2. The open parity game in Fig. 1(a) is the tuple $(\bar{m}, \bar{n}, Q, E, \rho, M, \omega)$ where

$$\begin{aligned} \bar{m} &= (1, 1), & \bar{n} &= (1, 0), & Q &= \{a, b\}, & \rho(a) &= \exists, & \rho(b) &= \forall, \\ 1Ea, & aE2, & aEb, & bEa, & bE1', & \omega(a) &= 1, & \omega(b) &= 2, & \text{and } M &= 2. \end{aligned}$$

In Fig. 1(a), the open end 1 is the entry position in $[m_r + n_q] = [1 + 0]$. The open ends 1' and 2 are the exit positions in $[n_r + m_q] = [1 + 1]$. The two boxes are internal positions in Q , with annotations on roles ρ and priorities ω . As usual, E is depicted by arrows.

Condition (3) of Def. 2.1 requires that a unique outgoing/incoming edge from/to an entry/exit position, respectively. This condition can be enforced by adding some dummy positions.

The following definition is a first step towards introducing a trace operator.

Definition 2.3 (rightward open parity game). An open parity game $\mathcal{A} = (\bar{m}, \bar{n}, Q, E, \rho, M, \omega)$ is *rightward* if $\bar{m} = (m_r, 0_q)$ and $\bar{n} = (n_r, 0_q)$ for some m_r and n_r .

In the last definition, we require each open end in \bar{m} and \bar{n} to be headed in the right. Note that we do not impose the same requirement on (internal) edges in E —a rightward open parity game may contain cycles.

2.2 A Traced Symmetric Monoidal Category of Rightward Open Parity Games

We shall first define the traced symmetric monoidal category \mathbf{OPG}_M^r of rightward open parity games. It yields the compact closed category \mathbf{OPG}_M of open parity games by the Int construction (see Fig. 3).

In fact, we do so restricting the priorities to be below a certain natural number M , talking about \mathbf{OPG}_M^r and \mathbf{OPG}_M . The reason for doing so is discussed in Rem. 4.4.

In what follows, we assume that a given rightward open parity game \mathcal{A} is of the form $\mathcal{A} = ((m^{\mathcal{A}}, 0), (n^{\mathcal{A}}, 0), Q^{\mathcal{A}}, E^{\mathcal{A}}, \rho^{\mathcal{A}}, M, \omega^{\mathcal{A}})$. The convention also applies to \mathcal{B} .

We need an equivalence relation on the set of rightward open parity games to define \mathbf{OPG}_M^r . For our purpose here, we define the equivalence in terms of structure-preserving bijections. It is easy to define an equivalence relation on open parity games in the same way.

Definition 2.4 (equivalence relation \sim on rightward open parity games). We define an equivalence relation \sim on the set of rightward open parity games as follows: $\mathcal{A} \sim \mathcal{B}$ if $m^{\mathcal{A}} = m^{\mathcal{B}}$, $n^{\mathcal{A}} = n^{\mathcal{B}}$, and there is a bijection $\eta : Q^{\mathcal{A}} \rightarrow Q^{\mathcal{B}}$ such that the following conditions are satisfied: (i) for $(s, t) \in ([m^{\mathcal{A}}] + Q^{\mathcal{A}}) \times ([n^{\mathcal{A}}] + Q^{\mathcal{A}})$, $(s, t) \in E^{\mathcal{A}} \iff (\bar{\eta}(s), \bar{\eta}(t)) \in E^{\mathcal{B}}$, (ii) for $s \in Q^{\mathcal{A}}$, $\rho^{\mathcal{A}}(s) = \rho^{\mathcal{B}}(\eta(s))$, and (iii) for $s \in Q^{\mathcal{A}}$, $\omega^{\mathcal{A}}(s) = \omega^{\mathcal{B}}(\eta(s))$. Here we extend η to $\bar{\eta} : (\mathbb{N} + Q^{\mathcal{A}}) \rightarrow (\mathbb{N} + Q^{\mathcal{B}})$ by $\bar{\eta}(n) = n$ for $n \in \mathbb{N}$.

We define the category \mathbf{OPG}_M^r as follows. Objects are natural numbers, and a morphism from m to n is an equivalence class $[\mathcal{A}]_{\sim}$ of rightward open parity games from $(m, 0)$ to $(n, 0)$. The identity and composition of morphisms are given by $\text{id}_n := [\mathcal{I}_n]_{\sim}$ and $[\mathcal{A}]_{\sim}; [\mathcal{B}]_{\sim} := [\mathcal{A}; \mathcal{B}]_{\sim}$, where \mathcal{I}_n and $\mathcal{A}; \mathcal{B}$ are given in Def. 2.5 and Def. 2.6 below, respectively.

Definition 2.5 (identity). For $n \in \mathbb{N}$, we define the *identity game* \mathcal{I}_n as $((n, 0), (n, 0), \emptyset, E, !_{\{\exists, \forall\}}^{\emptyset}, M, !_{\mathbb{N}_M}^{\emptyset})$ where $E = \{(a, a) \mid a \in [n]\}$.

Fig. 4 shows the identity id_3 .

Next, we define the sequential composition $\mathcal{A}; \mathcal{B}$ of rightward open parity games. The intuition is to connect each exit position of \mathcal{A} with the corresponding entry position of \mathcal{B} , and then to hide those interface open ends. Fig. 1(b) in the introduction illustrates this construction.

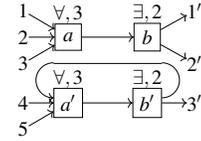
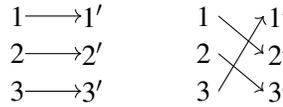


Figure 4: id_3 . Figure 5: $\sigma_{2,1}$. Figure 6: Parallel composition of Fig. 1(b) & Fig. 2(a).

Definition 2.6 (sequential composition). Let \mathcal{A} and \mathcal{B} be rightward open parity games and $n^{\mathcal{A}} = m^{\mathcal{B}}$. We define the sequential composition $\mathcal{A}; \mathcal{B}$ as follows: $\mathcal{A}; \mathcal{B} = ((m^{\mathcal{A}}, 0), (n^{\mathcal{B}}, 0), Q^{\mathcal{A}} + Q^{\mathcal{B}}, E^{\mathcal{A}; \mathcal{B}}, [\rho^{\mathcal{A}}, \rho^{\mathcal{B}}], M, [\omega^{\mathcal{A}}, \omega^{\mathcal{B}}])$, where $E^{\mathcal{A}; \mathcal{B}} = E^{\mathcal{A}} \setminus (([m^{\mathcal{A}}] + Q^{\mathcal{A}}) \times [n^{\mathcal{A}}]) + E^{\mathcal{B}} \setminus ([m^{\mathcal{B}}] \times ([n^{\mathcal{B}}] + Q^{\mathcal{B}})) + \{(s, s') \mid \text{there exists } a \in [n^{\mathcal{A}}] = [m^{\mathcal{B}}] \text{ such that } (s, a) \in E^{\mathcal{A}} \text{ and } (a, s') \in E^{\mathcal{B}}\}$.

We can show associativity and unitality up to structure-preserving bijection, which entails that \mathbf{OPG}_M^r is a category by Def. 2.4.

We also define a parallel (or vertical) composition \oplus of rightward open parity games, which gives a monoidal product structure of \mathbf{OPG}_M^r by $[\mathcal{A}]_{\sim} \oplus [\mathcal{B}]_{\sim} = [\mathcal{A} \oplus \mathcal{B}]_{\sim}$. Fig. 6 gives an example, notice that the open ends in the second game need to be shifted, for which we need the following definition: for $l \in \mathbb{N}$ and $s \in [m] + Q$, let $s^{\downarrow l} \in [l + m] + Q$ be defined by $s^{\downarrow l} = l + s$ if $s \in [m]$, and $s^{\downarrow l} = s$ if $s \in Q$.

Definition 2.7 (parallel composition). Let \mathcal{A} and \mathcal{B} be rightward open parity games. The parallel composition $\mathcal{A} \oplus \mathcal{B}$ is defined as follows: $\mathcal{A} \oplus \mathcal{B} = ((m^{\mathcal{A}} + m^{\mathcal{B}}, 0), (n^{\mathcal{A}} + n^{\mathcal{B}}, 0), Q^{\mathcal{A}} + Q^{\mathcal{B}}, E^{\mathcal{A} \oplus \mathcal{B}}, [\rho^{\mathcal{A}}, \rho^{\mathcal{B}}], M, [\omega^{\mathcal{A}}, \omega^{\mathcal{B}}])$, where $E^{\mathcal{A} \oplus \mathcal{B}}$ is given by $E^{\mathcal{A} \oplus \mathcal{B}} = E^{\mathcal{A}} + \{(s^{\downarrow m^{\mathcal{A}}}, t^{\downarrow n^{\mathcal{A}}}) \mid (s, t) \in E^{\mathcal{B}}\}$.

The following game swaps the order of entry positions and that of exit positions. This makes \mathbf{OPG}_M^r a symmetric monoidal category. Fig. 5 shows the swap game $\sigma_{2,1}$.

Definition 2.8. (swap) For any $m, n \in \mathbb{N}_{\geq 1}$, we define the *swap game* $\sigma_{m,n}$ as follows: $\sigma_{m,n} = ((m + n, 0), (n + m, 0), \emptyset, E^{\sigma_{m,n}}, !_{\{\exists, \forall\}}^{\emptyset}, M, !_{\mathbb{N}_M}^{\emptyset})$, where $E^{\sigma_{m,n}} = \{(a, n + a) \mid a \in [m]\} \cup \{(m + a, a) \mid a \in [n]\}$.

Cycles are essential to parity games: without them there would not be any infinite play. To introduce cycles in rightward open parity games, we use a trace operator on \mathbf{OPG}_M^r , as illustrated in Fig. 2(c).

Definition 2.9 (trace operator of \mathbf{OPG}_M^r). Let l, m , and n be objects in \mathbf{OPG}_M^r . We define the trace operator $\text{tr}_{l,m,n} : \mathbf{OPG}_M^r(l + m, l + n) \rightarrow \mathbf{OPG}_M^r(m, n)$ as follows. Let $\mathcal{A} \in \mathbf{OPG}_M^r(l + m, l + n)$, i.e., let $m^{\mathcal{A}} = l + m$ and $n^{\mathcal{A}} = l + n$. Then, $\text{tr}_{l,m,n}([\mathcal{A}]_{\sim}) := [\text{tr}_{l,m,n}(\mathcal{A})]_{\sim}$ where

$$\text{tr}_{l,m,n}(\mathcal{A}) = ((m, 0), (n, 0), Q^{\mathcal{A}}, E^{\text{tr}_{l,m,n}(\mathcal{A})}, \rho^{\mathcal{A}}, M, \omega^{\mathcal{A}}), \text{ where}$$

$$E^{\text{tr}_{l,m,n}(\mathcal{A})} = \{(s, s') \in ([m] + Q^{\mathcal{A}}) \times ([n] + Q^{\mathcal{A}}) \mid s^{\downarrow l} E^{\mathcal{A}} a_1 E^{\mathcal{A}} \dots E^{\mathcal{A}} a_k E^{\mathcal{A}} s^{\downarrow l} \text{ for some } k \in \mathbb{N}, (a_i)_i \in [l]^k\}.$$

Here is the main result of this section. With the given definitions, the proof is (lengthy but) routine work.

Theorem 2.10 (\mathbf{OPG}_M^r). *The data $(\mathbf{OPG}_M^r, \oplus, \emptyset, \sigma, \text{tr})$ defined so far constitutes a strict traced symmetric monoidal category, where \emptyset denotes the obvious empty game.* \square

2.3 A Compact Closed Category of Open Parity Games

To obtain the category \mathbf{OPG}_M of open parity games, we use the *Int construction* [21] (see also [18] for some correction). It is a *free* construction from a traced symmetric monoidal category \mathbb{C} to a compact closed category $\text{Int}(\mathbb{C})$. We briefly explain how it is defined here, but see the full version or [21] for more details.

Let \mathbf{CpCC} be the 2-category of (locally small) CpCCs, compact closed functors, and monoidal natural transformations. Note that its 2-cells automatically respect compact closed structures and are monoidal natural *isomorphisms* [20, Proposition 7.1]. Also, let \mathbf{TrSMC}_g be the 2-category of (locally small) TSMCs, traced symmetric strong monoidal functors, and monoidal natural *isomorphisms*.

Then the Int construction is a left biadjoint to the embedding $\mathbf{TrSMC}_g \rightarrow \mathbf{CpCC}$. Specifically, given a traced symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I}, \sigma, \text{tr})$, the category $\text{Int}(\mathbb{C})$ is defined as follows: An object of $\text{Int}(\mathbb{C})$ is a pair (X_+, X_-) of objects of \mathbb{C} . Then $\text{Int}(\mathbb{C})((X_+, X_-), (Y_+, Y_-)) := \mathbb{C}(X_+ \otimes Y_-, Y_+ \otimes X_-)$, and $\text{id}_{(X_+, X_-)} := \text{id}_{X_+ \otimes X_-}$. Notably, for $f \in \text{Int}(\mathbb{C})((X_+, X_-), (Y_+, Y_-))$ and $g \in \text{Int}(\mathbb{C})((Y_+, Y_-), (Z_+, Z_-))$, the composite of f and g is defined using the trace operator, namely by $\text{tr}_{Y_-, X_+ \otimes Z_-, Z_+ \otimes X_-}^{\mathbb{C}}((\sigma_{Z_+, Y_-} \otimes \text{id}_{X_-}) \circ (g \otimes \text{id}_{X_-}) \circ (\text{id}_{Y_+} \otimes \sigma_{X_-, Z_-}) \circ (f \otimes \text{id}_{Z_-}) \circ (\sigma_{Y_-, X_+} \otimes \text{id}_{Z_-}))$. See [21] for details, including diagrammatic illustration.

Definition 2.11 (\mathbf{OPG}_M). Let \mathbf{OPG}_M be the compact closed category $\text{Int}(\mathbf{OPG}_M^r)$ of open parity games.

The following proposition is trivial from the definition.

Proposition 2.12. *A morphism of \mathbf{OPG}_M is an \sim -equivalence class of open parity games (in Def. 2.1).* \square

The compact closed structure is the basis of our compositional approach to parity games. The structure serves as *game constructors*, and the compact closed functor $\mathscr{W}_M : \mathbf{OPG}_M \rightarrow \text{Int}(\mathbf{FinScott}_{!M}^{op})$ (see Fig. 3) given in §5 computes compositionally if an entry position in a composed game wins.

3 A Graphical Language of Open Parity Games

In this section, we introduce the category $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ as a graphical language for open parity games. The category is a *prop*, a symmetric monoidal category version of the notion of Lawvere theory whose use has been actively pursued recently [1, 5, 7]. It gives to open parity games introduced in §2 a language of string diagrams generated by certain generators and equations. Moreover, we find that the category $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is free in two senses: (i) as the prop induced by a theory $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ for open parity games; and (ii) as a compact closed category $\text{Int}(\mathbb{F}_{\text{tr}}(\Sigma_M^{\text{int}}))$ (see Fig. 3). The second freeness is exploited in the compositional definition of the interpretation functor $\llbracket - \rrbracket_M$ in Fig. 3.

3.1 The Graphical Language $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$

We define $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ as a colored prop constructed from a symmetric monoidal theory $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$. For the detail of this prop construction, the reader can consult, e.g., [7].

Definition 3.1 (*C-prop, morphism, C-Prop*). Let C be a set (of *colors*). A *C-prop* is a small strict symmetric monoidal category where the monoid of all the objects is the free monoid C^* of C . A *C-prop morphism* between C -props is a strict symmetric monoidal functor that is the identity on objects. We write **C-Prop** for the category of C -props and C -prop morphisms.

In this paper, we consider $\{\ulcorner, \llcorner\}$ -props. The colors \ulcorner and \llcorner represent “rightward” and “leftward”, respectively. They intuitively correspond to m_{\ulcorner} and m_{\llcorner} of $\bar{m} = (m_{\ulcorner}, m_{\llcorner})$ in an open parity game.

We want to define $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ as a free $\{\ulcorner, \llcorner\}$ -prop. A free C -prop is generated from a *C-symmetric monoidal theory* (*C-SMT* for short), i.e., a pair of a *C-signature* and a set of *C-equations*. Intuitively, given a *C-SMT*, morphisms of the corresponding free C -prop are terms built freely from the signature (as well as sequential and parallel composition) and quotiented by the equations.

Definition 3.2 (*C-signature, morphism, C-Sig*). A *C-signature* is a functor $\Sigma : C^* \times C^* \rightarrow \mathbf{Set}$ where the free monoid C^* is thought of as a discrete category. A *C-signature morphism* $f : \Sigma \rightarrow \Sigma'$ is a natural transformation from Σ to Σ' . We write **C-Sig** for the category of C -signatures and C -signature morphisms.

Thus, **C-Sig** = $\mathbf{Set}^{C^* \times C^*}$. We define an $\{\ulcorner, \llcorner\}$ -signature Σ_M^{opg} , which is used to define the graphical language $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$. In the signature Σ_M^{opg} , for each domain $w \in \{\ulcorner, \llcorner\}^*$, codomain $u \in \{\ulcorner, \llcorner\}^*$, role $r \in \{\forall, \exists\}$, and priority $p \in \mathbb{N}_M$, there is a single generator $n_{r,p}^{w,u}$ that represents the type of nodes in open parity games with these specific domain, codomain, role, and priority.

Definition 3.3 ($\{\ulcorner, \llcorner\}$ -signature Σ_M^{opg}). For $w, u \in \{\ulcorner, \llcorner\}^*$, we define $\Sigma_M^{\text{opg}}(w, u)$ as follows: Let $N_{w,u} = \{n_{r,p}^{w,u} \mid r \in \{\forall, \exists\}, p \in \mathbb{N}_M\}$. Then $\Sigma_M^{\text{opg}}(\varepsilon, \ulcorner \cdot \llcorner) = N_{\varepsilon, \ulcorner \cdot \llcorner} \cup \{d_{\ulcorner}\}$, $\Sigma_M^{\text{opg}}(\llcorner \cdot \ulcorner, \varepsilon) = N_{\llcorner \cdot \ulcorner, \varepsilon} \cup \{e_{\llcorner}\}$, and $\Sigma_M^{\text{opg}}(w, u) = N_{w,u}$ otherwise.

The generators d_{\ulcorner} and e_{\llcorner} intuitively represent a unit and counit over \ulcorner , respectively. We now turn to equations, for which we first need to define terms of a *C-SMT*. They are given by the following free construction. Let $\cup_{\text{sig}}^C : \mathbf{C-Prop} \rightarrow \mathbf{C-Sig}$ be the obvious forgetful functor.

Theorem 3.4 ([11, 16]). *The forgetful functor \cup_{sig}^C has a left adjoint $\mathbb{F}_{\text{sig}}^C : \mathbf{C-Sig} \rightarrow \mathbf{C-Prop}$.* \square

For the unit $\eta^{\text{sig}} : \text{Id}_{\mathbf{C-Sig}} \rightarrow \cup_{\text{sig}}^C \circ \mathbb{F}_{\text{sig}}^C$, we identify $(\eta_{\Sigma}^{\text{sig}})_{w,u}(f) \in \cup_{\text{sig}}^C(\mathbb{F}_{\text{sig}}^C(\Sigma))(w, u) = \mathbb{F}_{\text{sig}}^C(\Sigma)(w, u)$ with $f \in \Sigma(w, u)$ for simplicity of presentation.

Definition 3.5 (*C-SMT*) A *C-colored symmetric monoidal theory* (*C-SMT* for short) is a tuple (Σ, E, l, r) where Σ and E are C -signatures and $l, r : E \rightarrow \cup_{\text{sig}}^C(\mathbb{F}_{\text{sig}}^C(\Sigma))$ are C -signature morphisms.

We often write simply (Σ, E) for (Σ, E, l, r) . We call $\mathbb{F}_{\text{sig}}^C(\Sigma)$ the set of *terms* generated by Σ , and E the set of (*C*-)equations in Σ , where each $e \in E$ represents the equation $l(e) = r(e)$.

Definition 3.6 (*SMT* $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$). We complete the definition of the $\{\ulcorner, \llcorner\}$ -SMT $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}, l_M^{\text{opg}}, r_M^{\text{opg}})$ by giving the equations: $(e_{\ulcorner} \oplus \text{id}_{\llcorner}) \circ (\text{id}_{\ulcorner} \oplus d_{\llcorner}) = \text{id}_{\llcorner}$ and $(\text{id}_{\ulcorner} \oplus e_{\llcorner}) \circ (d_{\ulcorner} \oplus \text{id}_{\llcorner}) = \text{id}_{\ulcorner}$.

The $\{\ulcorner, \llcorner\}$ -SMT $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ describes open parity games, and the equations in Fig. 7(c) represent the coherence conditions of compact closed categories.

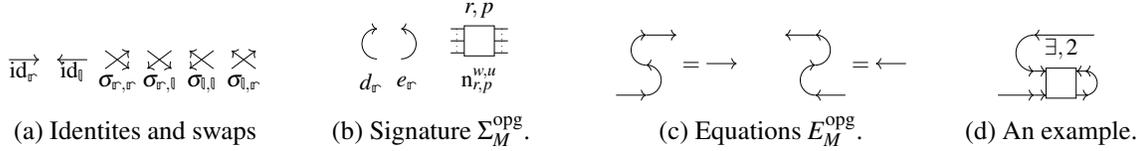
In general, a *C-SMT* induces a free C -prop $\mathbb{F}(\Sigma, E)$, whose arrows give a graphical language.

Definition 3.7 (free prop $\mathbb{F}(T)$ [1, 7]). Let $T = (\Sigma, E, l, r)$ be a *C-SMT*. We define a C -prop $\mathbb{F}(T)$ as the coequalizer of $l^\dagger, r^\dagger : \mathbb{F}_{\text{sig}}^C(E) \rightarrow \mathbb{F}_{\text{sig}}^C(\Sigma)$ in **C-Prop** where l^\dagger and r^\dagger are, respectively, the transposition of $l, r : E \rightarrow \cup_{\text{sig}}^C(\mathbb{F}_{\text{sig}}^C(\Sigma))$ in **C-Sig** by $\mathbb{F}_{\text{sig}}^C \dashv \cup_{\text{sig}}^C$.

Definition 3.8 (graphical language $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$). We define $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ by Def. 3.6 and Def. 3.7.

By definition, an object of $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is an element of $\{\ulcorner, \llcorner\}^*$, and a morphism is the class of terms generated by sequential and parallel composition applied to constants in Σ_M^{opg} , and quotiented by the congruence produced by the equations in E_M^{opg} (again under sequential and parallel composition). The prop $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is illustrated in Fig. 7(a), Fig. 7(b), and Fig. 7(c).

Example 3.9. Fig. 7(d) is the morphism $(\text{id}_{\llcorner} \oplus e_{\ulcorner}) \circ (\text{id}_{\llcorner} \oplus n_{\exists, 2}^{\ulcorner \cdot \ulcorner, \llcorner \cdot \ulcorner}) \circ (d_{\llcorner} \oplus \text{id}_{\ulcorner})$ from \ulcorner to \llcorner in $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$.

Figure 7: Illustration of $(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$.

3.2 Free Compact Closedness of $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ and the Full Functor \mathcal{R}_M

We show that the graphical language $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is a free compact closed category, so that we can freely define a compact closed functor from $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ to any compact closed category \mathbb{C} with additional structure. This way we obtain the realization functor $\mathcal{R}_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \mathbf{OPG}_M$ (see Fig. 3); we show that \mathcal{R}_M is full, meaning that every open parity game has a presentation in $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$. Due to the space limitation, we put the detailed information that is written here in the full version.

We need some definitions for proving that $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is a free compact closed category. We call an object of the category $\mathbf{Set}^{\{\ulcorner, \lrcorner\}^* \times \{\ulcorner, \lrcorner\}^*}$ a *compact closed signature (CCS, for short)*. Recall that the signature Σ_M^{opg} consists of nodes $n_{r,p}^{w,u}$ of open parity games and the unit d_r and counit e_r of compact closed structure. We define a CCS Σ_M that is a signature of open parity games without the compact closed structure d_r or e_r .

Definition 3.10 (CCS Σ_M). We define a CCS Σ_M by $\Sigma_M(w, u) := \{n_{r,p}^{w,u} \mid r \in \{\exists, \forall\} \text{ and } p \in \mathbb{N}_M\}$.

To state the free compact closedness of $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$, we define a *valuation*, which defines a way to interpret elements of a signature into a compact closed category.

Definition 3.11 (valuation). For a CCS Σ and a compact closed category \mathbb{C} , a *valuation* of Σ into \mathbb{C} is a pair $(V_r, (V_{w,u})_{w,u})$ such that (i) $V_r \in \text{ob}(\mathbb{C})$ and (ii) $V_{w,u} : \Sigma(w, u) \rightarrow \mathbb{C}(V_w^*, V_u^*)$ for $w, u \in \{\ulcorner, \lrcorner\}^*$ where V_w^* ($w \in \{\ulcorner, \lrcorner\}^*$) is defined as follows: $V_{d_1 \dots d_n}^* := V_{d_1} \otimes \dots \otimes V_{d_n}$ where $V_{\lrcorner} := V_r^\perp$.

Definition 3.12 (action on valuations). Given a compact closed functor $F : \mathbb{C} \rightarrow \mathbb{D}$ and a valuation V of Σ into \mathbb{C} , the *action* $(F \circ V)$ on V by F is defined by (i) $(F \circ V)_r := F(V_r)$ and (ii) $(F \circ V)_{w,u}(f) := (\phi_u^F)^{-1} \circ F(V_{w,u}(f)) \circ \phi_w^F$, where ϕ_w^F (and ϕ_u^F) are defined as follows: for any $w = \ulcorner^{i_1} \dots \ulcorner^{i_n} \in \{\ulcorner, \lrcorner\}^*$ where each \ulcorner^{i_j} is either \ulcorner or \ulcorner^\perp , the morphism $\phi_w^F : F(V_r)^{i_1} \otimes \dots \otimes F(V_r)^{i_n} \rightarrow F(V_r^{i_1} \otimes \dots \otimes V_r^{i_n})$ is the isomorphism given by the fact that F respects the compact closed structures.

Finally, we prove the free compact closedness of $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ by using the above definitions.

Theorem 3.13 (free compact closedness of $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$). *The prop $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is a strict compact closed category. Furthermore, $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ is a free compact closed category, i.e., there exists a valuation η_{Σ_M} of Σ_M into $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ such that, for any compact closed category \mathbb{C} and any valuation V of Σ_M into \mathbb{C} , there exists a unique (up to iso) compact closed functor $F : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \mathbb{C}$ such that $(F \circ \eta_{\Sigma_M}) = V$. \square*

By the general result (Thm. 3.13), we define the realization functor $\mathcal{R}_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \mathbf{OPG}_M$ (see Fig. 3):

Definition 3.14. (realization functor \mathcal{R}_M) We let the *realization functor* $\mathcal{R}_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \mathbf{OPG}_M$ be the functor determined by Thm. 3.13 with $\mathcal{R}_M(\ulcorner) = (1, 0)$ and $\mathcal{R}_M(n_{r,p}^{w,u}) = ((\overrightarrow{w}, \overleftarrow{w}), (\overrightarrow{u}, \overleftarrow{u}), \{\star\}, E, \{\star \mapsto r\}, M, \{\star \mapsto p\})$ where $E = \{(a, \star) \mid a \in [\overrightarrow{w} + \overleftarrow{u}]\} \cup \{(\star, a) \mid a \in [\overrightarrow{u} + \overleftarrow{w}]\}$.

The following theorem says that every open parity game in \mathbf{OPG}_M can be represented as a graphical one, i.e., a morphism in $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$.

Theorem 3.15 (fullness). *The functor $\mathcal{R}_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \mathbf{OPG}_M$ is full.* \square

Remark 3.16. The functor \mathcal{R}_M is not faithful (a one-node example can be easily given). In recent works on props such as [5, 30], the main interest is in the faithfulness of a semantics functor whose codomain is a well-known semantic category (that of linear relations [5], automata [30], etc.). In this case, faithfulness amounts to the *completeness* of equational axioms. We do not share this interest: in Fig. 3, the codomain of $\mathcal{R}_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \mathbf{OPG}_M$ is *not* a well-known category, and the value of a corresponding complete equational axiomatization is not clear. Faithfulness of the interpretation functor $\llbracket - \rrbracket_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \text{Int}(\mathbf{FinScottL}_{!_M}^{\text{op}})$ (introduced in §4) seems more interesting, since it amounts to an equational characterization of the equivalence of parity games in terms of who is winning. The problem seems challenging, however, given the complexity of solving parity games, and we leave it as future work. We note that, for our purpose of compositional solution of parity games (see e.g. Ex. 5.11), faithfulness of \mathcal{R}_M or $\llbracket - \rrbracket_M$ is not needed.

Kissinger gives a construction for free traced symmetric monoidal categories $\mathbb{F}_{\text{tr}}(\Sigma)$ [24]. We show an equivalence $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \simeq \text{Int}(\mathbb{F}_{\text{tr}}(\Sigma_M^{\text{int}}))$ in \mathbf{CpCC} for some 1-signature Σ_M^{int} (see the full version).

4 The Semantic Category of Open Parity Games

In this section, we define a semantic category of open parity games $\text{Int}(\mathbf{FinScottL}_{!_M}^{\text{op}})$. Grellois and Mellès restricted \mathbf{ScottL} to the full subcategory $\mathbf{FinScottL}$ [14, 15] of *finite* preordered sets, in order to introduce a fixpoint operator on $\mathbf{FinScottL}_{!_M}$ for some suitable comonad $!_M$ so that it forms a model of higher-order model checking. We use this fixpoint operator for the Int construction of $\text{Int}(\mathbf{FinScottL}_{!_M}^{\text{op}})$. We then define the *interpretation functor* $\llbracket - \rrbracket_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \text{Int}(\mathbf{FinScottL}_{!_M}^{\text{op}})$. (The reader may look at Ex. 5.11 for an example of computation of $\llbracket \mathcal{A} \rrbracket_M$ on a concrete open parity game \mathcal{A} .)

Definition 4.1 ($\mathbf{FinScottL}$ [14, 15]). The category $\mathbf{FinScottL}$ has as objects finite preordered sets $A = (|A|, \leq_A)$ and as morphisms $R : A \rightarrow B$ downward-closed binary relations R between A^{op} and B : i.e., a binary relation $R \subseteq |A| \times |B|$ such that if $a' \geq_A a$, aRb , $b \geq_B b'$, then $a'Rb'$. Composition is defined as usual: $a(S \circ R)c$ iff aRb and bSc for some $b \in |B|$. The identity on A is $\text{id}_A = \{(a', a) \mid a \leq_A a'\}$.

Note that $\mathbf{FinScottL}$ has a symmetric monoidal structure induced by the existence of finite cartesian products $1 = (\{*\}, =)$ and $(|A|, \leq_A) \times (|B|, \leq_B) = (|A| + |B|, \leq_A + \leq_B)$.

In open parity games, we have two roles: \exists and \forall . For player \exists , a choice by \forall is not predictable. This nondeterminism is represented by the finite powerset comonad \mathcal{P} .

Definition 4.2 (finite powerset comonad). The *finite powerset comonad* $(\mathcal{P}, \varepsilon^{\mathcal{P}}, \delta^{\mathcal{P}})$ on $\mathbf{FinScottL}$ is defined by $\mathcal{P}(|A|, \leq_A) := (P(|A|), \leq)$, where $X \leq Y$ iff for any $x \in X$ there exists $y \in Y$ such that $x \leq_A y$. For $R : A \rightarrow B$, $\mathcal{P}(R) := \{(X, Y) \in P(|A|) \times P(|B|) \mid \forall y \in Y, \exists x \in X, (x, y) \in R\}$. Then $\varepsilon_A^{\mathcal{P}} := \{(X, a) \in P(|A|) \times |A| \mid \exists x \in X, a \leq_A x\}$ and $\delta_A^{\mathcal{P}} := \{(X, \{Y_1, \dots, Y_n\}) \in P(|A|) \times P(P(|A|)) \mid Y_1 \cup \dots \cup Y_n \leq_{\mathcal{P}(A)} X\}$.

Priorities that are not greater than M are represented by the coloring comonad \square_M .

Definition 4.3 (coloring comonad [14, 15]). The *coloring comonad* $(\square_M, \varepsilon^{\square_M}, \delta^{\square_M})$ on $\mathbf{FinScottL}$ is defined as follows: $\square_M(|A|, \leq_A) = (\mathbb{N}_M \times |A|, \leq_{\square_M A})$ where $(p, a) \leq_{\square_M A} (q, b)$ iff $p = q$ and $a \leq_A b$. For $R : A \rightarrow B$, $\square_M(R) := \{((p, a), (p, b)) \in (\mathbb{N}_M \times |A|) \times (\mathbb{N}_M \times |B|) \mid p \in \mathbb{N}_M \text{ and } (a, b) \in R\}$. Then $\varepsilon_A^{\square_M} := \{((0, a), a') \in (\mathbb{N}_M \times |A|) \times |A| \mid a' \leq_A a\}$ and $\delta_A^{\square_M} := \{((\max(p, q), a), (p, (q, a')))) \in (\mathbb{N}_M \times |A|) \times (\mathbb{N}_M \times (\mathbb{N}_M \times |A|)) \mid a' \leq_A a\}$.

Remark 4.4. We expect that this comonad can be extended to a *graded comonad* [12, 23] so that its Kleisli category interprets all open parity games, without fixing the parity bound M . However, we do

not take this approach because it is reasonably harmless to fix the maximal parity, while the use of the complex notion of graded comonad makes it hard to see the essential idea.

Combining the above notions, we define the comonad $!_M$.

Definition 4.5 (comonad $!_M$ [14, 15]). We define a distributive law $\lambda : \mathcal{P} \circ \square_M \Rightarrow \square_M \circ \mathcal{P}$ on **FinScottL** by $\lambda_{(A, \leq_A)} := \{ (X, (p, Y)) \in P(\mathbb{N}_M \times |A|) \times (\mathbb{N}_M \times P(|A|)) \mid \forall y \in Y, \exists a \in A, (p, a) \in X \text{ and } y \leq_A a \}$, and we define a comonad $!_M = (!_M, \varepsilon^{!_M}, \delta^{!_M})$ on **FinScottL** by: (i) $!_M := \mathcal{P} \circ \square_M$, (ii) $\varepsilon^{!_M} := \varepsilon^{\mathcal{P}} \circ (\mathcal{P} * \varepsilon^{\square_M})$, and (iii) $\delta^{!_M} := (\mathcal{P} * \lambda * \square_M) \circ (\delta^{\mathcal{P}} * \delta^{\square_M})$ where $*$ is the horizontal composition of natural transformations.

In general, the Kleisli category of a comonad inherits the cartesian product from the original category, and so the Kleisli category **FinScottL** $_{!_M}$ has a cartesian products given by $(|A| + |B|, \leq_A + \leq_B)$. (Furthermore, **FinScottL** $_{!_M}$ is cartesian closed [14, 15], though we do not use this fact.)

In order to give a model of higher-order model checking, Grellois and Melliès introduced a fixpoint operator fix^{GM} on **FinScottL** $_{!_M}$ to deal with infinite plays [14, 15]. Its definition is based on the notion of *semantic run-tree*. From fix^{GM} , we get a trace operator on **FinScottL** $_{!_M}$, because having a fixpoint operator is equivalent to having a trace operator for a cartesian category [17]. We then get a trace operator tr^{GM} on **FinScottL** $_{!_M}^{\text{op}}$, since if \mathbb{C} is a traced symmetric monoidal category, then so is \mathbb{C}^{op} canonically.

Now we give the definition of tr^{GM} (the definition of fix^{GM} can be found in the full version). In order to do this, we first adapt the notion of semantic run-tree by Grellois and Melliès (which we also call *semantic run-tree*) through the correspondance above between fixpoint operators and trace operators.

Definition 4.6 (semantic run-tree for tr^{GM}). Let $R \in \mathbf{FinScottL}_{!_M}^{\text{op}}(D + A, D + B)$ and $a \in |A|$; then especially, $R \subseteq P(\mathbb{N}_M \times (|D| + |B|)) \times (|D| + |A|)$. A *semantic run-tree* ψ for R and a (for the trace operator) is a possibly infinite $(\mathbb{N}_M \times (|D| + |A| + |B|))$ -labeled tree ψ that satisfies the following conditions:

1. The label of the root of ψ is $(0, a) \in \mathbb{N}_M \times |A|$.
2. Any node of ψ that is neither a leaf nor the root has its label in $\mathbb{N}_M \times |D|$.
3. For any non-leaf node (possibly being the root that is not a leaf) of ψ with label $(p, x) \in \mathbb{N}_M \times (|D| + |A|)$, let $X \subseteq \mathbb{N}_M \times (|D| + |A| + |B|)$ be the set of the labels of all the children of the node. Then $(X, x) \in R$.
4. For any leaf node (possibly being the root that is a leaf) of ψ such that its label belongs to $\mathbb{N}_M \times (|D| + |A|)$ (rather than $\mathbb{N}_M \times |B|$) and is (p, x) , we have $(\emptyset, x) \in R$.

We write $\text{SRT}(A, B, D, R, a)$ for the set of semantic run-trees with respect to A, B, D, R and a . For a semantic run-tree $\psi \in \text{SRT}(A, B, D, R, a)$, we define $\text{leaves}(\psi) \in |\mathcal{P}(\square_M(B))| = P(\mathbb{N}_M \times |B|)$ as the set of elements $(p, b) \in \mathbb{N}_M \times |B|$ such that there exists a leaf ℓ of ψ such that: (i) the label of leaf ℓ is (p', b) for some $p' \in \mathbb{N}_M$ and (ii) p is the maximal priority encountered on the path from the leaf ℓ to the root of ψ .

A semantic run-tree is similar to a (usual) run for a parity game, except that (i) its branching models \forall 's choices, and (ii) it is induced by a suitable semantic construct R instead of a graph-theoretic notion of game. In our use of the notion (§5), R will be a ‘‘summary’’ of an open parity game, which retains the necessary data to decide who is winning yet is much smaller than the original open parity game.

Definition 4.7 (trace operator tr^{GM} [14, 15]). For every $A, B, D \in \mathbf{FinScottL}_{!_M}^{\text{op}}$, we define a trace operator $\text{tr}_{D;A,B}^{\text{GM}} : \mathbf{FinScottL}_{!_M}^{\text{op}}(D \otimes A, D \otimes B) \rightarrow \mathbf{FinScottL}_{!_M}^{\text{op}}(A, B)$ as follows:

$$\text{tr}_{D;A,B}^{\text{GM}}(R) := \{ (\text{leaves}(\psi), a) \mid \psi \in \text{SRT}(A, B, D, R, a) \text{ that meets the parity condition} \}$$

where a semantic run-tree meets the *parity condition* if for every infinite path $((p_i, x_i))_{i \in \mathbb{N}}$, the maximum priority met infinitely along the path is even (i.e., $\max\{q \mid \#\{i \mid p_i = q\} = \infty\}$ is even).

Thus $\mathbf{FinScottL}_{!M}^{op}$ is a traced symmetric monoidal category. The trace operator above is used in the (sequential) composition of $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$ given below.

Now we define the semantic category $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$ for open parity games. In §5, we explain how $\mathbf{FinScottL}_{!M}^{op}$ and $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$ serve as the semantic categories in the traced and compact closed structures, respectively, by giving a suitable *winning-position* functor \mathscr{W}_M^r from \mathbf{OPG}_M^r to $\mathbf{FinScottL}_{!M}^{op}$ and then by inducing \mathscr{W}_M (see Fig. 3).

Definition 4.8 (semantic category $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$). By applying the Int construction to the traced symmetric monoidal category $\mathbf{FinScottL}_{!M}^{op}$, we obtain the compact closed category $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$.

Remark 4.9. We have $\mathbf{FinScottL}_{!M}^{op} \cong \mathbf{FinPreord}_T$, where $\mathbf{FinPreord}$ is the category of finite pre-ordered sets and monotonic functions, and (a Kleisli morphism of) the monad T is of the following form: $\mathbf{FinScottL}_{!M}^{op}(A, B) \cong \mathbf{FinPreord}\left(A, (P^\uparrow(\mathcal{P}(\mathbf{N}_M \times |B|, \leq_{\square_{MB}})), \supseteq)\right)$, where P^\uparrow is the upward-closed powerset. This description is closed to the double-powerset style semantics for 2-player games, e.g., in [19].

We want to define an interpretation functor $\llbracket - \rrbracket_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \text{Int}(\mathbf{FinScottL}_{!M}^{op})$ that reflects the winning condition on open parity games. The idea is that, if $((j_k, p_k)_{k \in n}, i) \in \llbracket G \rrbracket_M$, then player \exists can force any play that starts from the entry position corresponding to i in G to end in one of the exit positions corresponding to the j_k 's while encountering a maximum priority of p_k . By Thm. 3.13, we obtain this functor as:

Definition 4.10 (interpretation functor $\llbracket - \rrbracket_M$). We define the *interpretation functor* $\llbracket - \rrbracket_M : \mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}}) \rightarrow \text{Int}(\mathbf{FinScottL}_{!M}^{op})$ to be the compact closed functor whose action on objects is generated by $\llbracket r \rrbracket_M = (([1], =), (\emptyset, =)) \in \text{Int}(\mathbf{FinScottL}_{!M}^{op})$ and whose action on morphisms is generated by:

$$\begin{aligned} \llbracket n_{\exists, p}^{a, b} \rrbracket_M &= \begin{cases} \emptyset & (\vec{b} + \overleftarrow{a} = 0) \\ \{(T, i) \mid (j, p) \in T \text{ for some } j \in [\vec{b} + \overleftarrow{a}]\} & (\vec{b} + \overleftarrow{a} \neq 0) \end{cases} \\ \llbracket n_{\forall, p}^{a, b} \rrbracket_M &= \begin{cases} P(\emptyset) \times [\vec{a} + \overleftarrow{b}] & (\vec{b} + \overleftarrow{a} = 0) \\ \{(T, i) \mid \{(j, p) \mid j \in [\vec{b} + \overleftarrow{a}]\} \subseteq T\} & (\vec{b} + \overleftarrow{a} \neq 0). \end{cases} \end{aligned}$$

Both morphisms above are from $\llbracket a \rrbracket_M = ([\vec{a}], [\overleftarrow{a}])$ to $\llbracket b \rrbracket_M = ([\vec{b}], [\overleftarrow{b}])$ in $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$, i.e., $P(\mathbf{N}_M \times [\vec{b} + \overleftarrow{a}]) \mapsto [\vec{a} + \overleftarrow{b}]$ in $\mathbf{FinScottL}$, where powerset is ordered by inclusion.

5 Strategies and the Winning Position Functor

In §5.1, we define notions of play and strategy for open parity games (in the traditional style of graph game), as well as winning, losing, and *pending* strategies and positions. We use these definitions in §5.2 to define the *winning-position* functor $\mathscr{W}_M^r : \mathbf{OPG}_M^r \rightarrow \mathbf{FinScottL}_{!M}^{op}$, which gives information that allows compositional computation of winning positions. We show that the diagram in Figure 3 commutes, which gives a justification of our compositional approach to parity games. In this section, we assume that a given rightward open parity game \mathcal{A} is of the form $\mathcal{A} = (m, n, Q, E, \rho, M, \omega)$.

5.1 Winning Strategies and Winning Positions for Open Parity Games

Here we give the notions of strategy and play. We also define the *denotation* of a strategy/position, which is how they win, lose, or are pending. These definitions are given only for *rightward* open parity

games, but we can readily extend them to general open parity games, because any open parity game is a rightward open parity game by definition: $\mathbf{OPG}_M((m_r, m_l), (n_r, n_l)) = \mathbf{OPG}_M^r(m_r + n_l, n_r + m_l)$.

First we define the notion of strategy. For an open parity game $\mathcal{A} : [m] \rightarrow [n]$, a family $(s_i)_{i \in I}$ of positions in $[m] + [n] + Q$ is called a *position sequence* if $I = \mathbb{N}_{\geq 1}$ or $I = \{1, \dots, k\}$ for some $k \in \mathbb{N}_{\geq 1}$, (in that case, we also write $(s_i)_{i \in I}$ as $s_1 \cdots s_k$).

Definition 5.1 (\exists -strategy and \forall -strategy). Let \mathcal{A} be a rightward open parity game from m to n . We define $\text{Play}_{\exists}(\mathcal{A}) := \{s_1 \cdots s_k \mid k \geq 1, s_i \in Q (i \in [k]), (s_i, s_{i+1}) \in E (i \in [k-1]), \rho(s_k) = \exists\}$. We often just write Play_{\exists} if there is no confusion. Then, an \exists -strategy on \mathcal{A} is a partial function $\tau : \text{Play}_{\exists} \rightarrow [n] + Q$ where for any $s_1 \cdots s_k \in \text{Play}_{\exists}$, (i) if $\tau(s_1 \cdots s_k) = s$, then $(s_k, s) \in E$, and (ii) if $\tau(s_1 \cdots s_k)$ is undefined, then for all $s \in [n] + Q$, $(s_k, s) \notin E$. A \forall -strategy on an open parity game \mathcal{A} is defined in the same way, by replacing the occurrence of \exists with \forall in the above definition. The sets of \exists -strategies and \forall -strategies on \mathcal{A} are $\text{Str}_{\exists}(\mathcal{A})$ and $\text{Str}_{\forall}(\mathcal{A})$, respectively.

A pair of an \exists -strategy and a \forall -strategy resolves the non-determinism in a game to induce a unique play:

Definition 5.2. (induced play $\text{play}_a^{\tau_{\exists}, \tau_{\forall}}$) Let \mathcal{A} be a rightward open parity game from m to n . The induced play $\text{play}_a^{\tau_{\exists}, \tau_{\forall}}$ from an entry position $a \in [m]$ by an \exists -strategy τ_{\exists} and a \forall -strategy τ_{\forall} is the (necessarily unique) maximal position-sequence $(s_i)_{i \in I}$ (for the prefix order) such that: (i) $aE s_1$, (ii) for any $i \in I$, if $\rho(s_i) = \exists$ and $\tau_{\exists}(s_1 \cdots s_i)$ is defined, then $\tau_{\exists}(s_1 \cdots s_i) = s_{i+1}$, and similarly (iii) for any $i \in I$, if $\rho(s_i) = \forall$ and $\tau_{\forall}(s_1 \cdots s_i)$ is defined, then $\tau_{\forall}(s_1 \cdots s_i) = s_{i+1}$.

The following notion for a play corresponds to the *winning condition* in (traditional) game theory, where the condition is two-valued, “win” or “lose”. Below \exists and \forall correspond to “win” and “lose”, but we have other intermediate results $(m, s_{|I})$ due to the openness, which we call *pending states*. In this paper, we call the following many-valued winning/losing/pending condition $\{\!-\!\}_{\mathcal{A}}$ on plays simply *winning condition*. An infinite position sequence $(s_i)_{i \in \mathbb{N}}$ satisfies the *parity condition* if the maximum of priorities that occur infinitely in the play is even. We apply the following notion $\{\!-\!\}_{\mathcal{A}}$ only to induced plays.

Definition 5.3 (winning condition $\{\!-\!\}_{\mathcal{A}}$ on plays). Let \mathcal{A} be a rightward open parity game. The denotation $\{\!(s_i)_{i \in I}\!\}_{\mathcal{A}}$ of a position sequence $(s_i)_{i \in I}$ is defined as

$$\begin{aligned} & (m, s_{|I}) \text{ if } I \text{ is finite, } m = \max\{\omega(s_i) : i \in I\}, \text{ and } s_{|I} \text{ is an open end,} \\ \exists & \quad \text{if } (I \text{ is finite and } \rho(s_{|I}) = \forall) \text{ or } (I \text{ is infinite and } (s_i)_{i \in I} \text{ satisfies the parity condition),} \\ \forall & \quad \text{if } (I \text{ is finite and } \rho(s_{|I}) = \exists) \text{ or } (I \text{ is infinite and } (s_i)_{i \in I} \text{ does not satisfy the parity condition).} \end{aligned}$$

We call the function $\{\!-\!\}_{\mathcal{A}}$ the *winning condition* of \mathcal{A} .

Next, we define the denotation of an \exists -strategy; note that an \exists -strategy is a strategy for the “player” while \forall -strategies are those for the “opponent”. The denotation is “lose” if there is a losing play, and otherwise is the collection of all the pending states; if the collection is the empty set, then the denotation is “win”.

Definition 5.4 (denotation of positions and \exists -strategies). Let \mathcal{A} be a rightward open parity game from m to n . The denotation $\{\!(a, \tau_{\exists})\!\}$ of an entry position $a \in [m]$ and an \exists -strategy τ_{\exists} is defined by

$$\{\!(a, \tau_{\exists})\!\} := \begin{cases} \text{lose} & \text{if there is } \tau_{\forall} \text{ such that } \{\!\text{play}_a^{\tau_{\exists}, \tau_{\forall}}\!\}_{\mathcal{A}} = \forall, \\ \{\!\{\!\text{play}_a^{\tau_{\exists}, \tau_{\forall}}\!\}_{\mathcal{A}} \in \mathbb{N}_M \times [n] \mid \tau_{\forall} \in \text{Str}_{\forall}(\mathcal{A}) \text{ and } \{\!\text{play}_a^{\tau_{\exists}, \tau_{\forall}}\!\}_{\mathcal{A}} \neq \exists\} & \text{otherwise.} \end{cases}$$

5.2 The Winning Position Functor \mathcal{W}_M

Now we give the central notion of this section, the *winning position* functor \mathcal{W}_M , which is a compact closed functor constructed by the Int-construction of a traced symmetric strong monoidal functor $\mathcal{W}_M^{\text{tr}}$. In the definition of $\mathcal{W}_M^{\text{tr}}$, if we fix an entry position a , then $\mathcal{W}_M^{\text{tr}}(\mathcal{A})$ (or precisely $\{T \mid (T, a) \in \mathcal{W}_M^{\text{tr}}(\mathcal{A})\}$) is the upward-closed set generated by the denotations $\{(a, \tau_{\exists})\}$ of a and all \exists -strategies τ_{\exists} that does not lose from a :

Definition 5.5 (the functor $\mathcal{W}_M^{\text{tr}}$). We define a functor $\mathcal{W}_M^{\text{tr}} : \mathbf{OPG}_M^{\text{tr}} \rightarrow \mathbf{FinScottL}_{!M}^{\text{op}}$. The mapping on objects is given by $\mathcal{W}_M^{\text{tr}}(m) := ([m], =)$, and for a morphism $\mathcal{A} \in \mathbf{OPG}_M^{\text{tr}}(m, n)$,

$$\mathcal{W}_M^{\text{tr}}(\mathcal{A}) := \{ (T, a) \in P(\mathbb{N}_M \times [n]) \times [m] \mid \{(a, \tau_{\exists})\} \neq \mathbf{lose} \text{ and } \{(a, \tau_{\exists})\} \subseteq T \text{ for some } \exists\text{-strategy } \tau_{\exists} \}.$$

The functor $\mathcal{W}_M^{\text{tr}}$ determines whether an entry position wins, but the precise perspective is as follows. As mentioned in the introduction, in the traditional notion of (non-open) parity games, a position is just either winning or losing, two-valued. With the new notion of open ends, however, we have the intermediate result of pending states. The following definition reflects this idea.

Definition 5.6 (winning/losing/pending positions). Let \mathcal{A} and a be a rightward open parity game and an entry position, respectively. (i) a is *winning* if $(\emptyset, a) \in \mathcal{W}_M^{\text{tr}}(\mathcal{A})$, (ii) a is *losing* if $(T, a) \notin \mathcal{W}_M^{\text{tr}}(\mathcal{A})$ for any T , and (iii) a is *pending* otherwise (i.e., if $(\emptyset, a) \notin \mathcal{W}_M^{\text{tr}}(\mathcal{A})$ and $(T, a) \in \mathcal{W}_M^{\text{tr}}(\mathcal{A})$ for some $T \neq \emptyset$).

There is an obvious transformation that maps a traditional parity game G and position x into an open parity game $\mathcal{A}_x^G : 1 \rightarrow 0 \in \mathbf{OPG}_M^{\text{tr}}$, where $1 \in [1]$ points to the internal position x . The notion of winning/losing defined above agrees with the traditional one in the following sense (n.b. there is no pending case):

Proposition 5.7. *Given a (traditional) parity game G and a position x in G , x is winning (resp. losing) in G iff x is winning (resp. losing) in \mathcal{A}_x^G .* \square

The main technical result of this section is stated below, and allows us to define a compact closed functor $\mathcal{W}_M : \mathbf{OPG}_M \rightarrow \mathbf{Int}(\mathbf{FinScottL}_{!M}^{\text{op}})$.

Theorem 5.8. *The functor $\mathcal{W}_M^{\text{tr}} : \mathbf{OPG}_M^{\text{tr}} \rightarrow \mathbf{FinScottL}_{!M}^{\text{op}}$ is a traced symmetric strict monoidal functor.* \square

Definition 5.9 (winning position functor \mathcal{W}_M). We define the *winning position functor* \mathcal{W}_M by $\mathbf{Int}(\mathcal{W}_M^{\text{tr}})$.

Summarizing all the main results in this paper, we obtain the following theorem:

Theorem 5.10. *The triangle in Fig. 3 commutes: $\llbracket - \rrbracket_M \simeq \mathcal{W}_M \circ \mathcal{R}_M$.* \square

We remark that we can obtain a similar result to the above in the TSMC setting by the freeness of $\mathbb{F}_{\text{tr}}(\Sigma_M^{\text{int}})$.

Given any open parity game, which can be represented also by a morphism in $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ by the fullness of \mathcal{R}_M (Thm. 3.15), the above Thm. 5.10 says that we can calculate whether an entry position is winning, losing, or pending, either (i) by calculating strategies (i.e., by \mathcal{W}_M), or *equivalently* (ii) by induction (i.e., by $\llbracket - \rrbracket_M$) without calculating strategies. An elaborated example on how we can compute the denotation of an entry position of an open parity game by the induction $\llbracket - \rrbracket_M$ can be found below.

Finally, note that the notion of winning/losing/pending *position* is defined for $\mathbf{Int}(\mathbf{FinScottL}_{!M}^{\text{op}})$, and hence is defined also for $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$ and \mathbf{OPG}_M , by using $\llbracket - \rrbracket_M$ and \mathcal{W}_M , respectively. On the other hand, the notion of winning/losing/pending *strategy* is defined for morphisms of \mathbf{OPG}_M (and hence of $\mathbb{F}(\Sigma_M^{\text{opg}}, E_M^{\text{opg}})$, too) but not of $\mathbf{Int}(\mathbf{FinScottL}_{!M}^{\text{op}})$. In particular, we can conclude that we have given an *abstract* (or extensional) semantics for open parity games, by eliminating the information of strategies.

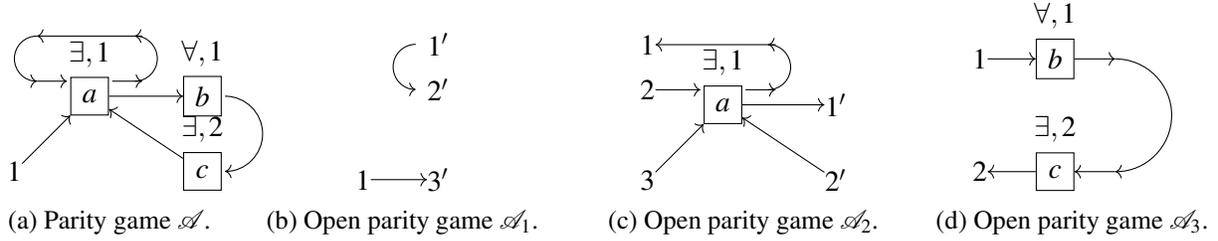


Figure 8: An extended example.

Example 5.11. Let \mathcal{A} be the open parity game in Fig. 8(a). We want to check whether the position 1 is winning by composing the interpretations of \mathcal{A} 's subgames.

Concretely, \mathcal{A} is divided as $\mathcal{A}_3 \circ \mathcal{A}_2 \circ \mathcal{A}_1$ with \mathcal{A}_1 , \mathcal{A}_2 , and \mathcal{A}_3 shown in Fig. 8(b), 8(c), and 8(d), respectively (note that open ends are labelled using prop-style ordering, and while the ordering in $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$ is different, we keep the same notations for readability). It follows directly by unfolding definitions and by compact closedness of $\llbracket - \rrbracket_M$ that $\llbracket \mathcal{A}_1 \rrbracket_M = \{(T, 1) \mid (0, 3') \in T\} \cup \{(T, 1') \mid (0, 2') \in T\}$, $\llbracket \mathcal{A}_2 \rrbracket_M = \{(T, i) \mid i \in \{2, 3, 2'\}, \exists j \in \{1, 1'\}, (1, j) \in T\}$, and $\llbracket \mathcal{A}_3 \rrbracket_M = \{(T, 1) \mid (2, 2) \in T\}$, which are indeed the expected results. For example, to compute $\llbracket \mathcal{A}_1 \rrbracket_M$, we can decompose \mathcal{A}_1 as $d_{\mathbb{1}} \oplus \text{id}_{\mathbb{r}}$, so $\llbracket \mathcal{A}_1 \rrbracket_M = d_{\llbracket \mathbb{1} \rrbracket_M} \oplus \text{id}_{\llbracket \mathbb{r} \rrbracket_M}$, which can easily be computed from the definition of the identity in $\mathbf{FinScottL}_{!M}^{op}$.

In order to compute the composition of two interpretations in $\text{Int}(\mathbf{FinScottL}_{!M}^{op})$, we need to compute a trace, and therefore semantic run-trees (we can avoid it in \mathcal{A}_2 and \mathcal{A}_3 above because they can be reorganized so that composition involves a trivial trace). In semantic run-trees corresponding to $\llbracket \mathcal{A}_2 \rrbracket_M \circ \llbracket \mathcal{A}_1 \rrbracket_M$, there must be no infinite path (corresponding to the intuition that the only infinite path is losing), and the only possible leaf is $1'$ (the only exit position), while conditions in Def. 4.6 (involving $\llbracket \mathcal{A}_2 \rrbracket_M$ and $\llbracket \mathcal{A}_1 \rrbracket_M$ above) ensure that $(1, 1')$ must be one of the leaves, which gives $\llbracket \mathcal{A}_2 \circ \mathcal{A}_1 \rrbracket_M = \{(T, i) \mid i \in \{1, 2'\}, (1, 1') \in T\}$.

Similarly, to compose $\llbracket \mathcal{A}_3 \rrbracket_M$ with $\llbracket \mathcal{A}_2 \circ \mathcal{A}_1 \rrbracket_M$, we also have to compute the corresponding semantic run-trees. Here, there can be no leaves (no exit positions), and the run-tree corresponding to taking the loop infinitely meets the parity condition (because all its nodes are $(2, 1')$, except for the root), so $(\emptyset, 1)$ is in the interpretation, whence $\llbracket \mathcal{A} \rrbracket_M = \{(T, 1) \mid \text{true}\} = \{(\emptyset, 1)\}$. Therefore, 1 is a winning position in G .

6 Conclusions and Future Work

We have given a compositional approach to parity games by exhibiting their underlying compact closed structure. Parity games can be composed by considering open ends, and we defined a prop that gives a graphical language to describe such open parity games. At the semantic level, we have given a notion of winning/losing positions that takes open ends into account. It retains enough information to be compositional, but is still extensional, as it can be computed without referring to starategies.

The current semantic category is a *strategy-insensitive* model, in that it only keeps track of the (non)existence of a winning strategy, while *strategy-sensitive* models should keep track of all strategies (perhaps up to some suitable equivalence). Strategy-sensitive models can easily be obtained when restricting to *history-free* strategies. One future work is to find some history-dependent strategy-sensitive model.

It could be fruitful to deepen the link between the existing body of work on props and our use of props in this work. For example, by showing equivalence between the graphical language and the category of open parity games, we could get decidability results on parity games from the syntax, as in [5].

Another possible future work is the coalgebraic treatment of open parity games. There seems a bijective correspondence between open parity games up to some notion of bisimilarity and pairs of functions $[m] \rightarrow Q$ and $Q \rightarrow P(Q + [n]) \times \{\exists, \forall\} \times \mathbb{N}_M$ up to the bisimilarity of coalgebras. Then there might exist a compact closed structure in the category of coalgebraic open parity games up to the bisimilarity. However, it seems subtle to give the same categorical structure in the category of, say, the above form of coalgebras up to *isomorphism*. Also, it might be interesting to give a functor (similar to \mathcal{R}_M) from some graphical category to the category of coalgebras up to bisimilarity by using bialgebraic methods [25, 33]. We did not take the bisimilarity approach in this paper because in game theory we basically consider the level up to isomorphism, say, for complexity.

References

- [1] John C. Baez, Brandon Coya & Franciscus Rebro (2018): *Props in network theory. Theory and Applications of Categories* 33(25), pp. 727–783. Available at <http://www.tac.mta.ca/tac/volumes/33/25/33-25abs.html>.
- [2] John C. Baez & Jade Master (2020): *Open Petri nets. Math. Struct. Comput. Sci.* 30(3), pp. 314–341, doi:10.1017/S0960129520000043.
- [3] Filippo Bonchi, Joshua Holland, Robin Piedeleu, Paweł Sobociński & Fabio Zanasi (2019): *Diagrammatic Algebra: From Linear to Concurrent Systems. Proc. ACM Program. Lang.* 3(POPL), doi:10.1145/3290338.
- [4] Filippo Bonchi, Robin Piedeleu, Paweł Sobocinski & Fabio Zanasi (2019): *Graphical Affine Algebra*. In: *34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019*, IEEE, pp. 1–12, doi:10.1109/LICS.2019.8785877.
- [5] Filippo Bonchi, Paweł Sobociński & Fabio Zanasi (2017): *Interacting Hopf algebras. Journal of Pure and Applied Algebra* 221(1), pp. 144–184, doi:10.1016/j.jpaa.2016.06.002.
- [6] Titouan Carette, Dominic Horsman & Simon Perdrix (2019): *SZX-calculus: Scalable Graphical Quantum Reasoning*. In: *MFCS 2019 - 44th International Symposium on Mathematical Foundations of Computer Science, Leibniz International Proceedings in Informatics (LIPIcs)* 138, Aachen, Germany, pp. 55:1–55:15, doi:10.4230/LIPIcs.MFCS.2019.55. Available at <https://hal.archives-ouvertes.fr/hal-02400070>.
- [7] Titouan Carette & Simon Perdrix (2020): *Colored props for large scale graphical reasoning. CoRR* abs/2007.03564. arXiv:2007.03564.
- [8] Edmund M. Clarke, David E. Long & Kenneth L. McMillan (1989): *Compositional Model Checking*. In: *Proceedings of the Fourth Annual Symposium on Logic in Computer Science (LICS '89), Pacific Grove, California, USA, June 5-8, 1989*, IEEE Computer Society, pp. 353–362, doi:10.1109/LICS.1989.39190.
- [9] Wojciech Czerwinski, Laure Daviaud, Nathanaël Fijalkow, Marcin Jurdzinski, Ranko Lazic & Paweł Parys (2019): *Universal trees grow inside separating automata: Quasi-polynomial lower bounds for parity games*. In Timothy M. Chan, editor: *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, SIAM, pp. 2333–2349, doi:10.1137/1.9781611975482.142.
- [10] Marcelo P. Fiore & Marco Devesas Campos (2013): *The Algebra of Directed Acyclic Graphs*. In Bob Coecke, Luke Ong & Prakash Panangaden, editors: *Computation, Logic, Games, and Quantum Foundations. The Many Facets of Samson Abramsky - Essays Dedicated to Samson Abramsky on the Occasion of His 60th Birthday, Lecture Notes in Computer Science* 7860, Springer, pp. 37–51, doi:10.1007/978-3-642-38164-5_4.

- [11] Yael Frégier, Martin Markl & Donald Yau (2008): *The L_∞ -deformation complex of diagrams of algebras*. *New York J. Math.* 15, p. 353–392.
- [12] Soichiro Fujii, Shin-ya Katsumata & Paul-André Melliès (2016): *Towards a Formal Theory of Graded Monads*. In Bart Jacobs & Christof Löding, editors: *Foundations of Software Science and Computation Structures - 19th International Conference, FOSSACS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings, Lecture Notes in Computer Science 9634*, Springer, pp. 513–530, doi:10.1007/978-3-662-49630-5_30.
- [13] Neil Ghani, Jules Hedges, Viktor Winschel & Philipp Zahn (2018): *Compositional Game Theory*. In Anuj Dawar & Erich Grädel, editors: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, ACM, pp. 472–481, doi:10.1145/3209108.3209165.
- [14] Charles Grellois (2016): *Semantics of linear logic and higher-order model-checking. (Sémantique de la logique linéaire et "model-checking" d'ordre supérieur)*. Ph.D. thesis, Paris Diderot University, France. Available at <https://tel.archives-ouvertes.fr/tel-01311150>.
- [15] Charles Grellois & Paul-André Melliès (2015): *Finitary Semantics of Linear Logic and Higher-Order Model-Checking*. In Giuseppe F. Italiano, Giovanni Pighizzini & Donald Sannella, editors: *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part I, Lecture Notes in Computer Science 9234*, Springer, pp. 256–268, doi:10.1007/978-3-662-48057-1_20.
- [16] Philip Hackney & Marcy Robertson (2012): *On the Category of Props*. *Applied Categorical Structures* 23, doi:10.1007/s10485-014-9369-4.
- [17] Masahito Hasegawa (1997): *Recursion from Cyclic Sharing: Traced Monoidal Categories and Models of Cyclic Lambda Calculi*. In Philippe de Groote, editor: *Typed Lambda Calculi and Applications, Third International Conference on Typed Lambda Calculi and Applications, TLCA '97, Nancy, France, April 2-4, 1997, Proceedings, Lecture Notes in Computer Science 1210*, Springer, pp. 196–213, doi:10.1007/3-540-62688-3_37.
- [18] Masahito Hasegawa & Shin-ya Katsumata (2010): *A note on the biadjunction between 2-categories of traced monoidal categories and tortile monoidal categories*. *Mathematical Proceedings of the Cambridge Philosophical Society* 148(1), p. 107–109, doi:10.1017/S0305004109002606.
- [19] Ichiro Hasuo (2014): *Generic Weakest Precondition Semantics from Monads Enriched with Order*. In Marcello M. Bonsangue, editor: *Coalgebraic Methods in Computer Science - 12th IFIP WG 1.3 International Workshop, CMCS 2014, Colocated with ETAPS 2014, Grenoble, France, April 5-6, 2014, Revised Selected Papers, Lecture Notes in Computer Science 8446*, Springer, pp. 10–32, doi:10.1007/978-3-662-44124-4_2.
- [20] André Joyal & Ross Street (1993): *Braided Tensor Categories*. *Advances in Mathematics* 102(1), pp. 20–78, doi:10.1006/aima.1993.1055. Available at <https://www.sciencedirect.com/science/article/pii/S0001870883710558>.
- [21] André Joyal, Ross Street & Dominic Verity (1996): *Traced monoidal categories*. *Mathematical Proceedings of the Cambridge Philosophical Society* 119(3), p. 447–468, doi:10.1017/S0305004100074338.
- [22] Shin-ya Katsumata (2010): *Categorical Descriptive Composition*. In Kazunori Ueda, editor: *Programming Languages and Systems - 8th Asian Symposium, APLAS 2010, Shanghai, China, November 28 - December 1, 2010. Proceedings, Lecture Notes in Computer Science 6461*, Springer, pp. 222–238, doi:10.1007/978-3-642-17164-2_16.
- [23] Shin-ya Katsumata (2014): *Parametric effect monads and semantics of effect systems*. In Suresh Jagannathan & Peter Sewell, editors: *The 41st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '14, San Diego, CA, USA, January 20-21, 2014*, ACM, pp. 633–646, doi:10.1145/2535838.2535846.

- [24] Aleks Kissinger (2014): *Abstract Tensor Systems as Monoidal Categories*. In Claudia Casadio, Bob Coecke, Michael Moortgat & Philip Scott, editors: *Categories and Types in Logic, Language, and Physics - Essays Dedicated to Jim Lambek on the Occasion of His 90th Birthday*, Lecture Notes in Computer Science 8222, Springer, pp. 235–252, doi:10.1007/978-3-642-54789-8_13.
- [25] Bartek Klin (2011): *Bialgebras for structural operational semantics: An introduction*. *Theor. Comput. Sci.* 412(38), pp. 5043–5069, doi:10.1016/j.tcs.2011.03.023.
- [26] Marta Z. Kwiatkowska, Gethin Norman, David Parker & Hongyang Qu (2013): *Compositional probabilistic verification through multi-objective model checking*. *Inf. Comput.* 232, pp. 38–65, doi:10.1016/j.ic.2013.10.001.
- [27] Elena Di Lavore, Jules Hedges & Paweł Sobociński (2021): *Compositional Modelling of Network Games*. In Christel Baier & Jean Goubault-Larrecq, editors: *29th EACSL Annual Conference on Computer Science Logic (CSL 2021)*, Leibniz International Proceedings in Informatics (LIPIcs) 183, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 30:1–30:24, doi:10.4230/LIPIcs.CSL.2021.30.
- [28] Saunders MacLane (1965): *Categorical algebra*. *Bulletin of the American Mathematical Society* 71(1), pp. 40–106, doi:10.1090/S0002-9904-1965-11234-4.
- [29] Roger Penrose (1971): *Applications of negative dimensional tensors*. *Combinatorial mathematics and its applications* 1, pp. 221–244.
- [30] Robin Piedeleu & Fabio Zanasi (2021): *A String Diagrammatic Axiomatisation of Finite-State Automata*. In Stefan Kiefer & Christine Tasson, editors: *Foundations of Software Science and Computation Structures - 24th International Conference, FOSSACS 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27 - April 1, 2021, Proceedings*, Lecture Notes in Computer Science 12650, Springer, pp. 469–489, doi:10.1007/978-3-030-71995-1_24.
- [31] Peter Selinger (2011): *A Survey of Graphical Languages for Monoidal Categories*. In Bob Coecke, editor: *New Structures for Physics*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 289–355, doi:10.1007/978-3-642-12821-9_4.
- [32] Takeshi Tsukada & C.-H. Luke Ong (2014): *Compositional higher-order model checking via ω -regular games over Böhm trees*. In: *Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS '14, Vienna, Austria, July 14 - 18, 2014*, ACM, pp. 78:1–78:10, doi:10.1145/2603088.2603133.
- [33] Daniele Turi & Gordon D. Plotkin (1997): *Towards a Mathematical Operational Semantics*. In: *Proceedings, 12th Annual IEEE Symposium on Logic in Computer Science*, Warsaw, Poland, June 29 - July 2, 1997, IEEE Computer Society, pp. 280–291, doi:10.1109/LICS.1997.614955.
- [34] Thomas Wilke (2001): *Alternating tree automata, parity games, and modal μ -calculus*. *Bulletin of the Belgian Mathematical Society - Simon Stevin* 8(2), pp. 359 – 391, doi:10.36045/bbms/1102714178.