Bisimulation Invariant Monadic-Second Order Logic in the Finite

Achim Blumensath[1]
Masaryk University Brno
blumens@fi.muni.cz

Felix Wolf[2]
Technische Universität Darmstadt, Institute TEMF
Graduate School of Excellence Computational Engineering
wolf@gsc.tu-darmstadt.de

Abstract

We consider bisimulation-invariant monadic second-order logic over various classes of finite transition systems. We present several combinatorial characterisations of when the expressive power of this fragment coincides with that of the modal μ-calculus. Using these characterisations we prove for some simple classes of transition systems that this is indeed the case. In particular, we show that, over the class of all finite transition systems with Cantor–Bendixson rank at most k, bisimulation-invariant MSO coincides with L_μ.

2012 ACM Subject Classification F.4.1 Mathematical Logic

Keywords and phrases bisimulation, monadic second-order logic, composition method

Digital Object Identifier 10.4230/LIPIcs.ICALP.2018.14

1 Introduction

A characterisation of the bisimulation-invariant fragment of a given classical logic relates this logic to a suitable modal logic. In this way, one obtains a correspondence between a family of classical logics and a family of modal logics. Such characterisation results therefore help with ordering the zoo of logics introduced (on both sides) over the years and with distinguishing between natural and artificial instances of such logics.

The study of bisimulation-invariant fragments of classical logics was initiated by a result of van Benthem [2] who proved that the bisimulation-invariant fragment of first-order logic coincides with standard modal logic. Inspired by this work, several other characterisations have been obtained. The table below summarises the results known so far.

bisimulation-invariant fragment	modal logic	reference
first-order logic	modal logic	[2]
monadic second-order logic	modal μ-calculus	[10]
monadic path logic	CTL*	[12, 13]
weak monadic second-order logic	continuous μ-calculus	[4]
weak chain logic	PDL	[4]

There are also similar characterisations for various variants of bisimulation like *guarded bisimulation* [1, 7] or bisimulation for *inquisitive modal logic* [5].

[1] Work supported by the Czech Science Foundation, grant No. GA17-01035S
[2] Work partially supported by the *Excellence Initiative* of the German Federal and State Governments and the Graduate School of Computational Engineering at Technische Universität Darmstadt

45th International Colloquium on Automata, Languages, and Programming (ICALP 2018).
Editors: Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Don Sannella; Article No. 14; pp. 14:1–14:22
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Researchers in finite model theory started to investigate to which extent these correspondences also hold when only considering *finite* structures, that is, whether every formula of a given classical logic that is bisimulation-invariant over the class of all finite transition systems is equivalent, over that class, to the corresponding modal logic. For first-order logic, a corresponding characterisation does indeed hold. Its proof by Rosen [15] uses tools from finite model theory and is very different to the proof by van Benthem.

The above mentioned result by Janin and Walukiewicz on bisimulation-invariant monadic second-order logic has so far defied all attempts at a similar transfer to the realm of finite structures. The main reason is that the original proof is based on automata-theoretic techniques and an essential ingredient is a reduction to trees, via the unravelling operation. As this operation produces infinite trees, we cannot use it for formulae that are only bisimulation-invariant over finite transition systems.

In this paper we start a fresh attempt at a finitary version of the result of Janin and Walukiewicz. Instead of automata-theoretic techniques we employ the composition method. For certain classes of very simple, finite transition systems we characterise the bisimulation-invariant fragments of monadic second-order logic over these classes. We hope that some day our techniques can be extended to the general case of *all* finite structures, but currently there are still a few technical obstacles to overcome.

We start in Section 2 by recalling the needed material on bisimulation and by listing all known results on bisimulation-invariant monadic second-order logic. We also collect some low-hanging fruit by proving two new results concerning (i) finite classes and (ii) the class of all finite trees. Finally, we lay the groundwork for the more involved proofs to follow by characterising bisimulation-invariance in terms of a combinatorial property called the *unravelling property.* In Section 3, we collect some tools from logic we will need. The emphasis in on so-called *composition lemmas.* Nothing in this section is new.

Finally we start in Section 4 in earnest by developing the technical machinery our proofs are based on. Sections 5 and 6 contain our first two applications: characterisations of bisimulation-invariant monadic second-order logic over (i) the class of lassos and (ii) certain classes of what we call *hierarchical* lassos. The former is already known and simply serves as an example of our techniques and to fix our notation for the second result, which is new.

Before presenting our last characterisation result, we develop in Section 7 some additional technical tools that allow us to reduce one characterisation result to another. This is then applied in Section 8 to the most complex of our results. We characterise bisimulation-invariant monadic second-order logic over the class of all transition systems of a given Cantor–Bendixson rank.

2 Bisimulation-invariance

We consider two logics in this paper: (i) *monadic second-order logic* (MSO), which is the extension of first-order logic by set variables and set quantifiers, and (ii) the *modal μ-calculus* (L_μ), which is the fixed-point extension of modal logic. A detailed introduction can be found, e.g., in [8]. Concerning the μ-calculus and bisimulation, we also refer to the survey [17]. *Transition systems* are directed graphs where the edges are labelled by elements of a given set A and vertices by elements of some set I. Formally, we consider a transition system as a structure of the form $\mathfrak{S} = \langle S, (E_a)_{a\in A}, (P_i)_{i\in I}, s_0\rangle$ where the $E_a \subseteq S \times S$ are (disjoint) binary edge relations, the $P_i \subseteq S$ are (disjoint) unary predicates, and s_0 is the initial state. We write $\mathfrak{S}, s$ to denote the transition system obtained from $\mathfrak{S}$ by declaring s to be the initial state.

A central notion in modal logic is *bisimilarity* since modal logics cannot distinguish between bisimilar systems.

▶ **Definition 2.1.** Let $\mathfrak{S}$ and $\mathfrak{T}$ be transition systems.

(a) A *bisimulation* between $\mathfrak{S}$ and $\mathfrak{T}$ is a binary relation $Z \subseteq S \times T$ such that all pairs $\langle s, t\rangle \in Z$ satisfy the following conditions.

(prop) $s \in P_i^{\mathfrak{S}}$ iff $t \in P_i^{\mathfrak{T}}$, for all $i \in I$.
(forth) For each edge $\langle s, s'\rangle \in E_a^{\mathfrak{S}}$, there is some $\langle t, t'\rangle \in E_a^{\mathfrak{T}}$ such that $\langle s', t'\rangle \in Z$.
(back) For each edge $\langle t, t'\rangle \in E_a^{\mathfrak{T}}$, there is some $\langle s, s'\rangle \in E_a^{\mathfrak{S}}$ such that $\langle s', t'\rangle \in Z$.

(b) Let s_0 and t_0 be the initial states of, respectively, $\mathfrak{S}$ and $\mathfrak{T}$. We say that $\mathfrak{S}$ and $\mathfrak{T}$ are *bisimilar* if there exists a bisimulation Z between $\mathfrak{S}$ and $\mathfrak{T}$ with $\langle s_0, t_0\rangle \in Z$. We denote this fact by $\mathfrak{S} \sim \mathfrak{T}$.

(c) We denote by $\mathcal{U}(\mathfrak{S})$ the *unravelling* of a transition system $\mathfrak{S}$. ⌟

The next two observations show that the unravelling operation is closely related to bisimilarity. In fact, having the same unravelling can be seen as a poor man's version of bisimilarity.

▶ **Lemma 2.2.** *Let $\mathfrak{S}$ and $\mathfrak{T}$ be transition systems.*

(a) $\mathcal{U}(\mathfrak{S}) \sim \mathfrak{S}$.
(b) $\mathfrak{S} \sim \mathfrak{T}$ *implies* $\mathcal{U}(\mathfrak{S}) \sim \mathcal{U}(\mathfrak{T})$.

As already mentioned modal logics cannot distinguish between bisimilar systems. They are *bisimulation-invariant* in the sense of the following definition.

▶ **Definition 2.3.** Let $\mathcal{C}$ be a class of transition systems.

(a) An MSO-formula φ is *bisimulation-invariant* over $\mathcal{C}$ if

$$\mathfrak{S} \sim \mathfrak{T} \quad \text{implies} \quad \mathfrak{S} \models \varphi \;\Leftrightarrow\; \mathfrak{T} \models \varphi, \quad \text{for all } \mathfrak{S}, \mathfrak{T} \in \mathcal{C}.$$

(b) We say that, *over the class* $\mathcal{C}$*, bisimulation-invariant* MSO *coincides with* L_μ if, for every MSO-formula φ that is bisimulation-invariant over the class $\mathcal{C}$, there exists an L_μ-formula ψ such that

$$\mathfrak{S} \models \varphi \quad \text{iff} \quad \mathfrak{S} \models \psi, \quad \text{for all } \mathfrak{S} \in \mathcal{C}.$$ ⌟

A straightforward induction over the structure of formulae shows that every L_μ-formula is bisimulation-invariant over all transition systems. Hence, bisimulation-invariance is a necessary condition for an MSO-formula to be equivalent to an L_μ-formula.

The following characterisations of bisimulation-invariant MSO have been obtained so far. We start with the result of Janin and Walukiewicz.

▶ **Theorem 2.4** (Janin, Walukiewicz [10])**.** *Over the class of all transition systems, bisimulation-invariant* MSO *coincides with* L_μ.

The main part of the proof consists in proving the following variant, which implies the case of all structures by a simple reduction.

▶ **Theorem 2.5** (Janin, Walukiewicz)**.** *Over the class of all trees, bisimulation-invariant* MSO *coincides with* L_μ.

There have already been two attempts at a finitary version. The first one is by Hirsch who considered the class of all regular trees, i.e., unravellings of finite transition systems. The proof is based on the fact that a formula is bisimulation-invariant over all trees if, and only if, it is bisimulation-invariant over regular trees.

▶ **Theorem 2.6** (Hirsch [9]). *Over the class of all regular trees, bisimulation-invariant* MSO *coincides with* L_μ.

The second result is by Dawar and Janin who considered the class of finite lassos, i.e., finite paths leading to a cycle. We will present a proof in Section 5 below.

▶ **Theorem 2.7** (Dawar, Janin [6]). *Over the class of all lassos, bisimulation-invariant* MSO *coincides with* L_μ.

In this paper, we will extend this last result to larger classes. We start with two easy observations. The first one is nearly trivial.

▶ **Theorem 2.8.** *Over every finite class $\mathcal{C}$ of finite transition systems, bisimulation-invariant* MSO *coincides with* L_μ.

The second observation is much deeper, but fortunately nearly all of the work has already been done by Janin and Walukiewicz.

▶ **Theorem 2.9.** *Over the class of all finite trees, bisimulation-invariant* MSO *coincides with* L_μ.

As a preparation for the more involved characterisation results to follow, we simplify our task by introducing the following property of a class $\mathcal{C}$ of transition systems, which will turn out to be equivalent to having a characterisation result for bisimulation-invariant MSO over $\mathcal{C}$.

▶ **Definition 2.10.** We say that a class $\mathcal{C}$ of transition systems has the *unravelling property* if, for every MSO-formula φ that is bisimulation-invariant over $\mathcal{C}$, there exists an MSO-formula $\hat{\varphi}$ that is bisimulation-invariant over trees such that

$$\mathfrak{S} \models \varphi \quad \text{iff} \quad \mathcal{U}(\mathfrak{S}) \models \hat{\varphi}\,, \quad \text{for all } \mathfrak{S} \in \mathcal{C}\,.$$ ⌟

Using Theorem 2.5, we can reformulate this definition as follows. This version will be our main tool to prove characterisation results for bisimulation-invariant MSO: it is sufficient to prove that the given class has the unravelling property.

▶ **Theorem 2.11.** *A class $\mathcal{C}$ of transition systems has the unravelling property if, and only if, over $\mathcal{C}$ bisimulation-invariant* MSO *coincides with* L_μ.

Let us also note the following result, which allows us to extend the unravelling property from a given class to certain superclasses.

▶ **Lemma 2.12.** *Let $\mathcal{C}_0 \subseteq \mathcal{C}$ be classes such that every system in $\mathcal{C}$ is bisimilar to one in $\mathcal{C}_0$. If $\mathcal{C}_0$ has the unravelling property, then so does $\mathcal{C}$.*

3 Composition lemmas

We have mentioned above that automata-theoretic methods have so far been unsuccessful at attacking the finite version of the Janin–Walukiewicz result. Therefore, we rely on the composition method instead. Let us recall how this method works.

▶ **Definition 3.1.** Let $\mathfrak{S}$ and $\mathfrak{T}$ be transition systems (or general structures) and $m < \omega$ a number. The *m-theory* $\mathrm{Th}_m(\mathfrak{S})$ of $\mathfrak{S}$ is the set of all MSO-formulae of quantifier-rank m that are satisfied by $\mathfrak{S}$. (The quantifier-rank of a formula is its nesting depths of (first-order and second-order) quantifiers.) We write

$$\mathfrak{S} \equiv_m \mathfrak{T} \quad :\text{iff} \quad \mathrm{Th}_m(\mathfrak{S}) = \mathrm{Th}_m(\mathfrak{T}) .$$
⌟

Roughly speaking the composition method provides some machinery that allows us to compute the m-theory of a given transition system by breaking it down into several components and looking at the m-theories of these components separately. This approach is based on the realisation that several operations on transition systems are compatible with m-theories in the sense that the m-theory of the result can be computed from the m-theories of the arguments. Statements to that effect are known as *composition theorems.* For an overview we refer the reader to [3] and [11]. The following basic operations and their composition theorems will be used below. We start with disjoint unions.

▶ **Definition 3.2.** The *disjoint union* of two structures $\mathfrak{A} = \langle A, R_0^{\mathfrak{A}}, \dots, R_m^{\mathfrak{A}} \rangle$ and $\mathfrak{B} = \langle B, R_0^{\mathfrak{B}}, \dots, R_m^{\mathfrak{B}} \rangle$ is the structure

$$\mathfrak{A} \oplus \mathfrak{B} := \big\langle A \cup B,\ R_0^{\mathfrak{A}} \cup R_0^{\mathfrak{B}}, \dots,\ R_m^{\mathfrak{A}} \cup R_m^{\mathfrak{B}},\ \mathrm{Left},\ \mathrm{Right} \big\rangle$$

obtained by forming the disjoint union of the universes and relations of $\mathfrak{A}$ and $\mathfrak{B}$ and adding two unary predicates $\mathrm{Left} := A$ and $\mathrm{Right} := B$ that mark whether an element belongs to $\mathfrak{A}$ or to $\mathfrak{B}$. If $\mathfrak{A}$ and $\mathfrak{B}$ are transition systems, the initial state of $\mathfrak{A} \oplus \mathfrak{B}$ is that of $\mathfrak{A}$. ⌟

The corresponding composition theorem looks as follows. It can be proved by a simple induction on m.

▶ **Lemma 3.3.** $\mathfrak{A} \equiv_m \mathfrak{A}'$ *and* $\mathfrak{B} \equiv_m \mathfrak{B}'$ *implies* $\mathfrak{A} \oplus \mathfrak{B} \equiv_m \mathfrak{A}' \oplus \mathfrak{B}'$.

Two other operations we need are interpretations and fusion operations.

▶ **Definition 3.4.** An *interpretation* is an operation τ on structures that is given by a list $\langle \delta(x), (\varphi_R(\bar{x}))_{R \in \Sigma} \rangle$ of MSO-formulae. Given a structure $\mathfrak{A}$, it produces the structure $\tau(\mathfrak{A})$ whose universe consists of all elements of $\mathfrak{A}$ satisfying the formula δ and whose relations are those defined by the formulae φ_R. The *quantifier-rank* of an interpretation is the maximal quantifier-rank of a formula in the list. An interpretation is *quantifier-free* if its quantifier-rank is 0. ⌟

▶ **Lemma 3.5.** *Let τ be an interpretation of quantifier-rank k. Then*

$$\mathfrak{A} \equiv_{m+k} \mathfrak{A}' \quad \textit{implies} \quad \tau(\mathfrak{A}) \equiv_m \tau(\mathfrak{A}') .$$

▶ **Definition 3.6.** Let P be a predicate symbol. The *fusion operation* fuse_P merges in a given structure all elements of the set P into a single element, i.e., all elements of P are replaced by a single new element and all edges incident with one of the old elements are attached to the new one instead. ⌟

▶ **Lemma 3.7.** $\mathfrak{A} \equiv_m \mathfrak{A}'$ *implies* $\mathrm{fuse}_P(\mathfrak{A}) = \mathrm{fuse}_P(\mathfrak{A}')$.

Using the composition theorems for these basic operations we can prove new theorems for derived operations. As an example let us consider *pointed paths,* i.e., paths where both end-points are marked by special colours.

▶ **Definition 3.8.** We denote the *concatenation* of two paths $\mathfrak{A}$ and $\mathfrak{B}$ by $\mathfrak{A} + \mathfrak{B}$. And we write $\mathfrak{A}^\bullet$ for the expansion of a path $\mathfrak{A}$ by two new constants for the end-points. ⌟

▶ **Corollary 3.9.** *Let $\mathfrak{A}, \mathfrak{A}', \mathfrak{B}, \mathfrak{B}'$ be paths. Then $\mathfrak{A}^\bullet \equiv_m \mathfrak{A}'^\bullet$ and $\mathfrak{B}^\bullet \equiv_m \mathfrak{B}'^\bullet$ implies $(\mathfrak{A} + \mathfrak{B})^\bullet \equiv_m (\mathfrak{A}' + \mathfrak{B}')^\bullet$.*

Proof. As the end-points are given by constants, we can construct a quantifier-free interpretation τ mapping $\mathfrak{A}^\bullet \oplus \mathfrak{B}^\bullet$ to $(\mathfrak{A} + \mathfrak{B})^\bullet$. ◀

Note that, since the concatenation operation is associative, it in particular follows that the set of m-theories of paths forms a semigroup.

Finally let us mention one more involved operation with a composition theorem. Let $\mathfrak{S}$ be a transition system and $\mathfrak{C} \subseteq \mathfrak{S}$ a subsystem. We say that $\mathfrak{C}$ is *attached* at the state $s \in S$ if there is a unique edge (in either direction) between a state in $S \setminus C$ and a state in C and this edge leads from s to the initial state of $\mathfrak{C}$.

▶ **Proposition 3.10.** *Let $\mathfrak{S}$ be a (possibly infinite) transition system and let $\mathfrak{S}'$ be the system obtained from $\mathfrak{S}$ by replacing an arbitrary number of attached subsystems by subsystems with the same m-theories (as the corresponding replaced ones). Then $\mathfrak{S} \equiv_m \mathfrak{S}'$.*

For a finite system $\mathfrak{S}$ this statement can be proved in the same way as Corollary 3.9 by expressing $\mathfrak{S}$ as a disjoint union followed by a quantifier-free interpretation. For infinite systems, we need a more powerful version of the disjoint union operation called a *generalised sum* (see [16]).

As presented above these tools work with m-theories, which is not quite what we need since we have to also account for bisimulation-invariance. To do so we modify the definitions as follows.

▶ **Definition 3.11.** Let $\mathcal{C}$ be a class of transition systems and $m < \omega$ a number.

(a) We denote by $\simeq^m_{\mathcal{C}}$ the transitive closure of the union $\equiv_m \cup \sim$ restricted to the class $\mathcal{C}$. Formally, we define $\mathfrak{S} \simeq^m_{\mathcal{C}} \mathfrak{T}$ if there exist systems $\mathfrak{C}_0, \dots, \mathfrak{C}_n \in \mathcal{C}$ such that

$$\mathfrak{C}_0 = \mathfrak{S}\,, \quad \mathfrak{C}_n = \mathfrak{T}\,, \quad \text{and} \quad \mathfrak{C}_i \equiv_m \mathfrak{C}_{i+1} \quad \text{or} \quad \mathfrak{C}_i \sim \mathfrak{C}_{i+1}\,, \qquad \text{for all } i < n\,.$$

(b) We denote by $\mathrm{Th}^m_{\mathcal{C}}(\mathfrak{S})$ the set of all MSO-formulae of quantifier-rank m that are bisimulation-invariant over $\mathcal{C}$ and that are satisfied by $\mathfrak{S}$, and we define

$$\mathfrak{S} \equiv^m_{\mathcal{C}} \mathfrak{S}' \quad :\text{iff} \quad \mathrm{Th}^m_{\mathcal{C}}(\mathfrak{S}) = \mathrm{Th}^m_{\mathcal{C}}(\mathfrak{S}')\,.$$

We also set $\mathrm{TH}^m_{\mathcal{C}} := \{\, \mathrm{Th}^m_{\mathcal{C}}(\mathfrak{S}) \mid \mathfrak{S} \in \mathcal{C} \,\}$. ⌟

Note that, up to logical equivalence, there are only finitely many formulae of a given quantifier-rank. Hence, each set $\mathrm{TH}^m_{\mathcal{C}}$ is finite and the relations $\equiv_m$, $\equiv^m_{\mathcal{C}}$ and $\simeq^m_{\mathcal{C}}$ have finite index.

▶ **Lemma 3.12.** *If φ is a MSO-formula of quantifier-rank m that is bisimulation-invariant over $\mathcal{C}$, then $\mathfrak{S} \simeq^m_{\mathcal{C}} \mathfrak{T}$ implies $\mathfrak{S} \models \varphi \Leftrightarrow \mathfrak{T} \models \varphi$.*

Some of the above composition theorems also hold for the relation $\simeq^m_{\mathcal{C}}$. This is immediate if the operation in question also preserves bisimilarity. We mention only two such results. The second one will be needed below.

▶ **Lemma 3.13.** *Let $\mathcal{C}$ be a class that is closed under disjoint unions.*

$$\mathfrak{A} \simeq^m_{\mathcal{C}} \mathfrak{A}' \quad \textit{and} \quad \mathfrak{B} \simeq^m_{\mathcal{C}} \mathfrak{B}' \quad \textit{implies} \quad \mathfrak{A} \oplus \mathfrak{B} \simeq^m_{\mathcal{C}} \mathfrak{A}' \oplus \mathfrak{B}'\,.$$

▶ **Proposition 3.14.** *Let $\mathcal{C}$ and $\mathcal{D}$ be two classes, $\mathfrak{S} \in \mathcal{C}$ a (possibly infinite) transition system and let $\mathfrak{S}'$ be the system obtained from $\mathfrak{S}$ by replacing an arbitrary number of attached subsystems by subsystems which are $\simeq^m_{\mathcal{D}}$-equivalent. Then $\mathfrak{S} \simeq^m_{\mathcal{C}} \mathfrak{S}'$ provided that the class $\mathcal{C}$ is closed under the operation of replacing attached subsystems in $\mathcal{D}$.*

4 Types

Our strategy to prove the unravelling property for a class $\mathcal{C}$ is as follows. For every quantifier-rank m, we assign to each tree $\mathfrak{T}$ a so-called *m-type* $\tau_m(\mathfrak{T})$. We choose the functions τ_m such that we can compute the theory $\mathrm{Th}^m_{\mathcal{C}}(\mathfrak{C})$ of a system $\mathfrak{C} \in \mathcal{C}$ from the m-type $\tau_m(\mathcal{U}(\mathfrak{C}))$ of its unravelling. Furthermore, we need to find MSO-formulae checking whether a tree has a given m-type. The formal definition is as follows.

▶ **Definition 4.1.** Let $\mathcal{C}$ be a class of transition systems and $\mathcal{T}$ the class of all trees.

(a) A *family of type functions* for $\mathcal{C}$ is a family of functions $\tau_m : \mathcal{T} \to \Theta_m$, for $m < \omega$, where the co-domains Θ_m are finite sets and each τ_m satisfies the following two axioms.

(S1) $\tau_m(\mathcal{U}(\mathfrak{C})) = \tau_m(\mathcal{U}(\mathfrak{C}'))$ implies $\mathrm{Th}^m_{\mathcal{C}}(\mathfrak{C}) = \mathrm{Th}^m_{\mathcal{C}}(\mathfrak{C}')$, for $\mathfrak{C}, \mathfrak{C}' \in \mathcal{C}$.

(S2) $\mathfrak{T} \sim \mathfrak{T}'$ implies $\tau_m(\mathfrak{T}) = \tau_m(\mathfrak{T}')$, for all $\mathfrak{T}, \mathfrak{T}' \in \mathcal{T}$.

(b) A family $(\tau_m)_m$ of type functions is *definable* if, for every $\theta \in \Theta_m$, there exists an MSO-formula ψ_θ such that

(S3) $\mathfrak{T} \models \psi_\theta$ iff $\tau_m(\mathfrak{T}) = \theta$, for all trees $\mathfrak{T}$. ⌟

Let us start by showing how to prove the unravelling property using type functions. The following characterisation theorem can be considered to be the main theoretical result of this article.

▶ **Theorem 4.2.** *Let $\mathcal{C}$ be a class of transition systems and $\mathcal{T}$ the class of all trees. The following statements are equivalent.*

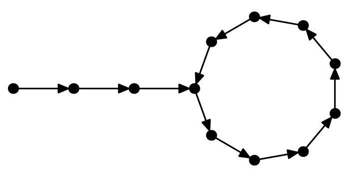
(1) *Over $\mathcal{C}$, bisimulation-invariant* MSO *coincides with* L_μ.
(2) *$\mathcal{C}$ has the unravelling property.*
(3) *There exists a definable family $(\tau_m)_m$ of type functions for $\mathcal{C}$.*
(4) *There exist functions $g : \omega \to \omega$ and $h_m : \mathrm{TH}^{g(m)}_{\mathcal{T}} \to \mathrm{TH}^m_{\mathcal{C}}$, for $m < \omega$, such that*

$$h_m\big(\mathrm{Th}^{g(m)}_{\mathcal{T}}(\mathcal{U}(\mathfrak{C}))\big) = \mathrm{Th}^m_{\mathcal{C}}(\mathfrak{C})\,, \quad \text{for all } \mathfrak{C} \in \mathcal{C}$$

(in other words, the $g(m)$-theory of $\mathcal{U}(\mathfrak{C})$ determines the m-theory of $\mathfrak{C}$).

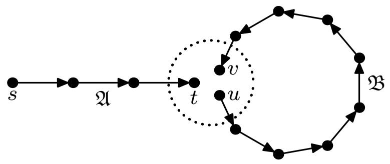
5 Lassos

As an application of type functions, we consider a very simple example, the class of *lassos*. Our proof is based on more or less the same arguments as that by Dawar and Janin [6], just the presentation differs. A lasso is a transition system consisting of a directed path ending in a cycle.



We allow the borderline cases where the initial path has length 0 or the cycle consists of only a single edge.

To define the type of a lasso, note that we can construct every lasso $\mathfrak{L}$ from two finite paths $\mathfrak{A}$ and $\mathfrak{B}$ by identifying three of their end-points.



The paths $\mathfrak{A}$ and $\mathfrak{B}$ are uniquely determined by $\mathfrak{L}$. We will refer to $\mathfrak{A}$ as the *tail* of the lasso and to $\mathfrak{B}$ as the *loop*. We introduce two kinds of types for lassos, a strong one and a weak one.

▶ **Definition 5.1.** The *strong m-type* of a lasso $\mathfrak{L}$ with tail $\mathfrak{A}$ and loop $\mathfrak{B}$ is the pair

$$\mathrm{stp}_m(\mathfrak{L}) := \langle \alpha, \beta \rangle\,, \quad \text{where} \quad \alpha := \mathrm{Th}_m(\mathfrak{A}^\bullet) \quad \text{and} \quad \beta := \mathrm{Th}_m(\mathfrak{B}^\bullet)\,. \quad ⌟$$

The strong m-type of a lasso uniquely determines its m-theory.

▶ **Lemma 5.2.** *Let $\mathfrak{L}_0$ and $\mathfrak{L}_1$ be lassos.*

$$\mathrm{stp}_m(\mathfrak{L}_0) = \mathrm{stp}_m(\mathfrak{L}_1) \quad \textit{implies} \quad \mathfrak{L}_0 \equiv_m \mathfrak{L}_1\,.$$

The problem with the strong type of a lasso $\mathfrak{L}$ is that we cannot recover it from the unravelling of $\mathfrak{L}$ as the decomposition of $\mathcal{U}(\mathfrak{L})$ into the parts of $\mathfrak{L}$ is uncertain. Therefore we introduce another notion of a type where this recovery is possible. For this we recall some facts from the theory of ω-semigroups.

Recall that we have noted in Corollary 3.9 that the m-theories of pointed paths form a finite semigroup with respect to concatenation. Furthermore, every element a of a finite semigroup has an *idempotent power* a^π, which is defined as the value a^n where n is the least natural number such that $a^n \cdot a^n = a^n$.

▶ **Definition 5.3.** (a) A *factorisation* of an infinite path $\mathfrak{A}$ is a sequence $(\mathfrak{A}_i)_{i<\omega}$ of finite paths whose concatenation is $\mathfrak{A}$. Such a factorisation has *m-type* $\langle \alpha, \beta \rangle$ if

$$\alpha := \mathrm{Th}_m(\mathfrak{A}_0^\bullet) \quad \text{and} \quad \beta := \mathrm{Th}_m(\mathfrak{A}_i^\bullet)\,, \quad \text{for } i > 0\,.$$

(b) Two pairs $\langle \alpha, \beta \rangle$ and $\langle \gamma, \delta \rangle$ of m-theories are *conjugate* if there are m-theories ξ and η such that

$$\gamma\delta^\pi = \alpha\beta^\pi\xi\,, \quad \beta^\pi = \xi\eta\,, \quad \text{and} \quad \delta^\pi = \eta\xi\,.$$

Being conjugate is an equivalence relation. We denote the equivalence class of a pair $\langle \alpha, \beta \rangle$ by $[\alpha, \beta]$.

(c) The *weak m-type* of a lasso $\mathfrak{L}$ with parts $\mathfrak{A}$ and $\mathfrak{B}$ is

$$\mathrm{wtp}_m(\mathfrak{L}) := [\alpha, \beta]\,, \quad \text{where} \quad \alpha := \mathrm{Th}_m(\mathfrak{A}^\bullet) \quad \text{and} \quad \beta := \mathrm{Th}_m(\mathfrak{B}^\bullet)\,.$$

(d) The *m-type* of an infinite tree $\mathfrak{T}$ is

$$\tau_m(\mathfrak{T}) := [\alpha, \beta]\,,$$

where α and β is an arbitrary pair of m-theories such that every branch of $\mathfrak{T}$ has a factorisation of m-type $\langle \alpha, \beta \rangle$. If there is no such pair, we set $\tau_m(\mathfrak{T}) := \bot$. ⌟

▶ **Lemma 5.4.** *Let $\mathcal{L}$ be the class of all lassos and let $\mathfrak{L}_0, \mathfrak{L}_1 \in \mathcal{L}$.*

$$\mathrm{wtp}_m(\mathfrak{L}_0) = \mathrm{wtp}_m(\mathfrak{L}_1) \quad \textit{implies} \quad \mathfrak{L}_0 \simeq^m_{\mathcal{L}} \mathfrak{L}_1 .$$

To show that the functions $(\tau_m)_m$ form a family of type functions, we need the following standard facts about factorisations and their types (see, e.g., Section II.2 of [14]).

▶ **Proposition 5.5.** *Let $\mathfrak{A}$ be an infinite path.*

(a) *$\mathfrak{A}$ has a factorisation of type $\langle \alpha, \beta \rangle$, for some α and β.*
(b) *If $\mathfrak{A}$ has factorisations of type $\langle \alpha, \beta \rangle$ and $\langle \gamma, \delta \rangle$, then $\langle \alpha, \beta \rangle$ and $\langle \gamma, \delta \rangle$ are conjugate.*

Note that these two statements imply in particular that the type $\tau_m(\mathfrak{T})$ of a tree $\mathfrak{T}$ is well-defined.

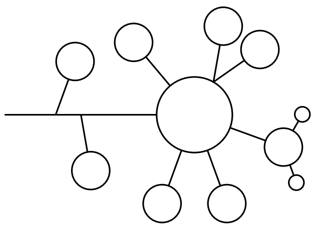
▶ **Lemma 5.6.** *The functions $(\tau_m)_m$ defined above form a definable family of type functions for the class of all lassos.*

By Theorem 4.2, it therefore follows that the class of lassos has the unravelling property.

▶ **Theorem 5.7.** *The class of all lassos has the unravelling property.*

6 Hierarchical Lassos

After the simple example in the previous section, let us give a more substantial application of the type machinery. We consider *hierarchical* (or *nested*) lassos. These are obtained from a lasso by repeatedly attaching sublassos to some states. More precisely, a 1-lasso is just an ordinary lasso, while inductively a $(k + 1)$-lasso is obtained from a k-lasso by attaching one or more lassos to some of the states. (Each state may have several sublassos attached.)



Alternatively, we can obtain a $(k + 1)$-lasso $\mathfrak{M}$ from a 1-lasso $\mathfrak{L}$ by attaching k-lassos. We will call this lasso $\mathfrak{L}$ the *main lasso* of $\mathfrak{M}$.

The types we use for k-lassos are based on the same principles as those for simple lassos, but we have to nest them in order to take the branching of a hierarchical lasso into account.

▶ **Definition 6.1.** Let $t : \mathrm{dom}(t) \to C$ be a labelled tree and $m < \omega$.
(a) For a branch β of t, we set

$$\mathrm{wtp}_m(\beta) := [\sigma, \tau] ,$$

if β has a factorisation of m-type $\langle \sigma, \tau \rangle$. (By Proposition 5.5, this is well-defined.)
(b) For $k < \omega$, we define

$$\mathrm{tp}^0_m(t) := \{\, \mathrm{wtp}_m(\beta) \mid \beta \text{ a branch of } t \,\} ,$$
$$\mathrm{tp}^{k+1}_m(t) := \mathrm{tp}^0_m(\mathrm{TP}^k_m(t)) ,$$

where $\mathrm{TP}^k_m(t) : T \to C \times \mathcal{P}(\Theta^k_m)$ is the tree with labelling

$$\mathrm{TP}^k_m(t)(v) := \big\langle t(v),\ \{\, \mathrm{tp}^k_m(t|_u) \mid u \text{ a successor of } v \,\}\big\rangle . \quad ⌟$$

We will prove that the functions tp^k_m form a family of type functions. Note that it follows immediately from the definition that they satisfy Properties (S2) and (S3). Hence, it only remains to check (S1).

▶ **Lemma 6.2.** (a) *Let $\mathfrak{M}$ be a k-lasso and $\mathfrak{N}$ a k'-lasso. Then*

$$\mathcal{U}(\mathfrak{M}) \sim \mathcal{U}(\mathfrak{N}) \quad \textit{implies} \quad \mathrm{tp}^k_m(\mathfrak{M}) = \mathrm{tp}^k_m(\mathfrak{N})\,.$$

(b) *For every type τ, there exists an* MSO*-formula φ such that*

$$\mathcal{U}(\mathfrak{M}) \models \varphi \quad \text{iff} \quad \mathrm{tp}^k_m(\mathfrak{M}) = \tau\,.$$

Thus, to prove that the class of k-lassos has the unravelling property it is sufficient to show that tp^k_m also satisfies Property (S1). We will do so by induction on k. The base case of this induction rests on the following lemma.

▶ **Lemma 6.3.** *Let $\mathcal{L}_k$ be the class of all k-lassos and let $\mathfrak{M}$ be a k-lasso such that, for every vertex v and all branches β and γ starting at a successor of v, we have $\mathrm{wtp}_m(\beta) = \mathrm{wtp}_m(\gamma)$. Then $\mathfrak{M} \simeq^m_{\mathcal{L}_k} \mathfrak{N}$, for some 1-lasso $\mathfrak{N}$.*

▶ **Proposition 6.4.** *Let $\mathfrak{M}$ be a k-lasso and $\mathfrak{N}$ a k'-lasso. For $m \geq 1$,*

$$\mathrm{tp}^k_m(\mathfrak{M}) = \mathrm{tp}^k_m(\mathfrak{N}) \quad \textit{implies} \quad \mathfrak{M} \simeq^m_{\mathcal{L}_K} \mathfrak{N}\,,$$

where $\mathcal{L}_K$ is the class of all K-lassos with $K := \max(k, k')$.

Using Theorem 4.2 we now immediately obtain the following statement.

▶ **Theorem 6.5.** *For every k, the class of all k-lassos has the unravelling property.*

7 Reductions

We would like to define reductions that allow us to prove that a certain class has the unravelling property when we already know that some other class has this property. To do so, we encode every transition system of the first class by some system in the second one. The main example we will be working with is a function ϱ that removes certain attached subsystems and uses additional vertex labels to remember the m-theories of all deleted system. Up to equivalence of m-theories, we can undo this operation by a function η that attaches to each vertex labelled by some m-theory θ some fixed system with theory θ. Let us give a general definition of such pairs of maps.

▶ **Definition 7.1.** Let $\mathcal{C}$ and $\mathcal{D}$ be classes of transition systems and $k, m < \omega$. A function $\varrho : \mathcal{C} \to \mathcal{D}$ is a (k,m)-*encoding map* if there exists a function $\eta : \mathcal{D} \to \mathcal{C}$ such that

(E1) $\varrho(\eta(\mathfrak{D})) \simeq^k_{\mathcal{D}} \mathfrak{D}\,, \quad$ for all $\mathfrak{D} \in \mathcal{D}\,.$
(E2) $\varrho(\mathfrak{C}) \simeq^k_{\mathcal{D}} \varrho(\mathfrak{C}') \quad$ implies $\quad \mathfrak{C} \simeq^m_{\mathcal{C}} \mathfrak{C}'\,, \quad$ for all $\mathfrak{C}, \mathfrak{C}' \in \mathcal{C}\,.$

In this case, we call the function η a (k,m)-*decoding map* for ϱ. ⌟

These two axioms imply dual axioms with the functions ϱ and η exchanged.

▶ **Lemma 7.2.** *Let $\eta : \mathcal{D} \to \mathcal{C}$ be a (k,m)-decoding map for $\varrho : \mathcal{C} \to \mathcal{D}$.*

(E3) $\eta(\varrho(\mathfrak{C})) \simeq^m_{\mathcal{C}} \mathfrak{C}\,, \quad$ *for all* $\mathfrak{C} \in \mathcal{C}\,.$
(E4) $\mathfrak{D} \simeq^k_{\mathcal{D}} \mathfrak{D}' \quad$ *implies* $\quad \eta(\mathfrak{D}) \simeq^m_{\mathcal{C}} \eta(\mathfrak{D}')\,, \quad$ *for all* $\mathfrak{D}, \mathfrak{D}' \in \mathcal{D}\,.$

The axioms of an encoding map were chosen to guarantee the property stated in the following lemma. It will be used below to prove that encoding maps can be used to transfer the unravelling property from one class to another.

▶ **Lemma 7.3.** *Let $\varrho : \mathcal{C} \to \mathcal{D}$ a (k, m)-encoding map and $\eta : \mathcal{D} \to \mathcal{C}$ a (k, m)-decoding map for ϱ. For every* MSO*-formula φ of quantifier-rank m that is bisimulation-invariant over $\mathcal{C}$, there exists an* MSO*-formula $\hat{\varphi}$ of quantifier-rank k that is bisimulation-invariant over $\mathcal{D}$ such that*

$$\mathfrak{C} \models \varphi \quad \text{iff} \quad \varrho(\mathfrak{C}) \models \hat{\varphi}\,, \quad \textit{for all } \mathfrak{C} \in \mathcal{C}\,.$$

It remains to show how to use encoding maps to transfer the unravelling property. Just the existence of such a map is not sufficient. It also has to be what we call definable.

▶ **Definition 7.4.** Let $\mathcal{C}$ be a class of transition systems.

(a) A (k, m)-encoding map $\varrho : \mathcal{C} \to \mathcal{D}$ is *definable* if, for every MSO-formula φ that is bisimulation-invariant over trees, there exists an MSO-formula $\hat{\varphi}$ that is bisimulation-invariant over trees such that

$$\mathcal{U}(\varrho(\mathfrak{C})) \models \varphi \quad \text{iff} \quad \mathcal{U}(\mathfrak{C}) \models \hat{\varphi}\,, \quad \text{for all } \mathfrak{C} \in \mathcal{C}\,.$$

(b) We say that $\mathcal{C}$ is *reducible* to a family $(\mathcal{D}_m)_{m<\omega}$ of classes if there exist a map $g : \omega \to \omega$ and, for each $m < \omega$, functions $\varrho_m : \mathcal{C} \to \mathcal{D}_m$ and $\eta_m : \mathcal{D}_m \to \mathcal{C}$ such that ϱ_m is a definable $(g(m), m)$-encoding map and η_m a corresponding $(g(m), m)$-decoding map. ⌟

(The only reason why we use a family of classes to reduce to, instead of a single one is so that we can have the labellings of systems in $\mathcal{D}_m$ depend on the quantifier-rank m.)

▶ **Theorem 7.5.** *Suppose that $\mathcal{C}$ is reducible to $(\mathcal{D}_m)_{m<\omega}$. If every class $\mathcal{D}_m$ has the unravelling property, so does $\mathcal{C}$.*

8 Finite Cantor–Bendixson rank

One common property of k-lassos is that the trees we obtain by unravelling them all have finite Cantor–Bendixson rank. In this section we will generalise our results to cover transition systems with this more general property. The proof below consists in a two-step reduction to the class of k-lassos.

▶ **Definition 8.1.** Let $\mathfrak{T}$ be a finitely branching tree. The *Cantor–Bendixson derivative* of $\mathfrak{T}$ is the tree $\mathfrak{T}'$ obtained from $\mathfrak{T}$ by removing all subtrees that have only finitely many infinite branches. The *Cantor-Bendixson rank* of a tree $\mathfrak{T}$ is the least ordinal α such that applying $\alpha + 1$ Cantor–Bendixson derivatives to $\mathfrak{T}$ results in an empty tree. The *Cantor–Bendixson rank* of a transition system $\mathfrak{S}$ is equal to the Cantor–Bendixson rank of its unravelling. ⌟

We can go from the class of k-lassos to that of systems with bounded Cantor–Bendixson rank in two steps.

▶ **Definition 8.2.** (a) A transition system is a *generalised k-lasso* if it is obtained from a finite tree by attaching (one or several) k-lassos to every leaf.

(b) A transition system $\mathfrak{T}$ is a *tree extension* of $\mathfrak{S}$ if $\mathfrak{T}$ is obtained from $\mathfrak{S}$ by attaching an arbitrary number of finite trees to some of the vertices. ⌟

With these two notions we can characterise the property of having bounded Cantor–Bendixson rank as follows.

▶ **Proposition 8.3.** *Let $\mathfrak{S}$ be a finite transition system.*

(a) *For every $k < \omega$, the following statements are equivalent.*

(1) *$\mathfrak{S}$ has Cantor–Bendixson rank at most k.*
(2) *$\mathfrak{S}$ is bisimilar to a tree extension of a generalised $(k+1)$-lasso.*

(b) *The following statements are equivalent.*

(1) *$\mathfrak{S}$ has finite Cantor–Bendixson rank.*
(2) *$\mathfrak{S}$ is bisimilar to a tree extension of a generalised k-lasso, for some $k < \omega$.*
(3) *Every strongly connected component of $\mathfrak{S}$ is either a singleton or a cycle.*

To prove the unravelling property for the transition systems of bounded Cantor–Bendixson rank, we proceed in two steps. First we consider generalised k-lassos and then their tree extensions.

▶ **Theorem 8.4.** *For fixed k, the class of all generalised k-lassos has the unravelling property.*

Using this intermediate step, we obtain the following proof for transition systems with bounded Cantor–Bendixson rank.

▶ **Theorem 8.5.** *The class of all finite transition systems of Cantor–Bendixson rank at most k has the unravelling property.*

▶ **Corollary 8.6.** *Over the class of all finite transition systems with Cantor–Bendixson rank at most k, bisimulation-invariant* MSO *coincides with* L_μ.

9 Conclusion

We have shown in several simple examples how to characterise bisimulation-invariant MSO in the finite. In particular, we have proved that it coincides with L_μ over

- every finite class (Theorem 2.8),
- the class of all finite trees (Theorem 2.9),
- the classes of all lassos, k-lassos, and generalised k-lassos (Theorems 5.7, 6.5, and 8.4),
- the class of all systems of Cantor–Bendixson rank at most k (Theorem 8.5).

Our main tool in these proofs was the unravelling property (Theorem 2.11). It will be interesting to see how far our methods can be extended to more complicated classes. For instance, can they be used to prove the following conjecture?

Conjecture. *If a class $\mathcal{C}$ of transition systems has the unravelling property, then so does the class of all subdivisions of systems in $\mathcal{C}$.*

A good first step seems to be the class of all finite transition systems that have Cantor–Bendixson rank k, for some $k < \omega$ that is not fixed.

In this paper we have considered only transition systems made out of paths with very limited branching. To extend our techniques to classes allowing for more branching seems to require new ideas. A simple test case that looks promising is the class of systems with a 'lasso-decomposition' of width k, i.e., something like a tree decomposition but where the pieces are indexed by a lasso instead of a tree.

References

1. H. Andréka, J. v. Benthem, and I. Németi, *Modal languages and bounded fragments of predicate logic*, Journal of Philosophical Logic, 27 (1998), pp. 217–274.
2. J. v. Benthem, *Modal Correspondence Theory*, Ph. D. Thesis, University of Amsterdam, Amsterdam, 1976.
3. A. Blumensath, T. Colcombet, and C. Löding, *Logical Theories and Compatible Operations*, in Logic and Automata: History and Perspectives, J. Flum, E. Grädel, and T. Wilke, eds., Amsterdam University Press, 2007, pp. 73–106.
4. F. Carreiro, *Fragments of Fixpoint Logics*, PhD Thesis, Institute for Logic, Language and Computation, Amsterdam, 2015.
5. I. Ciardelli and M. Otto, *Bisimulation in Inquisitive Modal Logic*, in Proc. 16th Conference on Theoretical Aspects of Rationality and Knowledge, TARK 2017, 2017, pp. 151–166.
6. A. Dawar and D. Janin, *On the bisimulation invariant fragment of monadic Σ_1 in the finite*, in Proc. of the 24th Int. Conf. on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2004, 2004, pp. 224–236.
7. E. Grädel, C. Hirsch, and M. Otto, *Back and Forth Between Guarded and Modal Logics*, ACM Transactions on Computational Logics, (2002), pp. 418–463.
8. E. Grädel, W. Thomas, and T. Wilke, *Automata, Logic, and Infinite Games*, LNCS 2500, Springer-Verlag, 2002.
9. C. Hirsch, *Guarded Logics: Algorithms and Bisimulation*, Ph. D. Thesis, RWTH Aachen, Aachen, 2002.
10. D. Janin and I. Walukiewicz, *On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic*, in Proc. of the 7th International Conference on Concurrency Theory, CONCUR 1996, 1996, pp. 263–277.
11. J. A. Makowsky, *Algorithmic aspects of the Feferman-Vaught Theorem*, Annals of Pure and Applied Logic, 126 (2004), pp. 159–213.
12. F. Moller and A. Rabinovitch, *On the expressive power of CTL**, in Proc. 14th IEEE Symp. on Logic in Computer Science, LICS, 1999, pp. 360–369.
13. ———, *Counting on CTL*: on the expressive power of monadic path logic*, Information and Computation, 184 (2003), pp. 147–159.
14. D. Perrin and J.-É. Pin, *Infinite Words – Automata, Semigroups, Logic and Games*, Elsevier, 2004.
15. E. Rosen, *Modal logic over finite structures*, Journal of Logic, Language and Information, 6 (1997), pp. 427–439.
16. S. Shelah, *The Monadic Second Order Theory of Order*, Annals of Mathematics, 102 (1975), pp. 379–419.
17. C. Stirling, *Bisimulation and logic*, in Advanced topics in Bisimulation and Coinduction, D. Sangiorgi and J. Rutten, eds., Cambridge University Press, 2011, pp. 172–196.