The Power of the Weak

Facundo Carreiro, University of Amsterdam
Alessandro Facchini, IDSIA, Switzerland
Yde Venema, University of Amsterdam
Fabio Zanasi, University College London

A landmark result in the study of logics for formal verification is Janin & Walukiewicz's theorem, stating that the modal μ-calculus ($\mu\mathrm{ML}$) is equivalent modulo bisimilarity to standard monadic second-order logic (here abbreviated as SMSO), over the class of labelled transition systems (LTSs for short). Our work proves two results of the same kind, one for the alternation-free or *noetherian* fragment of $\mu\mathrm{ML}$ ($\mu_N\mathrm{ML}$) and one for weak MSO (WMSO). Whereas it was known in the setting of binary trees that WMSO is equivalent to an appropriate version of $\mu_N\mathrm{ML}$, our analysis shows that the picture radically changes once we reason over arbitrary LTSs. The first theorem that we prove is that, over LTSs, $\mu_N\mathrm{ML}$ is equivalent modulo bisimilarity to *noetherian* MSO (NMSO), a newly introduced variant of SMSO where second-order quantification ranges over "well-founded" subsets only. Our second theorem starts from WMSO, and proves it equivalent modulo bisimilarity to a fragment of $\mu_N\mathrm{ML}$ defined by a notion of continuity. Analogously to Janin & Walukiewicz's result, our proofs are automata-theoretic in nature: as another contribution, we introduce classes of parity automata characterising the expressiveness of WMSO and NMSO (on tree models) and of $\mu_C\mathrm{ML}$ and $\mu_N\mathrm{ML}$ (for all transition systems).

CCS Concepts: •**Theory of computation** → **Logic; Formal languages and automata theory;**

Additional Key Words and Phrases: Modal μ-Calculus, Weak Monadic Second Order Logic, Tree Automata, Bisimulation

ACM Reference Format:
Facundo Carreiro, Alessandro Facchini, Yde Venema, Fabio Zanasi, 2017. The Power of the Weak. *ACM Trans. Comput. Logic* 0, 0, Article 0 (2017), 39 pages.
DOI: 0000001.0000001

1. ONE-STEP LOGICS, PARITY AUTOMATA AND μ-CALCULI

This section introduces and studies the type of parity automata that will be used in the characterisation of WMSO and NMSO on tree models. In order to define these automata in a uniform way, we introduce, at a slightly higher level of abstraction, the notion of a *one-step logic*, a concept from coalgebraic modal logic [Cîrstea and Pattinson 2004] which provides a nice framework for a general approach towards the theory of automata operating on infinite objects. As salient specimens of such one-step logics we will discuss monadic first-order logic with equality (FOE_1) and its extension with the infinity quantifier (FOE_1^∞). We then define, parametric in the language L_1 of such a one-step logic, the notions of an L_1-automaton and of a mu-calculus μL_1, and we show how various classes of L_1-automata effectively correspond to fragments of μL_1.

1.1. One-step logics and normal forms

Definition 1.1. Given a finite set A of *monadic predicates*, a *one-step model* is a pair (D, V) consisting of a *domain* set D and a *valuation* or *interpretation* $V : A \to \wp D$. Where $B \subseteq A$, we say that $V' : A \to \wp D$ is a *B-extension of* $V : A \to \wp D$, notation $V \leq_B V'$, if $V(b) \subseteq V'(b)$ for every $b \in B$ and $V(a) = V'(a)$ for every $a \in A \setminus B$.
A *one-step language* is a map assigning any set A to a collection $L_1(A)$ of objects called *one-step formulas* over A. We assume that one-step languages come with a *truth* relation $\models$ between one-step formulas and models, writing $(D, V) \models \varphi$ to denote that (D, V) satisfies φ.

Note that we do allow the (unique) one-step model that is based on the empty domain; we will simply denote this model as $(\varnothing, \varnothing)$.

Our chief examples of one-step languages will be variants of modal and first-order logic.

Definition 1.2. A very simple example of a one-step logic is the following *basic one-step modal logic* ML_1, of which the language is defined as follows, for a set A of monadic predicates:

$$\mathrm{ML}_1(A) := \{\Diamond a, \Box a \mid a \in A\}.$$

The semantics of these formulas is given by

$$\begin{array}{lll} (D, V) \models \Diamond a & \text{iff} & V(a) \neq \varnothing \\ (D, V) \models \Box a & \text{iff} & V(a) = D. \end{array}$$

Definition 1.3. The one-step language $\mathrm{FOE}_1(A)$ of *first-order logic with equality* on a set of predicates A and individual variables iVar is given by the sentences (formulas without free variables) generated by the following grammar, where $a \in A$ and $x, y \in \mathsf{iVar}$.:

$$\varphi ::= a(x) \mid \neg a(x) \mid x \approx y \mid x \not\approx y \mid \exists x.\varphi \mid \forall x.\varphi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \qquad (1)$$

We use FO_1 for the equality-free fragment, where we omit the clauses $x \approx y$ and $x \not\approx y$.

The interpretation of this language in a model (D, V) with $D \neq \varnothing$ is completely standard. Formulas of FO_1 and FOE_1 are interpreted inductively by augmenting the pair (D, V) with a variable assignment $g : \mathsf{iVar} \to D$. The semantics then defines the desired truth relation $(D, V), g \models \varphi$ between one-step models, assignments and one-step formulas. As usual, the variable assignment g can and will be omitted when we are dealing with sentences — and note that we only take sentences as one-step formulas. For the interpretation in one-step models with empty domain we refer to Definition 1.5.

We now introduce an extension of first-order logic with two additional quantifiers, which first appeared in the context of Mostowski's study [Mostowski 1957] of generalised quantifiers. The first, written $\exists^\infty x.\varphi$, expresses that there exist infinitely many elements satisfying a formula φ. Its dual, written $\forall^\infty x.\varphi$, expresses that there are *at most finitely many* elements *falsifying* the formula φ. Formally:

$$\begin{array}{lll} (D,V),g \models \exists^\infty x.\varphi(x) & \text{iff} & |\{s \in D \mid (D,V),g[x \mapsto s] \models \varphi(x)\}| \geq \omega \\ (D,V),g \models \forall^\infty x.\varphi(x) & \text{iff} & |\{s \in D \mid (D,V),g[x \mapsto s] \not\models \varphi(x)\}| < \omega \end{array} \tag{2}$$

Definition 1.4. The one-step language $\mathrm{FOE}_1^\infty(A)$ is defined by adding to the grammar (1) of $\mathrm{FOE}_1(A)$ the cases $\exists^\infty x.\varphi$ and $\forall^\infty x.\varphi$. In the case of non-empty models, the truth relation $(D,V),g \models \varphi$ is defined by extending the truth relation for $\mathrm{FOE}_1(A)$ with the clauses (2).

In the case of models with empty domain, we cannot give an inductive definition of the truth relation using variable assignments. Nevertheless, a definition of truth can be provided for formulas that are Boolean combinations of sentences of the form $Qx.\varphi$, where $Q \in \{\exists, \exists^\infty, \forall, \forall^\infty\}$ is a quantifier.

Definition 1.5. For the one-step model $(\varnothing,\varnothing)$ we define the truth relation as follows: For every sentence $Qx.\varphi$, where $Q \in \{\exists, \exists^\infty, \forall, \forall^\infty\}$, we set

$$\begin{array}{llll} (\varnothing,\varnothing) & \not\models & Qx.\varphi & \text{if} \;\; Q \in \{\exists, \exists^\infty\} \\ (\varnothing,\varnothing) & \models & Qx.\varphi & \text{if} \;\; Q \in \{\forall, \forall^\infty\}, \end{array}$$

and we extend this definition to arbitrary FOE_1^∞-sentences via the standard clauses for the boolean connectives.

For various reasons it will be convenient to assume that our one-step languages are closed under taking *(boolean) duals*. Here we say that the one-step formulas φ and ψ are boolean duals if for every one-step model we have $(D,V) \models \varphi$ iff $(D,V^c) \not\models \psi$, where V^c is the complement valuation given by $V^c(a) := D \setminus V(a)$, for all a.

As an example, it is easy to see that for the basic one-step modal logic ML_1 the formulas $\Diamond a$ and $\Box a$ are each other's dual. In the case of the monadic predicate logics FO_1, FOE_1 and FOE_1^∞ we can define the boolean dual of a formula φ by a straightforward induction.

Definition 1.6. For $L_1 \in \{\mathrm{FO}_1, \mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$, we define the following operation on formulas:

$$\begin{array}{rlrl} (a(x))^\delta & := a(x) & (\neg a(x))^\delta & := \neg a(x) \\ (\top)^\delta & := \bot & (\bot)^\delta & := \top \\ (x \approx y)^\delta & := x \not\approx y & (x \not\approx y)^\delta & := x \approx y \\ (\varphi \wedge \psi)^\delta & := \varphi^\delta \vee \psi^\delta & (\varphi \vee \psi)^\delta & := \varphi^\delta \wedge \psi^\delta \\ (\exists x.\psi)^\delta & := \forall x.\psi^\delta & (\forall x.\psi)^\delta & := \exists x.\psi^\delta \\ (\exists^\infty x.\psi)^\delta & := \forall^\infty x.\psi^\delta & (\forall^\infty x.\psi)^\delta & := \exists^\infty x.\psi^\delta \end{array}$$

We leave it for the reader to verify that the operation $(\cdot)^\delta$ indeed provides a boolean dual for every one-step sentence.

The following semantic properties will be essential when studying the parity automata and μ-calculi associated with one-step languages.

Definition 1.7. Given a one-step language $L_1(A)$, $\varphi \in L_1(A)$ and $B \subseteq A$,

— φ is *monotone* in B if for all pairs of one step models (D,V) and (D,V') with $V \leq_B V'$, $(D,V) \models \varphi$ implies $(D,V'),g \models \varphi$.

— φ is B-*continuous* if φ is monotone in B and, whenever $(D,V) \models \varphi$, then there exists $V'\colon A \to \wp(D)$ such that $V' \leq_B V$, $(D,V') \models \varphi$ and $V'(b)$ is finite for all $b \in B$.
— φ is B-*cocontinuous* if its dual φ^δ is continuous in B.

We recall from [Carreiro et al. 2018] syntactic characterisations of these semantic properties, relative to the monadic predicate logics FO_1, FOE_1 and FOE_1^∞. We first discuss characterisations of monotonicity and (co)continuity given by grammars.

Definition 1.8. For $L_1 \in \{\mathrm{FO}_1, \mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$, we define the *positive* fragment of $L_1(A)$, written $L_1^+(A)$, as the set of sentences generated by the grammar we obtain by leaving out the clause $\neg a(x)$ from the grammar for L_1.

For $B \subseteq A$, the B-*continuous* fragment of $\mathrm{FOE}_1^+(A)$, written $\mathtt{Con}_B(\mathrm{FOE}_1(A))$, is the set of sentences generated by the following grammar, for $b \in B$ and $\psi \in \mathrm{FOE}_1^+(A \setminus B)$:

$$\varphi ::= b(x) \mid \psi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x.\varphi.$$

If $\psi \in \mathrm{FO}_1^+(A \setminus B)$ in the condition above, we then obtain the B-continuous fragment $\mathtt{Con}_B(\mathrm{FO}_1(A))$ of $\mathrm{FO}_1^+(A)$. The B-*continuous* fragment of $\mathrm{FOE}_1^{\infty +}(A)$, written $\mathtt{Con}_B(\mathrm{FOE}_1^\infty(A))$, is defined by adding to the above grammar the clause $\mathbf{W}x.(\varphi,\psi)$, which is a shorthand for $\forall x.(\varphi(x) \vee \psi(x)) \wedge \forall^\infty x.\psi(x)$.[1] For $L_1 \in \{\mathrm{FO}_1, \mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$ and $B \subseteq A$, the B-*cocontinuous* fragment of $L_1^+(A)$, written $\mathtt{CoCon}_B(L_1(A))$, is the set $\{\varphi \mid \varphi^\delta \in \mathtt{Con}_B(L_1(A))\}$.

Note that we do allow the clause $x \not\approx y$ in the positive fragments of FOE_1 and FOE_1^∞. The following result provides syntactic characterizations for the mentioned semantics properties.

THEOREM 1.9 ([CARREIRO ET AL. 2018]). *For $L_1 \in \{\mathrm{FO}_1, \mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$, we have let $\varphi \in L_1(A)$ be a one-step formula. Then*

(1) $\varphi \in L_1(A)$ is A-monotone iff it is equivalent to some $\psi \in L_1^+(A)$.
(2) $\varphi \in L_1(A)$ is B-continuous iff it is equivalent to some $\psi \in \mathtt{Con}_B(L_1(A))$.
(3) $\varphi \in L_1(A)$ is B-cocontinuous iff it is equivalent to some $\psi \in \mathtt{CoCon}_B(L_1(A))$.

PROOF. The first two statements are proved in [Carreiro et al. 2018]. The third one can be verified by a straightforward induction on φ. $\square$

In some of our later proofs we need more precise information on the shape of formulas belonging to certain syntactic fragments. For this purpose we introduce normal forms for positive sentences in FO_1, FOE_1 and FOE_1^∞.

Definition 1.10. A *type* T is just a subset of A. It defines a FOE_1-formula

$$\tau_T^+(x) := \bigwedge_{a \in T} a(x).$$

Given a one-step model (D,V), $s \in D$ *witnesses* a type T if $(D,V), g[x \mapsto s] \models \tau_T^+(x)$ for any g. The predicate $\mathrm{diff}(\overline{\mathbf{y}})$, stating that the elements $\overline{\mathbf{y}}$ are distinct, is defined as $\mathrm{diff}(y_1,\dots,y_n) := \bigwedge_{1 \leq m < m' \leq n}(y_m \not\approx y_{m'})$.

A formula $\varphi \in \mathrm{FO}_1(A)$ is said to be in *basic form* if $\varphi = \bigvee \nabla_{\mathrm{FO}}^+(\Sigma,\Sigma)$, where for sets Σ, Π of types, the formula $\nabla_{\mathrm{FO}}^+(\Sigma,\Pi)$ is defined as

$$\nabla_{\mathrm{FO}}^+(\Sigma,\Pi) := \bigwedge_{S \in \Sigma} \exists x\, \tau_{T_i}^+(x) \wedge \forall z. \bigvee_{S \in \Pi} \tau_S^+(z)$$

[1] In words, $\mathbf{W}x.(\varphi,\psi)$ says: “every element of the domain validates $\varphi(x)$ or $\psi(x)$, but only finitely many need to validate $\varphi(x)$ (where $b \in B$ may occur). Thus $\forall^\infty$ makes a certain use of $\forall$ compatible with continuity.

We say that $\varphi \in \mathrm{FOE}_1(A)$ is in *basic form* if $\varphi = \bigvee \nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi)$ where each disjunct is of the form

$$\nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi) := \exists \overline{\mathbf{x}}.\big(\mathrm{diff}(\overline{\mathbf{x}}) \wedge \bigwedge_i \tau^+_{T_i}(x_i) \wedge \forall z.(\mathrm{diff}(\overline{\mathbf{x}}, z) \to \bigvee_{S \in \Pi} \tau^+_S(z))\big)$$

such that $\overline{\mathbf{T}} \in \wp(A)^k$ for some k and $\Pi \subseteq \overline{\mathbf{T}}$.

Finally, we say that $\varphi \in \mathrm{FOE}_1^\infty(A)$ is in *basic form* if $\varphi = \bigvee \nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$ where each disjunct is of the form

$$\nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma) := \nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi \cup \Sigma) \wedge \nabla^+_\infty(\Sigma)$$
$$\nabla^+_\infty(\Sigma) := \bigwedge_{S \in \Sigma} \exists^\infty y.\tau^+_S(y) \wedge \forall^\infty y. \bigvee_{S \in \Sigma} \tau^+_S(y)$$

for some sets of types $\Pi, \Sigma \subseteq \wp A$ and $T_1, \ldots, T_k \subseteq A$.

Intuitively, the basic FO_1-formula $\nabla^+_{\mathrm{FO}}(\Sigma, \Sigma)$ simply states that Σ covers a one-step model, in the sense that each element of its domain witnesses some type S of Σ and each type S of Σ is witnessed by some element. The formula $\nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi)$ says that each one-step model satisfying it admits a partition of its domain in two parts: distinct elements $t_1, \ldots, t_n$ witnessing types $T_1, \ldots, T_n$, and all the remaining elements witnessing some type S of Π. The formula $\nabla^+_\infty(\Sigma)$ extends the information given by $\nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi \cup \Sigma)$ by saying that (1) for every type $S \in \Sigma$, there are infinitely many elements witnessing each $S \in \Sigma$ and (2) only finitely many elements do not satisfy any type in Σ.

The next theorem states that the basic formulas indeed provide normal forms.

THEOREM 1.11 ([CARREIRO ET AL. 2018]). *For each $L_1 \in \{\mathrm{FO}_1, \mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$ there is an effective procedure transforming any sentence $\varphi \in L_1^+(A)$ into an equivalent sentence $\varphi^\bullet$ in basic L_1-form.*

One may use these normal forms to provide a tighter syntactic characterisation for the notion of continuity, in the cases of FO_1 and FOE_1^∞.

THEOREM 1.12 ([CARREIRO ET AL. 2018]).

(1) *A formula $\varphi \in \mathrm{FO}_1(A)$ is continuous in $B \subseteq A$ iff it is equivalent to a formula, effectively obtainable from φ, in the basic form $\bigvee \nabla^+_{\mathrm{FO}}(\Sigma', \Sigma)$ where we require that $B \cap \bigcup \Sigma = \varnothing$ for every Σ.*

(2) *A formula $\varphi \in \mathrm{FOE}_1^\infty(A)$ is continuous in $B \subseteq A$ iff it is equivalent to a formula, effectively obtainable from φ, in the basic form $\bigvee \nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$, where we require that $B \cap \bigcup \Sigma = \varnothing$ for every Σ.*

Remark 1.13. We focussed on normal form results for monotone and (co)continuous sentences, as these are the ones relevant to our study of parity automata. However, generic sentences both of FO_1, FOE_1 and FOE_1^∞ also enjoy normal form results, with the syntactic formats given by variations of the "basic form" above. The interested reader may find in [Carreiro et al. 2018] a detailed overview of these results.

We finish this section with a disucssion of the notion of *separation*.

Definition 1.14. Fix a one-step language L_1, and two sets A and B with $B \subseteq A$. Given a one-step model (D, V), we say that $V : A \to \wp D$ *separates* B if $|V^{-1}(d) \cap B| \leq 1$, for every $d \in D$. A formula $\varphi \in L_1(A)$ is *B-separating* if φ is monotone in B and, whenever $(D, V) \models \varphi$, then there exists a B-separating valuation $V' \colon A \to \wp(D)$ such that $V' \leq_B V$ and $(D, V') \models \varphi$.

Intuitively, a formula φ is B-separating if its truth in a monadic model never requires an element of the domain to satisfy two distinct predicates in B at the same time; any valuation violating this constraint can be reduced to a valuation satisfying it, without sacrificing the truth of φ. We do not need a full syntactic characterisation of this notion, but the following sufficient condition is used later on.

PROPOSITION 1.15.

(1) Let $\varphi \in \mathrm{FOE}_1^+(A)$ *be a formula in basic form,* $\varphi = \bigvee \nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi)$. *Then* φ *is* B*-separating if, for each disjunct,* $|S \cap B| \leq 1$ *for each* $S \in \{T_1, \dots, T_k\} \cup \Pi$.

(2) Let $\varphi \in \mathrm{FOE}_1^{\infty +}(A)$ *be a formula in basic form,* $\varphi = \bigvee \nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$. *Then* φ *is* B*-separating if, for each disjunct,* $|S \cap B| \leq 1$ *for each* $S \in \{T_1, \dots, T_k\} \cup \Pi \cup \Sigma$.

PROOF. We only discuss the case $L_1 = \mathrm{FOE}_1^\infty$: a simplification of the same argument yields the case $L_1 = \mathrm{FOE}_1$. Aassume that $(D, V) \models \varphi$ for some model (D, V). We want to construct a valuation $V' \leq_B V$ witnessing the B-separation property. First, we fix one disjunct $\psi = \nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$ of $\varphi^\bullet$ such that $(D, V) \models \psi$. The syntactic shape of ψ implies that (D, V) can be partitioned in three sets D_1, D_2 and D_3 as follows: D_1 contains elements $s_1, \dots, s_k$ witnessing types $T_1, \dots, T_k$, respectively; among the remaining elements, there are infinitely many witnessing some $S \in \Sigma$ (these form D_2), and finitely many not witnessing any $S \in \Sigma$ but each witnessing some $R \in \Pi$ (these form D_3). In other words, we have assigned to each $d \in D$ a type $S_d \in \{T_1, \dots, T_k\} \cup \Pi \cup \Sigma$ such that d witnesses S_d. Now consider the valuation U that we obtain by pruning V to the extent that $U(a) := V(a)$ for $a \in A \setminus B$, while $U(b) := \{d \in D \mid b \in S_d\}$. It is then easy to see that we still have $(D, U) \models \psi$, while it is obvious that U separates B and that $U \leq_B A$. Therefore ψ is B-separating and so φ is too. $\square$

1.2. Parity automata

Throughout the rest of the section we fix, next to a set P of proposition letters, a one-step language L_1, as defined in Subsection 1.1. In light of the results therein, we assume that we have isolated fragments $L_1^+(A)$, $\mathtt{Con}_B(L_1(A))$ and $\mathtt{CoCon}_B(L_1(A))$ consisting of one-step formulas in $L_1(A)$ that are respectively monotone, B-continuous and B-co-continuous, for $B \subseteq A$.

We first recall the definition of a general parity automaton, adapted to this setting.

Definition 1.16 (*Parity Automata*). A *parity automaton* based on the one-step language L_1 and the set P of proposition letters, or briefly: an L_1-*automaton*, is a tuple $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ such that A is a finite set of states, $a_I \in A$ is the initial state, $\Delta : A \times \wp(\mathsf{P}) \to L_1^+(A)$ is the transition map, and $\Omega : A \to \mathbb{N}$ is the priority map. The class of such automata will be denoted by $Aut(L_1)$.

Acceptance of a P-transition system $\mathbb{S} = \langle T, R, \kappa, s_I \rangle$ by $\mathbb{A}$ is determined by the *acceptance game* $\mathcal{A}(\mathbb{A}, \mathbb{S})$ of $\mathbb{A}$ on $\mathbb{S}$. This is the parity game defined according to the rules of the following table.

Position	Player	Admissible moves	Priority
$(a, s) \in A \times T$	$\exists$	$\{V : A \to \wp(R[s]) \mid (R[s], V) \models \Delta(a, \kappa(s))\}$	$\Omega(a)$
$V : A \to \wp(T)$	$\forall$	$\{(b, t) \mid t \in V(b)\}$	0

$\mathbb{A}$ *accepts* $\mathbb{S}$ if $\exists$ has a winning strategy in $\mathcal{A}(\mathbb{A}, \mathbb{S})@(a_I, s_I)$, and *rejects* $\mathbb{S}$ if (a_I, s_I) is a winning position for $\forall$. We write $\mathsf{Mod}(\mathbb{A})$ for the class of transition systems that are accepted by $\mathbb{A}$ and $\mathsf{TMod}(\mathbb{A})$ for the class of tree models in $\mathsf{Mod}(\mathbb{A})$.

Explained in words, the acceptance game $\mathcal{A}(\mathbb{A}, \mathbb{S})$ proceeds in rounds, each round moving from one basic position $(a, s) \in A \times T$ to the next. At such a basic position, it is $\exists$'s task to turn the set $R(s)$ of successors of s into the domain of a one-step model for the formula $\Delta(a,\kappa(s)) \in L_1(A)$. That is, she needs to come up with a valuation $V : A \to \wp(R[s])$ such that $(R[s], V) \models \Delta(a,\kappa(s))$ (and if she cannot find such a valuation, she looses immediately). One may think of the set $\{(b,t) \mid t \in V(b)\}$ as a collection of witnesses to her claim that, indeed, $(R[s], V) \models \Delta(a,\kappa(s))$. The round ends with $\forall$ picking one of these witnesses, which then becomes the basic position at the start of the next round. (Unless, of course, $\exists$ managed to satisfy the formula $\Delta(a,\kappa(s))$ with an empty set of witnesses, in which case $\forall$ gets stuck and looses immediately.)

Many properties of parity automata can already be determined at the one-step level. An important example concerns the notion of complementation, which will be used later in this section. Recall the notion of *dual* of a one-step formula (Definition 1.1). Following ideas from [Muller and Schupp 1987; Kissig and Venema 2009], we can use duals, together with a *role switch* between $\forall$ and $\exists$, in order to define a negation or complementation operation on automata.

Definition 1.17. Assume that, for some one-step language L_1, the map $(\cdot)^\delta$ provides, for each set A, a dual $\varphi^\delta \in L_1(A)$ for each $\varphi \in L_1(A)$. We define the *complement* of a given L_1-automaton $\mathbb{A} = \langle A, \Delta, \Omega, a_I\rangle$ as the automaton $\mathbb{A}^\delta := \langle A, \Delta^\delta, \Omega^\delta, a_I\rangle$ where $\Delta^\delta(a,c) := (\Delta(a,c))^\delta$, and $\Omega^\delta(a) := 1 + \Omega(a)$, for all $a \in A$ and $c \in \wp(\mathsf{P})$.

PROPOSITION 1.18. *Let L_1 and $(\cdot)^\delta$ be as in the previous definition. For each $\mathbb{A} \in Aut(L_1)$ and $\mathbb{S}$ we have that $\mathbb{A}^\delta$ accepts $\mathbb{S}$ if and only if $\mathbb{A}$ rejects $\mathbb{S}$.*

The proof of Proposition 1.18 is based on the fact that the *power* of $\exists$ in $\mathcal{A}(\mathbb{A}^\delta, \mathbb{S})$ is the same as that of $\forall$ in $\mathcal{A}(\mathbb{A}, \mathbb{S})$, as defined in [Kissig and Venema 2009]. As an immediate consequence, one may show that if the one-step language L_1 is closed under duals, then the class $Aut(L_1)$ is closed under taking complementation. Further on we will use Proposition 1.18 to show that the same may apply to some subclasses of $Aut(L_1)$.

The automata-theoretic characterisation of WMSO and NMSO will use classes of parity automata constrained by two additional properties. To formulate these we first introduce the notion of a *cluster*.

Definition 1.19. Let L_1 be a one-step language, and let $\mathbb{A} = \langle A, \Delta, \Omega, a_I\rangle$ be in $Aut(L_1)$. Write $\prec$ for the reachability relation in $\mathbb{A}$, i.e., the transitive closure of the "occurrence relation" $\{(a,b) \mid b \text{ occurs in } \Delta(a,c) \text{ for some } c \in \wp(\mathsf{P})\}$; in case $a \prec b$ we say that b is *active* in a. A *cluster* of $\mathbb{A}$ is a cell of the equivalence relation generated by the relation $\prec \cap \succ$ (i.e., the intersection of $\prec$ with its converse). A cluster is called *degenerate* if it consists of a single element which is not active in itself.

Observe that any cluster of an automaton is either degenerate, or else each of its states is active in itself and in any other state of the cluster. Observe too that there is a natural order on clusters: we may say that one cluster is *higher* than another if each member of the second cluster if active in each member of the first. We may assume without loss of generality that the initial state belongs to the highest cluster of the automaton.

We can now formulate the mentioned requirements on L_1-automata as follows.

Definition 1.20. Let $\mathbb{A} = \langle A, \Delta, \Omega, a_I\rangle$ be some L_1-automaton. We say that $\mathbb{A}$ is *weak* if $\Omega(a) = \Omega(b)$ whenever a and b belong to the same cluster. For the property of *continuity* we require that, for any cluster M, any state $a \in M$ and any $c \in \wp\mathsf{P}$, we have that $\Omega(a) = 1$ implies $\Delta(a,c) \in \mathtt{Con}_M(L_1(A))$ and $\Omega(a) = 0$ implies $\Delta(a,c) \in \mathtt{CoCon}_M(L_1(A))$.

We call a parity automaton $\mathbb{A} \in Aut(L_1)$ *weak-continuous* if it satisfies both properties, weakness and continuity. The classes of weak and weak-continuous automata are denoted as $Aut_w(L_1)$ and $Aut_{wc}(L_1)$, respectively.

Intuitively, weakness forbids an automaton to register non-trivial properties concerning the vertical 'dimension' of input trees, whereas continuity expresses a constraint on how much of the horizontal 'dimension' of an input tree the automaton is allowed to process. In terms of second-order logic, they correspond respectively to quantification over 'vertically' finite (i.e. included in well-founded subtrees) and 'horizontally' finite (i.e. included in finitely branching subtrees) sets. The conjunction of weakness and continuity thus corresponds to quantification over finite sets.

Remark 1.21. Any weak parity automaton $\mathbb{A}$ is equivalent to a special weak automaton $\mathbb{A}'$ with $\Omega : A' \to \{0, 1\}$. This is because *(weakness)* prevents states of different parity to occur infinitely often in acceptance games; so we may just replace any even priority with 0, and any odd priority with 1. We shall assume such a restricted priority map for weak parity automata.

1.3. μ-Calculi

We now see how to associate, with each one-step language L_1, the following variant μL_1 of the modal μ-calculus. These logics are of a fairly artificial nature; their main use is to smoothen the translations from automata to second-order formulas further on.

Definition 1.22. Given a one-step language L_1, we define the language μL_1 of the *μ-calculus over* L_1 by the following grammar:

$$\varphi ::= q \mid \neg q \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc_\alpha(\varphi_1, \ldots, \varphi_n) \mid \mu p.\varphi' \mid \nu p.\varphi',$$

where $p, q \in \mathsf{Prop}$, $\alpha(a_1, \ldots, a_n) \in L_1^+$ and φ' is monotone in p.

As in the case of the modal μ-calculus $\mu\mathrm{ML}$, we will freely use standard syntactic concepts and notations related to this language.

Observe that the language μL_1 generally has a wealth of modalities: one for each one-step formula in L_1.

The semantics of this language is given as follows.

Definition 1.23. Let $\mathbb{S}$ be a transition system. The satisfaction relation $\Vdash$ is defined in the standard way, with the following clause for the modality $\bigcirc_\alpha$:

$$\mathbb{S} \Vdash \bigcirc_\alpha(\varphi_1, \ldots, \varphi_n) \quad \text{iff} \quad (R[s_I], V_{\overline{\varphi}}) \models \alpha(a_1, \ldots, a_n), \tag{3}$$

where $V_{\overline{\varphi}}$ is the one-step valuation given by

$$V_{\overline{\varphi}}(a_i) := \{t \in R[s_I] \mid \mathbb{S}.t \Vdash \varphi_i\}. \tag{4}$$

Example 1.24.

(1) If we identify the modalities $\bigcirc_{\Diamond a}$ and $\bigcirc_{\Box a}$ of the basic modal one-step language ML_1 (cf. Definition 1.2) with the standard $\Diamond$ and $\Box$ operators, we may observe that $\mu(\mathrm{ML}_1)$ corresponds to the standard modal μ-calculus: $\mu(\mathrm{ML}_1) = \mu\mathrm{ML}$.

(2) Consider the one-step formulas $\alpha = \exists x(a_1(x) \wedge \forall y\, a_2(y))$, $\beta = \exists xy(x \not\approx y \wedge a_1(x) \wedge a_1(y))$, and $\gamma = \mathbf{W}x(a_1(x), a_2(x))$. Then $\bigcirc_\alpha(\varphi_1, \varphi_2)$ is equivalent to the modal formula $\Diamond\varphi_1 \wedge \Box\varphi_2$ and $\bigcirc_\beta(\varphi)$ expresses that the current state has at least two successors where φ holds. The formula $\bigcirc_\gamma(\varphi_1, \varphi_2)$ holds at a state s if all its successors satisfy φ_1 or φ_2, while at most finitely many successors refute φ_2. Neither $\bigcirc_\beta$ nor $\bigcirc_\gamma$ can be expressed in standard modal logic.

(3) If the one-step language L_1 is closed under taking disjunctions (conjunctions, respectively), it is easy to see that $\bigcirc_{\alpha\vee\beta}(\overline{\varphi}) \equiv \bigcirc_\alpha(\overline{\varphi}) \vee \bigcirc_\beta(\overline{\varphi})$ ($\bigcirc_{\alpha\wedge\beta}(\overline{\varphi}) \equiv \bigcirc_\alpha(\overline{\varphi}) \wedge \bigcirc_\beta(\overline{\varphi})$, respectively).

Alternatively but equivalently, one may interpret the language game-theoretically.

Definition 1.25. Given a μL_1-formula φ and a model $\mathbb{S}$ we define the *evaluation game* $\mathcal{E}(\varphi, \mathbb{S})$ as the two-player infinite game whose rules are given in the next table.

Position	Player	Admissible moves
(q, s), with $q \in FV(\varphi) \cap \kappa(s)$	$\forall$	$\emptyset$
(q, s), with $q \in FV(\varphi) \setminus \kappa(s)$	$\exists$	$\emptyset$
$(\neg q, s)$, with $q \in FV(\varphi) \cap \kappa(s)$	$\exists$	$\emptyset$
$(\neg q, s)$, with $q \in FV(\varphi) \setminus \kappa(s)$	$\forall$	$\emptyset$
$(\psi_1 \vee \psi_2, s)$	$\exists$	$\{(\psi_1, s), (\psi_2, s)\}$
$(\psi_1 \wedge \psi_2, s)$	$\forall$	$\{(\psi_1, s), (\psi_2, s)\}$
$(\bigcirc_\alpha(\varphi_1, \dots, \varphi_n), s)$	$\exists$	$\{Z \subseteq \{\varphi_1, \dots, \varphi_n\} \times R[s] \mid (R[s], V_Z^*) \models \alpha(\overline{a})\}$
$Z \subseteq Sfor(\varphi) \times S$	$\forall$	$\{(\psi, s) \mid (\psi, s) \in Z\}$
$(\mu p.\varphi, s)$	$-$	$\{(\varphi, s)\}$
$(\nu p.\varphi, s)$	$-$	$\{(\varphi, s)\}$
(p, s), with $p \in BV(\varphi)$	$-$	$\{(\delta_p, s)\}$

For the admissible moves at a position of the form $(\bigcirc_\alpha(\varphi_1, \dots, \varphi_n), s)$, we consider the valuation $V_Z^* : \{a_1, \dots, a_n\} \to \wp(R[s])$, given by $V_Z^*(a_i) := \{t \in R[s] \mid (\varphi_i, t) \in Z\}$. The winning conditions of $\mathcal{E}(\varphi, \mathbb{S})$ are standard: $\exists$ wins those infinite matches of which the highest variable that is unfolded infinitely often during the match is a μ-variable.

The following proposition, stating the adequacy of the evaluation game for the semantics of μL_1, is formulated explicitly for future reference. We omit the proof, which is completely routine.

FACT 1.26 (ADEQUACY). *For any formula $\varphi \in \mu L_1$ and any model $\mathbb{S}$ the following equivalence holds:*

$$\mathbb{S} \Vdash \varphi \quad \textit{iff} \quad (\varphi, s_I) \textit{ is a winning position for } \exists \textit{ in } \mathcal{E}(\varphi, \mathbb{S}).$$

We will be specifically interested in two fragments of μL_1, associated with the properties of being noetherian and continuous, respectively, and with the associated variants of the μ-calculus μL_1 where the use of the fixpoint operator μ is restricted to formulas belonging to these two respective fragments.

Definition 1.27. Let Q be a set of proposition letters. We first define the fragment $\mathtt{Noe}_{\mathsf{Q}}(\mu L_1)$ of μL_1 of formulas that are syntactically *noetherian* in Q by the following grammar:

$$\varphi ::= q \mid \psi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc_\alpha(\varphi_1, \dots, \varphi_n) \mid \mu p.\varphi'$$

where $q \in \mathsf{Q}$, ψ is a Q-free μML-formula, $\alpha(a_1, \dots, a_n) \in L_1^+$ and $\varphi' \in \mathtt{Noe}_{\mathsf{Q} \cup \{p\}}(\mu L_1)$. The *co-noetherian* fragment $\mathtt{CoNoe}_Q(\mu L_1)$ is defined dually.

Similarly, we define the fragment $\mathtt{Con}_{\mathsf{Q}}(\mu L_1)$ of μL_1-formulas that are syntactically *continuous* in Q as follows:

$$\varphi ::= q \mid \psi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc_\alpha(\varphi_1, \dots, \varphi_k, \psi_1, \dots, \psi_m) \mid \mu p.\varphi'$$

where $p \in \mathsf{Prop}$, $q \in \mathsf{Q}$, ψ, ψ_i are Q-free μL_1-formula, $\alpha(a_1, \dots, a_k, b_1, \dots, b_m) \in \mathtt{Con}_{\overline{a}}(L_1)(\overline{a}, \overline{b})$, and $\varphi' \in \mathtt{Con}_{\mathsf{Q} \cup \{p\}}(\mu L_1)$. The *co-continuous* fragment $\mathtt{CoCon}_Q(\mu L_1)$ is defined dually.

Based on this we can now define the mentioned variants of the μ-calculus μL_1 where the use of the least (greatest) fixpoint operator can only be applied to formulas that belong to, respectively, the noetherian (co-noetherian) and continuous (co-continuous) fragment of the language that we are defining.

Definition 1.28. The formulas of the *alternation-free* μ-calculus $\mu_D L_1$ are defined by the following grammar:

$$\varphi ::= q \mid \neg q \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc_\alpha(\varphi_1, \ldots, \varphi_n) \mid \mu p.\varphi' \mid \nu p.\varphi'',$$

where $\alpha(a_1, \ldots, a_n) \in L_1^+$, $\varphi' \in \mu_D L_1 \cap \mathtt{Noe}_p(\mu L_1)$ and dually $\varphi'' \in \mu_D L_1 \cap \mathtt{CoNoe}_p(\mu L_1)$.

Similarly, the formulas of the *continuous* μ-calculus $\mu_C L_1$ are given by the grammar

$$\varphi ::= q \mid \neg q \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc_\alpha(\varphi_1, \ldots, \varphi_n) \mid \mu p.\varphi' \mid \nu p.\varphi'',$$

where $\alpha(a_1, \ldots, a_n) \in L_1^+$, $\varphi' \in \mu_C L_1 \cap \mathtt{Con}_p(\mu L_1)$ and dually $\varphi'' \in \mu_C L_1 \cap \mathtt{CoCon}_p(\mu L_1)$.

Example 1.29. Following up on Example 1.24, it is easy to verify that $\mu_D \mathrm{ML}_1 = \mu_N \mathrm{ML}$ and $\mu_C \mathrm{ML}_1 = \mu_C \mathrm{ML}$.

1.4. From automata to formulas

It is well-known that there are effective translations from automata to formulas and vice versa [Grädel et al. 2002]. The first result on L_1-automata that we need in this paper is the following.

THEOREM 1.30. *There is an effective procedure that, given an automaton $\mathbb{A}$ in $Aut(L_1)$, returns a formula $\xi_{\mathbb{A}} \in \mu L_1$ which satisfies the following properties:*

(1) $\xi_{\mathbb{A}}$ is equivalent to $\mathbb{A}$;
(2) $\xi_{\mathbb{A}} \in \mu_D L_1$ if $\mathbb{A} \in Aut_w(L_1)$;
(3) $\xi_{\mathbb{A}} \in \mu_C L_1$ if $\mathbb{A} \in Aut_{wc}(L_1)$.

In the remainder of this subsection we focus on the proof of this theorem, which is (a refinement of) a variation of the standard proof showing that any modal automaton can be translated into an equivalent formula in the modal μ-calculus (see for instance [Venema 2012, Section 6]). For this reason we will not go into the details of showing that $\mathbb{A}$ and $\xi_{\mathbb{A}}$ are equivalent, but we will provide a detailed definition of the translation, and pay special attention to showing that the translations of weak and of weak-continuous L_1-automata land in the right fragments of μL_1.

The definition of $\xi_{\mathbb{A}}$ is by induction on the number of clusters of $\mathbb{A}$, with a subinduction based on the number of states in the top cluster of $\mathbb{A}$. For this inner induction we need to widen the class of L_1-automata, and it will also be convenient to introduce the notion of a preautomaton (which is basically an automaton without initital state).

Definition 1.31. A *preautomaton* based on L_1 and P, or briefly: a *preautomaton*, is a triple $\mathbb{A} = \langle A, \Delta, \Omega \rangle$ such that A is a (possibly empty) finite set of states, $\Delta : A \times \wp(\mathsf{P}) \to L_1^+(A)$ and $\Omega : A \to \mathbb{N}$.

Given a set X of propositional variables, a *generalized preautomaton* over P and X is a triple $\mathbb{A} = \langle A, \Delta, \Omega \rangle$ such that $\Omega : A \to \mathbb{N}$ is a priority map on the finite state set A, while the transition map is of the form $\Delta : A \times \wp(\mathsf{P}) \to L_1^+(A \cup X)$.

Since we will not prove the semantic equivalence of $\mathbb{A}$ and $\xi_{\mathbb{A}}$, we confine our attention to the semantics of generalised automata to the following remark.

Remark 1.32. Generalised automata operate on $\mathsf{P} \cup X$-models; it will be convenient to denote these structures as quintuples of the form $\mathbb{S} = \langle S, R, \kappa, U, s_I \rangle$, where $\kappa : \mathsf{P} \to \wp S$ is a P-colouring and $U : S \to \wp X$ is an X-valuation on S. The acceptance game $\mathcal{A}(\mathbb{A}, \mathbb{S})$ associated with a generalised automaton $\mathbb{A} = \langle A, a_I, \Delta, \Omega \rangle$ and a $\mathsf{P} \cup X$-model $\mathbb{S}$ is a minor variation of the one associated with a standard automaton. At a basic position of the form $(a, s) \in A \times S$, as before $\exists$ needs to come up with a valuation V turning the set $R[s]$ into the domain of a one-step model of the formula $\Delta(a, \kappa(s))$. The difference with standard automata is that the formula $\Delta(a, \kappa(s))$ may now involve variables from the set X, and that the interpretation of these is already fixed by the valuation U of $\mathbb{S}$, namely by the restriction $U_s : x \mapsto R[s] \cap U(x)$ to the collection of successors of s. In table, we can present this game as follows:

Position	Player	Admissible moves	Priority
$(a, s) \in A \times S$	$\exists$	$\{V : A \to \wp(R[s]) \mid (R[s], V \cup U_s) \models \Delta(a, \kappa(s))\}$	$\Omega(a)$
$V : A \to \wp(S)$	$\forall$	$\{(b, t) \mid t \in V(b)\}$	0

where $V \cup U_s$ is the obviously defined $A \cup X$-valuation on $R[s]$.

We now turn to the definition of the translation. We will use the same notation for substitution as for the standard μ-calculus, cf. Subsection 2.3. In addition we use the following notation.

Definition 1.33. Consider, for some preautomaton $\mathbb{A} = \langle A, \Delta, \Omega\rangle$, some state $a \in A$, and some colour $c \in \wp(\mathsf{P})$, the one-step formula $\Delta(a, c) \in L_1(A)$. Suppose that for some subset $B \subseteq A$ we have a collection of μL_1-formulas $\{\varphi_b \mid b \in B\}$. Without loss of generality we may write $\Delta(a, c) = \alpha(a_1, \ldots, a_m, b_1, \ldots, b_n)$, where the a_i and b_j belong to $A \setminus B$ and B respectively. Then we will denote the μL_1-formula $\bigcirc_\alpha(a_1, \ldots, a_m, \varphi_1, \ldots, \varphi_n)$ as follows:

$$\bigcirc_{\Delta(a,c)}(\varphi_b/b \mid b \in B) := \bigcirc_\alpha(a_1, \ldots, a_m, \varphi_1, \ldots, \varphi_n).$$

We can now define the desired translation from L_1-automata to μL_1-formulas.

Definition 1.34. By induction on the number of clusters of a preautomaton $\mathbb{A} = \langle A, \Delta, \Omega\rangle$ we define a map

$$\mathtt{tr}_{\mathbb{A}} : A \to \mu L_1(P).$$

Based on this definition, for an automaton $\mathbb{A} = \langle A, \Delta, \Omega, a_I\rangle$ we put

$$\xi_{\mathbb{A}} := \mathtt{tr}_{\langle A,\Delta,\Omega\rangle}(a_I).$$

In the base case of the definition of $\mathtt{tr}$ the preautomaton $\mathbb{A}$ has no clusters at all, which means in particular that $A = \varnothing$. In this case we let $\mathtt{tr}_{\mathbb{A}}$ be the empty map.

In the inductive case we assume that $\mathbb{A} = \langle A, \Delta, \Omega, a_I\rangle$ does have clusters. Let $B \neq \varnothing$ be the highest cluster, and let $\mathbb{A}^-$ denote the preautomaton with carrier $A \setminus B$, obtained by restricting the maps Δ and Ω to the set $A \setminus B$. Then inductively we may assume a translation $\mathtt{tr}_{\mathbb{A}^-} : (A \setminus B) \to \mu L_1(P)$, and we will define

$$\mathtt{tr}_{\mathbb{A}}(a) := \mathtt{tr}_{\mathbb{A}^-}(a), \quad \text{if } a \in A \setminus B.$$

To extend this definition to the states in B, we make a case distinction. If B is a degenerate cluster, that is, $B = \{b\}$ for some state b which is not active in itself, then we define

$$\mathtt{tr}_{\mathbb{A}}(b) := \bigvee_{c \in \wp\mathsf{P}} \bigcirc_{\Delta(b,c)}(\mathtt{tr}_{\mathbb{A}^-}(a)/a \mid a \in A \setminus B).$$

The main case of the definition is where B is not degenerate. Fix an enumeration $b_1, \ldots, b_n$ of B such that $i \leq j$ implies $\Omega(b_i) \leq \Omega(b_j)$. Let $\mathbb{A}_k$ be the generalized preautomaton[2] obtained from $\mathbb{A}$ by restricting the transition and priority map to the set

$$A_k := (A \setminus B) \cup \{b_1, \ldots, b_k\},$$

[2]Here we see the reason to generalise the notion of an automaton: in the structure $\mathbb{A}_k$ $(0 \leq k \leq n)$ the objects $b_{k+1}, \ldots, b_n$ are no longer states, but in the formulas $\Delta_k(a, c)$ they still occur at the position of states.

so that $\mathbb{A}^0 = \mathbb{A}^-$ and $\mathbb{A}^n = \mathbb{A}$. Where $B_k := \{b_1, \ldots, b_k\}$, we now define, by induction on k, a map

$$\mathtt{tr}^k : B \to \mu L_1(P \cup (B \setminus B_k)).$$

In the base case of this definition we set

$$\mathtt{tr}^0(b) := \bigvee_{c \in \wp \mathsf{P}} \bigcirc_{\Delta(b,c)}(\mathtt{tr}_{\mathbb{A}^-}(a)/a \mid a \in A \setminus B),$$

and in the inductive case we first define $\eta_{k+1} := \mu$ if $\Omega(b_{k+1})$ is odd, and $\eta_{k+1} := \nu$ if $\Omega(b_{k+1})$ is even, and then set

$$\begin{array}{lll} \mathtt{tr}^{k+1}(b_{k+1}) & := & \eta_{k+1} b_{k+1}.\mathtt{tr}^0(b_{k+1})[\mathtt{tr}^k(b_i)/b_i \mid 1 \leq i \leq k] \\ \mathtt{tr}^{k+1}(b_i) & := & \mathtt{tr}^k(b_i)[\mathtt{tr}^{k+1}(b_{k+1})/b_{k+1}] \qquad \text{for } i \neq k+1. \end{array}$$

Finally, we complete the definition of $\mathtt{tr}_{\mathbb{A}}$ by putting

$$\mathtt{tr}_{\mathbb{A}}(b) := \mathtt{tr}^n(b),$$

for any $b \in B$.

In the proof of Theorem 1.30 we will need the following closure property of the fragments $\mathtt{Noe}_{\mathsf{Q}}(\mu L_1)$ and $\mathtt{CoCon}_{\mathsf{Q}}(\mu L_1)$.

PROPOSITION 1.35. *Let $\mathsf{R} \subseteq \mathsf{Q}$ be sets of proposition letters, and let φ and φ_q, for each $q \in \mathsf{Q}$, be formulas in μL_1.*

(1) If φ and each φ_q belongs to $\mathtt{Noe}_{\mathsf{Q}\setminus\mathsf{R}}(\mu L_1)$ ($\mathtt{CoNoe}_{\mathsf{Q}\setminus\mathsf{R}}(\mu L_1)$), then so does $\varphi[\varphi_q/q \mid q \in \mathsf{Q}]$.

(2) If φ and each φ_q belongs to $\mathtt{Con}_{\mathsf{Q}\setminus\mathsf{R}}(\mu L_1)$ ($\mathtt{CoCon}_{\mathsf{Q}\setminus\mathsf{R}}(\mu L_1)$), then so does $\varphi[\varphi_q/q \mid q \in \mathsf{Q}]$.

Both items of this proposition can be proved by a straightforward formula induction — we omit the details.

PROOF OF THEOREM 1.30. As mentioned, the verification of the equivalence of $\xi_{\mathbb{A}}$ and $\mathbb{A}$ is a standard exercise in the theory of parity automata and mu-calculi, and so we omit the details. We also skip the proof of item (2), completely focussing on the (harder) third item.

To prove this item, it suffices to take an arbitrary continuous-weak L_1-preautomaton $\mathbb{A} = \langle A, \Delta, \Omega \rangle$ for the set P, and to show that

$$\mathtt{tr}_{\mathbb{A}}(a) \in \mu_C L_1(\mathsf{P}) \tag{5}$$

for all $a \in A$. We will prove this by induction on the number of clusters of $\mathbb{A}$.

Since there is nothing to prove in the base case of the proof, we immediately move to the inductive case. Let B be the highest cluster of $\mathbb{A}$. By the induction hypothesis we have $\mathtt{tr}_{\mathbb{A}}(a) = \mathtt{tr}_{\mathbb{A}^-}(a) \in \mu_C L_1(\mathsf{P})$ for all $a \in A \setminus B$, where $\mathbb{A}^-$ is as in Definition 1.34. To show that (5) also holds for all $b \in B$, we distinguish cases.

If B is a degenerate cluster, say, $B = \{b\}$, then for every $c \in \wp(\mathsf{P})$, the variables occurring in the formula $\Delta(b, c) \in L_1(A)$ are all from $A \setminus \{b\}$. Given the definition of $\mathtt{tr}_{\mathbb{A}}(b)$, it suffices to show that all formulas of the form $\bigcirc_{\Delta(b,c)}(\mathtt{tr}_{\mathbb{A}}(a)/a \mid a \in A \setminus \{b\})$ belong to the set $\mu_C L_1(\mathsf{P})$, but this is immediate by the induction hypothesis and the definition of the language.

If, on the other hand, B is nondegenerate, let $b_1, \ldots, b_n$ enumerate B, and let, for $0 \leq k \leq n$, the map $\mathtt{tr}^k : B \to \mu L_1$ be as in Definition 1.34. We only consider the case where B is an odd cluster, i.e., $\Omega(b)$ is odd for all $b \in B$. Our key claim here is that

$$\mathtt{tr}^k(b_i) \in \mu_C L_1(\mathsf{P} \cup \{b_{k+1}, \ldots, b_n\}) \cap \mathtt{Con}_{\{b_{k+1},\ldots,b_n\}}(\mu L_1), \tag{6}$$

for all k and i with $0 \leq k \leq n$ and $0 < i \leq n$. We will prove this statement by induction on k — this is the 'inner' induction that we announced earlier on.

In the base case of this inner induction we need to show that $\mathtt{tr}^0(b_i)$ belongs to both $\mu_C L_1(\mathsf{P} \cup B)$ and $\mathtt{Con}_B(\mu L_1)$. Showing the first membership relation is straightforward; for the second, the key observation is that by our assumption on $\mathbb{A}$, every one-step formula of the form $\Delta(b_i, c)$ is syntactically continuous in every variable $b \in B$. Furthermore, by the outer inductive hypothesis we have $\mathtt{tr}_{\mathbb{A}^-}(a) \in \mu_C L_1(\mathsf{P}) \subseteq \mathtt{Con}_B(\mu L_1)$, for every $a \in A \setminus B$, and we trivially have that every variable $b \in B$ belongs to the set $\mathtt{Con}_B(\mu L_1)$. But then it is immediate by the definition of $\mathtt{Con}_B(\mu L_1)$ that this fragment contains the formula $\bigcirc_{\Delta(b_i,c)}(\mathtt{tr}_{\mathbb{A}^-}(b) \mid b \in B)$, and since this fragment is closed under taking disjunctions, we find that $\mathtt{tr}^0(b_i) \in \mathtt{Con}_B(\mu L_1)$ indeed. This finishes the proof of the base case of the inner induction.

For the inner induction step we fix a k and assume that (6) holds for this k and for all i with $0 < i \leq k$. We will prove that

$$\mathtt{tr}^{k+1}(b_i) \in \mu_C L_1(\mathsf{P}) \cap \mathtt{Con}_{\{b_{k+2},\ldots,b_n\}}(\mu L_1). \tag{7}$$

first for $i = k+1$, and then for an arbitrary $i \neq k+1$. To prove (7) for the case $i = k+1$, first note that

$$\mathtt{tr}^0(b_{k+1}) \ \in \ \mu_C L_1(\mathsf{P} \cup B) \cap \mathtt{Con}_B(\mu L_1),$$

as we just saw in the base case of the inner induction. But then it is immediate by Proposition 1.35 and the induction hypothesis on the formulas $\mathtt{tr}^k(b_i)$ that

$$\mathtt{tr}^0(b_{k+1})[\mathtt{tr}^k(b_i)/b_i \mid 1 \leq i \leq k] \in \mathtt{Con}_{\{b_{k+1},\ldots,b_n\}}(\mu L_1),$$

and from this it easily follows by the definition of $\mathtt{Con}_{\{b_{k+1},\ldots,b_n\}}(\mu L_1)$ that

$$\mu b_{k+1} b_{k+1}.\mathtt{tr}^0(b_{k+1})[\mathtt{tr}^k(b_i)/b_i \mid 1 \leq i \leq k] \in \mathtt{Con}_{\{b_{k+2},\ldots,b_n\}}(\mu L_1)$$

as well. That is,

$$\mathtt{tr}^{k+1}(b_{k+1}) \in \mathtt{Con}_{\{b_{k+2},\ldots,b_n\}}(\mu L_1).$$

This is the crucial step in proving (7) for the case $i = k+1$, the proof that $\mathtt{tr}^{k+1}(b_{k+1}) \in \mu_C L_1(\mathsf{P})$ is easy.

Second, to prove (7) for the case $i \neq k+1$, we first recall that by the induction hypothesis we have

$$\mathtt{tr}^k(b_i) \in \mu_C L_1(\mathsf{P}) \cap \mathtt{Con}_{\{b_{k+1},\ldots,b_n\}}(\mu L_1),$$

while we just saw that $\mathtt{tr}^{k+1}(b_{k+1}) \in \mathtt{Con}_{\{b_{k+2},\ldots,b_n\}}(\mu L_1)$. But from the latter two statements it is immediate by Proposition 1.35 that

$$\mathtt{tr}^{k+1}(b_i) = \mathtt{tr}^k(b_i)[\mathtt{tr}^{k+1}(b_{k+1})/b_{k+1}] \in \mathtt{Con}_{\{b_{k+2},\ldots,b_n\}}(\mu L_1)$$

so that we have indeed proved (6) for the case $i \neq k+1$. This finishes the proof of the inner induction.

Finally, it follows from (6), instantiated with $k = n$, that for all $b \in B$ we have

$$\mathtt{tr}_{\mathbb{A}}(b) = \mathtt{tr}^n(b) \in \mu_C L_1(\mathsf{P}),$$

as required for proving the outer induction step. In other words, we are finished with the proof of (5), and hence, finished with the proof of the theorem. □

1.5. From Formulas to Automata

In this subsection we focus on the meaning preserving translation in the opposite direction, viz., from formula to automata. In our set-up we need the one-step language to be closed under conjunctions and disjunctions.

THEOREM 1.36. *Let L_1 be a on-step language that is closed under taking conjunctions and disjunctions. Then there is an effective procedure that, given a formula $\xi \in \mu L_1$ returns an automaton $\mathbb{A}_\xi$ in $Aut(L_1)$, which satisfies the following properties:*

(1) $\mathbb{A}_\xi$ is equivalent to ξ;
(2) $\mathbb{A}_\xi \in Aut_w(L_1)$ if $\xi \in \mu_D L_1$;
(3) $\mathbb{A}_\xi \in Aut_{wc}(L_1)$ if $\xi \in \mu_C L_1$.

As in the case of the translation from automata to formulas, the proof of part (1) of this theorem, is a straightforward variation of the standard proof showing that any fixpoint modal formula can be translated into an equivalent modal automaton (see for instance [Venema 2012, Section 6]). The point is to show that this standard construction transforms formulas from $\mu_D L_1$ and $\mu_C L_1$ into automata of the right kind. In fact, we will confine our attention to proving the (second and) third part of the theorem; the fact that the input formula belongs to the alternation-free fragment of μL_1 enables a slightly simplified presentation of the construction.

Our first observation is that without loss of generality we may confine attention to *guarded* formulas.

Definition 1.37. An occurrence of a bound variable p in $\xi \in \mu L_1$ is called *guarded* if there is a modal operator between its binding definition and the variable itself. A formula $\xi \in \mu L_1$ is called *guarded* if every occurrence of every bound variable is guarded.

There is a standard construction, going back to [Kozen 1983], which transforms any formula ξ in μL_1 into an equivalent guarded $\xi^\flat \in \mu L_1$, and it is easily verified that the construction $(\cdot)^\flat$ restricts to the fragments $\mu_D L_1$ and $\mu_C L_1$. It therefore suffices to show that any guarded formula in $\mu_D L_1$ ($\mu_C L_1$, respectively) can be transformed into an equivalent continuous-weak L_1-automaton.

In the remainder of this section we will show that any guarded formula ξ in $\mu_C L_1$ can be transformed into an equivalent continuous-weak L_1-automaton $\mathbb{A}_\xi$; the analogous result for $\mu_D L_1$ will be obvious from our construction.

We will prove the result by induction on the so-called weak alternation depth of the formula ξ. We only consider the μ-case of the inductive proof step; that is, we assume that ξ can be obtained as a formula in the following grammar, for some set Q of variables:

$$\varphi ::= q \mid \psi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc_\alpha(\overline{\varphi}, \overline{\psi}) \mid \mu q.\varphi, \tag{8}$$

where $q \in \mathsf{Q}$, every ψ, ψ_j is Q-free, and $\alpha(\overline{a}, \overline{b}) \in \mathtt{Con}_{\overline{a}}(L_1)$. We write $\varphi \trianglelefteq' \xi$ if φ is a subformula of ξ according to the grammar (8) (that is, we consider Q-free formulas ψ to be atomic). Furthermore, we inductively assume that for every Q-free formula $\psi \trianglelefteq' \xi$ we have already constructed an equivalent continuous-weak L_1-automaton $\mathbb{A}_\psi = \langle A_\psi, a_\psi, \Delta_\psi, \Omega_\psi \rangle$. (Observe that every subformula of a guarded formula it itself guarded, so that we are justified to apply the induction hypothesis.)

Define $\Psi := \{\psi \trianglelefteq' \xi \mid \psi \text{ Q-free}\}$, let Φ consist of all formulas $\varphi \notin \Psi$ that occur as some $\varphi = \varphi_i$ in a formula $\bigcirc_\alpha(\overline{\varphi}, \overline{\psi}) \trianglelefteq' \xi$, and set $A_\Psi := \{a_\psi \mid \psi \in \Psi\}$ and $A_\Phi := \{a_\varphi \mid \varphi \in \Phi\}$. It now follows by guardedness of ξ that there is a unique map $(\cdot)^o$ assigning a one-step formula $\chi^o \in L_1(A_\Psi \cup A_\Phi)$ to every formula $\chi \trianglelefteq' \xi$ such that

$$\begin{array}{lcl}
q^o & = & \delta_q^o \\
\psi^o & = & a_\psi \\
(\varphi_0 \vee \varphi_1)^o & = & \varphi_0^o \vee \varphi_1^o \\
(\varphi_0 \wedge \varphi_1)^o & = & \varphi_0^o \wedge \varphi_1^o \\
\bigcirc_\alpha(\varphi_1, \ldots, \varphi_n, \psi_1, \ldots, \psi_k)^o & = & \alpha(a_{\varphi_1}, \ldots, a_{\varphi_n}, a_{\psi_1}, \ldots, a_{\psi_k}) \\
(\mu q.\delta_q)^o & = & \delta_q^o,
\end{array}$$

where δ_q is the unique formula δ such that $\mu q.\delta \trianglelefteq' \xi$.

It is easy to verify that for every $\chi \trianglelefteq' \xi$, the formula χ^o is continuous in A_Φ. We may thus assume without loss of generality that every χ^o belongs to the syntactic fragment $\mathtt{Con}_{A_\Phi}(L_1)$. (Should this not be the case, then by Theorem 1.12 we may replace χ^o with an equivalent formula that does belong to this fragment.) We are now ready to define the automaton $\mathbb{A}_\xi = \langle A_\xi, a_I, \Delta_\xi, \Omega_\xi \rangle$ by putting

$$\begin{array}{lcl}
A_\xi & := & \{a_\xi\} \cup A_\Phi \cup \bigcup_{\psi \trianglelefteq' \xi} A_\psi \\
a_I & := & a_\xi \\
\Delta_\xi(a, c) & := & \begin{cases} \chi^o & \text{if } a = a_\chi \text{ with } \chi \in \{\xi\} \cup \Phi \\ \Delta_\psi(a, c) & \text{if } a \in A_\psi. \end{cases} \\
\Omega_\xi(a) & := & \begin{cases} 1 & \text{if } a \in \{a_\xi\} \cup A_\Phi \\ \Omega_\psi(a) & \text{if } a \in A_\psi. \end{cases}
\end{array}$$

It is then straightforward to check that $\mathbb{A}_\xi$ is a continuous-weak L_1-automaton. Proving the equivalence of ξ and $\mathbb{A}_\xi$ is a routine exercise.

2. AUTOMATA FOR WMSO

In this section we start looking at the automata-theoretic characterisation of WMSO. That is, we introduce the following automata, corresponding to this version of monadic second-order logic; these WMSO-*automata* are the continuous-weak automata for the one-step language FOE_1^∞, cf. Definition 1.20.

Definition 2.1. A WMSO-*automaton* is a continuous-weak automaton for the one-step language FOE_1^∞.

Recall that our definition of continuous-weak automata is syntactic in nature, i.e., if $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ is a WMSO-automaton, then for any pair of states a, b with $a \prec b$ and $b \prec a$, and any $c \in C$, we have $\Delta(a, c) \in \mathtt{Con}_b(\mathrm{FOE}_1^\infty(A)^+)$ if $\Omega(a)$ is odd and $\Delta(a, c) \in \mathtt{CoCon}_b(\mathrm{FOE}_1^\infty(A)^+)$ if $\Omega(a)$ is even.

The main result of this section states one direction of the automata-theoretic characterisation of WMSO.

THEOREM 2.2. *There is an effective construction transforming a* WMSO-*formula* φ *into a* WMSO-*automaton* $\mathbb{A}_\varphi$ *that is equivalent to* φ *on the class of trees.*

The proof proceeds by induction on the complexity of φ. For the inductive steps, we will need to verify that the class of WMSO-automata is closed under the boolean operations and finite projection. The latter closure property requires most of the work: we devote Section 2.1 to a simulation theorem that puts WMSO-automata in a suitable shape for the projection construction. To this aim, it is convenient to define a closure operation on classes of tree models corresponding to the semantics of WMSO quantification. The inductive step of the proof of Theorem 2.2 will show that the classes that are recognizable by WMSO-automata are closed under this operation.

Definition 2.3. Fix a set P of proposition letters, a proposition letter $p \notin P$ and a language C of $\mathsf{P} \cup \{p\}$-labeled trees. The *finitary projection* of C over p is the language of P-labeled trees defined as

$$\exists_F p.\mathsf{C} := \{\mathbb{T} \mid \text{ there is a finite } p\text{-variant } \mathbb{T}' \text{ of } \mathbb{T} \text{ with } \mathbb{T}' \in \mathsf{C}\}.$$

A collection of classes of tree models is *closed under finitary projection over* p if it contains the class $\exists_F p.\mathsf{C}$ whenever it contains the class C itself.

2.1. Simulation theorem for WMSO-automata

Our next goal is a *projection construction* that, given a WMSO-automaton $\mathbb{A}$, provides one recognizing $\exists_F p.\mathsf{TMod}(\mathbb{A})$. For SMSO-automata, the analogous construction crucially uses the following *simulation theorem*: every SMSO-automaton $\mathbb{A}$ is equivalent to a *non-deterministic* automaton $\mathbb{A}'$ [Walukiewicz 1996]. Semantically, non-determinism yields the appealing property that every node of the input model $\mathbb{T}$ is associated with at most one state of $\mathbb{A}'$ during the acceptance game— that means, we may assume $\exists$'s strategy f in $\mathcal{A}(\mathbb{A}', \mathbb{T})$ to be *functional* (*cf.* Definition 2.9 below). This is particularly helpful in case we want to define a p-variant of $\mathbb{T}$ that is accepted by the projection construct on $\mathbb{A}'$: our decision whether to label a node s with p or not, will crucially depend on the value $f(a, s)$, where a is the unique state of $\mathbb{A}'$ that is associated with s. Now, in the case of WMSO-automata we are interested in guessing *finitary* p-variants, which requires f to be functional only on a *finite* set of nodes. Thus the idea of our simulation theorem is to turn a WMSO-automaton $\mathbb{A}$ into an equivalent one $\mathbb{A}^F$ that behaves non-deterministically on a *finite* portion of any accepted tree.

For SMSO-automata, the simulation theorem is based on a powerset construction: if the starting automaton has carrier A, the resulting non-deterministic automaton is based on "macro-states" from the set $\wp A$. Analogously, for WMSO-automata we will associate the non-deterministic behaviour with macro-states. However, as explained above, the automaton $\mathbb{A}^F$ that we construct has to be non-deterministic just on finitely many nodes of the input and may behave as $\mathbb{A}$ (i.e. in "alternating mode") on the others. To this aim, $\mathbb{A}^F$ will be "two-sorted", roughly consisting of a copy of $\mathbb{A}$ (with carrier A) together with a variant of its powerset construction, based both on A and $\wp A$. For any accepted $\mathbb{T}$, the idea is to make any match π of $\mathcal{A}(\mathbb{A}^F, \mathbb{T})$ consist of two parts:

(Non-deterministic mode). For finitely many rounds π is played on macro-states, i.e. positions belong to the set $\wp A \times T$. In her strategy player $\exists$ assigns macro-states (from $\wp A$) only to *finitely many* nodes, and states (from A) to the rest. Also, her strategy is functional in $\wp A$, i.e. it assigns *at most one macro-state* to each node.

(Alternating mode). At a certain round, π abandons macro-states and turns into a match of the game $\mathcal{A}(\mathbb{A}, \mathbb{T})$, i.e. all subsequent positions are from $A \times T$ (and are played according to a not necessarily functional strategy).

Therefore successful runs of $\mathbb{A}^N$ will have the property of processing only a *finite* amount of the input with $\mathbb{A}^N$ being in a macro-state and all the rest with $\mathbb{A}^N$ behaving exactly as $\mathbb{A}$. We now proceed in steps towards the construction of $\mathbb{A}^N$. First, recall from Definition 1.10 that a *A-type* is just a subset of A. We now define a notion of liftings for sets of types, which is instrumental in translating the transition function from states on macro-states.

Definition 2.4. The *lifting* of a type $S \in \wp A$ is defined as the following $\wp A$-type:

$$S^{\Uparrow} := \begin{cases} \{S\} & \text{if } S \neq \varnothing \\ \varnothing & \text{if } S = \varnothing. \end{cases}$$

This definition is extended to sets of A-types by putting $\Sigma^{\Uparrow} := \{S^{\Uparrow} \mid S \in \Sigma\}$.

The distinction between empty and non-empty elements of Σ is to ensure that the empty type on A is lifted to the empty type on $\wp A$. Notice that the resulting set $\Sigma^{\Uparrow}$ is either empty or contains exaclty one $\wp A$-type. This property is important for functionality, see below.

Next we define a translation on the sentences associated with the transition function of the original WMSO-automaton. Following the intuition given above, we want to work with sentences that can be made true by assigning macro-states (from $\wp A$) to finitely many nodes in the model, and ordinary states (from A) to all the other nodes. Moreover, each node should be associated with *at most one* macro-state, because of functionality. These desiderata are expressed for one-step formulas as *$\wp A$-continuity* and *$\wp A$-separability*, see the Definitions 1.7 and 1.14. For the language FOE_1^{∞}, Theorem 1.11 and Proposition 1.15 guarantee these properties when formulas are in a certain syntactic shape. The next definition will provide formulas that conform to this particular shape.

Definition 2.5. Let $\varphi \in \mathrm{FOE}_1^{\infty+}(A)$ be a formula of shape $\nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$ for some $\Pi, \Sigma \subseteq \wp A$ and $\overline{\mathbf{T}} = \{T_1, \ldots, T_k\} \subseteq \wp A$. We define $\varphi^F \in \mathrm{FOE}_1^{\infty+}(A \cup \wp A)$ as the formula $\nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}^{\Uparrow}, \Pi^{\Uparrow} \cup \Sigma^{\Uparrow}, \Sigma)$, that means,

$$\begin{aligned} \varphi^F := \exists \overline{\mathbf{x}}.\Big(\mathrm{diff}(\overline{\mathbf{x}}) \wedge \bigwedge_{0 \leq i \leq n} \tau^+_{T_i^{\Uparrow}}(x_i) \wedge \forall z.(\mathrm{diff}(\overline{\mathbf{x}}, z) \to \bigvee_{S \in \Pi^{\Uparrow} \cup \Sigma^{\Uparrow} \cup \Sigma} \tau^+_S(z))\Big) \\ \wedge \bigwedge_{P \in \Sigma} \exists^\infty y.\tau^+_P(y) \wedge \forall^\infty y. \bigvee_{P \in \Sigma} \tau^+_P(y) \end{aligned} \tag{9}$$

We combine the previous definitions to form the transition function for macro-states.

Definition 2.6. Let $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ be a WMSO-automaton. Fix $c \in C$ and $Q \in \wp A$. By Theorem 1.11, for some $\Pi, \Sigma \subseteq \wp A$ and $T_i \subseteq A$, there is a sentence $\Psi_{Q,c} \in \mathrm{FOE}_1^{\infty+}(A)$ in the basic form $\bigvee \nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$ such that $\bigwedge_{a \in Q} \Delta(a,c) \equiv \Psi_{Q,c}$. By definition $\Psi_{Q,c}$ is of the form $\bigvee_i \varphi_i$, with each φ_i of shape $\nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$. We put $\Delta^\sharp(Q,c) := \bigvee_i \varphi_i^F$, where the translation $(-)^F$ is given as in Definition 2.5. Observe that $\Delta^\sharp(Q,c)$ is of type $\mathrm{FOE}_1^{\infty+}(A \cup \wp A)$.

We have now all the ingredients to define our two-sorted automaton.

Definition 2.7. Let $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ be a WMSO-automaton. We define the *finitary construct over* $\mathbb{A}$ as the automaton $\mathbb{A}^F = \langle A^F, \Delta^F, \Omega^F, a_I^F \rangle$ given by

$$\begin{array}{lllllll} A^F & := & A \cup \wp A & \Omega^F(a) & := & \Omega(a) & \Delta^F(a,c) \ := \ \Delta(a,c) \\ a_I^F & := & \{a_I\} & \Omega^F(R) & := & 1 & \Delta^F(Q,c) \ := \ \Delta^\sharp(Q,c) \vee \bigwedge_{a \in Q} \Delta(a,c). \end{array}$$

Remark 2.8. In the standard powerset construction of non-deterministic parity automata ([Walukiewicz 2002], see also [Venema 2012; Arnold and Niwiński 2001]) macro-states are required to be *relations* rather than sets in order to determine whether a run through macro-states is accepting. This is not needed in our construction: macro-states will never be visited infinitely often in accepting runs, thus they may simply be assigned the priority 1.

The idea behind this definition is that $\mathbb{A}^F$ is enforced to process only a finite portion of any accepted tree while in the non-deterministic mode. This is encoded in game-theoretic terms through the notion of functional and finitary strategy.

Definition 2.9. Given a WMSO-automaton $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ and transition system $\mathbb{T}$, a strategy f for $\exists$ in $\mathcal{A}(\mathbb{A}, \mathbb{T})$ is *functional in* $B \subseteq A$ (or simply functional, if $B = A$) if for each node s in $\mathbb{T}$ there is at most one $b \in B$ such that (b,s) is a reachable position in an f-guided match. Also f is *finitary* in B if there are only finitely many nodes s in $\mathbb{T}$ for which a position (b,s) with $b \in B$ is reachable in an f-guided match.

The next proposition establishes the desired properties of the finitary construct.

THEOREM 2.10 (SIMULATION THEOREM FOR WMSO-AUTOMATA). *Let* $\mathbb{A}$ *be a* WMSO-*automaton and* $\mathbb{A}^F$ *its finitary construct.*

(1) $\mathbb{A}^F$ *is a* WMSO-*automaton.*
(2) For any tree model $\mathbb{T}$, *if* (a_I^F, s_I) *is a winning position for* $\exists$ *in* $\mathcal{A}(\mathbb{A}^F, \mathbb{T})$ *, then she has a winning strategy that is both functional and finitary in* $\wp A$.
(3) $\mathbb{A} \equiv \mathbb{A}^F$.

PROOF.

(1) Observe that any cluster of $\mathbb{A}^F$ involves states of exactly one sort, either A or $\wp A$. For clusters on sort A, weakness and continuity of $\mathbb{A}^F$ follow by the same properties of $\mathbb{A}$. For clusters on sort $\wp A$, weakness follows by observing that all macro-states in $\mathbb{A}^F$ have the same priority. Concerning continuity, by definition of Δ^F any macro-state can only appear inside a formula of the form $\varphi^F = \nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}^{\Uparrow}, \Pi^{\Uparrow} \cup \Sigma^{\Uparrow}, \Sigma)$ as in (9). Because $\wp A \cap \bigcup \Sigma = \varnothing$, by Theorem 1.12 φ^F is continuous in each $Q \in \wp A$.
(2) Let f be a (positional) winning strategy for $\exists$ in $\mathcal{A}(\mathbb{A}^F, \mathbb{T})@(a_I^F, s_I)$. We define a strategy f' for $\exists$ in the same game as follows:
   (a) On basic positions of the form $(a,s) \in A \times T$, let $V : A \to \wp R[s]$ be the valuation suggested by f. We let the valuation suggested by f' be the restriction V' of V to A. Observe that, as no predicate from $A^F \setminus A = \wp A$ occurs in $\Delta^F(a, \kappa(s)) = \Delta(a, \kappa(s))$, then V' also makes that sentence true in $R[s]$.

(b) For winning positions of the form $(R, s) \in \wp A \times T$, let $V_{R,s} : (\wp A \cup A) \to \wp R[s]$ be the valuation suggested by f. As f is winning, $\Delta^F(R, \kappa(s))$ is true in the model $V_{R,s}$. If this is because the disjunct $\bigwedge_{a \in R} \Delta(a, \kappa(s))$ is made true, then we can let f' suggest the restriction to A of $V_{R,s}$, for the same reason as in (a).
Otherwise, the disjunct $\Delta^\sharp(R, \kappa(s)) = \bigvee_i \varphi_i^F$ is made true. This means that, for some i, $(R[s], V_{R,s}) \models \varphi_i^F$. Now, by construction of φ_i^F as in (9), we have $\wp A \cap \bigcup \Sigma = \varnothing$. By Theorem 1.12, this implies that φ_i^F is continuous in $\wp A$. Thus we have a restriction $V'_{R,s}$ of $V_{R,s}$ that verifies φ_i^F and assigns only finitely many nodes to predicates from $\wp A$. Moreover, by construction of φ_i^F, for each $S \in \{T_1^\Uparrow, \ldots, T_k^\Uparrow\} \cup \in \Pi^\Uparrow \cup \Sigma^\Uparrow$, S contains at most one element from $\wp A$). Thus, by Proposition 1.15, φ_i^F is $\wp A$-separable. But then we may find a separating valuation $V''_{R,s} \leq_{\wp A} V''_{R,s}$ such that $V''_{R,s}$ verifies φ_i^F. Separation means that $V''_{R,s}$ associates with each node at most one predicate from $\wp A$, and the fact that $V''_{R,s} \leq_{\wp A} V''_{R,s}$, combined with the $\wp A$-continuity of $V'_{R,s}$ ensures $\wp A$-continuity of $V''_{R,s}$. In this case we let f' suggest $V''_{R,s}$ at position (R, s).

The strategy f' defined as above is immediately seen to be surviving for $\exists$. It is also winning, since at every basic winning position for $\exists$, the set of possible next basic positions offered by f' is a subset of those offered by f. By this observation it also follows that any f'-guided match visits basic positions of the form $(R, s) \in \wp A \times C$ only finitely many times, as those have odd parity. By definition, the valuation suggested by f' only assigns finitely many nodes to predicates in $\wp A$ from positions of that shape, and no nodes from other positions. It follows that f' is finitary in $\wp A$. Functionality in $\wp A$ also follows immediately by definition of f'.

(3) For the direction from left to right, it is immediate by definition of $\mathbb{A}^F$ that a winning strategy for $\exists$ in $\mathcal{G} = \mathcal{A}(\mathbb{A}, \mathbb{T})@(a_I, s_I)$ is also winning for $\exists$ in $\mathcal{G}^F = \mathcal{A}(\mathbb{A}^F, \mathbb{T})@(a_I^F, s_I)$.

For the direction from right to left, let f be a winning strategy for $\exists$ in $\mathcal{G}^F$. The idea is to define a strategy f' for $\exists$ in stages, while playing a match π' in $\mathcal{G}$. In parallel to π', a shadow match π in $\mathcal{G}^F$ is maintained, where $\exists$ plays according to the strategy f. For each round z_i, we want to keep the following relation between the two matches:

Either
(1) positions of the form $(Q, s) \in \wp A \times T$ and $(a, s) \in A \times T$ occur respectively in π and π', with $a \in Q$,
or
(2) the same position of the form $(a, s) \in A \times T$ occurs in both matches. (‡)

The key observation is that, because f is winning, a basic position of the form $(Q, s) \in \wp A \times T$ can occur only for finitely many initial rounds $z_0, \ldots, z_n$ that are played in π, whereas for all successive rounds $z_n, z_{n+1}, \ldots$ only basic positions of the form $(a, s) \in A \times T$ are encountered. Indeed, if this was not the case then either $\exists$ would get stuck or the highest priority occurring infinitely often would be odd, since states from $\wp A$ all have priority 1.
It follows that enforcing a relation between the two matches as in (‡) suffices to prove that the defined strategy f' is winning for $\exists$ in π'. For this purpose, first observe that (‡).1 holds at the initial round, where the positions visited in π' and π are respectively $(a_I, s_I) \in A \times T$ and $(\{a_I\}, s_I) \in A^F \times T$. Inductively, consider any round z_i that is played in π' and π, respectively with basic positions $(a, s) \in A \times T$ and $(q, s) \in A^F \times T$. To define the suggestion of f' in π', we distinguish two cases.

— First suppose that (q, s) is of the form $(Q, s) \in \wp A \times T$. By (‡) we can assume that a is in Q. Let $V_{Q,s} : A^F \to \wp(R[s])$ be the valuation suggested by f, verifying the sentence $\Delta^F(Q, \kappa(s))$. We distinguish two further cases, depending on which disjunct of $\Delta^F(Q, \kappa(s))$ is made true by $V_{Q,s}$.

(i) If $(R[s], V_{Q,s}) \models \bigwedge_{b \in Q} \Delta(b, \kappa(s))$, then we let $\exists$ pick the restriction to A of the valuation $V_{Q,s}$.

(ii) If $(R[s], V_{Q,s}) \models \Delta^\sharp(Q, \kappa(s))$, we let $\exists$ pick a valuation $V_{a,s} : A \to \wp(R[s])$ defined by putting, for each $b \in A$:

$$V_{a,s}(b) := \bigcup_{b \in Q'} \{t \in R[s] \mid t \in V_{Q,s}(Q')\} \cup \{t \in R[s] \mid t \in V_{Q,s}(b)\}.$$

It can be readily checked that the suggested move is legitimate for $\exists$ in π, i.e. it makes $\Delta(a, \kappa(s))$ true in $R[s]$.

For case (ii), observe that the nodes assigned to b by $V_{Q,s}$ have to be assigned to b also by $V_{a,s}$, as they may be necessary to fulfill the condition, expressed with $\exists^\infty$ and $\forall^\infty$ in $\Delta^\sharp$, that infinitely many nodes witness (or that finitely many nodes do not witness) some type.

We now show that (‡) holds at round z_{i+1}. If (i) is the case, any next position $(b, t) \in A \times T$ picked by player $\forall$ in π' is also available for $\forall$ in π, and we end up in case (‡.2). Suppose instead that (ii) is the case. Given a move $(b, t) \in A \times T$ by $\forall$, by definition of $V_{a,s}$ there are two possibilities. First, (b, t) is also an available choice for $\forall$ in π, and we end up in case (‡.2) as before. Otherwise, there is some $Q' \in \wp A$ such that b is in Q' and $\forall$ can choose (Q', t) in the shadow match π. By letting π advance at round z_{i+1} with such a move, we are able to maintain (‡.1) also in z_{i+1}.

— In the remaining case, inductively we are given the same basic position $(a, s) \in A \times T$ both in π and in π'. The valuation V suggested by f in π verifies $\Delta^F(a, \kappa(s)) = \Delta(a, \kappa(s))$, thus we can let the restriction of V to A be the valuation chosen by $\exists$ in the match π'. It is immediate that any next move of $\forall$ in π' can be mirrored by the same move in π, meaning that we are able to maintain the same position –whence the relation (‡.1)– also in the next round.

In both cases, the suggestion of strategy f' was a legitimate move for $\exists$ maintaining the relation (‡) between the two matches for any next round z_{i+1}. It follows that f' is a winning strategy for $\exists$ in $\mathcal{G}$.

□

2.2. From formulas to automata

In this subsection we conclude the proof of Theorem 2.2. We first focus on the case of projection with respect to finite sets, which exploits our simulation result, Theorem 2.10. The definition of the projection construction is formulated more generally for parity automata, as it will be later applied to classes other than $Aut_{wc}(\mathrm{FOE}_1^\infty)$. It clearly preserves the weakness and continuity conditions.

Definition 2.11. Let $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ be a parity automaton on alphabet $\wp(\mathsf{P} \cup \{p\})$. We define the automaton $\exists p.\mathbb{A} = \langle A, \Delta^\exists, \Omega, a_I \rangle$ on alphabet $\wp\mathsf{P}$ by putting

$$\Delta^\exists(a, c) := \Delta(a, c) \qquad \Delta^\exists(Q, c) := \Delta(Q, c) \vee \Delta(Q, c \cup \{p\}).$$

The automaton $\exists p.\mathbb{A}$ is called the *finitary projection construct of* $\mathbb{A}$ *over* p.

LEMMA 2.12. *Let $\mathbb{A}$ be a* WMSO-*automaton on alphabet* $\wp(\mathsf{P} \cup \{p\})$*. Then* $\mathbb{A}^F$ *is a* WMSO-*automaton on alphabet* $\wp\mathsf{P}$ *which satisfies*

$$\mathsf{TMod}(\exists p.\mathbb{A}^F) \equiv \exists_F p.\mathsf{TMod}(\mathbb{A}).$$

PROOF. Unraveling definitions, we need to show that for any tree $\mathbb{T} = \langle T, R, \kappa \colon \mathsf{P} \to \wp T, s_I \rangle$:

$$\exists p.\mathbb{A}^F \text{ accepts } \mathbb{T} \text{ iff there is a finite } p \text{ -variant } \mathbb{T}' \text{of } \mathbb{T} \text{ such that } \mathbb{A} \text{ accepts } \mathbb{T}'.$$

For the direction from left to right, by the equivalence between $\mathbb{A}$ and $\mathbb{A}^F$ it suffices to show that if $\exists p.\mathbb{A}^F$ accepts $\mathbb{T}$ then there is a finite p-variant $\mathbb{T}'$ of $\mathbb{T}$ such that $\mathbb{A}^F$ accepts $\mathbb{T}'$. First, we first observe that the properties stated by Theorem 2.10, which hold for $\mathbb{A}^F$ by assumption, by construction hold for $\exists p.\mathbb{A}^F$ as well. Thus we can assume that the given winning strategy f for $\exists$ in $\mathcal{G}_\exists = \mathcal{A}(\exists_F p.\mathbb{A}^F, \mathbb{T})@(a_I^F, s_I)$ is functional and finitary in $\wp A$. Functionality allows us to associate with each node s either none or a unique state $Q_s \in \wp A$ such that (Q_s, s) is winning for $\exists$. We now want to isolate the nodes that f treats "as if they were labeled with p". For this purpose, let V_s be the valuation suggested by f from a position $(Q_s, s) \in \wp A \times T$. As f is winning, V_s makes $\Delta^\exists(Q, \kappa(s))$ true in $R[s]$. We define a p-variant $\mathbb{T}' = \langle T, R, \kappa' \colon \mathsf{P} \cup \{p\} \to \wp T, s_I \rangle$ of $\mathbb{T}$ by defining $\kappa' := \kappa[p \mapsto X_p]$, that is, by colouring with p all nodes in the following set:

$$X_p := \{s \in T \mid (R[s], V_s) \models \Delta^F(Q_s, \kappa(s) \cup \{p\})\}. \tag{10}$$

The fact that f is finitary in $\wp A$ guarantees that X_p is finite, whence $\mathbb{T}'$ is a finite p-variant. It remains to show that $\mathbb{A}^F$ accepts $\mathbb{T}'$: we claim that f itself is winning for $\exists$ in $\mathcal{G} = (\mathbb{A}^F, \mathbb{T}')@(a_I, s_I)$. In order to see that, let us construct in stages an f-guided match π of $\mathcal{G}$ and an f-guided shadow match $\tilde{\pi}$ of $\mathcal{G}_\exists$. The inductive hypothesis we want to bring from one round to the next is that the same basic position occurs in both matches, as this suffices to prove that f is winning for $\exists$ in $\mathcal{G}$.

First we consider the case of a basic position $(Q, s) \in A^F \times T$ where $Q \in \wp A$. By assumption f provides a valuation V_s that makes $\Delta^\exists(Q, \kappa(s))$ true in $R[s]$. Thus V_s verifies either $\Delta^F(Q, \kappa(s))$ or $\Delta^F(Q, \kappa(s) \cup \{p\})$. Now, the match π^F is played on the p-variant $\mathbb{T}'$, where the labeling $\kappa'(s)$ is decided by the membership of s to X_p. According to (10), if V_s verifies $\Delta^F(Q, \kappa(s) \cup \{p\})$ then s is in X_p, meaning that it is labeled with p in $\mathbb{T}'$, i.e. $\kappa'(s) = \kappa(s) \cup \{p\}$. Therefore V_s also verifies $\Delta^F(Q, \kappa'(s))$ and it is a legitimate move for $\exists$ in match π^F. In the remaining case, V_s verifies $\Delta^F(Q, \kappa(s))$ but falsifies $\Delta^F(Q, \kappa(s) \cup \{p\})$, implying by definition that s is not in X_p. This means that s is not labeled with p in $\mathbb{T}'$, i.e. $\kappa'(s) = \kappa(s)$. Thus again V_s verifies $\Delta^F(Q, \kappa'(s))$ and it is a legitimate move for $\exists$ in match π^F.

It remains to consider the case of a basic position $(a, s) \in A^F \times T$ with $a \in A$ a state. By definition $\Delta^\exists(a, \kappa(s))$ is just $\Delta^F(a, \kappa(s))$. As (a, s) is winning, we can assume that no position (Q, s) with Q a macro-state is winning according to the same f, as making $\Delta^\exists$-sentences true never forces $\exists$ to mark a node both with a state and a macro-state. Therefore, s is not in X_p either, meaning that it it is not labeled with p in the p-variant $\mathbb{T}'$ and thus $\kappa'(s) = \kappa(s)$. This implies that f makes $\Delta^F(a, \kappa'(s)) = \Delta^F(a, \kappa(s))$ true in $R[s]$ and its suggestion is a legitimate move for $\exists$ in match π^F. In order to conclude the proof, observe that for all positions that we consider the same valuation is suggested to $\exists$ in both games: this means that any next position that is picked by player $\forall$ in π^F is also available for $\forall$ in the shadow match $\tilde{\pi}$.

We now show the direction from right to left of the statement. Let $\mathbb{T}'$ be a finite p-variant of $\mathbb{T}$, with labeling function κ', and g a winning strategy for $\exists$ in $\mathcal{G} = \mathcal{A}(\mathbb{A}, \mathbb{T}')@(a_I, s_I)$. Our goal is to define a strategy g' for $\exists$ in $\mathcal{G}_\exists$. As usual, g' will be constructed in stages, while playing a match π' in $\mathcal{G}_\exists$. In parallel to π', a *bundle* $\mathcal{B}$ of

g-guided shadow matches in $\mathcal{G}$ is maintained, with the following condition enforced for each round z_i:

(‡)

(1) If the current basic position in π' is of the form $(Q, s) \in \wp A \times T$, then for each $a \in Q$ there is an g-guided (partial) shadow match π_a at basic position $(a, s) \in A \times T$ in the current bundle $\mathcal{B}_i$. Also, either $\mathbb{T}'_s$ is not p-free (i.e., it does contain a node s' with $p \in \kappa'(s')$) or s has some sibling t such that $\mathbb{T}'_t$ is not p-free.
(2) Otherwise, the current basic position in π' is of the form $(a, s) \in A \times T$ and $\mathbb{T}'_s$ is p-free. Also, the bundle $\mathcal{B}_i$ only consists of a single g-guided match π_a whose current basic position is also (a, s).

We recall the idea behind (‡). Point (‡.1) describes the part of match π' where it is still possible to encounter nodes which are labeled with p in $\mathbb{T}'$. As $\Delta^{\exists}$ only takes the letter p into account when defined on macro-states in $\wp A$, we want π' to visit only positions of the form $(Q, s) \in \wp A \times T$ in that situation. Anytime we visit such a position (Q, s) in π', the role of the bundle is to provide one g-guided shadow match at position (a, s) for each $a \in Q$. Then g' is defined in terms of what g suggests from those positions.

Point (‡.2) describes how we want the match π' to be played on a p-free subtree: as any node that one might encounter has the same label in $\mathbb{T}$ and $\mathbb{T}'$, it is safe to let $\exists_F p.\mathbb{A}^F$ behave as $\mathbb{A}$ in such situation. Provided that the two matches visit the same basic positions, of the form $(a, s) \in A \times T$, we can let g' just copy g.

The key observation is that, as $\mathbb{T}'$ is a *finite* p-variant of $\mathbb{T}$, nodes labeled with p are reachable only for finitely many rounds of π'. This means that, provided that (‡) hold at each round, (‡.1) will describe an initial segment of π', whereas (‡.2) will describe the remaining part. Thus our proof that g' is a winning strategy for $\exists$ in $\mathcal{G}_{\exists}$ is concluded by showing that (‡) holds for each stage of construction of π' and $\mathcal{B}$.

For this purpose, we initialize π' from position $(a^{\sharp}_I, s) \in \wp A \times T$ and the bundle $\mathcal{B}$ as $\mathcal{B}_0 = \{\pi_{a_I}\}$, with π_{a_I} the partial g-guided match consisting only of the position $(a_I, s) \in A \times T$. The situation described by (‡.1) holds at the initial stage of the construction. Inductively, suppose that at round z_i we are given a position $(q, s) \in A^F \times T$ in π^F and a bundle $\mathcal{B}_i$ as in (‡). To show that (‡) can be maintained at round z_{i+1}, we distinguish two cases, corresponding respectively to situation (‡.1) and (‡.2) holding at round z_i.

(A) If (q, s) is of the form $(Q, s) \in \wp A \times T$, by inductive hypothesis we are given with g-guided shadow matches $\{\pi_a\}_{a \in Q}$ in $\mathcal{B}_i$. For each match π_a in the bundle, we are provided with a valuation $V_{a,s} : A \to \wp(R[s])$ making $\Delta(a, \kappa'(s))$ true. Then we further distinguish the following two cases.
   (i) Suppose first that $\mathbb{T}'_s$ is not p-free. We let the suggestion $V' \colon A^F \to \wp(R[s])$ of g' from position (Q, s) be defined as follows:

$$V'(q') := \begin{cases} \bigcap\limits_{\substack{(a,b) \in q', \\ a \in Q}} \{t \in R[s] \mid t \in V_{a,s}(b)\} & q' \in \wp A \\ \bigcup\limits_{a \in Q} \{t \in R[s] \mid t \in V_{a,s}(q') \text{ and } \mathbb{T}'.t \text{ is } p\text{-free}\} & q' \in A. \end{cases}$$

   The definition of V' on $q' \in \wp A$ is standard (*cf.* [Zanasi 2012, Prop. 2.21]) and guarantees a correspondence between the states assigned by the valuations $\{V_{a,s}\}_{a \in Q}$ and the macro-states assigned by V'. The definition of V' on $q' \in A$ aims at fulfilling the conditions, expressed via $\exists^{\infty}$ and $\forall^{\infty}$, on the number of nodes in $R[s]$ witnessing (or not) some A-types. Those conditions are the ones

that $\Delta^\sharp(Q,\kappa'(s))$ –and thus also $\Delta^F(Q,\kappa'(s))$– "inherits" by $\bigwedge_{a\in R}\Delta(a,\kappa'(s))$, by definition of $\Delta^\sharp$. Notice that we restrict $V'(q')$ to the nodes $t \in V_{a,s}(q')$ such that $\mathbb{T}'.t$ is p-free. As $\mathbb{T}'$ is a *finite* p-variant, only *finitely many* nodes in $V_{a,s}(q')$ will not have this property. Therefore their exclusion, which is crucial for maintaining condition (‡) (*cf.* case (a) below), does not influence the fulfilling of the cardinality conditions expressed via $\exists^\infty$ and $\forall^\infty$ in $\Delta^\sharp(Q,\kappa'(s))$.
On the base of these observations, one can check that V' makes $\Delta^\sharp(Q,\kappa'(s))$–and thus also $\Delta^F(Q,\kappa'(s))$–true in $R[s]$. In fact, to be a legitimate move for $\exists$ in π', V' should make $\Delta^\exists(Q,\kappa(s))$ true: this is the case, for $\Delta^F(Q,\kappa'(s))$ is either equal to $\Delta^F(Q,\kappa(s))$, if $p \notin \kappa'(s)$, or to $\Delta^F(Q,\kappa(s)\cup\{p\})$ otherwise. In order to check that we can maintain (‡), let $(q',t) \in A^F \times T$ be any next position picked by $\forall$ in π' at round z_{i+1}. As before, we distinguish two cases:

(a) If q' is in A, then, by definition of V', $\forall$ can choose (q',t) in some shadow match π_a in the bundle $\mathcal{B}_i$. We dismiss the bundle –i.e. make it a singleton– and bring only π_a to the next round in the same position (q',t). Observe that, by definition of V', $\mathbb{T}'.t$ is p-free and thus (‡.2) holds at round z_{i+1}.

(b) Otherwise, q' is in $\wp A$. The new bundle $\mathcal{B}_{i+1}$ is given in terms of the bundle $\mathcal{B}_i$: for each $\pi_a \in \mathcal{B}_i$ with $a \in Q$, we look if for some $b \in q'$ the position (b,t) is a legitimate move for $\forall$ at round z_{i+1}; if so, then we bring π_a to round z_{i+1} at position (b,t) and put the resulting (partial) shadow match π_b in $\mathcal{B}_{i+1}$. Observe that, if $\forall$ is able to pick such position (q',t) in π', then by definition of V' the new bundle $\mathcal{B}_{i+1}$ is non-empty and consists of an g-guided (partial) shadow match π_b for each $b \in q'$. In this way we are able to keep condition (‡.1) at round z_{i+1}.

(ii) Let us now consider the case in which $\mathbb{T}'_s$ is p-free. We let g' suggest the valuation V' that assigns to each node $t \in R[s]$ all states in $\bigcup_{a\in Q}\{b \in A \mid t \in V_{a,s}(b)\}$. It can be checked that V' makes $\bigwedge_{a\in Q}\Delta(a,\kappa'(s))$ – and then also $\Delta^F(Q,\kappa'(s))$ – true in $R[s]$. As $p \notin \kappa(s) = \kappa'(s)$, it follows that V' also makes $\Delta^\exists(Q,\kappa(s))$ true, whence it is a legitimate choice for $\exists$ in π'. Any next basic position picked by $\forall$ in π' is of the form $(b,t) \in A \times T$, and thus condition (‡.2) holds at round z_{i+1} as shown in (i.a).

(B) In the remaining case, (q,s) is of the form $(a,s) \in A \times T$ and by inductive hypothesis we are given with a bundle $\mathcal{B}_i$ consisting of a single f-guided (partial) shadow match π_a at the same position (a,s). Let $V_{a,s}$ be the suggestion of $\exists$ from position (a,s) in π_a. Since by assumption s is p-free, we have that $\kappa'(s) = \kappa(s)$, meaning that $\Delta^\exists(a,\kappa(s))$ is just $\Delta(a,\kappa(s)) = \Delta(a,\kappa'(s))$. Thus the restriction V' of V to A makes $\Delta(a,\kappa'(t))$ true and we let it be the choice for $\exists$ in $\tilde{\pi}$. It follows that any next move made by $\forall$ in $\tilde{\pi}$ can be mirrored by $\forall$ in the shadow match π_a.

□

2.2.1. Closure under Boolean operations. Here we show that the collection of $Aut(\mathrm{WMSO})$-recognizable classes of tree models is closed under the Boolean operations. For union, we use the following result, leaving the straightforward proof as an exercise to the reader.

LEMMA 2.13. *Let* $\mathbb{A}_0$ *and* $\mathbb{A}_1$ *be* WMSO*-automata. Then there is a* WMSO*-automaton* $\mathbb{A}$ *such that* $\mathsf{TMod}(\mathbb{A})$ *is the union of* $\mathsf{TMod}(\mathbb{A}_0)$ *and* $\mathsf{TMod}(\mathbb{A}_1)$.

For closure under complementation we reuse the general results established in Section 1 for parity automata.

LEMMA 2.14. *Let $\mathbb{A}$ be an* WMSO*-automaton. Then the automaton $\overline{\mathbb{A}}$ defined in Definition 1.17 is a* WMSO*-automaton recognizing the complement of* $\mathsf{TMod}(\mathbb{A})$.

PROOF. It suffices to check that Proposition 1.18 restricts to the class $Aut_{wc}(\mathrm{FOE}_1^\infty)$ of WMSO-automata. First, the fact that FOE_1^∞ is closed under Boolean duals (Definition 1.6) implies that it holds for the class $Aut(\mathrm{FOE}_1^\infty)$. It then remains to check that the dual automata construction $\overline{(\cdot)}$ preserves weakness and continuity. But this is straightforward, given the self-dual nature of these properties. □

We are now finally able to conclude the direction from formulas to automata of the characterisation theorem.

PROOF OF THEOREM 2.2. The proof is by induction on φ.

— For the base case, we consider the atomic formulas $\Downarrow p$, $p \sqsubseteq q$ and $R(p,q)$.
  — The WMSO-automaton $\mathbb{A}_{\Downarrow p} = \langle A, \Delta, \Omega, a_I \rangle$ is given by putting

$$A := \{a_0, a_1\} \qquad a_I := a_0 \qquad \Omega(a_0) := 0 \qquad \Omega(a_1) := 0$$

$$\Delta(a_0, c) := \begin{cases} \forall x.a_1(x) & \text{if } p \in c \\ \bot & \text{otherwise.} \end{cases} \qquad \Delta(a_1, c) := \begin{cases} \forall x.a_1(x) & \text{if } p \notin c \\ \bot & \text{otherwise.} \end{cases}$$

  — The WMSO-automaton $\mathbb{A}_{p \sqsubseteq q} = \langle A, \Delta, \Omega, a_I \rangle$ is given by $A := \{a\}$, $a_I := a$, $\Omega(a) := 0$ and $\Delta(a,c) := \forall x\, a(x)$ if $p \notin c$ or $q \in c$, and $\Delta(a,c) := \bot$ otherwise.
  — The WMSO-automaton $\mathbb{A}_{R(p,q)} = \langle A, \Delta, \Omega, a_I \rangle$ is given below:

$$A := \{a_0, a_1\} \qquad a_I := a_0 \qquad \Omega(a_0) := 0 \qquad \Omega(a_1) := 1$$

$$\Delta(a_0, c) := \begin{cases} \exists x.a_1(x) \wedge \forall y.a_0(y) & \text{if } p \in c \\ \forall x\ (a_0(x)) & \text{otherwise.} \end{cases} \qquad \Delta(a_1, c) := \begin{cases} \top & \text{if } q \in c \\ \bot & \text{otherwise} \end{cases}$$

— For the Boolean cases, where $\varphi = \psi_1 \vee \psi_2$ or $\varphi = \neg\psi$ we refer to the Boolean closure properties that we just established in the Lemmas 2.13 and 2.14, respectively.
— The case $\varphi = \exists p.\psi$ follows by the following chain of equivalences, where $\mathbb{A}_\psi$ is given by the inductive hypothesis and $\exists_F p.\mathbb{A}_\psi$ is constructed according to Definition 2.11:

$$\begin{aligned} \exists_F p.\mathbb{A}_\psi \text{ accepts } \mathbb{T} & \text{ iff } \mathbb{A}_\psi \text{ accepts } \mathbb{T}[p \mapsto X], \text{ for some } X \subseteq_\omega T && \text{(Lemma 2.12)} \\ & \text{ iff } \mathbb{T}[p \mapsto X] \models \psi, \text{ for some } X \subseteq_\omega T && \text{(induction hyp.)} \\ & \text{ iff } \mathbb{T} \models \exists p.\psi && \text{(semantics WMSO)} \end{aligned}$$

□

3. AUTOMATA FOR NMSO

In this section we introduce the automata that capture NMSO.

Definition 3.1. A NMSO-*automaton* is a weak automaton for the one-step language FOE_1.

Aanalogous to the previous section, our main goal here is to construct an equivalent NMSO-automaton for every NMSO-formula.

THEOREM 3.2. *There is an effective construction transforming a* NMSO-*formula* φ *into a* NMSO-*automaton* $\mathbb{A}_\varphi$ *that is equivalent to* φ *on the class of trees.*

The proof for Theorem 3.2 will closely follow the steps for proving the analogous result for WMSO (Theorem 2.2). Again, the crux of the matter is to show that the collection of classes of tree models that are recognisable by some NMSO-automaton, is closed under the relevant notion of projection. Where this was finitary projection for WMSO (Def. 2.3), the notion mimicking NMSO-quantification is *noetherian* projection.

Definition 3.3. Given a set P of proposition letters, $p \notin P$ and a class C of $\mathsf{P} \cup \{p\}$-labeled trees, we define the *noetherian projection* of C over p as the language of P)-labeled trees given as

$$\exists_N p.\mathsf{C} := \{\mathbb{T} \mid \text{ there is a noetherian } p\text{-variant } \mathbb{T}' \text{ of } \mathbb{T} \text{ with } \mathbb{T}' \in \mathsf{C}\}.$$

A collection of classes of tree modelss is *closed under noetherian projection over* p if it contains the class $\exists_N p.\mathsf{C}$ whenever it contains the class C itself.

3.1. Simulation theorem for NMSO-automata

Just as for WMSO-automata, also for NMSO-automata the projection construction will rely on a simulation theorem, constructing a two-sorted automaton $\mathbb{A}^N$ consisting of a copy of the original automaton, based on states A, and a variation of its powerset construction, based on macro-states $\wp A$. For any accepted $\mathbb{T}$, we want any match π of $\mathcal{A}(\mathbb{A}^N, \mathbb{T})$ to split in two parts:

(Non-deterministic mode). for finitely many rounds π is played on macro-states, i.e. positions are of the form $\wp A \times T$. The strategy of player $\exists$ is functional in $\wp A$, i.e. it assigns *at most one macro-state* to each node.

(Alternating mode). At a certain round, π abandons macro-states and turns into a match of the game $\mathcal{A}(\mathbb{A}, \mathbb{T})$, i.e. all next positions are from $A \times T$ (and are played according to a non-necessarily functional strategy).

The only difference with the two-sorted construction for WMSO-automata is that, in the non-deterministic mode, the cardinality of nodes to which $\exists$'s strategy assigns macro-states is irrelevant. Indeed, NMSO's finiteness is only on the vertical dimension: assigning an odd priority to macro-states will suffice to guarantee that the non-deterministic mode processes just a well-founded portion of any accepted tree.

We now proceed in steps towards the construction of $\mathbb{A}^N$. First, the following lifting from states to macro-states parallels Definition 2.5, but for the one-step language FOE_1 proper of NMSO-automata. It is based on the basic form for FOE_1-formulas, see Definition 1.10.

Definition 3.4. Let $\varphi \in {\mathrm{FOE}_1}^+(A)$ be of shape $\nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi)$ for some $\Pi \subseteq \wp A$ and $\overline{\mathbf{T}} = \{T_1, \ldots, T_k\} \subseteq \wp A$. We define φ^N as $\nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}^{\Uparrow}, \Pi^{\Uparrow}) \in {\mathrm{FOE}_1}^+(\wp A)$, that means,

$$\varphi^N \;:=\; \exists \overline{\mathbf{x}}.\big(\mathrm{diff}(\overline{\mathbf{x}}) \wedge \bigwedge_{0 \leq i \leq n} \tau^+_{T_i^{\Uparrow}}(x_i) \wedge \forall z.(\mathrm{diff}(\overline{\mathbf{x}}, z) \to \bigvee_{S \in \Pi^{\Uparrow}} \tau^+_S(z))\big) \tag{11}$$

It is instructive to compare (11) with its WMSO-counterpart (9): the difference is that, because the quantifiers $\exists^\infty$ and $\forall^\infty$ are missing, the sentence does not impose any cardinality requirement, but only enforces $\wp A$-separability — *cf.* Section 1.1.

LEMMA 3.5. *Let $\varphi \in \mathrm{FOE_1}^+(A)$ and $\varphi^N \in \mathrm{FOE_1}^+(\wp A)$ be as in Definition 3.4. Then φ^N is separating in $\wp A$.*

PROOF. Each element of $\overline{\mathbf{T}}^{\Uparrow}$ and $\Pi^{\Uparrow}$ is by definition either the empty set or a singleton $\{Q\}$ for some $Q \in \wp A$. Then the statement follows from Proposition 1.15 . $\square$

We are now ready to define the transition function for macro-states. The following adapts Definition 2.6 to the one-step language FOE_1 of NMSO-automata, and its normal form result, Theorem 1.11.

Definition 3.6. Let $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ be a NMSO-automaton. Fix any $c \in C$ and $Q \in \wp A$. By Theorem 1.11 there is a sentence $\Psi_{Q,c} \in \mathrm{FOE_1}^+(A)$ in the basic form $\bigvee \nabla_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi)$, for some $\Pi \subseteq \wp A$ and $T_i \subseteq A$, such that $\bigwedge_{a \in Q} \Delta(a, c) \equiv \Psi_{Q,c}$. By definition, $\Psi_{Q,c} = \bigvee_n \varphi_n$, with each φ_n of shape $\nabla_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Pi)$. We put $\Delta^\flat(Q, c) := \bigvee_n \varphi_n^N \in \mathrm{FOE_1}^+(\wp A)$, where the translation $(\cdot)^N$ is as in Definition 3.4.

We now have all the ingredients for the two-sorted construction over NMSO-automata.

Definition 3.7. Let $\mathbb{A} = \langle A, \Delta, \Omega, a_I \rangle$ be a NMSO-automaton. We define the *noetherian construct over* $\mathbb{A}$ as the automaton $\mathbb{A}^N = \langle A^N, \Delta^N, \Omega^N, a_I^N \rangle$ given by

$$\begin{array}{rclrclrcl} A^N & := & A \cup \wp A & \Omega^N(a) & := & \Omega(a) & \Delta^N(a,c) & := & \Delta(a,c) \\ a_I^N & := & \{a_I\} & \Omega^N(R) & := & 1 & \Delta^N(Q,c) & := & \Delta^\flat(Q,c) \vee \bigwedge_{a \in Q} \Delta(a,c). \end{array}$$

The construction is the same as the one for WMSO-automata (Definition 2.7) but for the definition of the transition function for macro-states, which is now free of any cardinality requirement.

Definition 3.8. We say that a strategy f in an acceptance game $\mathcal{A}(\mathbb{A}, \mathbb{T})$ is *noetherian* in $B \subseteq A$ when in any f-guided match there can be only finitely many rounds played at a position of shape (q, s) with $q \in B$.

THEOREM 3.9 (SIMULATION THEOREM FOR NMSO-AUTOMATA). *Let $\mathbb{A}$ be an NMSO-automaton and $\mathbb{A}^N$ its noetherian construct.*

(1) $\mathbb{A}^N$ is an NMSO-automaton.
(2) For any $\mathbb{T}$, if $\exists$ has a winning strategy in $\mathcal{A}(\mathbb{A}^N, \mathbb{T})$ from position (a_I^N, s_I) then she has one that is functional in $\wp A$ and noetherian in $\wp A$.
(3) $\mathbb{A} \equiv \mathbb{A}^N$.

PROOF. The proof follows the same steps as the one of Proposition 2.10, minus all the concerns about continuity of the constructed automaton and any associated winning strategy f being finitary. One still has to show that f is noetherian in $\wp A$ ("vertically finitary"), but this is enforced by macro-states having an odd parity: visiting one of them infinitely often would mean $\exists$'s loss. $\square$

Remark 3.10. As mentioned, the class $Aut(\mathrm{FOE}_1)$ of automata characterising SMSO [Janin and Walukiewicz 1996] also enjoys a simulation theorem [Walukiewicz 1996], turning any automaton into an equivalent non-deterministic one. Given that the class $Aut_w(\mathrm{FOE}_1)$ only differs for the weakness constraint, one may wonder if the simulation result for $Aut(\mathrm{FOE}_1)$ could not actually be restricted to $Aut_w(\mathrm{FOE}_1)$, making our two-sorted construction redundant. This is actually not the case: not only does Walukiewicz's simulation theorem [Walukiewicz 1996] fail to preserve the weakness constraint, but even without this failure our purposes would not be served: A fully non-deterministic automaton is instrumental in guessing a p-variant of any accepted tree, but it does not guarantee that the p-variant is also noetherian, as the two-sorted construct does.

3.2. From formulas to automata

We can now conclude one direction of the automata characterisation of NMSO.

LEMMA 3.11. *For each* NMSO-*automaton* $\mathbb{A}$ *on alphabet* $\wp(\mathsf{P} \cup \{p\})$, *let* $\mathbb{A}^N$ *be its noetherian construct. We have that*

$$\mathsf{TMod}(\exists p.\mathbb{A}^N) \equiv \exists_N p.\mathsf{TMod}(\mathbb{A}).$$

PROOF. The argument is the same as for WMSO-automata (Lemma 2.12). As in that proof, the inclusion from left to right relies on the simulation result (Theorem 3.9): $\exists p.\mathbb{A}^N$ is two-sorted and its non-deterministic mode can be used to guess a noetherian p-variant of any accepted tree. □

PROOF OF THEOREM 3.2. As for its WMSO-counterpart Theorem 2.2, the proof is by induction on $\varphi \in \mathrm{NMSO}$. The boolean inductive cases are handled by the NMSO-versions of Lemma 2.13 and 2.14. The projection case follows from Lemma 3.11. □

4. FIXPOINT OPERATORS AND SECOND-ORDER QUANTIFIERS

In this section we will show how to translate some of the mu-calculi that we encountered until now into the appropriate second-order logics. Given the equivalence between automata and fixpoint logics that we established in Section 1, and the embeddings of WMSO and NMSO into, respectively, the automata classes $Aut_{wc}(\mathrm{FOE}_1^\infty)$ and $Aut_w(\mathrm{FOE}_1)$ that we provided in the Sections 2 and 3 for the class of tree models, the results here provide the missing link in the automata-theoretic characterizations of the monadic second order logics WMSO and NMSO:

$$\mu_C(\mathrm{FOE}_1^\infty) \equiv \mathrm{WMSO} \qquad \text{(over the class of all tree models)}$$
$$\mu_D(\mathrm{FOE}_1) \equiv \mathrm{NMSO} \qquad \text{(over the class of all tree models).}$$

4.1. Translating μ-calculi into second-order logics

More specifically, our aim in this Section is to prove the following result.

THEOREM 4.1.

(1) There is an effective translation $(\cdot)^* : \mu_D\mathrm{FOE}_1 \to \mathrm{NMSO}$ *such that* $\varphi \equiv \varphi^*$ *for every* $\varphi \in \mu_D\mathrm{FOE}_1$*; that is:*

$$\mu_D\mathrm{FOE}_1 \leq \mathrm{NMSO}.$$

(2) There is an effective translation $(\cdot)^* : \mu_C\mathrm{FOE}_1^\infty \to \mathrm{WMSO}$ *such that* $\varphi \equiv \varphi^*$ *for every* $\varphi \in \mu_C\mathrm{FOE}_1^\infty$*; that is:*

$$\mu_C\mathrm{FOE}_1^\infty \leq \mathrm{WMSO}.$$

Two immediate observations on this Theorem are in order. First, note that we use the same notation $(\cdot)^*$ for both translations; this should not cause any confusion since the maps agree on formulas belonging to their common domain. Consequently, in the remainder we will speak of a single translation $(\cdot)^*$. Second, as the target language of the translation $(\cdot)^*$ we will take the *two-sorted* version of second-order logic, as discussed in section 3.1, and thus we will need Fact **??** to obtain the result as formulated in Theorem 4.1, that is, for the one-sorted versions of MSO. We reserve a fixed individual variable v for this target language, i.e., every formula of the form φ^* will have this v as its unique free variable; the equivalence $\varphi \equiv \varphi^*$ is to be understood accordingly.

The translation $(\cdot)^*$ will be defined by a straightforward induction on the complexity of fixpoint formulas. The two clauses of this definition that deserve some special attention are the ones related to the fixpoint operators and the modalities.

Fixpoint operators. It is important to realise that our clause for the fixpoint operators *differs* from the one used in the standard inductive translation $(\cdot)^s$ of μML into standard MSO, where we would inductively translate $(\mu p.\varphi)^*$ as

$$\forall p \left(\forall w \left(\varphi^*[w/v] \to p(w)\right) \to p(v)\right), \tag{12}$$

which states that v belongs to any prefixpoint of φ with respect to p. To understand the problem with this translation in the current context, suppose, for instance, that we want to translate some continuous μ-calculus into WMSO. Then the formula in (12) expresses that v belongs to the intersection of all *finite* prefixpoints of φ, whereas the least fixpoint is identical to the intersection of *all* prefixpoints. As a result, (12) does not give the right translation for the formula $\mu p.\varphi$ into WMSO.

To overcome this problem, we will prove that least fixpoints in restricted calculi like $\mu_D\mathrm{FOE}_1$, $\mu_C\mathrm{FOE}_1^\infty$ and many others, in fact satisfy a rather special property, which enables an alternative translation. We need the following definition to formulate this property.

Definition 4.2. Let $F : \wp(S) \to \wp(S)$ be a functional; for a given $X \subseteq S$ we define the *restricted* map $F{\restriction_X} : \wp(S) \to \wp(S)$ by putting $F{\restriction_X}(Y) := FY \cap X$.

The observations formulated in the proposition below provide the crucial insight underlying our embedding of various alternation-free and continuous μ-calculi into, respectively, NMSO and WMSO.

PROPOSITION 4.3. *Let $\mathbb{S}$ be an LTS, and let r be a point in $\mathbb{S}$.*

(1) For any formula φ with $\mu p.\varphi \in \mu_D\mathrm{FOE}_1$ we have

$$r \in [\![\mu p.\varphi]\!]^{\mathbb{S}} \textit{ iff there is a noetherian set } X \textit{ such that } r \in LFP.(\varphi^{\mathbb{S}}_p){\restriction_X}. \tag{13}$$

(2) For any formula φ with $\mu p.\varphi \in \mu_C\mathrm{FOE}_1^{\infty}$ we have

$$r \in [\![\mu p.\varphi]\!]^{\mathbb{S}} \textit{ iff there is a finite set } X \textit{ such that } r \in LFP.(\varphi^{\mathbb{S}}_p){\restriction_X}. \tag{14}$$

Remark 4.4. In fact, the statements in Proposition 4.3 can be generalised to the setting of a fixpoint logic μL_1 associated with an arbitrary one-step language L_1.

The right-to-left direction of both (13) and (14) follow from the following, more general, statement, which can be proved by a routine transfinite induction argument.

PROPOSITION 4.5. *Let $F : \wp(S) \to \wp(S)$ be monotone. Then for every subset $X \subseteq S$ it holds that $LFP.F{\restriction_X} \subseteq LFP.F$.*

The left-to-right direction of (13) and (14) will be proved in the next two sections. Note that in the continuous case we will in fact prove a slightly stronger result, which applies to *arbitrary* continuous functionals.

The point of Proposition 4.3 is that it naturally suggests the following translation for the least fixpoint operator, as a subtle but important variation of (12):

$$(\mu p.\varphi)^* := \exists q \Big(\forall p \subseteq q.\ \big(p \in PRE((\varphi^{\mathbb{S}}_p){\restriction_q}) \to p(v)\big)\Big), \tag{15}$$

where $p \in PRE((\varphi^{\mathbb{S}}_p){\restriction_q})$ expresses that $p \subseteq q$ is a prefixpoint of the map $(\varphi^{\mathbb{S}}_p){\restriction_q}$, that is:

$$p \in PRE((\varphi^{\mathbb{S}}_p){\restriction_q}) := \forall w \Big((q(w) \wedge \varphi^*[w/v]) \to p(w)\Big).$$

Modalities. Finally, before we can give the definition of the translation $(\cdot)^*$, we briefly discuss the clause involving the modalities. Here we need to understand the role of the *one-step formulas* in the translation. First an auxiliary definition.

Definition 4.6. Let $\mathbb{S} = \langle T, R, \kappa, s_I\rangle$ be a P-LTS, A be a set of new variables, and $V : A \to \wp(X)$ be a valuation on a subset $X \subseteq T$. The $\mathsf{P} \cup A$-LTS $\mathbb{S}^V := \langle T, R, \kappa^V, s_I\rangle$ given by defining the marking $\kappa^V : T \to \wp(\mathsf{P} \cup A)$ where

$$\kappa^V(s) := \begin{cases} \kappa(s) & \text{if } s \notin X \\ \kappa(s) \cup \{a \in A \mid s \in V(a)\} & \text{else,} \end{cases}$$

is called the V-expansion of $\mathbb{S}$.

The following proposition states that at the one-step level, the formulas that provide the semantics of the modalities of $\mu\mathrm{FOE}_1$ and $\mu\mathrm{FOE}_1^{\infty}$ can indeed be translated into, respectively NMSO and WMSO.

PROPOSITION 4.7. *There is a translation $(\cdot)^\dagger : \mathrm{FOE}_1^{\infty}(A) \to \mathrm{WMSO}$ such that for every model $\mathbb{S}$ and every valuation $V : A \to \wp(R[s_I])$:*

$$(R[s_I], V) \models \alpha \textit{ iff } \mathbb{S}^V \models \alpha^\dagger[s_I].$$

Moreover, $(\cdot)^\dagger$ *restricts to first-order logic, i.e.,* $\alpha^\dagger$ *is a first-order formula if* $\alpha \in \mathrm{FOE}_1$.

PROOF. Basically, the translation $(\cdot)^\dagger$ *restricts* all quantifiers to the collection of successors of v. In other words, $(\cdot)^\dagger$ is the identity on basic formulas, it commutes with the propositional connectives, and for the quantifiers $\exists$ and $\exists^\infty$ we define:

$$\begin{array}{lll} (\exists x\, \alpha)^\dagger & := & \exists x\, (Rvx \wedge \alpha^\dagger) \\ (\exists^\infty x\, \alpha)^\dagger & := & \forall p \exists x\, (Rvx \wedge \neg p(x) \wedge \alpha^\dagger) \end{array}$$

We leave it for the reader to verify the correctness of this definition — observe that the clause for the infinity quantifier $\exists^\infty$ is based on the equivalence between WMSO and FOE^∞, established by Väänänen [Väänänen 1977]. □

We are now ready to define the translation used in the main result of this section.

Definition 4.8. By an induction on the complexity of formulas we define the following translation $(\cdot)^*$ from $\mu\mathrm{FOE}^\infty$-formulas to formulas of monadic second-order logic:

$$\begin{array}{lll} p^* & := & p(v) \\ (\neg\varphi)^* & := & \neg\varphi^* \\ (\varphi \vee \psi)^* & := & \varphi^* \vee \psi^* \\ (\bigcirc_\alpha(\overline{\varphi}))^* & := & \alpha^\dagger[\varphi_i^*/a_i \mid i \in I], \end{array}$$

where $\alpha^\dagger$ is as in Proposition 4.7, and $[\varphi_i^*/a_i \mid i \in I]$ is the substitution that replaces every occurrence of an atomic formula of the form $a_i(x)$ with the formula $\varphi_i^*(x)$ (i.e. the formula φ_i^*, but with the free variable v substituted by x).

Finally, the inductive clause for a formula of the form $\mu p.\varphi$ is given as in (15).

PROOF OF THEOREM 4.1. First of all, it is clear that in both cases the translation $(\cdot)^*$ lands in the correct language. For both parts of the theorem, we thence prove that $(\cdot)^*$ is truth preserving by a straightforward formula induction. E.g., for part (2) we need to show that, for an arbitrary formula $\varphi \in \mu_C\mathrm{FOE}_1^\infty$ and an arbitrary model $\mathbb{S}$:

$$\mathbb{S} \Vdash \varphi \text{ iff } \mathbb{S} \models \varphi^*[s_I]. \tag{16}$$

As discussed in the main text, the two critical cases concern the inductive steps for the modalities and the least fixpoint operators. Let $L_1^+ \in \{\mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$. We start verifying the case of modalities. Hence, consider the formula $\bigcirc_\alpha(\varphi_1, \ldots, \varphi_n)$ with $\alpha(a_1, \ldots, a_n) \in L_1^+$. By induction hypothesis, $\varphi_\ell \equiv \varphi_\ell^*$, for $\ell = 1, \ldots, n$. Now, let $\mathbb{S}$ be a transition system. We have that

$$\begin{array}{lll} \mathbb{S} \Vdash \bigcirc_\alpha(\varphi_1, \ldots, \varphi_n) & \text{iff } (R[s_I], V_{\overline{\varphi}}) \models \alpha(a_1, \ldots, a_n) & (\text{ by (3)}) \\ & \text{iff } \mathbb{S}^{V_{\overline{\varphi}}} \models \alpha^\dagger[s_I] & (\text{ by Prop. 4.7}) \\ & \text{iff } \mathbb{S} \models \alpha^\dagger[\varphi_i^*/a_i \mid i \in I][s_I] & (\text{ by (4), Def. 4.6 and IH}) \end{array}$$

The inductive step for the least fixpoint operator will be justified by Proposition 4.3. In more detail, given a formula of the form $\mu x.\psi \in \mu_Y L_1^+$, with $Y = D$ for $L_1^+ = \mathrm{FOE}_1$, and $Y = C$ for $L_1^+ = \mathrm{FOE}_1^\infty$, consider the following chain of equivalences:

$$
\begin{aligned}
& s_I \in [\![\mu p.\psi]\!]^{\mathbb{S}} \\
\text{iff } & s_I \in LFP.(\psi_p^{\mathbb{S}})\restriction_Q \text{ for some } \begin{cases}\text{finite}\\ \text{noetherian}\end{cases} \text{set } Q && (\text{by (13)/(14)}) \\
\text{iff } & s_I \in \bigcap \Big\{ P \subseteq Q \mid P \in PRE((\psi_p^{\mathbb{S}})\restriction_Q) \Big\} \text{ for some } \begin{cases}\text{finite}\\ \text{noetherian}\end{cases} \text{set } Q \\
\text{iff } & \mathbb{S} \models \exists q. \Big(\forall p \subseteq q. \big(p \in PRE((\psi_p^{\mathbb{S}})\restriction_q) \to p(s_I) \big) \Big) \\
\text{iff } & \mathbb{S} \models (\mu p.\psi)^*[s_I]. && (\text{IH})
\end{aligned}
$$

This concludes the proof of (16). □

4.2. Fixpoints of continuous maps

It is well-known that continuous functionals are *constructive*. That is, if we construct the least fixpoint of a continuous functional $F : \wp(S) \to \wp(S)$ using the ordinal approximation $\varnothing, F\varnothing, F^2\varnothing, \ldots, F^\alpha\varnothing, \ldots$, then we reach convergence after at most ω many steps, implying that $LFP.F = F^\omega\varnothing$. We will see now that this fact can be strengthened to the following observation, which is the crucial result needed in the proof of Proposition 4.3.

THEOREM 4.9. *Let $F : \wp(S) \to \wp(S)$ be a continuous functional. Then for any $s \in S$:*

$$s \in LFP.F \textit{ iff } s \in LFP.F\restriction_X, \textit{ for some finite } X \subseteq S. \tag{17}$$

PROOF. The direction from right to left of (17) is a special case of Proposition 4.5. For the opposite direction of (17) a bit more work is needed. Assume that $s \in LFP.F$; we claim that there are sets $U_1, \ldots, U_n$, for some $n \in \omega$, such that $s \in U_n$, $U_1 \subseteq_\omega F(\varnothing)$, and $U_{i+1} \subseteq_\omega F(U_i)$, for all i with $1 \leq i < n$.

To see this, first observe that since F is continuous, we have $LFP.F = F^\omega(\varnothing) = \bigcup_{n\in\omega} F^n(\varnothing)$, and so we may take n to be the least natural number such that $s \in F^n(\varnothing)$. By a downward induction we now define sets $U_n, \ldots, U_1$, with $U_i \subseteq F^i(\varnothing)$ for each i. We set up the induction by putting $U_n := \{s\}$, then $U_n \subseteq F^n(\varnothing)$ by our assumption on n. For $i < n$, we define U_i as follows. Using the inductive fact that $U_{i+1} \subseteq_\omega F^{i+1}(\varnothing) = F(F^i(\varnothing))$, it follows by continuity of F that for each $u \in U_{i+1}$ there is a set $V_u \subseteq_\omega F^i(\varnothing)$ such that $u \in F(V_u)$. We then define $U_i := \bigcup\{V_u \mid u \in U_{i+1}\}$, so that clearly $U_{i+1} \subseteq_\omega F(U_i)$ and $U_i \subseteq_\omega F^i(\varnothing)$. Continuing like this, ultimately we arrive at stage $i = 1$ where we find $U_1 \subseteq F(\varnothing)$ as required.

Finally, given the sequence $U_n, \ldots, U_1$, we define

$$X := \bigcup_{0<i\leq n} U_i.$$

It is then straightforward to prove that $U_i \subseteq LFP.F\restriction_X$, for each i with $0 < i \leq n$, and so in particular we find that $s \in U_n \subseteq LFP.F\restriction_X$. This finishes the proof of the implication from left to right in (17). □

As an almost immediate corollary of this result we obtain the second part of Proposition 4.3.

PROOF OF PROPOSITION 4.3(2). Take an arbitrary formula $\mu p.\varphi \in \mu_C\mathrm{FOE}_1^\infty$, then by definition we have $\varphi \in \mu_C\mathrm{FOE}_1^\infty \cap \mathtt{Con}_p(\mu\mathrm{FOE}_1^\infty)$. But it follows from a routine inductive proof that every formula $\psi \in \mu_C\mathrm{FOE}_1^\infty \cap \mathtt{Con}_{\mathsf{Q}}(\mu\mathrm{FOE}_1^\infty)$ is continuous in each variable in Q. Thus φ is continuous in p, and so the result is immediate by Theorem 4.9. □

4.3. Fixpoints of noetherian maps

We will now see how to prove Proposition 4.3(1), which is the key result that we need to embed alternation-free μ-calculi such as $\mu_D\mathrm{FOE}_1$ and $\mu_D\mathrm{ML}$ into noetherian second-order logic. Perhaps suprisingly, this case is slightly more subtle than the characterisation of fixpoints of continuous maps.

We start with stating some auxiliary definitions and results on monotone functionals, starting with a game-theoretic characterisation of their least fixpoints [Venema 2012].

Definition 4.10. Given a monotone functional $F : \wp(S) \to \wp(S)$ we define the *unfolding game* $\mathcal{U}_F$ as follows:

— at any position $s \in S$, $\exists$ needs to pick a set X such that $s \in FX$;
— at any position $X \in \wp(S)$, $\forall$ needs to pick an element of X
— all infinite matches are won by $\forall$.

A positional strategy $f : S \to \wp(S)$ for $\exists$ in $\mathcal{U}_F$ is *descending* if, for all ordinals α,

$$s \in F^{\alpha+1}(\varnothing) \text{ implies } f(s) \subseteq F^{\alpha}(\varnothing). \tag{18}$$

It is not the case that *all* positional winning strategies for $\exists$ in $\mathcal{U}_F$ are descending, but the next result shows that there always is one.

PROPOSITION 4.11. *Let $F : \wp(S) \to \wp(S)$ be a monotone functional.*

(1) For all $s \in S$, $s \in \mathrm{Win}_\exists(\mathcal{U}_F)$ iff $s \in LFP.F$;
(2) If $s \in LFP.F$, then $\exists$ has a descending winning strategy in $\mathcal{U}_F@s$.

PROOF. Point (1) corresponds to [Venema 2012, Theorem 3.14(2)]. For part (2) one can simply take the following strategy. Given $s \in LFP.F$, let α be the least ordinal such that $s \in F^{\alpha}(\varnothing)$; it is easy to see that α must be a successor ordinal, say $\alpha = \beta + 1$. Now simply put $f(s) := F^{\beta}(\varnothing)$. ◻

Definition 4.12. Let $F : \wp(S) \to \wp(S)$ be a monotone functional, let f be a positional winning strategy for $\exists$ in $\mathcal{U}_F$, and let $r \in S$. Define $T_{f,r} \subseteq S$ to be the set of states in S that are f-reachable in $\mathcal{U}_F@r$. This set has a tree structure induced by the map f itself, where the children of $s \in T_{f,r}$ are given by the set $f(s)$; we will refer to $T_{f,r}$ as the *strategy tree* of f.

Note that a strategy tree $T_{f,r}$ will have no infinite paths, since we define the notion only for a *winning* strategy f.

PROPOSITION 4.13. *Let $F : \wp(S) \to \wp(S)$ be a monotone functional, let $r \in S$, and let f be a descending winning strategy for $\exists$ in $\mathcal{U}_F$. Then*

$$r \in LFP.F \text{ implies } r \in LFP.F{\restriction_{T_{f,r}}}. \tag{19}$$

PROOF. Let F, r and f be as in the formulation of the proposition. Assume that $r \in LFP.F$, then clearly $r \in F^{\alpha}(\varnothing)$ for some ordinal α; furthermore, $T_{f,r}$ is defined and clearly we have $r \in T_{f,r}$. Abbreviate $T := T_{f,r}$. It then suffices to show that for all ordinals α we have

$$F^{\alpha}(\varnothing) \cap T \subseteq (F{\restriction_T})^{\alpha}(\varnothing). \tag{20}$$

We will prove (20) by transfinite induction. The base case, where $\alpha = 0$, and the inductive case where α is a limit ordinal are straightforward, so we focus on the case where α is a successor ordinal, say $\alpha = \beta + 1$. Take an arbitrary state $u \in F^{\beta+1}(\varnothing) \cap T$, then we find $f(u) \subseteq F^{\beta}(\varnothing)$ by our assumption (18), and $f(u) \subseteq T$ by definition of T. Then the induction hypothesis yields that $f(u) \subseteq (F{\restriction_T})^{\beta}(\varnothing)$, and so we have

$f(u) \subseteq (F{\restriction_T})^{\beta}(\varnothing) \cap T$. But since f is a winning strategy, and u is a winning position for $\exists$ in $\mathcal{U}_F$ by Claim 4.11(i), $f(u)$ is a legitimate move for $\exists$, and so we have $u \in F(f(u))$. Thus by monotonicity of F we obtain $u \in F((F{\restriction_T})^{\beta}(\varnothing) \cap T)$, and since $u \in T$ by assumption, this means that $u \in (F{\restriction_T})^{\beta+1}(\varnothing)$ as required. $\square$

We now turn to the specific case where we consider the least fixed point of a functional F which is induced by some formula $\varphi(p) \in \mu_D L_1$ on some LTS $\mathbb{S}$. By Proposition 4.11 and Fact 1.26, $\exists$ has a winning strategy in $\mathcal{E}(\mu p.\varphi(p), \mathbb{S})@(\mu p.\varphi(p), s)$ if and only if she has a winning strategy in $\mathcal{U}_F@s$ too, where $F := \varphi_p^{\mathbb{S}}$ is the monotone functional defined by $\varphi(p)$. The next Proposition makes this correspondence explicit when $L_1 = \mathrm{FOE}$.

First, we need to introduce some auxiliary concepts and notations. Given a winning strategy f for $\exists$ in $\mathcal{E}(\mu p.\varphi, \mathbb{S})@(\mu p.\varphi, s)$, we denote by $B(f)$ the set of all finite f-guided, possibly partial, matches in $\mathcal{E}(\psi, \mathbb{S})@(\psi, s)$ in which no position of the form $(\nu q.\psi, r)$ is visited. Let f be a positional winning strategies for $\exists$ in $\mathcal{U}_F@s$ and f' a winning strategy for her in $\mathcal{E}(\mu p.\varphi, \mathbb{S})@(\mu p.\varphi, s)$. We call f and f' *compatible* if each point in $T_{f,s}$ occurs on some path belonging to $B(f')$.

PROPOSITION 4.14. *Let $\varphi(p) \in \mu_D\mathrm{FOE}p$ and $s \in [\![\mu p.\varphi]\!]^{\mathbb{S}}$. Then there is a descending winning strategy for $\exists$ in $\mathcal{U}_F@s$ compatible with a winning strategy for $\exists$ in $\mathcal{E}(\mu p.\varphi, \mathbb{S})@(\mu p.\varphi, s)$*

PROOF. Let $F := \varphi_p^{\mathbb{S}}$ be the monotone functional defined by $\varphi(p)$. From $s \in [\![\mu p.\varphi]\!]^{\mathbb{S}}$, we get that $s \in LFP.F$. Applying Proposition 4.11 to the fact that $s \in LFP.F$ yields that $\exists$ has a descending winning strategy $f : S \to \wp(S)$ in $\mathcal{U}_F@s$. We define $\exists$'s strategies f' in $\mathcal{E}(\mu p.\varphi, \mathbb{S})@(\mu p.\varphi, s)$, and f^* in $\mathcal{U}_F@s$ as follows:

(1) In the evaluation games $\mathcal{E}$, after the initial automatic move, the position of the match is (φ, s); there $\exists$ first plays her positional winning strategy f_s from $\mathcal{E}(\varphi(p), \mathbb{S}[p \mapsto f(s)])@(\varphi(p), s)$, and we define her move $f^*(s)$ in the unfolding game $\mathcal{U}$ as the set of all nodes $t \in f(s)$ such that there is a f_s-guided match in $B(f_s)$ whose last position is (p, t).
(2) Each time a position (p, t) is reached in the evaluation games $\mathcal{E}$, distinguisg cases:
   (a) if $t \in \mathrm{Win}_\exists(\mathcal{U}_F)$, then $\exists$ continues with the positional winning strategy f_t from $\mathcal{E}(\varphi(p), \mathbb{S}[p \mapsto f(t)])@(\varphi(p), t)$, and we define her move $f^*(t)$ in $\mathcal{U}$ as the set of all nodes $w \in f(t)$ such that there is a f_t-guided match in $B(f_t)$ whose last position (p, w);
   (b) if $t \notin \mathrm{Win}_\exists(\mathcal{U}_F)$, then $\exists$ continues with a random positional strategy and we define $f^*(t) := \varnothing$.
(3) For any position (p, t) that was not reached in the previous steps, $\exists$ sets $f^*(t) := \varnothing$.

By construction, f' and f^* are compatible. Moreover, $f^*(t) \subseteq f(t)$, for $t \in S$, meaning that f^* is descending. We verify that both f' and f^* are actually winning strategies for $\exists$ in the respective games.

First of all, observe that every position of the form (p, t) reached during a f'-guided match, we have $t \in \mathrm{Win}_\exists(\mathcal{U}_F)$. This can be proved by induction on the number of position of the form (p, t) visited during an f'-guided match. For the inductive step, assume $w \in \mathrm{Win}_\exists(\mathcal{U}_F)$. Hence f_w is winning for $\exists$ in $\mathcal{E}(\varphi, \mathbb{S}[p \mapsto f(w)])@(\varphi, w)$. This means that if a position of the form (p, t) is reached, the variable p must be true at t in the model $\mathbb{S}[p \mapsto f(w)]$, meaning that it belongs to the set $f(w)$. By assumption f is a winning strategy for $\exists$ in $\mathcal{U}_F$, and therefore any element of $f(w)$ is again a member of the set $\mathrm{Win}_\exists(\mathcal{U}_F)$.

Finally, let π be an arbitrary f'-guided match of $\mathcal{E}(\varphi, \mathbb{S}[p \mapsto f(w)])@(\varphi, w)$. We verify that π is winning for $\exists$. First observe that since f is winning for her in $\mathcal{U}_F@s$, the fixpoint variable p is unfolded only finitely many times during π. Let (p,t) be the last basic position in π where p occurs. Then from now on f' and f_t coincide, yielding that the match is winning for $\exists$.

We finally verify that f^* is winning for $\exists$ in the unfolding game $\mathcal{U}_F@s$. First of all, since f' is winning, $B(f')$ does not contain an infinite ascending chain of f'-guided matches, and thence any f^*-guided match in $\mathcal{U}_F@s$ is finite. It therefore remains to verify that for every f^*-guided match π in $\mathcal{U}_F@s$ such that $\mathsf{last}(\pi)$ is an $\exists$ position, she can always move. We do it by induction on the length of a f^*-guided match. At each step, we use compatibility and thus keep track of the corresponding position in the evaluation game $\mathcal{E}(\mu p.\varphi,\mathbb{S})@(\mu p.\varphi,s)$. The initial position for her is $s \in S$. Notice that $f^*(s) = f(s) \cap B(\xi')$ and therefore f' corresponds to f_s on $\mathcal{E}(\varphi(p),\mathbb{S}[p \mapsto f^*(s)])@(\varphi(p),s)$ and it is therefore winning for $\exists$. In particular, this means that $s \in F(f^*(s))$. Hence, as initial move, $\exists$ is allowed to play $f^*(s)$. Moreover any subsequent choice for $\forall$ is such that there is a winning match $\pi \in B(\xi_s)$ for $\exists$ such that $\mathsf{last}(\pi) = (p,w)$. For the induction step, assume $\forall$ has chosen $t \in f^*(w)$, where $f^*(w) = f(w) \cap B(\xi')$, f' corresponds to the winning strategy f_w on $\mathcal{E}(\varphi(p),\mathbb{S}[p \mapsto f^*(w)])@(\varphi(p),w)$, and there is a winning match $\pi \in B(\xi_w)$ for $\exists$ such that $\mathsf{last}(\pi) = (p,w)$. By construction, f' corresponds to the winning strategy f_t for $\exists$ on $\mathcal{E}(\varphi(p),\mathbb{S}[p \mapsto f(t)])@(\varphi(p),t)$. Because $f^*(t) = f(s) \cap B(\xi')$, f_t is also winning for her in $\mathcal{E}(\varphi(p),\mathbb{S}[p \mapsto f^*(t)])@(\varphi(p),t)$, meaning that $s \in F(f^*(s))$. The move $f^*(t)$ is therefore admissible, and any subsequent choice for $\forall$ is such that there is a winning match $\pi \in B(\xi_t)$ for $\exists$ with $\mathsf{last}(\pi) = (p,w)$. $\square$

PROOF OF PROPOSITION 4.3(1). Let $\mathbb{S}$ be an LTS and $\varphi(p) \in \mu_D\mathrm{FOE}_1 p$.

The right-to-left direction of (13) being proved by Proposition 4.5, we check the left-to-right direction. We first verify that winning strategies in evaluation games for noetherian fixpoint formulas naturally induce bundles. More precisely:

CLAIM. Let $B^{\mathbb{S}}(f)$ be the projection of $B(f)$ on S, that is the set of all paths in $\mathbb{S}$ that are a projection on S of a f-guided (partial) match in $B(f)$. Then $B^{\mathbb{S}}(f)$ is a bundle.

PROOF OF CLAIM. Assume towards a contradiction that $B^{\mathbb{S}}(f)$ contains an infinite ascending chain $\pi_0 \sqsubset \pi_1 \sqsubset \cdots$. Let π be the limit of this chain and consider the set of elements in $B(f)$ that, projected on S, are prefixes of π. By König's Lemma, this set contains an infinite ascending chain whose limit is an infinite f-guided match in $\mathcal{E}(\mu p.\varphi,\mathbb{S})$ which starts at $(\mu p.\varphi,s)$, and of which π is the projection on S. By definition of $B(f)$, the highest bound variable of $\mu p.\varphi$ that gets unravelled infinitely often in ρ is a μ-variable, meaning that the match is winning for $\forall$, a contradiction. ◀

Assume that $s \in [\![\mu p.\varphi]\!]^{\mathbb{S}}$, and let $F := \varphi^{\mathbb{S}}_p$ be the monotone functional defined by $\varphi(p)$. By Proposition 4.14, $\exists$ has a winning strategy f' in $\mathcal{E}(\mu p.\varphi,\mathbb{S})@(\mu p.\varphi,s)$ compatible with a descending winning strategy f in $\mathcal{U}_F@s$. By Proposition 4.13, we obtain that $s \in LFP.F{\upharpoonright_{T_{f,s}}}$. Because of compatibility, every node in $T_{f,s}$ occurs on some path of $B(f')$. From the Claim we known that $B^{\mathbb{S}}(f')$ is a bundle, meaning that $T_{f,s}$ is noetherian as required. $\square$

5. EXPRESSIVENESS MODULO BISIMILARITY

In this Section we use the tools developed in the previous parts to prove the main results of the paper on expressiveness modulo bisimilarity, viz., Theorem 1.1 stating

$$\mu_N\mathrm{ML} \;\equiv\; \mathrm{NMSO}/{\underline{\leftrightarrow}} \tag{21}$$

$$\mu_C\mathrm{ML} \;\equiv\; \mathrm{WMSO}/{\underline{\leftrightarrow}} \tag{22}$$

PROOF OF THEOREM 1.1. The structure of the proof is the same for the statements (21) and (22). In both cases, we will need three steps to establish a link between the modal language on the left hand side of the equation to the bisimulation-invariant fragment of the second-order logic on the right hand side.

The first step is to connect the fragments $\mu_N\mathrm{ML}$ and $\mu_C\mathrm{ML}$ of the modal μ-calculus to, respectively, the weak and the continuous-weak automata for first-order logic without equality. That is, in Theorem 5.1 below we prove the following:

$$\mu_N\mathrm{ML} \;\equiv\; Aut_w(\mathrm{FO}_1) \tag{23}$$

$$\mu_C\mathrm{ML} \;\equiv\; Aut_{wc}(\mathrm{FO}_1) \tag{24}$$

Second, the main observations that we shall make in this section is that

$$Aut_w(\mathrm{FO}_1) \;\equiv\; Aut_w(\mathrm{FOE}_1)/{\underline{\leftrightarrow}} \tag{25}$$

$$Aut_{wc}(\mathrm{FO}_1) \;\equiv\; Aut_{wc}(\mathrm{FOE}_1^\infty)/{\underline{\leftrightarrow}} \tag{26}$$

That is, for (25) we shall see in Theorem 5.4 below that a weak FOE_1-automaton $\mathbb{A}$ is bisimulation invariant iff it is equivalent to a weak FO_1-automaton $\mathbb{A}^\Diamond$ (effectively obtained from $\mathbb{A}$); and similarly for (26).

Finally, we use the automata-theoretic characterisations of NMSO and WMSO that we obtained in earlier sections:

$$Aut_w(\mathrm{FOE}_1) \;\equiv\; \mathrm{NMSO} \tag{27}$$

$$Aut_{wc}(\mathrm{FOE}_1^\infty) \;\equiv\; \mathrm{WMSO} \tag{28}$$

Then it is obvious that the equation (21) follows from (23), (25) and (27), while similarly (22) follows from (24), (26) and (28). □

It is left to prove the equations (23) and (24), and (25) and (26); this we will take care of in the two subsections below.

5.1. Automata for $\mu_N\mathrm{ML}$ and $\mu_C\mathrm{ML}$

In this subsection we consider the automata corresponding to the continuous and the alternation-free μ-calculus. That is, we verify the equations (23) and (24).

THEOREM 5.1.

(1) *There is an effective construction transforming a formula* $\varphi \in \mu\mathrm{ML}$ *into an equivalent automaton in* $Aut(\mathrm{FO}_1)$*, and vice versa.*
(2) *There is an effective construction transforming a formula* $\varphi \in \mu_N\mathrm{ML}$ *into an equivalent automaton in* $Aut_w(\mathrm{FO}_1)$*, and vice versa.*
(3) *There is an effective construction transforming a formula* $\varphi \in \mu_C\mathrm{ML}$ *into an equivalent automaton in* $Aut_{wc}(\mathrm{FO}_1)$*, and vice versa.*

PROOF. In each of these cases the direction from left to right is easy to verify, so we omit details. For the opposite direction, we focus on the hardest case, that is, we will only prove that $Aut_{wc}(\mathrm{FO}_1) \leq \mu_C\mathrm{ML}$. By Theorem 1.30 it suffices to show that $\mu_C\mathrm{FO}_1 \leq \mu_C\mathrm{ML}$, and we will in fact provide a direct, inductively defined, truth-preserving translation $(\cdot)^t$ from $\mu_C\mathrm{FO}_1(\mathsf{P})$ to $\mu_C\mathrm{ML}(\mathsf{P})$. Inductively we will ensure that, for every set $\mathsf{Q} \subseteq \mathsf{P}$:

$$\varphi \in \mathtt{Con}_{\mathsf{Q}}(\mu\mathrm{FO}_1) \text{ implies } \varphi^t \in \mathtt{Con}_{\mathsf{Q}}(\mu\mathrm{ML}) \tag{29}$$

and that the dual property holds for cocontinuity.

Most of the clauses of the definition of the translation $(\cdot)^t$ are completely standard: for the atomic clause we take $p^t := p$ and $(\neg p)^t := \neg p)$, for the boolean connectives we define $(\varphi_0 \vee \varphi_1)^t := \varphi_0^t \vee \varphi_1^t$ and $(\varphi_0 \wedge \varphi_1)^t := \varphi_0^t \wedge \varphi_1^t$, and for the fixpoint operators we take $(\mu p.\varphi)^t := \mu p.\varphi^t$ and $(\nu p.\varphi)^t := \nu p.\varphi^t$ — to see that the latter clauses indeed provide formulas in $\mu_C\mathrm{ML}$ we use (29) and its dual. In all of these cases it is easy to show that (29) holds (or remains true, in the inductive cases).

The only interesting case is where φ is of the form $\bigcirc_\alpha(\varphi_1, \ldots, \varphi_n)$. By definition of the language $\mu_C\mathrm{FO}_1$ we may assume that $\alpha(a_1, \ldots, a_n) \in \mathtt{Con}_B(\mathrm{FO}_1(A))$, where $A = \{a_1, \ldots, a_n\}$ and $B = \{a_1, \ldots, a_k\}$, that for each $1 \leq i \leq k$ the formula φ_i belongs to the set $\mathtt{Con}_{\mathsf{Q}}(\mu_C\mathrm{FO}_1)$ and that for each $k+1 \leq j \leq n$ the formula φ_j is Q-free. It follows by the induction hypothesis that $\varphi_l \equiv \varphi_l^t \in \mu_C\mathrm{ML}$ for each l, that $\varphi_i^t \in \mathtt{Con}_{\mathsf{Q}}(\mu\mathrm{ML})$ for each $1 \leq i \leq k$, and that the formula φ_j^t is Q-free for each $k+1 \leq j \leq n$. The key observation is now that by Theorem 1.12 we may without loss of generality assume that α is in *normal form*; that is, a disjunction of formulas of the form $\alpha_\Sigma = \nabla^+_{\mathrm{FO}}(\Sigma, \Pi)$, where every Σ and Π is a subset of $\wp(A)$, $B \cap \bigcup \Pi = \varnothing$ for every Π, and

$$\nabla^+_{\mathrm{FO}}(\Sigma, \Pi) := \bigwedge_{S \in \Sigma} \exists x \bigwedge_{a \in S} a(x) \ \wedge\ \forall x \bigvee_{S \in \Pi} \bigwedge_{a \in S} a(x)$$

We now define

$$\begin{aligned} \bigvee \left(\bigcirc_{\alpha_\Sigma}(\overline{\varphi})\right)^t &:= \textstyle\bigwedge_{S \in \Sigma} \Diamond \bigwedge_{a_l \in S} \varphi_l^t \ \wedge\ \Box \bigvee_{S \in \Pi} \bigwedge_{a_j \in S} \varphi_j^t \\ \varphi^t &:= \textstyle\bigvee_\Sigma \left(\bigcirc_{\alpha_\Sigma}(\overline{\varphi})\right)^t \end{aligned}$$

It is then obvious that φ and φ^t are equivalent, so it remains to verify (29). But this is immediate by the observation that all formulas φ_j^t in the scope of the $\Box$ are associated with an a_j belonging to a set $S \subseteq A$ that has an empty intersection with the set B; that is, each a_j belongs to the set $\{a_{k+1}, \ldots, a_n\}$ and so φ_j^t is Q-free. $\Box$

5.2. Bisimulation invariance, one step at a time

In this subsection we will show how the bisimulation invariance results in this paper can be proved by automata-theoretic means. Following Janin & Walukiewicz [Janin and Walukiewicz 1996], we will define a construction that, for $L_1 \in \{\mathrm{FOE}_1, \mathrm{FOE}_1^\infty\}$, transforms an arbitrary L_1-automaton $\mathbb{A}$ into an FO_1-automaton $\mathbb{A}^\Diamond$ such that $\mathbb{A}$ is bisimulation invariant iff it is equivalent to $\mathbb{A}^\Diamond$. In addition, we will make sure that this transformation preserves both the weakness and the continuity condition. The operation $(\cdot)^\Diamond$ is completely determined by the following translation at the one-step level.

Definition 5.2. Recall from Theorem 1.11 that any formula in ${\mathrm{FOE}_1}^+(A)$ is equivalent to a disjunction of formulas of the form $\nabla^+_{\mathrm{FOE}}(\overline{\mathbf{T}}, \Sigma)$, whereas any formula in ${\mathrm{FOE}_1^\infty}^+(A)$ is equivalent to a disjunction of formulas of the form $\nabla^+_{\mathrm{FOE}^\infty}(\overline{\mathbf{T}}, \Pi, \Sigma)$. Based on these normal forms, for both one-step languages $L_1 = \mathrm{FOE}_1$ and $L_1 = \mathrm{FOE}_1^\infty$, we define the translation $(\cdot)^{\diamond} : {L_1}^{+}(A) \to \mathrm{FO}_1^{+}(A)$ by setting

$$\left.\begin{array}{l}\left(\nabla^{+}_{\mathrm{FOE}}(\overline{\mathbf{T}},\Sigma)\right)^{\diamond}\\ \left(\nabla^{+}_{\mathrm{FOE}^{\infty}}(\overline{\mathbf{T}},\Pi,\Sigma)\right)^{\diamond}\end{array}\right\} := \bigwedge_i \exists x_i.\tau^{+}_{T_i}(x_i) \wedge \forall x. \bigvee_{S\in\Sigma} \tau^{+}_{S}(x),$$

and for $\alpha = \bigvee_i \alpha_i$ we define $\alpha^{\diamond} := \bigvee \alpha_i^{\diamond}$.

This definition propagates to the level of automata in the obvious way.

Definition 5.3. Let $L_1 \in \{\mathrm{FOE}_1, \mathrm{FOE}_1^{\infty}\}$ be a one-step language. Given an automaton $\mathbb{A} = \langle A, \Delta, \Omega, a_I\rangle$ in $Aut(L_1)$, define the automaton $\mathbb{A}^{\diamond} := \langle A, \Delta^{\diamond}, \Omega, a_I\rangle$ in $Aut(\mathrm{FO}_1)$ by putting, for each $(a, c) \in A \times C$:

$$\Delta^{\diamond}(a, c) := (\Delta(a, c))^{\diamond}.$$

The main result of this section is the theorem below. For its formulation, recall that $\mathbb{S}^{\omega}$ is the ω-unravelling of the model $\mathbb{S}$ (as defined in the preliminaries). As an immediate corollary of this result, we see that (25) and (26) hold indeed.

THEOREM 5.4. *Let $L_1 \in \{\mathrm{FOE}_1, \mathrm{FOE}_1^{\infty}\}$ be a one-step language and let $\mathbb{A}$ be an L_1-automaton.*

(*1*) *The automata $\mathbb{A}$ and $\mathbb{A}^{\diamond}$ are related as follows, for every model $\mathbb{S}$:*

$$\mathbb{A}^{\diamond} \textit{ accepts } \mathbb{S} \textit{ iff } \mathbb{A} \textit{ accepts } \mathbb{S}^{\omega}. \tag{30}$$

(*2*) *The automaton $\mathbb{A}$ is bisimulation invariant iff $\mathbb{A} \equiv \mathbb{A}^{\diamond}$.*

(*3*) *If $\mathbb{A} \in Aut_w(L_1)$ then $\mathbb{A}^{\diamond} \in Aut_w(\mathrm{FO}_1)$, and if $\mathbb{A} \in Aut_{wc}(\mathrm{FOE}_1^{\infty})$ then $\mathbb{A}^{\diamond} \in Aut_{wc}(\mathrm{FO}_1)$.*

The remainder of this section is devoted to the proof of Theorem 5.4. The key proposition is the following observation on the one-step translation, that we take from the companion paper [Carreiro et al. 2018].

PROPOSITION 5.5. *Let $L_1 \in \{\mathrm{FOE}_1, \mathrm{FOE}_1^{\infty}\}$. For every one-step model (D, V) and every $\alpha \in L_1^{+}(A)$ we have*

$$(D, V) \models \alpha^{\diamond} \textit{ iff } (D \times \omega, V_{\pi}) \models \alpha, \tag{31}$$

where V_{π} is the induced valuation given by $V_{\pi}(a) := \{(d, k) \mid d \in V(a), k \in \omega\}$.

PROOF OF THEOREM 5.4. The proof of the first part is based on a fairly routine comparison, based on Proposition 5.5, of the acceptance games $\mathcal{A}(\mathbb{A}^{\diamond}, \mathbb{S})$ and $\mathcal{A}(\mathbb{A}, \mathbb{S}^{\omega})$. (In a slightly more general setting, the details of this proof can be found in [Venema 2014].)

For part 2, the direction from right to left is immediate by Theorem 5.1. The opposite direction follows from the following equivalences, where we use the bisimilarity of $\mathbb{S}$ and $\mathbb{S}^{\omega}$ (Fact 2.4):

$$\begin{array}{rlr}\mathbb{A} \text{ accepts } \mathbb{S} & \text{iff } \mathbb{A} \text{ accepts } \mathbb{S}^{\omega} & (\mathbb{A} \text{ bisimulation invariant})\\ & \text{iff } \mathbb{A}^{\diamond} \text{ accepts } \mathbb{S} & (\text{equivalence } (30))\end{array}$$

It remains to be checked that the construction $(\cdot)^{\diamond}$, which has been defined for arbitrary automata in $Aut(L_1)$, transforms both WMSO-automata and NMSO-automata into automata of the right kind. This can be verified by a straightforward inspection at the one-step level. □

Remark 5.6. In fact, we are dealing here with an instantiation of a more general phenomenon that is essentially coalgebraic in nature. In [Venema 2014] it is proved that if L_1 and L_1' are two one-step languages that are connected by a translation

$(\cdot)^{\diamond} : L_1' \to L_1$ satisfying a condition similar to (31), then we find that $Aut(L_1)$ corresponds to the bisimulation-invariant fragment of $Aut(L_1')$: $Aut(L_1) \equiv Aut(L_1')/\underline{\leftrightarrow}$. This subsection can be generalized to prove similar results relating $Aut_w(L_1)$ to $Aut_w(L_1')$, and $Aut_{wc}(L_1)$ to $Aut_{wc}(L_1')$.

REFERENCES

André Arnold and Damian Niwinski. 1992. Fixed point characterization of weak monadic logic definable sets of trees. In *Tree Automata and Languages*. 159–188.

André Arnold and Damian Niwiński. 2001. *Rudiments of μ-calculus*. Studies in Logic and the Foundations of Mathematics, Vol. 146. North-Holland. 1–39 pages.

Facundo Carreiro. 2015. *Fragments of Fixpoint Logics*. Ph.D. Dissertation. Institute for Logic, Language and Computation, Universiteit van Amsterdam.

Facundo Carreiro, Alessandro Facchini, Yde Venema, and Fabio Zanasi. 2014. Weak MSO: automata and expressiveness modulo bisimilarity. 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*. 27:1–27:27. DOI:http://dx.doi.org/10.1145/2603088.2603101

Facundo Carreiro, Alessandro Facchini, Yde Venema, and Fabio Zanasi. 2018. *Model Theory of Monadic Predicate Logic with the Infinity Quantifier*. Technical Report. arXiv.

Corina Cîrstea and Dirk Pattinson. 2004. Modular Construction of Modal Logics. In *Proceedings of the 15th International Conference on Concurrency Theory (CONCUR 2004) (LNCS)*, Ph. Gardner and N. Yoshida (Eds.), Vol. 3170. Springer, 258–275.

Anuj Dawar and Martin Otto. 2009. Modal characterisation theorems over special classes of frames. *Ann. Pure Appl. Logic* 161, 1 (2009), 1–42.

Ernest Allen Emerson and Charanjit S. Jutla. 1991. Tree Automata, μ-Calculus and Determinacy (Extended Abstract). In *FOCS*. IEEE Computer Society, 368–377.

Ernest Allen Emerson and Chin-Laung Lei. 1986. Efficient Model Checking in Fragments of the Propositional μ-Calculus (Extended Abstract). In *LICS*. 267–278.

Alessandro Facchini, Yde Venema, and Fabio Zanasi. 2013. A Characterization Theorem for the Alternation-Free Fragment of the Modal μ-Calculus. In *LICS*. IEEE Computer Society, 478–487.

Gaëlle Fontaine. 2008. Continuous Fragment of the mu-Calculus. In *CSL (Lecture Notes in Computer Science)*, Michael Kaminski and Simone Martini (Eds.), Vol. 5213. Springer, 139–153.

Gaëlle Fontaine and Yde Venema. 2018. Some model theory for the modal μ-calculus: syntactic characterisations of semantic properties. *Logical Methods in Computer Science* 14, 1 (2018).

Erich Grädel, Wolfgang Thomas, and Thomas Wilke (Eds.). 2002. *Automata, Logics, and Infinite Games: A Guide to Current Research*. Lecture Notes in Computer Science, Vol. 2500. Springer.

David Janin and Igor Walukiewicz. 1996. On the Expressive Completeness of the Propositional μ-Calculus with Respect to Monadic Second Order Logic. In *Proceedings of the 7th International Conference on Concurrency Theory (CONCUR '96)*. Springer-Verlag, London, UK, 263–277. http://portal.acm.org/citation.cfm?id=646731.703838

Christian Kissig and Yde Venema. 2009. Complementation of Coalgebra Automata. In *CALCO (Lecture Notes in Computer Science)*, Alexander Kurz, Marina Lenisa, and Andrzej Tarlecki (Eds.), Vol. 5728. Springer, 81–96.

Dexter Kozen. 1983. Results on the propositional μ-calculus. *Theoretical Computer Science* 27, 3 (1983), 333 – 354. DOI:http://dx.doi.org/DOI:10.1016/0304-3975(82)90125-6

David E. Long, Anca Browne, Edmund M. Clarke, Somesh Jha, and Wilfredo R. Marrero. 1994. An Improved Algorithm for the Evaluation of Fixpoint Expressions. In *Computer Aided Verification, 6th International Conference, CAV '94, Stanford, California, USA, June 21-23, 1994, Proceedings*. 338–350. DOI:http://dx.doi.org/10.1007/3-540-58179-0_66

Andrzej Mostowski. 1957. On a generalization of quantifiers. *Fundamenta Mathematicae* 44, 1 (1957), 12–36. http://eudml.org/doc/213418

Andrzej Mostowski. 1991. *Games with forbidden positions*. Technical Report 78. University of Gdansk.

David E. Muller, Ahmed Saoudi, and Paul E. Schupp. 1992. Alternating Automata, the Weak Monadic Theory of Trees and its Complexity. *Theor. Comput. Sci.* 97, 2 (1992), 233–244.

David E. Muller and Paul E. Schupp. 1987. Alternating automata on infinite trees. *Theoretical Computer Science* 54, 2-3 (1987), 267–276. DOI:http://dx.doi.org/10.1016/0304-3975(87)90133-2

Damian Niwinski. 1986. On Fixed-Point Clones (Extended Abstract). In *ICALP (Lecture Notes in Computer Science)*, Laurent Kott (Ed.), Vol. 226. Springer, 464–473.

David Park. 1980. On the Semantics of Fair Parallelism. In *Proceedings of the Abstract Software Specifications, 1979 Copenhagen Winter School*. Springer-Verlag, London, UK, UK, 504–526. http://dl.acm.org/citation.cfm?id=647448.727229

Eric Rosen. 1997. Modal logic over finite structures. *Journal of Logic, Language and Information* 6 (1997), 427–439.

Balder ten Cate and Alessandro Facchini. 2011. Characterizing EF over Infinite Trees and Modal Logic on Transitive Graphs. In *MFCS (Lecture Notes in Computer Science)*, Filip Murlak and Piotr Sankowski (Eds.), Vol. 6907. Springer, 290–302.

Jouko Väänänen. 1977. Remarks on generalized quantifiers and second-order logics. In *Set theory and hierarchy theory*. Prace Naukowe Instytutu Matematyki Politechniki Wroclawskiej, Wroclaw, 117–123.

Johan van Benthem. 1977. *Modal Correspondence Theory*. Ph.D. Dissertation. Universiteit van Amsterdam.

Yde Venema. 2014. Expressiveness modulo bisimilarity: a coalgebraic perspective. (2014). To appear in: A. Baltag and S. Smets (eds.), *Johan van Benthem, Outstanding Contributions Series*, Springer.

Yde Venema. October 2012. Lectures on the modal μ-calculus. (October 2012). Lecture Notes, Institute for Logic, Language and Computation, University of Amsterdam.

Igor Walukiewicz. 1996. Monadic Second Order Logic on Tree-Like Structures. In *STACS (Lecture Notes in Computer Science)*, Claude Puech and Rüdiger Reischuk (Eds.), Vol. 1046. Springer, 401–413.

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

Fabio Zanasi. 2012. *Expressiveness of Monadic Second Order Logics on Infinite Trees of Arbitrary Branching Degree*. Master's thesis. ILLC, Universiteit van Amsterdam, the Netherlands.

Received July 2018; revised XXX; accepted XXX