Modal μ-calculus with Atoms*

Bartek Klin and Mateusz Łełyk

University of Warsaw

Abstract

We introduce an extension of modal μ-calculus to sets with atoms and study its basic properties. Model checking is decidable on orbit-finite structures, and a correspondence to parity games holds. On the other hand, satisfiability becomes undecidable. We also show some limitations to the expressiveness of the calculus and argue that a naive way to remove these limitations results in a logic whose model checking is undecidable.

1998 ACM Subject Classification F.4.1 Mathematical Logic–Temporal logic, D.2.4 Software/-Program Verification–Model checking

Keywords and phrases modal μ-calculus, sets with atoms

Digital Object Identifier 10.4230/LIPIcs.CSL.2017.30

1 Introduction

Modal μ-calculus [1, 4, 5, 29] is perhaps the best known formalism for describing properties of labeled transition systems or Kripke models. It combines a simple syntax with a mathematically elegant semantics and it is expressive enough to specify many interesting properties of systems. For example, the property "[in the current state] the predicate p holds, and there exists a transition path where it holds again some time in the future" is defined by a μ-calculus formula:

$$p \wedge \Diamond \mu X.(p \vee \Diamond X).$$

Other similar formalisms, such as the logic CTL* [17], can be encoded in the modal μ-calculus. However, decision problems such as model checking ("does a given formula hold in a given (finite) model?") or satisfiability ("does a given formula hold in some model?") are decidable.

Formulas of the μ-calculus are built over some fixed set of basic predicates, such as p above, whose semantics is provided in every model. In principle the set of such basic predicates may be infinite, but this generality is hardly useful: since a formula of the μ-calculus is a finite object, it may only refer to finitely many predicates.

In modeling systems, this finiteness may sometimes seem restrictive. Real systems routinely operate on data coming from potentially infinite domains, such as numbers or character strings. Basic predicates observed about a system may reasonably include ones like "a number n was input", denoted here p_n, for every number n. If one considers properties such as "there exists a transition path where the currently input number is input again some time in the future", one is in trouble writing finite formulas to define them. Were infinitary connectives allowed in the formalism, the formula

$$\bigvee_{n \in \mathbb{N}} (p_n \wedge \Diamond \mu X.(p_n \vee \Diamond X))$$

* This work is supported by Poland's National Science Centre grant 2012/07/B/ST6/01497.

would do the job; however, few good properties of the μ-calculus transport to a naively construed infinitary setting. In particular, an obvious obstruction to any kind of decidability results is that a general infinitary formula is not readily presented as input to an algorithm.

In this paper we introduce μ-calculus with atoms, where infinitary propositional connectives are allowed in a restricted form that includes formulas such as the one above. Roughly speaking, basic properties in formulas, and indeed whole Kripke models, are assumed to be built of *atoms* from a fixed infinite set. Connectives in μ-formulas are indexed by sets that are potentially infinite but are finitely definable in terms of atoms. This makes formulas finitely presentable.

Several basic properties of the standard μ-calculus hold in the atom-based setting. Syntax and semantics of the calculus is defined in a standard way. The model checking problem remains decidable, although this is not trivial, as formulas and models are now, strictly speaking, infinite. A correspondence between μ-formulas and parity games with atoms holds.

Some other properties of the classical calculus fail. The satisfiability problem becomes undecidable. The atomic μ-calculus also turns out to be less expressive than might be expected. In particular, the property "there exists a path where no basic predicate holds more than once", although decidable, is not definable. This means that, unlike in the classical setting, an atomic extension of CTL* (where explicit quantification over paths makes such properties easy to define) is not a fragment of the atomic μ-calculus. As it turns out, for atomic CTL* even the model-checking problem is undecidable.

Our approach is a part of a wider programme of extending various computational models to sets with atoms, also known as nominal sets [34]. For example, in [2] the classical notion of finite automaton was reinterpreted in the universe of sets with atoms, with the result related to register automata [21, 31], an established model of automata over infinite alphabets.

Temporal logics over structures extended with data from infinite domains, and their connections to various types of automata, have been extensively studied in the literature. For example, the linear time logic LTL has been extended with a *freeze quantifier* [15, 16, 20, 35] which, for structures where every position is associated with a single data item, can store the current item for future reference. This can serve as a mechanism for detecting repeated data values (see also [13, 14]). Another known idea is to extend temporal logics with local constraints over data from a fixed infinite domain, see e.g. [7, 12]. In [19, 25], alternating register automata on data words and trees were studied, closely connected to μ-calculi.

In all these works, the main goal is to study the decidability border for satisfiability (or nonemptiness, in the case of automata). To that end, the authors impose various, sometimes complex restrictions on their logics or automata, and inevitably limit their expressiveness to some extent. In contrast to that, our aim is to lift the classical modal μ-calculus to data-equipped structures in the most syntactically economic way, and to achieve an expressive formalism. As a price for this, the satisfiability problem quickly becomes undecidable. However, we believe that this does not disqualify the atomic μ-calculus as a practical formalism: most applications of temporal logics in system verification rely only on solving the model checking problem, and that remains decidable here on a wide class of structures.

It would be easy to give up even the decidability of model checking. This was done in [33], in a setting very similar to ours, by extending a basic multimodal logic with infinitary boolean connectives subject only to a finite support condition. The resulting logic has few good properties except its huge expressive power (indeed, it can immediately encode our atomic μ-calculus, and much more): its set of formulas is not even countable. Instead, our boolean connectives are subject to a more restrictive condition of orbit finiteness, an idea first used in [3] in the context of first-order logic with atoms.

Another branch of related work is centered around algebras of name-passing processes such as the π-calculus, and variants of the μ-calculus aimed at specific transition systems induced by those algebras. This line of work was started in [10], where a version of the μ-calculus for model checking properties of π-calculus processes was proposed, with a sound-and-complete proof system. Other efforts in this direction, resulting in logics fine-tuned to specific infinite models, include [18, 30], and a more abstract calculus was proposed in [11]. Finally, [23, 24] introduced a μ-calculus extended with predicates over arbitrary infinite data domains, with no hope for any general decidability results and with a focus on pragmatic usability.

The structure of this paper is as follows. In Sect. 2 we briefly recall the modal μ-calculus and related logics in the classical setting. In Sect. 3 we recall the basics of sets with atoms, including the notions of finite support and orbit finiteness. In Sect. 4 we introduce the syntax and semantics of the atomic μ-calculus. In Sect. 5 we prove that model-checking is decidable, and study a correspondence with parity games. In Sect. 6 we prove various undecidability results, and in Sect. 7 we study the undefinable path property mentioned above. A list of future work directions is in Sect. 8.

Acknowledgments. We are very grateful to A. Facchini for collaboration in initial stages of this work, to M. Bojańczyk, S. Lasota, S. Toruńczyk and B. Wcisło for valuable discussions, and to anonymous reviewers for their thorough work.

2 μ-calculus and related logics

To fix the notation and terminology, we begin by recalling basic definitions and properties of the μ-calculus in the classical setting. For a more detailed exposition see e.g. [1, 4–6, 36, 38].

▶ **Definition 2.1** (Syntax). Let $\mathbb{P}$ be an infinite set of *basic propositions* and $\mathbb{X}$ an infinite set of *variables*. The set $\mathcal{L}_\mu$ of μ-calculus formulas is generated by the grammar:

$$\phi ::= p \mid X \mid \phi \vee \phi \mid \neg\phi \mid \Diamond\phi \mid \mu X.\phi$$

where p ranges over $\mathbb{P}$ and X over $\mathbb{X}$. We only allow formulas $\mu X.\phi$ where X occurs only positively in ϕ (i.e. under an even number of negations).

We put $\top = p \vee \neg p$, $\phi \wedge \psi := \neg(\neg\phi \vee \neg\psi)$, $\Box\phi := \neg\Diamond\neg\phi$, $\nu X.\phi := \neg\mu X.\neg\phi[X := \neg X]$.

▶ **Definition 2.2** (Kripke model). A *Kripke model* is a triple $\langle K, R, W\rangle$ such that
1. $\langle K, R\rangle$ is a directed graph, i.e., R is a binary relation on the set K of *states*,
2. W is a function from $\mathbb{P}$ to $\mathcal{P}(K)$.

If $\mathcal{K} = \langle K, R, W\rangle$ is a Kripke model and $k \in K$ then $\langle \mathcal{K}, k\rangle$ is called a *pointed Kripke model*.

▶ **Definition 2.3** (Semantics). Formulas of μ-calculus are interpreted in the context of a Kripke model $\mathcal{K} = \langle K, R, W\rangle$ and an environment, i.e., a partial function $\rho : \mathbb{X} \rightharpoonup \mathcal{P}(K)$. For any formula ϕ, assuming ρ is defined on all free variables in ϕ, the interpretation $[\![\phi]\!]_\rho \subseteq K$ is defined by induction:

- $[\![p]\!]_\rho = W(p)$,
- $[\![X]\!]_\rho = \rho(X)$,
- $[\![\neg\phi]\!]_\rho = K \setminus [\![\phi]\!]_\rho$,
- $[\![\phi \vee \psi]\!]_\rho = [\![\phi]\!]_\rho \cup [\![\psi]\!]_\rho$,
- $[\![\Diamond\phi]\!]_\rho = \{k \in K \mid \exists s \in [\![\phi]\!]_\rho.\ \langle k, s\rangle \in R\}$,
- $[\![\mu X.\phi]\!]_\rho = \bigcap\{L \subseteq K \mid [\![\phi]\!]_{\rho[X\mapsto L]} \subseteq L\}$,

where $\rho[X \mapsto L]$ denotes the function that maps X to L and acts as ρ on all other arguments.

We write $[\![\phi]\!]$ (or $[\![\phi]\!]_\mathcal{K}$ if $\mathcal{K}$ is less clear from context) instead of $[\![\phi]\!]_\rho$ if ρ is empty. We say that ϕ *holds* in a state $k \in K$ if $k \in [\![\phi]\!]$ and denote it $k \models \phi$.

▶ **Example 2.4.** The formula $\mu X.(p \vee \Diamond X)$ holds for pointed Kripke structures $(\mathcal{K}, k)$ such that from k there is a finite path to a state where p holds. The formula $\nu X.(\Box X \wedge \mu Y.(p \vee \Box Y))$ holds in those pointed Kripke structures where on every path p holds infinitely often.

For applications in system verification, the following two problems are considered:
Model checking: given a finite pointed Kripke model $\langle \mathcal{K}, k \rangle$ and a formula $\phi \in \mathcal{L}_\mu$, decide if $k \models \phi$.
Satisfiability: given a formula $\phi \in \mathcal{L}_\mu$, decide if there exists a pointed Kripke model $\langle \mathcal{K}, k \rangle$ s.t. $k \models \phi$.

It is very easy to see that model checking is decidable: in a finite Kripke model, one can compute the semantics of an $\mathcal{L}_\mu$-formula inductively, calculating least fixpoints using approximants. A more efficient procedure can be derived from a correspondence of μ-calculus with parity games (see e.g. [6, 38]).

Some well-known logics used in system verification can be translated into fragments of the μ-calculus. We give two important examples:

▶ **Definition 2.5** (CTL*)**.** In the logic CTL* we distinguish state formulas Φ and path formulas ϕ, formed according to the following grammar:

$$\Phi ::= p \mid \Phi \vee \Phi \mid \neg\Phi \mid \exists\phi \qquad\qquad \phi ::= \Phi \mid \phi \vee \phi \mid \neg\phi \mid \phi \mathsf{U} \phi \mid \mathsf{X}\phi$$

where p comes from some fixed set of propositional variables.

Standard notational conventions are $\forall\phi = \neg\exists\neg\phi$, $\phi \mathsf{R} \psi = \neg(\neg\phi \mathsf{U} \neg\psi)$, $\mathsf{F}\phi = \top \mathsf{U} \phi$, $\mathsf{G}\phi = \neg\mathsf{F}\neg\phi$.

CTL* formulas are interpreted in Kripke models $\mathcal{K}$: state formulas are interpreted over states, and path formulas over paths. In the following definition of a satisfaction relation $\models$, for a path $\pi = \langle k_0, k_1, k_2, \ldots \rangle$, by $\pi[n..]$ we denote the subpath starting at k_n.

- $k \models p \iff k \in W(p)$.
- $k \models \exists\phi \iff$ for some path π starting at k, $\pi \models \phi$.
- $\pi \models \Phi \iff \pi[0] \models \Phi$.
- $\pi \models \mathsf{X}\phi \iff \pi[1..] \models \phi$
- $\pi \models \phi \mathsf{U} \psi \iff$ there exists $j \geq 0$ s.t. $\pi[j..] \models \psi$ and for all $i < j$, $\pi[i..] \models \phi$.

Boolean connectives are interpreted as expected.

The logic LTL is the fragment of CTL* where the symbol $\exists$ does not occur, with semantics inherited from CTL*. Usually LTL (where the distinction between state and path formulas disappears) is interpreted in models that are infinite paths. Slightly more generally (and more conveniently for our purposes), we will interpret them over pointed *deterministic* Kripke models, i.e., ones where for each $k \in K$ there is exactly one $s \in K$ such that $\langle k, s \rangle \in R$. This makes little difference, since in such a model every state uniquely determines an infinite path.

3 Sets with atoms

We now recall the basic notions and results concerning sets with atoms, also known as nominal sets [34]. There are several essentially equivalent ways to introduce these; we follow the set-theoretic presentation of [22], culminating in the notion of orbit-finite sets [2, 34] and computable operations on them.

Fix a countably infinite set $\mathbb{A}$, whose elements we shall call *atoms*. A bijection on $\mathbb{A}$ will be called an atom automorphism, and the group of atom automorphisms is denoted $\mathrm{Aut}(\mathbb{A})$.

Loosely speaking, a set with atoms is a set that can have atoms, or other sets with atoms, as elements. Formally, the universe $\mathcal{U}^{\mathbb{A}}$ of sets with atoms is defined by a von Neumann-like hierachy, by transfinite induction on ordinal numbers α:

$$\mathcal{U}^{\mathbb{A}}_0 = \emptyset, \qquad \mathcal{U}^{\mathbb{A}}_{\alpha+1} = \mathcal{P}(\mathcal{U}^{\mathbb{A}}_\alpha) + \mathbb{A}, \qquad \mathcal{U}^{\mathbb{A}}_\beta = \bigcup_{\alpha<\beta} \mathcal{U}^{\mathbb{A}}_\alpha \quad \text{for } \beta \text{ a limit ordinal,}$$

where $+$ denotes disjoint union of sets.

We are interested in sets that depend only on a finite number of atoms, in the following sense. Atom automorphisms act on $\mathcal{U}^{\mathbb{A}}$ by consistently renaming all atoms in a given set. Formally, this is again defined by transfinite induction. This defines a group action:

$$\_ \cdot \_ : \mathcal{U}^{\mathbb{A}} \times \mathrm{Aut}(\mathbb{A}) \to \mathcal{U}^{\mathbb{A}}.$$

For a finite set $S \subset \mathbb{A}$, let $\mathrm{Aut}_S(\mathbb{A})$ be the group of those automorphisms of $\mathbb{A}$ that fix every element of S. We say that S *supports* a set x if $x \cdot \pi = x$ for every $\pi \in \mathrm{Aut}_S(\mathbb{A})$. A set is *equivariant* if it is supported by the empty set. If x has a finite support then it has the least finite support (see [34] for a proof), denoted $\mathrm{supp}(x)$.

Relations, functions etc. are sets in the standard sense, so the notions of support and equivariance applies to them as well. Unfolding the definitions, for equivariant sets X and Y, a relation $R \subseteq X \times Y$ is equivariant if $\langle x, y\rangle \in R$ implies $\langle x \cdot \pi, y \cdot \pi\rangle \in R$ and a function $f : X \to Y$ is equivariant if $f(x \cdot \pi) = f(x) \cdot \pi$, for every $\pi \in \mathrm{Aut}(\mathbb{A})$.

From now on, we shall only consider sets with atoms that are *hereditarily finitely supported*, i.e., ones that have a finite support, whose every element has some finite support and so on.

For any x with atoms, the *S-orbit* of x is the set $\{x \cdot \pi \mid \pi \in \mathrm{Aut}_S(\mathbb{A})\}$. For example, if S supports x then the S-orbit of x is the singleton $\{x\}$.

For any S, S-orbits form a partition of the universe $\mathcal{U}^{\mathbb{A}}$. Moreover, for any S-supported set X, the S-orbits of its elements form a partition of X. We call such X *S-orbit-finite* if it is a union of finitely many S-orbits. If $S \subseteq T$ are finite, S supports X and X is S-orbit-finite, then (T supports X and) X is also T-orbit-finite. Thanks to this observation, we may drop the qualifier and simply call X *orbit-finite*, meaning "S-orbit-finite for any/every S that supports X".

▶ **Example 3.1.**

- Any classical set (without atoms) is an equivariant set with atoms. Since its every element is also equivariant, it forms its own orbit. Therefore, a classical set is orbit-finite if and only if it is finite.
- An atom $a \in \mathbb{A}$ has no elements, and it is supported by $\{a\}$. Every finite set of atoms $S \subseteq \mathbb{A}$ is supported by S, and every element of it forms a singleton S-orbit. Its complement $\mathbb{A} \setminus S$ is also supported by S, and is a single S-orbit. A subset of $\mathbb{A}$ that is neither finite nor co-finite, is not finitely supported.
- The set $\mathbb{A}$ of atoms, the set $\binom{\mathbb{A}}{2}$ of two-element sets of atoms, and the set $\mathbb{A}^2$ of all ordered pairs of atoms, are equivariant sets. The first two have a single orbit each, and the last one is a union of two orbits:

  $$\mathbb{A}^2 = \{\langle a, a\rangle \mid a \in \mathbb{A}\} \cup \{\langle a, b\rangle \mid a \neq b \in \mathbb{A}\}.$$

  Similarly, $\mathbb{A}^n$ is orbit-finite for every $n \in \mathbb{N}$. The set $\mathbb{A}^*$ of finite sequences of atoms is hereditarily finitely supported, but not orbit-finite. The powerset $\mathcal{P}(\mathbb{A})$ is equivariant itself, but it contains elements that are not finitely supported, and therefore is not considered a legal set with atoms.
- There are four equivariant binary relations on $\mathbb{A}$: the empty relation, the equality relation, the inequality relation and the full relation.

- There is no equivariant function from $\binom{\mathbb{A}}{2}$ to $\mathbb{A}$, but $\{\langle\{a,b\},a\rangle \mid a \neq b \in \mathbb{A}\}$ is a legal equivariant relation, and the function constant at an atom a is supported by $\{a\}$. The only equivariant function from $\mathbb{A}$ to $\mathbb{A}$ is identity. The only equivariant functions from $\mathbb{A}^2$ to $\mathbb{A}$ are the two projections, and the only equivariant function from $\mathbb{A}$ to $\mathbb{A}^2$ is the diagonal $a \mapsto \langle a,a\rangle$.

Orbit-finite sets, although usually infinite, can be presented by finite means and are therefore amenable to algorithmic manipulation. There are a few ways to do this. In [2] (with the idea going back to [8]), it was observed that every single-orbit equivariant set is in an equivariant bijection with a set of k-tuples of distinct atoms, suitably quotiented by a subgroup $G \leq \mathrm{Sym}(k)$ of the symmetric group on k elements. Thus such a set can be presented by a number k and the finite group G, and an orbit-finite set is a formal union of such presentations. A somewhat more readable and concise scheme was used in [26], where orbit-finite sets are presented by set-builder expressions of the form

$$\{e \mid v_1, \ldots, v_n \in \mathbb{A}, \phi\}$$

where e is again an expression, v_i are bound atom variables and ϕ is a first-order formula with equality. We refer to [26] for a precise formulation (and to [32] for a proof that all orbit-finite sets can be presented this way); suffice it to say that the expressions in Example 3.1 are of this form, and other similar expressions are allowed. The set defined by such an expression is supported by the atoms that appear freely in the expression.

It is not entirely trivial whether two representations define the same set or not. For example, the two-orbit equivariant set A^2 from Example 3.1 can be represented in two different ways (with some light syntactic sugar added for representing ordered pairs):

$$\{\langle a,b\rangle \mid a,b \in \mathbb{A}, \top\} \qquad \text{or} \qquad \{\langle a,a\rangle \mid a \in \mathbb{A}, \top\} \cup \{\langle a,b\rangle \mid a,b \in \mathbb{A}, a \neq b\}.$$

However, set equality and other basic operations on orbit-finite sets are computable on their representations, including:

- checking whether one set is an element (or a subset) of another,
- union and intersection of sets, cartesian product, set difference,
- applying an orbit-finite function to an argument, composing functions or relations,
- finding the image of a subset along a relation,
- checking whether a finite set S supports a given set, calculating the least support of a set,
- partitioning a given set into S-orbits, calculating the S-orbit of a given element.

These basic operations have been implemented as components of atomic programming languages [27, 28].

4 μ-calculus with atoms

Syntactically, the μ-calculus with atoms (or *atomic μ-calculus*) is simply an extension of the classical formalism with orbit-finite propositional connectives.

From now on, fix a countably infinite set $\mathbb{X}$ of variables. (Until Remark 4.3 below it is convenient to think of it simply as a discrete countable set as in the classical μ-calculus.)

▶ **Definition 4.1.** Let $\mathbb{P}$ be an equivariant set with atoms of basic propositions. The set $\mathcal{L}_\mu^{\mathbb{A}}$ of formulas of the atomic μ-calculus is generated by the following grammar:

$$\phi ::= p \mid X \mid \bigvee \Phi \mid \neg\phi \mid \Diamond\phi \mid \mu X.\phi$$

where p ranges over $\mathbb{P}$, X ranges over $\mathbb{X}$ and Φ ranges over *orbit-finite* sets of formulas. As usual we only allow $\mu X.\phi$ where X occurs only positively in ϕ.

The "orbit-finite set of formulas" above refers to a canonical action of $\mathrm{Aut}(\mathbb{A})$ on formulas, extending the action on $\mathbb{P}$ inductively. Note that, despite ostensibly infinite disjunctions, every formula in $\mathcal{L}_\mu^{\mathbb{A}}$ has a finite depth, since two formulas in the same orbit necessarily have the same depth. Thanks to this, no need arises for transfinite induction in reasoning about the syntax of $\mathcal{L}_\mu^{\mathbb{A}}$ formulas.

As expected, we put $\Box\phi$ as a shorthand for $\neg\Diamond\neg\phi$, $\nu X.\phi$ for $\neg\mu X.\neg\phi[X := \neg X]$ and $\bigwedge\Phi$ for $\neg\bigvee\{\neg\phi \mid \phi \in \Phi\}$. With these conventions, every formula can be written in the negation normal form, where negation occurs only in front of a basic proposition or a variable.

Often it is notationally convenient to view an orbit-finite set Φ as a family of formulas indexed by a simpler orbit-finite set. For example, we may write

$$\bigvee_{a\in\mathbb{A}} \Diamond a \qquad \text{to mean} \qquad \bigvee\{\Diamond a \mid a \in \mathbb{A}\}.$$

▶ **Example 4.2.** Put $\mathbb{P} = \mathbb{A}$. For every $a \in \mathbb{A}$ let $\Phi_a = \{\neg b \mid b \in \mathbb{A} \setminus \{a\}\}$. Then Φ_a is supported by $\{a\}$ and orbit-finite, hence $\bigwedge\Phi_a := \bigwedge_{b\neq a} \neg b$ is a formula in $\mathcal{L}_\mu^{\mathbb{A}}$. The set

$$\Psi = \left\{\Diamond\left(a \wedge \bigwedge_{b\neq a} \neg b\right) \mid a \in \mathbb{A}\right\}$$

is equivariant and orbit-finite, hence

$$\bigwedge\Psi = \bigwedge_{a\in A} \Diamond\left(a \wedge \bigwedge_{b\neq a} \neg b\right)$$

is also a legal formula in $\mathcal{L}_\mu^{\mathbb{A}}$.

▶ **Remark 4.3.** In the classical μ-calculus, one often wants a formula to be *clean*, or *well-named*, meaning that every bound variable is bound only once. In the presence of infinitary connectives this may seem problematic. For example, in the formula

$$\bigwedge_{a\in\mathbb{A}} (\mu X.a \vee \Diamond X)$$

the variable X occurs infinitely many times, and naively replacing each binding occurrence with a completely fresh variable would result in an orbit-infinite conjunction. An easy solution is to allow a nontrivial action of the group $\mathrm{Aut}(\mathbb{A})$ on the set of variables. One way to do this is to replace each (occurrence of) a bound variable with a formal occurrence of it as a subformula, i.e., a (finite) nested sequence of subformulas that contains that occurrence.

The following syntactic properties are easily proved by induction on the depth of formulas:

- Every formula ϕ is finitely supported.
- For every formula ϕ, the set of its subformulas is finitely supported and orbit-finite.
- The relation $\trianglelefteq \subseteq \mathcal{L}_\mu^{\mathbb{A}} \times \mathcal{L}_\mu^{\mathbb{A}}$ of being a subformula is equivariant, i.e. for every $\pi \in \mathrm{Aut}(\mathbb{A})$ and every $\phi, \psi \in \mathcal{L}_\mu^{\mathbb{A}}$ we have $\phi \trianglelefteq \psi$ if and only if $\phi \cdot \pi \trianglelefteq \psi \cdot \pi$.

Semantics of the atomic μ-calculus is a straightforward extension of the classical one: we simply require all sets and relations to be sets with atoms, and replace finite models with orbit-finite ones.

▶ **Definition 4.4.** An *atomic Kripke model* is a triple $\mathcal{K} = \langle K, R, W\rangle$ where

1. K is a set with atoms,
2. R is a finitely supported binary relation on K,
3. W is a finitely supported subset of $\mathbb{P} \times K$.

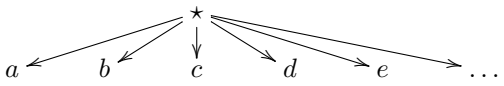
For every $p \in \mathbb{P}$ we denote $W(p) = \{k \in K \mid \langle p, k\rangle \in W\}$. Thus W can be equivalently seen as a finitely supported function from $\mathbb{P}$ to $\mathcal{P}_{\text{fs}}(K)$, the set of finitely supported subsets of K.

For any $k \in K$, the pair $\langle \mathcal{K}, k\rangle$ is called a *pointed Kripke model*. We shall say that an atomic Kripke model $\langle K, R, W\rangle$ is *orbit-finite* if the set K is so.

▶ **Example 4.5.** $\mathcal{K} = \langle K, R, W\rangle$, where:

$$K = \{\star\} \cup \mathbb{A},\ R = \{\langle \star, a\rangle \mid a \in \mathbb{A}\},\ W = \{\langle a, a\rangle \mid a \in \mathbb{A}\}$$

(with $\star$ is an equivariant element, i.e., an atom-less object that forms a singleton orbit by itself) is an infinite but orbit-finite, equivariant atomic Kripke model which can be drawn as:



▶ **Definition 4.6.** For an atomic Kripke model $\mathcal{K} = \langle K, R, W\rangle$, the meaning of a formula $\phi \in \mathcal{L}_\mu^{\mathbb{A}}$ in a variable environment $\rho : \mathbb{X} \rightharpoonup \mathcal{P}_{\text{fs}}(K)$ is defined exactly as in Definition 2.3, with the obvious modification:

$$\left[\!\left[\bigvee \Phi\right]\!\right]_\rho = \bigcup\{[\![\psi]\!]_\rho \mid \psi \in \Phi\}. \tag{1}$$

Following the philosophy of considering only finitely supported sets with atoms, the case of fixpoint formulas should also be modified to:

$$[\![\mu X.\phi]\!]_\rho = \bigcap\{L \subseteq K \mid L \text{ is finitely supported and } [\![\phi]\!]_{\rho[X\mapsto L]} \subseteq L\}. \tag{2}$$

The following easy lemma says that the meaning of every formula is a finitely supported set of states; in particular, it implies that if $\mathcal{K}$ is an equivariant model then for any formula ϕ without free variables, $[\![\phi]\!]_{\mathcal{K}}$ is supported by $\text{supp}(\phi)$.

▶ **Lemma 4.7.** *For every $\phi \in \mathcal{L}_\mu^{\mathbb{A}}$, an atomic Kripke model $\mathcal{K} = \langle K, R, W\rangle$ and a variable environment ρ, the set $[\![\phi]\!]_\rho \subseteq K$ is supported by $S = \text{supp}(\phi) \cup \text{supp}(\mathcal{K}) \cup \text{supp}(\rho) \subseteq \mathbb{A}$.*

Proof (sketch). By structural induction on ϕ. For example, consider the case of orbit-finite disjunction above, i.e., $\phi = \bigvee \Phi$. By the inductive assumption, the set $\{[\![\psi]\!]_\rho \mid \psi \in \Phi\}$ is supported by S. Since $\bigcup$ is an equivariant operation, the lemma follows.

The most interesting case is that of the fixpoint operator, $\phi = \mu X.\psi$. Recall the definition (2). The inclusion $[\![\psi]\!]_{\rho[X\mapsto L]} \subseteq L$, considered as a property of subsets $L \subseteq K$, is supported by S. Indeed, take any atom automorphism $\pi \in \text{Aut}_S(\mathbb{A})$. Since $\text{supp}(\phi) \subseteq S$, we have $\phi \cdot \pi = \phi$, hence $X \cdot \pi = X$ and $\psi \cdot \pi = \psi$. Since $\text{supp}(\rho) \subseteq S$, also $\rho \cdot \pi = \rho$. As a result, $[\![\psi]\!]_{\rho[X\mapsto L]} \subseteq L$ implies:

$$[\![\psi]\!]_{\rho[X\mapsto L\cdot\pi]} = [\![\psi \cdot \pi]\!]_{(\rho\cdot\pi)[X\cdot\pi\mapsto L\cdot\pi]} = ([\![\psi]\!]_{\rho[X\mapsto L]}) \cdot \pi \subseteq L \cdot \pi$$

where the last inclusion holds since the inclusion relation $\subseteq$ is equivariant.

Since the property of being finitely supported is equivariant, it follows that the family on the right of (2), hence its intersection, is supported by S. ◀

Atomic CTL* and atomic LTL are defined by analogy to atomic μ-calculus, extending Definition 2.5 with orbit-finite disjunctions, with semantics extended by analogy to (1) in Definition 4.6. With CTL* a design decision is to be made: in the semantic clause

- $k \models \exists\phi \iff$ for some path π starting at $k, \pi \models \phi$,

do we require the path π to be finitely supported or not? Note that even an equivariant Kripke model can contain infinite paths that are not finitely supported. We do not commit to a particular variant for now. Similar remarks apply to atomic LTL.

5 Basic properties

This section studies some basic results about the classical μ-calculus transported to the atomic setting. Proofs in this section are not difficult, but they illustrate standard techniques used to generalize properties of finite structures to orbit-finite ones.

5.1 Model-checking

▶ **Theorem 5.1.** *Model-checking problem for atomic μ-calculus over orbit-finite atomic Kripke models is decidable.*

Proof. Let us fix an orbit-finite atomic Kripke model $\mathcal{K} = \langle K, R, W\rangle$. We shall show that the meaning of any formula ϕ in $\mathcal{K}$ (under a variable environment ρ) can be computed from ϕ, by structural induction on ϕ and using basic operations listed at the end of Section 3.

The cases of basic propositions, variables, negation and the modality $\Diamond$ are straightforward. For the case of orbit-finite disjunction $\phi = \bigvee \Phi$, first calculate $S = \mathrm{supp}(\Phi) \cup \mathrm{supp}(\mathcal{K}) \cup \mathrm{supp}(\rho)$. Then partition Φ into S-orbits (there are finitely many of them), and select a system of representatives $\phi_1, \ldots, \phi_n$, one from each orbit. Using the inductive assumption, calculate $P_i = [\![\phi_i]\!]_\rho$ for each i. Then compute the S-orbit $\mathcal{O}_i$ of each P_i; each $\mathcal{O}_i$ is an S-supported family of subsets of K. The union of all $\bigcup \mathcal{O}_i$ is the desired set $[\![\phi]\!]_\rho$.

The most interesting case is computing $[\![\mu X.\phi]\!]_\rho$. This is done by approximating the least fixpoint by the following standard procedure:

(1) Put $L = \emptyset$,
(2) Extend ρ by mapping the variable X to L,
(3) Using the inductive assumption, calculate $[\![\phi]\!]_{\rho[X\mapsto L]}$ and put it as a new value of L,
(4) Repeat steps (2)-(3) until L stabilizes.

By the Knaster-Tarski theorem (see [1]) all we need to show is that this procedure terminates. Note that, by Lemma 4.7, each value assigned to L is a subset of K supported by $S = \mathrm{supp}(\phi) \cup \mathrm{supp}(\mathcal{K}) \cup \mathrm{supp}(\rho)$. In other words, L is the union of some selected S-orbits of K. Since K is orbit-finite, L can take on only finitely many values, therefore the above procedure terminates after finitely many steps. ◀

5.2 Parity games with atoms

The definition of atomic parity game is essentially as in the classical case (see e.g. [6, 38]):

▶ **Definition 5.2.** An *atomic parity game* $\mathcal{G}$ is a G quadruple $\langle V, V_\exists, R, \Omega\rangle$ such that

1. V is a set with atoms, $V_\exists$ is a finitely supported subset and we put $V_\forall = V \setminus V_\exists$;
2. $R \subseteq V^2$ is a finitely supported relation;
3. $\Omega : V \to \mathbb{N}$ is a bounded, finitely supported parity function.

The game is called *orbit-finite* if V is orbit-finite.

The notions of a match, partial match, strategy, positional strategy, winning strategy and determinacy are exactly as in the classical case.

Atomic parity games are obviously (forgetting about the action of atom automorphisms) parity games in the classical sense, so they are positionally determined. However, it might happen that in some atomic parity games no winning strategy is finitely supported.

▶ **Example 5.3.** Consider an atomic parity game where:

$$V = \binom{\mathbb{A}}{2} \cup \mathbb{A}, \quad V_\exists = \binom{\mathbb{A}}{2}, \quad R = \{\langle\{a,b\}, a\rangle \mid a \neq b \in \mathbb{A}\} \cup \{\langle a, \{b,c\}\rangle \mid a,b,c \in \mathbb{A}, b \neq c\}$$
$$\Omega(v) = 0 \text{ for all } v \in V.$$

Since $\exists$ wins every infinite play and every state has a successor with respect to R, it is clear that every state is winning for $\exists$. However, no winning strategy for $\exists$ is finitely supported. Indeed, such a strategy would determine a finitely supported function from $\binom{\mathbb{A}}{2}$ to $\mathbb{A}$ such as $f(C) \in C$ for all $C \in \binom{\mathbb{A}}{2}$, and it is easy to see that no such function exists.

In spite of this, winning regions in orbit-finite parity games are computable. Indeed, every orbit-finite game can be effectively transformed into a finite game in the following way. For an orbit-finite parity game $\mathcal{G} = \langle V, V_\exists, R, \Omega \rangle$, let S be any finite set of atoms that supports $\mathcal{G}$. Let V/S, $V_\exists/S$ be the sets of S-orbits of V and $V_\exists$, respectively; let $[v]$ denote the S-orbit of $v \in V$. Obviously $V_\exists/S \subseteq V/S$. Define $R/S \subseteq V/S \times V/S$ and $\Omega/S : V/S \to \mathbb{N}$ by:

$$\begin{aligned} \langle [v], [w] \rangle \in R/S \quad &\text{if} \quad \langle x, y \rangle \in R \text{ for some } x \in [v], y \in [w] \\ \Omega/S([v]) = n \quad &\text{if} \quad \Omega(x) = n \text{ for some } x \in [v] \end{aligned}$$

This is well defined since S supports both R and Ω. In particular, Ω/S is a function. We call $\mathcal{G}/S = \langle V/S, V_\exists/S, R/S, \Omega/S \rangle$ the *orbit game* of $\mathcal{G}$.

▶ **Lemma 5.4.** *The quotient function* Π *defined for every* $v \in V$ *by* $\Pi(v) = [v]$ *is a bisimulation between labeled Kripke models* $\langle V, R, \Omega \rangle$ *and* $\langle V/S, R/S, \Omega/S \rangle$.

Proof. First, if $\Pi(w) = [v]$, then $w \in [v]$ and consequently all the elements of $[v]$ have label $\Omega(w)$. So, by definition $\Omega/S([v]) = \Omega(w)$.

Now suppose $\Pi(w) = [v]$ and $\langle [v], [z] \rangle \in R/S$. It means that there are $x \in [v]$, $y \in [z]$ such that $\langle x, y \rangle \in R$. Since w and x are in the same S-orbit, pick a $\pi \in \mathrm{Aut}_S(\mathbb{A})$ such that $x \cdot \pi = w$. Since S supports R, we get $\langle w, y \cdot \pi \rangle \in R$. But $y \cdot \pi \in [z]$, so $\Pi(y \cdot \pi) = [z]$.

For the opposite direction, let $\Pi(w) = [v]$ and for some x, $\langle w, x \rangle \in R$. Then by the definition of R/S, $\langle [v], [x] \rangle \in R/S$ and obviously $\Pi(x) = [x]$. ◀

Moreover, for every $v \in V$, $v \in V_\exists$ if and only if $[v] \in V_\exists/S$. As a result, $\exists$ has a (positional) winning strategy from v in $\mathcal{G}$ if and only if she has a (positional) winning strategy from $[v]$ in $\mathcal{G}/S$. This implies that one can effectively decide whether a player has a winning strategy in an orbit-finite atomic parity game $\mathcal{G}$ by calculating first $S = \mathrm{supp}(\mathcal{G})$, then $\mathcal{G}/S$, and finally solving the problem in the finite parity game obtained, using standard methods.

A correspondence of the atomic μ-calculus with atomic parity games is proved essentially in the same way as in the classical case (see e.g. [6, 38]) using the notion of unfolding games. Together with the orbit game construction, this gives an alternative route to decidability of the model-checking problem for the atomic μ-calculus.

5.3 Failure of the orbit-finite model property

The classical modal μ-calculus enjoys the so-called finite model property: every satisfiable formula has a finite model. (In fact a stronger *small model property* holds, useful for complexity upper bounds.) There is no chance for this property to hold in the atomic setting, but since orbit-finite sets play the role of finite sets in the universe of sets with atoms, one might hope that an *orbit-finite model property* holds, i.e., that every satisfiable formula in $\mathcal{L}_\mu^{\mathbb{A}}$ has an orbit-finite model. However, even that weaker property fails, as we shall now prove.

Over a vocabulary of basic propositions that includes a proposition p_a for each atom a, consider the following three properties:

P1: every state reachable from the current state has at least one successor;
P2: in every state reachable from the current state some p_a holds;
P3: on every path starting in the current state, no p_a holds more than once.

All these are definable in the atomic μ-calculus. **P1** is simply $\nu X.(\Diamond\top \wedge \Box X)$, **P2** is $\nu X.\left(\bigvee_{a\in\mathbb{A}} p_a \wedge \Box X\right)$, and **P3** is

$$\neg(\mu X.(\psi \vee \Diamond X)) \qquad \text{where} \qquad \psi = \bigvee_{a\in\mathbb{A}} (p_a \wedge \Diamond \mu Y.(p_a \vee \Diamond Y)).$$

To build an atomic Kripke model for the conjunction of **P1**, **P2** and **P3**, put as states finite, nonempty sequences of distinct atoms. These form an equivariant set with atoms which, however, has infinitely many orbits (sequences of different lengths fall into separate orbits). For the transition relation put

$$R = \{\langle w, wa\rangle \mid a \notin w\},$$

that is, a sequence w can make a step to another valid sequence by appending a single atom a. The basic predicates are interpreted so that each p_a holds in exactly those sequences that end with the atom a. Properties **P1**, **P2** and **P3** hold in (every state of) this model.

However, the conjunction **P1**, **P2** and **P3** has no orbit-finite models. Indeed, assume that some state x_0 in such a model (with the transition relation denoted by R) satisfies all three properties. By **P1**, there exists an infinite path in the model:

$$x_0, x_1, x_2, x_3, x_4, \ldots \qquad \text{such that} \quad \langle x_i, x_{i+1}\rangle \in R.$$

By **P2**, each state on this path satisfies some predicate p_a, and by **P3** no such predicate is satisfied more than once.

Since the model is orbit-finite, there exists a global upper bound on the size of the least supports $\mathrm{supp}(x_i)$. This implies that there exists a number j and atoms $a \neq b$ such that:

- p_a holds in some x_i where $i < j$,
- p_b holds in some x_k where $j < k$, and
- $a, b \notin \mathrm{supp}(x_j)$.

Let $\pi \in \mathrm{Aut}(\mathbb{A})$ be the atom automorphism that swaps a and b and leaves all other atoms untouched; then $x_j \cdot \pi = x_j$, therefore $\langle x_{j-1}, x_j \cdot \pi\rangle \in R$. As a result:

$$x_0, x_1, x_2, \ldots, x_i, \ldots, x_{j-1}, x_j \cdot \pi, \ldots, x_k \cdot \pi, \ldots$$

is a legal path. But p_a holds both in x_i and in $x_k \cdot \pi$, so **P3** is violated on this path.

6 Undecidability results

▶ **Theorem 6.1.** *It is undecidable whether a given atomic LTL formula is satisfiable.*

Proof. See Appendix A. The proof follows the lines of the proof from [31] of the undecidability of the universality problem for register automata. ◀

▶ **Theorem 6.2.** *It is undecidable whether a given formula of the atomic μ-calculus is satisfiable.*

Proof. See Appendix B for details. Use a translation M from LTL into $\mathcal{L}_\mu^{\mathbb{A}}$ such that:

(i) In every word model, if a state satisfies ϕ then it satisfies $M(\phi)$,

(ii) In every Kripke model K, if a state x satisfies $M(\phi)$ then every path in K that starts from x, considered as a word model, satisfies ϕ.

Then apply Theorem 6.1. ◀

▶ **Theorem 6.3.** *The model checking problem for atomic CTL* is undecidable.*

Proof. Easy reduction from the satisfiability problem for atomic LTL; see Appendix C. ◀

7 Expressiveness limitations

In this section, consider Kripke models over a vocabulary of basic propositions that includes a proposition p_a for each atom a. For a state x in such a model, define $\mathrm{pred}(x) \subseteq \mathbb{A}$ to be the set of those atoms a for which p_a holds in x. Note that $\mathrm{pred}(x)$ ignores all propositions that are not of the form p_a.

Denote the property "there exists an infinite path where no p_a holds more than once", by #Path. Such properties of states in Kripke models have potentially significant practical importance. For example, one may imagine a system equipped with a token (e.g. password) generator where one needs to verify that, on every path where no token is generated more than once, the security of the system is never breached.

We shall show that although #Path is decidable, it is not definable in atomic μ-calculus. This is in contrast to the similar but definable property **P3** from Section 5.3.

▶ **Theorem 7.1.** #Path *is decidable on orbit-finite Kripke models.*

Proof. For simplicity, assume that a given orbit-finite Kripke model $\mathcal{K} = \langle K, R, W\rangle$ is equivariant; a generalization to finitely supported models is straightforward.

Notice that for every state $x \in K$, the set $\mathrm{pred}(x)$ is either finite (and contained in $\mathrm{supp}(x)$) or co-finite. Moreover, a single orbit of K only contains states of one of these two kinds. It is not difficult to decide the existence of a desired path where at least one state is of the second kind. Indeed, two such states cannot occur on the path at all, and even if exactly one of them occurs, almost all other states on the path must satisfy none of the predicates p_a. The existence of such a path from a given state x is straightforward to decide.

Once the existence of such paths is excluded, all (orbits of) states x with co-finite $\mathrm{pred}(x)$ may be safely deleted from the model. From now on, assume that $\mathrm{pred}(x) \subseteq \mathrm{supp}(x)$ for each $x \in K$.

Derived from $\mathcal{K}$, construct a new orbit-finite Kripke model $\hat{\mathcal{K}} = \langle \hat{K}, \hat{R}, \emptyset\rangle$, over the empty set of basic predicates, defined as follows:

$$\hat{K} = \{\langle x, S\rangle \mid x \in K,\ S \subseteq \mathrm{supp}(x) \setminus \mathrm{pred}(x)\}$$
$$\hat{R} = \{\langle\langle x, S\rangle, \langle y, T\rangle\rangle \mid \ \langle x, y\rangle \in R,\ (S \cup \mathrm{pred}(x)) \cap \mathrm{supp}(y) \subseteq T\}.$$

The intuition is that in a state $\langle x, S\rangle$, atoms in S are marked as having had occurred previously on a path, and are forbidden from occurring in the future. Note that this marking is restricted to atoms from the support of the current state x only.

In Appendix D we prove that a state $x \in K$ admits an infinite path where no p_a holds more than once, if and only if $(x, \emptyset) \in \hat{K}$ admits any infinite path. The theorem follows since the latter property is decidable (indeed, it is definable in the atomic μ-calculus, whose model-checking problem is decidable by Theorem 5.1). ◀

We shall now show that, in spite of its decidability and intuitive simplicity, #Path is not definable by a formula of the atomic μ-calculus. To this end, we introduce a hierarchy of bisimulations on atomic Kripke models.

Denote by $\mathbb{A}^{(\leq k)}$ the set of ordered tuples of pairwise distinct atoms of length at most k. Elements of such sets will be denoted with vector notation: $\vec{a}, \vec{b}$ etc. We shall write $x \sim y$ to say that x and y are in the same orbit.

▶ **Definition 7.2.** For a number $k \in \mathbb{N}$, a *k-bisimulation* on a Kripke model $\mathcal{K} = \langle K, R, W\rangle$ is a symmetric relation B on $K \times \mathbb{A}^{(\leq k)}$ such that, whenever $\langle x, \vec{a}\rangle B \langle y, \vec{b}\rangle$ then:

(i) $\langle \mathrm{pred}(x), \vec{a}\rangle \sim \langle \mathrm{pred}(y), \vec{b}\rangle$,

(ii) for every x' such that $\langle x, x'\rangle \in R$ there is a y' such that $\langle y, y'\rangle \in R$ and $\langle x', \vec{a}\rangle B \langle y', \vec{b}\rangle$,
(iii) for every $\vec{c} \in \mathbb{A}^{(\leq k)}$ there exists a $\vec{d} \in \mathbb{A}^{(\leq k)}$ such that $\langle \vec{a}, \vec{c}\rangle \sim \langle \vec{b}, \vec{d}\rangle$ and $\langle x, \vec{c}\rangle B \langle y, \vec{d}\rangle$.

Two states are called *k-bisimilar* if they are related by a k-bisimulation.

Some properties of k-bisimulations are straightforward to check. For example, by a standard argument, k-bisimilarity on a Kripke model is an equivalence relation. From condition (i) above it immediately follows that if $\langle x, \vec{a}\rangle$ and $\langle y, \vec{b}\rangle$ are k-bisimilar then $|\vec{a}| = |\vec{b}|$ and there exists a $\pi \in \mathrm{Aut}(\mathbb{A})$ such that $\vec{a} \cdot \pi = \vec{b}$. Furthermore, if shorter tuples $\vec{a}'$ and $\vec{b}'$ arise from $\vec{a}$ and $\vec{b}$ respectively by selecting the same subset of positions, then $\langle x, \vec{a}'\rangle$ and $\langle y, \vec{b}'\rangle$ are k-bisimilar as well. Finally, for $l < k$, the restriction of a k-bisimulation to the set $K \times \mathbb{A}^{(\leq l)}$ is an l-bisimulation.

The following result shows that k-bisimilar states cannot be distinguished by formulas from a certain fragment of the atomic μ-calculus. Call a formula $\phi \in \mathcal{L}_\mu^{\mathbb{A}}$ *globally k-supported* if every subformula of it (including ϕ itself) has a support of size at most k.

▶ **Theorem 7.3.** *For a globally k-supported formula ϕ, if*
- *$\langle x, \vec{a}\rangle$ and $\langle y, \vec{b}\rangle$ are k-bisimilar,*
- *$\pi \in Aut(\mathbb{A})$ is such that $\vec{a} \cdot \pi = \vec{b}$,*
- *$supp(\phi) \subseteq \vec{a}$, and $x \models \phi$*

then $y \models \phi \cdot \pi$.

▶ **Corollary 7.4.** *Assume that $\langle x, \epsilon\rangle$ and $\langle y, \epsilon\rangle$ are k-bisimilar. For every equivariant, globally k-supported formula ϕ, $x \models \phi$ if and only if $y \models \phi$.*

Proof. The Corollary is simply a special case of the Theorem, for $\vec{a} = \vec{b} = \epsilon$. To prove the Theorem, first apply a standard translation of the formula ϕ to infinitary modal logic, where no fixpoint operators are allowed, but boolean connectives of any arity are admitted. The translation is defined by induction. In the only interesting case, consider $\phi = \mu X.\psi$. For each ordinal α define a formula ϕ_α by:

$$\phi_0 = \bot, \qquad \phi_{\alpha+1} = \psi[X \mapsto \phi_\alpha], \qquad \phi_\beta = \bigvee_{\alpha<\beta} \phi_\alpha \text{ for a limiting ordinal } \beta.$$

It is then a standard result that $x \models \phi$ if and only if $x \models \bigvee_{\alpha<\kappa} \phi_\alpha$, where κ is the first cardinal larger than the Kripke model in question. Note that every ϕ_α is supported by $\mathrm{supp}(\phi)$, and so is the entire infinite alternative. As a result, if an original formula ϕ is globally k-supported then the resulting infinitary formula is also globally k-supported.

From now on, assume that ϕ is an infinitary modal logic formula. The Theorem is proved by transfinite induction on the depth of ϕ, as follows:
- Assume $\phi = p_c$ for some basic predicate p_c. Since $x \models p_c$ (hence $c \in \mathrm{pred}(x)$) and $\langle x, \vec{a}\rangle$ and $\langle y, \vec{b}\rangle$ are k-bisimilar then, by condition (i) in Defn. 7.2, $c \cdot \pi \in \mathrm{pred}(y)$ and $y \models p_c \cdot \pi$.
- Assume $\phi = \Diamond\psi$. Then $x \models \phi$ means that $\langle x, x'\rangle \in R$ for some x' such that $x' \models \psi$. By condition (ii) in Defn. 7.2, there is some y' such that $\langle y, y'\rangle \in R$ and $\langle x', \vec{a}\rangle$ and $\langle y', \vec{b}\rangle$ are k-bisimilar. Since $\mathrm{supp}(\psi) = \mathrm{supp}(\phi) \subseteq \vec{a}$, by the inductive assumption we get $y' \models \psi \cdot \pi$. As a result, $y \models \Diamond(\psi \cdot \pi)$, and the latter formula equals $\phi \cdot \pi$.
- Assume $\phi = \neg\psi$, and towards a contradiction that $y \models \psi \cdot \pi$. Notice that $\mathrm{supp}(\psi \cdot \pi) \subseteq \vec{b}$. Applying the inductive assumption to $\psi \cdot \pi$, by the symmetry of k-bisimilarity we obtain $x \models \psi \cdot \pi\pi^{-1}$, and the latter formula is ψ, contradicting the assumption that $x \models \phi$.
- Assume $\phi = \bigvee \Phi$, where Φ is a set of formulas such that $\mathrm{supp}(\Phi) \subseteq \vec{a}$. Pick $\psi \in \Phi$ such that $x \models \psi$, and let $\vec{c}$ be the set $\mathrm{supp}(\psi)$, ordered in an arbitrary way. Note that $|\vec{c}| \leq k$ by the assumption that ϕ is globally k-supported. Pick a tuple $\vec{d}$ that exists by condition

(iii) in Defn. 7.2. Since $\langle \vec{a}, \vec{c} \rangle \sim \langle \vec{b}, \vec{d} \rangle$, there exists a $\sigma \in \mathrm{Aut}(\mathbb{A})$ such that $\vec{a} \cdot \sigma = \vec{b}$ and $\vec{c} \cdot \sigma = \vec{d}$. Moreover, since $\langle x, \vec{c} \rangle$ and $\langle y, \vec{d} \rangle$ are k-bisimilar, we can apply the inductive assumption on ψ to obtain $y \models \psi \cdot \sigma$. This means that $y \models \phi \cdot \sigma$. However, since σ and π are equal on $\vec{a}$ and $\mathrm{supp}(\phi) \subseteq \vec{a}$, we get $\phi \cdot \sigma = \phi \cdot \pi$.
Note that we did not assume that Φ is an orbit-finite set; the reasoning works for any set of formulas as long as it is supported by $\vec{a}$. ◀

Note that every formula of the atomic μ-calculus is globally k-supported for some number k. Therefore, by Cor. 7.4, to prove that #Path is not definable it is enough to construct, for every number k, an orbit-finite Kripke model $\mathcal{K} = \langle K, R, W \rangle$ over the language of basic predicate symbols $\{p_a \mid a \in A\}$, and two states $x, y \in K$ such that $\langle x, \epsilon \rangle$ and $\langle y, \epsilon \rangle$ are k-bisimilar and #Path fails for x but holds for y. To this end, choose any $S \subseteq \mathbb{A}$ s.t $|S| = 2k + 1$. Let $K = \mathbb{A}$ and define the transition relation R by:

$$\langle a, b \rangle \in R \iff (a \in S \iff b \in S).$$

In words, R forms a disjoint sum of a finite clique of size $2k + 1$ and an infinite clique. Interpret the basic predicates by:

$$a \models p_b \iff a = b.$$

Finally, pick some $x \in S$ and $y \notin S$. It is easy to see that #Path fails for x and holds for y.

It remains to be proved that $\langle x, \epsilon \rangle$ and $\langle y, \epsilon \rangle$ are k-bisimilar. To this end, consider a relation B on $K \times \mathbb{A}^{(\leq k)}$:

$$\langle u, \vec{a} \rangle B \langle v, \vec{b} \rangle \iff (|\vec{a}| = |\vec{b}|) \wedge (u \in S \iff v \notin S) \wedge \\ \wedge (\forall i.\ a_i \in S \iff b_i \notin S) \wedge (\forall i.\ a_i = u \iff b_i = v).$$

Obviously B is a symmetric relation and $\langle x, \epsilon \rangle B \langle y, \epsilon \rangle$. We shall show that B is a k-bisimulation by checking, for any $\langle u, \vec{a} \rangle B \langle v, \vec{b} \rangle$, the three conditions of Defn. 7.2 in turn:

(i) We have $\mathrm{pred}(u) = \{u\}$ and $\mathrm{pred}(v) = \{v\}$, and from the definition of B it immediately follows that $\langle \{u\}, \vec{a} \rangle \sim \langle \{v\}, \vec{b} \rangle$.

(ii) Consider any u' such that $\langle u, u' \rangle \in R$. If $u' = a_i$ for some i then put $v' = b_i$. On the other hand, if u' does not appear in $\vec{a}$ then take v' to be any atom that does not appear in $\vec{b}$ and such that $u' \in S \iff v' \notin S$. (For $u' \notin S$, this is possible since $|\vec{b}| \leq k < |S|$.) Either way, $\langle v, v' \rangle \in R$ and $\langle u', \vec{a} \rangle B \langle v', \vec{b} \rangle$ as required.

(iii) For any $\vec{c} \in \mathbb{A}^{(\leq k)}$ construct $\vec{d} \in \mathbb{A}^{(\leq k)}$ as follows:
- if $c_i = a_j$ for some (necessarily unique) j then put $d_i = b_j$,
- if $c_i = u$ then put $d_i = v$,
- otherwise put d_i to be any atom that does not occur in $\vec{b}$, is different from v and such that $c_i \in S$ iff $d_i \notin S$. At the same time, ensure that $d_i \neq d_j$ for $i \neq j$. This is possible since $|S| = 2k + 1$, so there are at least k distinct atoms in S that do not occur in $\vec{b}$ and are different from v.

For $\vec{d}$ constructed this way, $\langle \vec{a}, \vec{c} \rangle \sim \langle \vec{b}, \vec{d} \rangle$ and $\langle u, \vec{c} \rangle B \langle v, \vec{d} \rangle$ as required.

The above Kripke model is not equivariant; its support is the chosen set S. However, a similar effect can be achieved in an equivariant model. Indeed, given a number k, it is enough to take the disjoint union, indexed by the family of all finite sets S of size $2k + 1$, of the models as above. The resulting model is equivariant with $2k + 2$ orbits of states, and the above reasoning holds for it without significant changes.

▶ **Remark 7.5.** The property #Path is easy to define in atomic CTL* with path formulas interpreted over arbitary paths, by a formula:

$$\exists \left(\bigwedge_{a \in \mathbb{A}} \mathsf{G}(p_a \to \mathsf{X}(\mathsf{G} \neg p_a)) \right).$$

This is, however, of little practical use due to Theorem 6.3. It also means that, contrary to the classical atom-less setting, atomic CTL* is not a fragment of the atomic μ-calculus.

8 Future work

We list some interesting aspects of the atomic μ-calculus that are best left for future work.

Complexity issues. For the decidability results we presented, in particular for the model checking problem over orbit-finite structures, one immediately asks about the complexity of the algorithms proposed. The answer depends on the way one measures the size of input structures. One obvious option is to consider the length of their representation with set-builder expressions and first-order formulas. With this view, most basic operations listed at the end of Section 3 become Pspace-hard, because the first-order theory of pure equality is Pspace-complete. As a result, the complexity of basic operations dwarfs the distinction between the two algorithmic approaches to model checking based on direct fixpoint computation and on parity games.

Another approach is to measure orbit-finite structures by the number of their orbits, and the (hereditary) size of their least support. Note that the number of orbits of a set can be exponentially bigger than the size of its logical representation; for example, the number of orbits of the set $\mathbb{A}^n$, whose representation has size linear in n, is equal to the n-th Bell number. In this view the difference between the two approaches to model checking becomes more prominent.

We defer precise complexity analyses until we have a better general understanding of various time and space complexity models on atomic structures.

Other atoms. As advocated in [2], much of the theory of sets with atoms can be generalized to other relational structures $\mathbb{A}$. For example, one can consider *ordered atoms* $\mathbb{A} = \langle \mathbb{Q}, < \rangle$, where sets such as the interval $\{b \in \mathbb{A} \mid 2 < b < 5\}$ become finitely supported. Other atom structures are also possible; see [2] for details.

If the first-order theory of $\mathbb{A}$ is decidable then most definitions and results in this paper generalize to other structures of atoms studied in [2]. An interesting exception is the undefinability of #Path studied in Section 7. Our construction there does not work for ordered atoms; indeed, in the model $\mathcal{K}$ constructed after Corollary 7.4, the states $\langle x, \epsilon \rangle$ and $\langle y, \epsilon \rangle$ are not even 1-bisimilar.

The property #Path does not mention the order of atoms in any way, so it would be quite surprising if it turned out to be definable in the μ-calculus with ordered atoms. We leave this as an open problem.

Defining #Path. The fact that #Path is not definable in the μ-calculus with equality atoms is disappointing, since it looks like a property of potential practical importance in system verification. Since we know that #Path is decidable, it is desirable to extend atomic μ-calculus in some well-structured and syntactically economic way that would allow one to define such properties while preserving the decidability of model checking. The property of "global freshness" has been studied in the context of automata with atoms [37], and one may look for inspiration there. Our proof of Theorem 7.1 also suggests some promising options. We leave this for future work.

References

1 A. Arnold and D. Niwiński. *Rudiments of μ-calculus.* Studies in logic and the foundations of mathematics. London, Amsterdam, 2001.

2 M. Bojańczyk, B. Klin, and S. Lasota. Automata theory in nominal sets. *Log. Meth. Comp. Sci.*, 10, 2014.

3 M. Bojańczyk and T. Place. Toward model theory with data values. In *Procs. ICALP 2012 Part II*, volume 7392 of *Lecture Notes in Computer Science*, pages 116–127, 2012.

4 J. Bradfield and C.Stirling. Modal mu-calculi. In *Handbook of Modal Logic*, volume 3, pages 721 – 756. Elsevier, 2007.

5 J. Bradfield and C. Stirling. Modal logics and mu-calculi: an introduction. In *Handbook of Process Algebra*, pages 293–330. North-Holland, 2001.

6 J. Bradfield and I. Walukiewicz. The mu-calculus and model-checking. In H. Veith E. Clarke, T. Henzinger, editor, *Handbook of Model Checking*. Springer-Verlag, 2015.

7 C. Carapelle and M. Lohrey. Temporal logics with local constraints (invited talk). In *Procs. CSL 2015*, volume 41 of *LIPIcs*, pages 2–13, 2015.

8 V. Ciancia and U. Montanari. Symmetries, local names and dynamic (de)-allocation of names. *Inf. Comput.*, 208(12):1349–1367, 2010.

9 S. Cranen, J. F. Groote, and M. Reniers. A linear translation from CTL* to the first-order modal μ-calculus. *Theoretical Computer Science*, 412(28):3129 – 3139, 2011.

10 Mads Dam. Model checking mobile processes. *Information and Computation*, 129(1):35–51, 1996.

11 Rocco De Nicola and Michele Loreti. Multiple-labelled transition systems for nominal calculi and their logics. *Mathematical Structures in Computer Science*, 18(01):107–143, 2008.

12 S. Demri and D. D'Souza. An automata-theoretic approach to constraint LTL. *Inf. Comput.*, 205(3):380–415, 2007.

13 S. Demri, D. D'Souza, and R. Gascon. Temporal logics of repeating values. *J. Log. Comput.*, 22(5):1059–1096, 2012.

14 S. Demri, D. Figueira, and M. Praveen. Reasoning about data repetitions with counter systems. *Logical Methods in Computer Science*, 12(3), 2016.

15 S. Demri and R. Lazic. LTL with the freeze quantifier and register automata. *ACM Trans. Comput. Log.*, 10(3):16:1–16:30, 2009.

16 S. Demri, R. Lazic, and D. Nowak. On the freeze quantifier in constraint LTL: decidability and complexity. *Inf. Comput.*, 205(1):2–24, 2007.

17 E. A. Emerson and J. Y. Halpern. "sometimes" and "not never" revisited: On branching versus linear time temporal logic. *J. ACM*, 33(1):151–178, 1986.

18 Gian-Luigi Ferrari, Stefania Gnesi, Ugo Montanari, and Marco Pistore. A model-checking verification environment for mobile processes. *ACM Transactions on Software Engineering and Methodology*, 12(4):440–473, 2003.

19 D. Figueira. Alternating register automata on finite words and trees. *Logical Methods in Computer Science*, 8(1), 2012.

20 D. Figueira and L. Segoufin. Future-looking logics on data words and trees. In *Procs. MFCS 2009*, volume 5734 of *Lecture Notes in Computer Science*, pages 331–343, 2009.

21 N. Francez and M. Kaminski. Finite-memory automata. *TCS*, 134(2):329–363, 1994.

22 M. Gabbay and A. M. Pitts. A new approach to abstract syntax with variable binding. *Formal Asp. Comput.*, 13(3-5):341–363, 2002.

23 Jan Friso Groote and Radu Mateescu. Verification of temporal properties of processes in a setting with data. In *Procs. AMAST'99*, pages 74–90, 1999.

24 Jan Friso Groote and Tim AC Willemse. Model-checking processes with data. *Science of Computer Programming*, 56(3):251–273, 2005.

25 M. Jurdzinski and R. Lazic. Alternating automata on data trees and xpath satisfiability. *ACM Trans. Comput. Log.*, 12(3):19:1–19:21, 2011.

26 B. Klin, E. Kopczyński, J. Ochremiak, and S. Toruńczyk. Locally finite constraint satisfaction problems. In *Procs. LICS 2015*, pages 475–486, 2015.

27 B. Klin and M. Szynwelski. SMT solving for functional programming over infinite structures. In *MFSP*, volume 207, pages 57–75, 2016. doi:10.4204/EPTCS.207.3.

28 E. Kopczyński and S. Toruńczyk. Lois: Syntax and semantics. In *Procs. of POPL 2017*, pages 586–598, 2017.

29 D. Kozen. Results on the propositional μ-calculus. *Theor. Comp. Sci.*, 27(3):333 – 354, 1983. doi:http://dx.doi.org/10.1016/0304-3975(82)90125-6.

30 Hui-Min Lin. Predicate μ-calculus for mobile ambients. *Journal of Computer Science and Technology*, 20(1):95–104, 2005.

31 F. Neven, T. Schwentick, and V. Vianu. Towards regular languages over infinite alphabets. In *MFCS*, pages 560–572, 2001.

32 J. Ochremiak. *Extended constraint satisfaction problems*. PhD thesis, University of Warsaw, 2016.

33 J. Parrow, J. Borgström, L.-H. Eriksson, R. Gutkovas, and T. Weber. Modal logics for nominal transition systems. In *Procs. CONCUR 2015*, volume 42 of *LIPIcs*, pages 198–211, 2015.

34 A. M. Pitts. *Nominal Sets: Names and Symmetry in Computer Science*. Cambridge University Press, 2013.

35 L. Segoufin. Automata and logics for words and trees over an infinite alphabet. In *Procs. CSL 2006*, volume 4207 of *Lecture Notes in Computer Science*, pages 41–57, 2006.

36 W. Thomas and T. Wilke. Automata, logics, and infinite games: A guide to current research. *Bulletin of Symbolic Logic*, 10(1):114–115, 2004.

37 N. Tzevelekos. Fresh-register automata. *SIGPLAN Not.*, 46(1):295–306, 2011.

38 Y. Venema. *Lectures on the modal μ-calculus*. ILLC, Univ. of Amsterdam, 2007.

A Proof of Theorem 6.1

Given a Turing machine $\mathcal{M}$ with a (finite) set of states Q, a (finite) working alphabet Γ (including a blank symbol $\flat$), initial and accepting states $q_{init}, q_{acc} \in Q$ and a transition relation δ, we shall build an atomic LTL formula that is satisfiable if and only if $\mathcal{M}$ accepts the empty word. Without loss of generality assume that $\mathcal{M}$, upon reaching an accepting configuration, enters an infinite loop in that configuration.

Consider the following language of basic predicate symbols:

- a special symbol \$,
- $\mathtt{atom}_a$ for each $a \in \mathbb{A}$,
- $\mathtt{tape}_\gamma$ for each $\gamma \in \Gamma$,
- $\mathtt{head}_q$ for each $q \in Q$.

Our formula shall be a conjunction of several properties. First, we enforce that every state in a model either satisfies \$ (and no other basic predicates) or satisfies exactly one predicate $\mathtt{atom}_a$, exactly one predicate $\mathtt{tape}_\gamma$ and at most one predicate $\mathtt{head}_q$. This is ensured by a conjunction of formulas such as:

$$\mathsf{G}\left(\$ \vee \bigvee_{a\in\mathbb{A}} \mathtt{atom}_a\right) \quad \text{and} \quad \bigwedge_{a\neq b\in\mathbb{A}} \mathsf{G}(\mathtt{atom}_a \to \neg\mathtt{atom}_b)$$

and so on. We also ensure that the initial state of the model satisfies \$.

Furthermore, we ensure that as far as predicates \$ and $\mathtt{atom}_a$ are concerned, the model can be presented as an infinite word:

$$\$w\$w\$w\cdots$$

where w is a finite sequence of predicate symbols of the form $\mathtt{atom_a}$ such that no single predicate appears in w more than once. This is achieved by a conjunction of formulas such as:

- $\bigwedge_{a,b\in\mathbb{A}} \mathsf{G}(\mathtt{atom}_a \wedge \mathsf{X}\mathtt{atom}_b \to \mathsf{G}(\mathtt{atom}_a \to \mathsf{X}\mathtt{atom}_b))$
- $\bigwedge_{a\in\mathbb{A}} \mathsf{G}(\mathtt{atom}_a \to (\neg\mathtt{atom}_a \mathsf{U}\$))$.

Portions of the model between two consecutive occurrences of \$ will store configurations of $\mathcal{M}$. To this end, we ensure that each portion contains exactly one state that satisfies some `head` predicate:

$$\mathsf{G}(\$ \to \mathsf{X}((\neg\psi \wedge \neg\$)\mathsf{U}(\psi \wedge \mathsf{X}(\neg\psi\mathsf{U}\$))))$$

where $\psi = \bigvee_{q\in Q} \mathtt{head}_q$.

We then ensure that every two consecutive configurations encode a legal step of $\mathcal{M}$. Predicates `atom` are useful for this, as they trace single tape cells in subsequent configurations. To ensure that the letters on the tape do not change unless the machine head is directly over them, we state for each $\gamma \in \Gamma$:

$$\bigwedge_{a\in\mathbb{A}} \mathsf{G}(\mathtt{atom}_a \wedge \mathtt{tape}_\gamma \wedge \neg\psi \to \mathsf{X}(\neg\mathtt{atom}_a \mathsf{U}(\mathtt{atom}_a \wedge \mathtt{tape}_\gamma)))$$

where ψ is as before. Furthermore, for every "head to the right" rule

$$\langle q, \gamma, q', \gamma', \Rightarrow\rangle \in \delta$$

we add a formula

$$\bigwedge_{a\in\mathbb{A}} \mathsf{G}(\mathtt{atom}_a \wedge \mathtt{head}_q \wedge \mathtt{tape}_\gamma \to \mathsf{X}(\neg\mathtt{atom}_a \mathsf{U}(\mathtt{atom}_a \wedge \mathtt{tape}_{\gamma'} \wedge \mathsf{X}\mathtt{head}_{q'})))$$

and similarly for "head to the left" transition rules.

Finally we ensure the correct form of the initial configuration and that an accepting state is reached, by adding formulas $\mathsf{X}\mathtt{head}_{\mathtt{q_{init}}}$, $\mathsf{X}(\mathtt{tape}_\flat \mathsf{U}\$)$ and $\mathsf{F}\mathtt{head}_{q_{acc}}$.

Models of the conjunction of all these formulas correspond to accepting runs of $\mathcal{M}$ on the empty input word. As a result, it is undecidable whether an LTL formula has a model.

B Proof of Theorem 6.2

For any LTL formula ϕ in the negation normal form (i.e., one where negations occur only in front of basic predicates, and where conjunction, disjunction, X, U and R modal operators are used), construct an atomic μ-calculus formula $M(\phi)$ by induction as follows:

$$M(\top) = \top, \qquad M(\bot) = \bot, \qquad M(p) = p, \qquad M(\neg p) = \neg p, \qquad M(\mathsf{X}\phi) = \Box M(\phi)$$
$$M(\phi \vee \psi) = M(\phi) \vee M(\psi), \qquad M(\phi \wedge \psi) = M(\phi) \wedge M(\psi)$$
$$M(\phi\mathsf{U}\psi) = \mu q.(M(\psi) \vee (M(\phi) \wedge \Box q)), \qquad M(\phi\mathsf{R}\psi) = \nu q.(M(\psi) \wedge (M(\phi) \vee \Box q))$$

This translation does not have all properties that may be desired (see e.g. [9]) but it is sufficient for our purposes:

(i) In every word model, if a state satisfies ϕ then it satisfies $M(\phi)$,
(ii) In every Kripke model K, if a state x satisfies $M(\phi)$ then every path in K that starts from x, considered as a word model, satisfies ϕ.

Note that the converse to the implication in (ii) does not hold in general (consider e.g. $\phi = \mathsf{X}p \vee \mathsf{X}q$ and its translation $M(\phi) = \Box p \vee \Box q$). Both properties (i) and (ii) are proved by induction, for example for (ii):

- Assume $x \models M(\phi \vee \psi)$. Without loss of generality, assume $x \models M(\phi)$. By the inductive assumption, every path starting from x satisfies ϕ, so it satisfies $\phi \vee \psi$.
- Assume $x \models M(\phi \mathsf{U} \psi)$. By definition of M, on every path π starting from x the formula $M(\phi)$ holds in every state y until at some point $M(\psi)$ holds. Now, for each state y the path π has a sub-path that starts at y, and by the inductive assumption ϕ holds for all these subpaths until at some point ψ holds. As a result, $\phi \mathsf{U} \psi$ holds for the path π.

Properties (i) and (ii) immediately imply that $M(\phi)$ is satisfiable if and only if ϕ is. By Theorem 6.1, satisfiability of atomic μ-calculus formulas is undecidable.

C Proof of Theorem 6.3

We reduce the satisfiability problem for atomic LTL, with some insight into the proof of Theorem 6.1. Given any Turing machine $\mathcal{M}$ as considered there, consider a Kripke model with the set of states:

$$(\mathbb{A} \times \Gamma \times (Q \cup \{\texttt{nohead}\})) \cup \{\$\}$$

with basic predicates defined in the obvious way, and with transitions going both ways between every two states. This model is orbit-finite and equivariant.

Now, for the atomic LTL formula ϕ obtained from $\mathcal{M}$ as in the proof of Theorem 6.1, the formula $\exists\phi$ holds in this model in the state \$ if and only if ϕ is satisfiable.

This works regardless of whether we interpret CTL* path formulas over arbitrary paths or over finitely supported ones, because the formula ϕ in the proof of Theorem 6.1 forces its model to be finitely supported.

D Details of the proof of Theorem 7.1

Note that $\hat{\mathcal{K}}$ is indeed equivariant and orbit-finite: every orbit of K gives rise to at most 2^k orbits in $\hat{K}$, where k is the size of the least support of any (equivalently, every) element in the orbit. Moreover, (a representation of) $\hat{\mathcal{K}}$ is computable from $\mathcal{K}$: for each orbit in K one can enumerate all corresponding orbits in $\hat{K}$, and orbits of transitions in $\hat{R}$ are also easy to enumerate.

We prove that a state $x \in K$ admits an infinite path where no p_a holds more than once, if and only if $(x, \emptyset) \in \hat{K}$ admits any infinite path.

For the left-to-right implication, assume an infinite path in $\mathcal{K}$, i.e., a sequence $x = x_0, x_1, x_2, x_3 \ldots$, such that $\langle x_i, x_{i+1}\rangle \in R$ for each $i \in \mathbb{N}$, and $\mathrm{pred}(x_i) \cap \mathrm{pred}(x_j) = \emptyset$ for each $i \neq j \in \mathbb{N}$. Define

$$y_i = \langle x_i, S_i\rangle, \quad \text{where} \quad S_i = \mathrm{supp}(x_i) \cap \bigcup_{j=0}^{i-1} \mathrm{pred}(x_j).$$

In particular, $y_0 = \langle x, \emptyset\rangle$. Then

$$\langle\langle x_i, S_i\rangle, \langle x_{i+1}, S_{i+1}\rangle\rangle \in \hat{R} \tag{3}$$

for each $i \in \mathbb{N}$. Indeed, calculate

$$(S_i \cup \mathrm{pred}(x_i)) \cap \mathrm{supp}(x_{i+1}) = \left(\left(\mathrm{supp}(x_i) \cap \bigcup_{j=0}^{i-1} \mathrm{pred}(x_j)\right) \cup \mathrm{pred}(x_i)\right) \cap \mathrm{supp}(x_{i+1})$$
$$\subseteq \left(\bigcup_{j=0}^{i} \mathrm{pred}(x_j)\right) \cap \mathrm{supp}(x_{i+1}) = S_{i+1}.$$

As a result, the pairs

$$(\langle x, \emptyset\rangle =) \langle x_0, S_0\rangle, \langle x_1, S_1\rangle, \langle x_2, S_2\rangle, \dots \tag{4}$$

form an infinite path in $\hat{\mathcal{K}}$.

For the right-to-left implication, assume any infinite sequence as in (4), for some x_i and S_i such that the condition (3) holds for every $i \in \mathbb{N}$. We construct sequences

$$y_0, y_1, \dots \in K \quad T_0, T_1, \dots \subseteq \mathbb{A} \quad \pi_1, \pi_2, \dots \in \mathrm{Aut}(\mathbb{A})$$

by simultaneous induction as follows:

- $y_0 = x_0$ and $T_0 = S_0$,
- π_{i+1} is an atom automorphism such that:
  - $\pi_{i+1}(a) = a$ for $a \in \mathrm{supp}(y_i)$, and
  - $\pi_{i+1}(a) \notin \bigcup_{j=0}^{i} \mathrm{supp}(y_j)$ for $a \in \mathrm{supp}(x_{i+1} \cdot \pi_1\pi_2\cdots\pi_i) \setminus \mathrm{supp}(y_i)$,

  and acting in an arbitrary way on the remaining atoms,
- $y_{i+1} = x_{i+1} \cdot \pi_1\pi_2\cdots\pi_i\pi_{i+1}$,
- $T_{i+1} = S_{i+1} \cdot \pi_1\pi_2\cdots\pi_i\pi_{i+1}$.

Notice that, since $\langle x_{i+1}, S_{i+1}\rangle$ is a legal state in $\hat{K}$, by equivariance so is $\langle y_{i+1}, T_{i+1}\rangle$. Moreover,

$$\langle\langle y_i, T_i\rangle, \langle y_{i+1}, T_{i+1}\rangle\rangle \in \hat{R}$$

for each $i \in \mathbb{N}$, therefore the sequence

$$\langle y_0, T_0\rangle, \langle y_1, T_1\rangle, \langle y_2, T_2\rangle, \dots$$

forms an infinite path in $\hat{K}$. To see this, note that (by equivariance of $\hat{R}$)

$$\langle\langle y_i \cdot \pi_{i+1}, T_i \cdot \pi_{i+1}\rangle, \langle y_{i+1}, T_{i+1}\rangle\rangle \in \hat{R}, \qquad y_i \cdot \pi_{i+1} = y_i \quad \text{and} \quad T_i \cdot \pi_{i+1} = T_i$$

since π_{i+1} by definition fixes $\mathrm{supp}(y_i)$ and $T_i \subseteq \mathrm{supp}(y_i)$.

As a consequence, the sequence $y_0, y_1, y_2, \dots$ forms a path in $\mathcal{K}$. A useful property of this path, easy to infer from the definition of y_i, is that:

$$(\mathrm{supp}(y_{i+1}) \setminus \mathrm{supp}(y_i)) \cap \bigcup_{j=0}^{i} \mathrm{supp}(y_j) = \emptyset. \tag{5}$$

In words, whenever a locally fresh atom appears in some y_i, then it does not appear anywhere earlier in the path.

We shall show that no predicate p_a holds on this path more than once. Assume towards a contradiction that $a \in \mathrm{pred}(y_i) \cap \mathrm{pred}(y_j)$ for some $a \in \mathbb{A}$ and $i < j$. Then obviously $a \in \mathrm{supp}(y_i)$ and $a \in \mathrm{supp}(y_j)$, and by induction on the difference $j - i$, using (5), we get that $a \in \mathrm{supp}(y_k)$ for all k between i and j. Again by induction, and by definition of $\hat{R}$, a belongs to all sets $T_{i+1}, T_{i+2}, \dots, T_j$. But this means that $a \in T_j \cap \mathrm{pred}(y_j)$, which contradicts the fact that $\langle y_j, T_j\rangle$ is a legal state in $\hat{K}$. This completes the proof of the right-to-left implication, and of the entire theorem.