Temporal logics over linear time domains are in PSPACE

Alexander Rabinovich

The Blavatnik School of Computer Science
Tel Aviv University, Tel Aviv, Israel 69978

Abstract

We investigate the complexity of the satisfiability problem of temporal logics with a finite set of modalities definable in the existential fragment of monadic second-order logic. We show that the problem is in PSPACE over the class of all linear orders. The same techniques show that the problem is in PSPACE over many interesting classes of linear orders.

1. Introduction

A major result concerning linear-time temporal logics is Kamp's theorem [12, 10, 9] which states that $TL(\mathsf{Until}, \mathsf{Since})$, the temporal logic having Until and Since as the only modalities, is expressively complete for first-order monadic logic of order over the class of Dedekind-complete linear orders.

The order of natural numbers $\omega = (\mathbb{N}, <)$ and the order of the real numbers $(\mathbb{R}, <)$ are both Dedekind-complete. Another important class of Dedekind-complete orders is the class of ordinals. However, the order of the rationals is not Dedekind-complete. Stavi introduced two modalities Until_{Stavi} and Since_{Stavi} and proved that the temporal logic having the four modalities Until, Since, Until_{Stavi} and Since_{Stavi} is expressively complete for first-order monadic logic of order over the class of all linear orders [10, 9].

Our concern in this paper will be with the complexity of the satisfiability problem for temporal logics over various classes of linear orders.

Sistla and Clarke [22] proved that the satisfiability problem for $TL(\mathsf{Until}, \mathsf{Since})$ over ω-models is PSPACE-complete. This proof was based on automata theoretical techniques.

Burgess and Gurevich [5] proved that $TL(\mathsf{Until}, \mathsf{Since})$ is decidable over the reals. They provided two proofs. The first involves an indirect reduction to Rabin's theorem on the decidability of the monadic second-order logic over the full binary tree [14]. The second one is based on the model-theoretical composition method. Both proofs provide algorithms of non-elementary complexity.

Reynolds [17, 16] proved that the satisfiability problem for $TL(\mathsf{Until}, \mathsf{Since})$ over the reals is PSPACE-complete and that the temporal logic with only the Until modality is PSPACE-complete over the class of all linear orders. The proofs in [17, 16] use temporal mosaics and are very non-trivial and difficult to grasp.

One of our objectives was to provide a simple proof of Reynolds' remarkable result [17] obtained in 1999. We also wanted a proof which can be applied to prove in a uniform way a PSPACE upper bound for other time domains. In [7], we showed that the satisfiability problem for $TL(\mathsf{Until}, \mathsf{Since})$ over the class of all ordinals is PSPACE-complete. This proof was based on automata theoretical techniques, and it is considerably simpler than Reynolds' proof of PSPACE-completeness for the satisfiability problem for $TL(\mathsf{Until}, \mathsf{Since})$ over the reals. However, the ordinals are simpler than the reals.

Cristau [6] provided a very unexpected translation from the temporal logic having the four modalities Until, Since, Until_{Stavi} and Since_{Stavi} into automata which work over arbitrary linear orders and as a consequence established a double exponential space algorithm for the satisfiability problem of this temporal logic over the class of all linear orders.

Let TL be a temporal logic with a finite set of modalities definable in the existential fragment of monadic second-order logic. We prove in this paper in a uniform manner that the satisfiability problem for TL is in

PSPACE over the following classes of time domains: (1) all linear orders, (2) ordinals, (3) scattered linear orders, (4) Dedekind-complete linear orders, (5) continuous orders, (6) rationals, (7) reals.

The proofs are based both on the composition method and on automata theoretical techniques and are easily adapted to various classes of structures and temporal and modal logics.

Recently, Reynolds [18] proved PSPACE upper bound for most of these classes, by reducing the satisfiability problem for these classes to the satisfiability problem over the reals.

Our proof uses several reductions. The first reduction uses the following notion. Let $\varphi(X_1, \dots, X_k)$ be a formula with free set variables among $X_1, \dots, X_k$. An instance of φ is a formula obtained by replacing $X_1, \dots, X_k$ by monadic predicate names. Let Φ be a set of formulas. A Φ-conjunctive formula is a conjunction of instances of formulas from Φ.

Our first reduction shows that for every temporal logic $\mathcal{L}$ with a finite set of modalities definable in the existential fragment of monadic second-order logic there is a finite set Φ of first-order formulas and a linear time algorithm that reduces the satisfiability problem for $\mathcal{L}$ to the satisfiability problem for Φ-conjunctive formulas. This algorithm is based on a simple unnesting procedure and works as it is for a much broader class of modal logics.

Next, we introduce recursively definable classes of structures. Our second reduction shows that for every finite set Φ of first-order formulas and every recursively definable class of structures $\mathcal{C}$ the satisfiability problem for the Φ-conjunctive formulas over $\mathcal{C}$ is in EXPTIME. Like the first reduction, this reduction is quite general; it relies on the composition method and is sound not only for linear orders. The first two reductions give an almost free EXPTIME algorithm for many temporal and modal logics with finite sets of modalities.

To obtain a PSPACE upper bound we need more subtle arguments. We assign a rank to every structure in a recursively definable class. An algorithm similar to the algorithm in the second reduction shows that for every polynomial p the problem whether a Φ-conjunctive formula φ is satisfiable over the structures of rank $p(|\varphi|)$ is in PSPACE. The main effort to show that the satisfiability problem for a recursively definable class is in PSPACE is to establish that if a formula is satisfiable, then it is satisfiable over the structures of a polynomial rank in the size of the formula. We prove such a bound for many interesting classes of linear orders. Our proof uses an automata-theoretical characterization of the temporal logic with Stavi's modalities over the linear orders found by Cristau [6].

The paper is organized as follows. The next section recalls basic definitions about monadic second-order logic, its fragments and temporal logics. Sect. 3 states a linear reduction from temporal logics to conjunctive formulas. Sect. 4 reviews basic notions about the compositional method. Sect. 5 introduces recursively defined classes of structures and Sect. 6 presents an exponential algorithm for the satisfiability of conjunctive formulas over these classes. Sect. 7 presents a PSPACE algorithm for the satisfiability of conjunctive formulas over the class of all linear orders and states a small rank property lemma needed for its complexity analysis. Sect. 8 introduces finite base automata over arbitrary linear orders. Sect. 9 states the main technical lemma (Lemma 9.1) about runs of automata and proves the small rank property lemma which was used in the proof of PSPACE bound of our algorithm. Sect 10 is the most technical part of the paper. It develops compositional methods for the automata types and proves the main technical lemma. Sect 11 considers temporal logics with any finite set of automata definable modalities and shows that the satisfiability problem for such logics over the class of countable linear orders is in PSPACE. Sect. 12 proves in a "plug-and-play" manner a PSPACE upper bound over several interesting classes of linear orders and discusses related works. Sect. 13 discusses the related results of Mark Reynolds [16, 17, 18]. Sect. 14 contains conclusion and further results.

Our results were obtained in 2007 using only the composition method and the proofs were considerably simplified in July 2009, relying on the automata theoretical results of Cristau [6]. An extended abstract of this paper was published in [15].

2. Monadic Logics and Temporal Logics

2.1. Monadic second-order logic

Monadic second-order logic (MSO) is the fragment of the full second-order logic allowing quantification only over elements and monadic predicates. One way to define the monadic second-order language for a signature Δ (notation MSO(Δ)) is to augment the first-order language for Δ by quantifiable monadic predicate variables (set variables) and by new atomic formulas $X(t)$, where t is a first-order variable and X is a monadic predicate variable. The monadic predicate variables range over all subsets of a structure for Δ.

The *quantifier depth* of a formula φ is defined as usual and is denoted by $\mathrm{qd}(\varphi)$.

We will use lower case letters t, t' for the first-order variables and upper case letters X, Y, Z for the monadic variables.

An MSO formula is existential if it is of the form $\exists X_1 \ldots \exists X_n \varphi$, where φ does not contain second-order quantifiers. The existential fragment of MSO consists of existential MSO formula and is denoted by $\exists$-MSO.

The first-order fragment of MSO contains formulas without the second-order quantifiers. These formulas might contain free second-order variables which play the same role as monadic predicate names. Hence, a formula in this fragment is interpreted over expansions of Δ-structures by predicates which provide meaning for the monadic variables. Sometimes, these free variables will serve as metavariables. If $\varphi(X_1, X_2)$ is a formula and P, Q are monadic predicate names, we will say that the formula obtained from φ by replacing X_1 by P and X_2 by Q is an *instance* of φ.

2.2. Temporal Logics and Truth Tables

Temporal logics use logical constructs called *"modalities"* to create a language free from quantifiers. Below is the general logical framework to define temporal logics:

The syntax of the Temporal Logic $TL(O_1^{(k_1)}, \ldots, O_n^{(k_n)})$ has in its vocabulary *monadic predicate variables* $X_1, X_2, \ldots$ and a sequence of *modality names* with a prescribed arity, $O_1^{(k_1)}, \ldots, O_n^{(k_n)}$ (the arity notation is usually omitted). The formulas of this temporal logic are given by the grammar:

$$\varphi \ ::= \ X \mid \neg\varphi \mid \varphi \wedge \varphi \mid O^{(k)}(\varphi_1, \cdots, \varphi_k)$$

When particular modality names are unimportant or are clear from the context, we omit them and write TL instead of $TL(O_1^{(k_1)}, \ldots, O_n^{(k_n)})$.

Structures for TL are partial orders with monadic predicates $\mathcal{M} = \langle A, <, P_1, P_2, \ldots, P_n, \ldots \rangle$, where the predicate P_i is assigned to a predicate variable X_i. Every modality $O^{(k)}$ is interpreted in every structure $\mathcal{M}$ as an operator $O_{\mathcal{M}}^{(k)} : [\mathcal{P}(A)]^k \to \mathcal{P}(A)$ which assigns "the set of points where $O^{(k)}[S_1 \ldots S_k]$ holds" to the k-tuple $\langle S_1 \ldots S_k \rangle \in \mathcal{P}(A)^k$. (Here, $\mathcal{P}$ is the power set notation, and $\mathcal{P}(A)$ denotes the set of all subsets of the domain A of $\mathcal{M}$.) Once every modality corresponds to an operator, the relation "φ holds in $\mathcal{M}$ at an element a" (denoted $\langle \mathcal{M}, a \rangle \models \varphi$) is defined as follows:

- for atomic formulas $\langle \mathcal{M}, a \rangle \models X$ iff $a \in P$, where the monadic predicate P is assigned to X.
- for Boolean combinations the definition is the usual one.
- for modalities: $\langle \mathcal{M}, a \rangle \models O^{(k)}(\varphi_1, \cdots, \varphi_k)$ iff $a \in O_{\mathcal{M}}^{(k)}(P_{\varphi_1}, \cdots, P_{\varphi_k})$, where $P_\varphi = \{ b \mid \langle \mathcal{M}, b \rangle \models \varphi \}$.

Usually, we are interested in a more restricted case; for the modality to be of interest the operator $O^{(k)}$ should reflect some intended connection between the sets A_{φ_i} of points satisfying φ_i and the set of points $O[A_{\varphi_1}, \ldots, A_{\varphi_k}]$. The intended meaning is usually given by a formula in an appropriate predicate logic.

Truth Tables: A formula $\overline{O}(t_0, X_1, \ldots X_k)$ in the predicate logic L is a *Truth Table* for the modality O if for every structure $\mathcal{M}$ and subsets $P_1, \ldots, P_k$ of $\mathcal{M}$

$$O_{\mathcal{M}}(P_1, \ldots, P_k) = \{a : \ \mathcal{M} \models \overline{O}[a, P_1, \ldots, P_k]\} \ .$$

Thus, the modality $\Diamond X$, "*eventually* X", is defined by

$$\varphi(t_0, X) \equiv \exists t > t_0 (t \in X).$$

The modality $X \mathsf{Until}\, Y$, "X *strict until* Y", is defined by

$$\exists t_1 (t_0 < t_1 \wedge t_1 \in Y \wedge \forall t (t_0 < t < t_1 \to t \in X)).$$

A truth table $\varphi(t, Y_1, \cdots, Y_k)$ defines in every structure a function from k-tuples of subsets. It associates with the tuple $Y_1, \cdots, Y_k$ of subsets of a structure $\mathcal{M}$, the set of elements t in $\mathcal{M}$ that satisfy $\varphi(t, Y_1, \cdots, Y_k)$ in $\mathcal{M}$. This is a special case of a more general way to define a function on all the structures in a given class of structures. Here is the formal notion of a definable functional.

Definition 2.1. 1. *Let L be a first-order or monadic second-order logic language, and let $\mathcal{M}$ be a structure.*
Let $\varphi(X, Y_1, \cdots, Y_k)$ be a formula in L with no free first-order variables, and with no set variables except for those specified. φ is an implicit definition of the functional $X = f_\varphi^{\mathcal{M}}(Y_1, \cdots, Y_k)$ *if for any k subsets $Y_1, \cdots, Y_k$ of $\mathcal{M}$, X is the only subset of $\mathcal{M}$ for which $\mathcal{M} \models \varphi(X, Y_1, \cdots, Y_k)$.*

2. *A modality $\mathcal{O}(Y_1, \cdots, Y_k)$ of a temporal logic has a* generalized truth table $\varphi(X, Y_1, \cdots, Y_k)$ *in a structure $\mathcal{M}$ if φ implicitly defines the operator of $\mathcal{O}$; i.e., given subsets $Y_1, \cdots, Y_k$ of a structure $\mathcal{M}$,*

$$\langle \mathcal{M}, a \rangle \models \mathcal{O}(Y_1, \cdots, Y_k) \quad \textit{iff} \quad a \in f_\varphi^{\mathcal{M}}(Y_1, \cdots, Y_k).$$

φ is a generalized truth table for $\mathcal{O}$ in a class $\mathcal{C}$ of structures if φ is a generalized truth table for $\mathcal{O}$ in every $\mathcal{M} \in \mathcal{C}$.

If the logic is a second-order logic, then this definition is a special case of the classical definition of a function defined by a formula. Note that if $\theta(t_0, Y_1, \cdots, Y_k)$ is a truth table for a modality $\mathcal{O}$, then $\forall t[X(t) \leftrightarrow \theta(t, Y_1, \cdots, Y_k)]$ is a generalized truth table for $\mathcal{O}$. Therefore, the notion of a generalized truth table is more general than that of a truth table. It is strictly more general. For example, it is well known that there is no first-order formula $\varphi(t, X)$ which defines over the naturals the set of points preceded by an even number of points in X; however, it is easy to write a first-order formula $\psi(Y, X)$ which defines this modality over $(\mathbb{N}, <)$.

If a modality O has a generalized truth table $\varphi(X, Y_1, \cdots, Y_k)$, where φ is an existential monadic second-order formula, then $\exists X\big((X(t_0)) \land \varphi\big)$ is an $\exists$-MSO truth table for O. Hence, a modality has an $\exists$-MSO truth table iff it has an $\exists$-MSO generalized truth table and we will say that it is $\exists$-MSO *definable*.

There are $\exists$-MSO definable modalities which are not definable even by generalized truth tables of the first-order logic. For example, there is an $\exists$-MSO formula $\varphi(Y, X)$ that expresses "Y holds at t if $\neg X(t)$ and t is preceded by a block of X of length $3m$ some $m > 0$", i.e., $X(t-1)$, $X(t-2), \ldots X(t-3m)$ and $\neg X(t-3m-1)$. However, there is no first-order formula equivalent to φ over $(\mathbb{N}, <)$.

Modal logics Temporal logics are examples of modal logics. The syntax of modal logics is defined exactly like the syntax of temporal logics. However, modal logics can be interpreted not only over linear or partial orders, but over structures of a more general signature Δ. Every modality $O^{(k)}$ is interpreted in every Δ-structure $\mathcal{M}$ as an operator $O^{(k)}_{\mathcal{M}} : [\mathcal{P}(\mathcal{M})]^k \to \mathcal{P}(\mathcal{M})$. Generalized truth tables are defined by formulas over Δ. We state our results for temporal logics; however, they hold for more general modal logics as well.

3. From Temporal Logic to Conjunctive Formulas

Let $\varphi(X_1, \ldots, X_k)$ be a formula with free set variables among $X_1, \ldots, X_k$. An *instance* of φ is a formula obtained by replacing $X_1, \ldots, X_k$ by monadic predicate names or monadic variables. Let Φ be a set of formulas. A Φ-*conjunctive* formula is a conjunction of instances of formulas from Φ.

Our first reduction shows that for every temporal logic $\mathcal{L}$ with a finite set of $\exists$-MSO definable modalities there is a finite set Φ of first-order formulas and a linear time algorithm that reduces the satisfiability problem for $\mathcal{L}$ to the satisfiability problem for Φ-conjunctive formulas.

Proposition 3.1. *Let TL be a temporal logic with a finite set of modalities. Assume that every modality of TL is $\exists$-MSO definable. Then there is a finite set Φ of first-order formulas, and a linear time algorithm which for every formula $\varphi(P_1, \ldots, P_m) \in TL$ computes a Φ-conjunctive formula $\psi(P_1, \ldots, P_m, Q_1, \ldots, Q_s)$ such that for every structure $\mathcal{M}$ in the signature $\{<, P_1, \ldots, P_m\}$, φ is satisfiable in $\mathcal{M}$ iff ψ is satisfiable in an expansion of $\mathcal{M}$ by monadic predicates (which are the interpretations of $Q_1, \ldots, Q_s$).*

The proof of this proposition is based on a simple unnesting procedure. A similar proposition holds for modal logics.

PROOF. Let $TL := TL(O_1, \dots, O_n)$ be a temporal logic. Assume that O_i has generalized truth table $\exists \overline{Z}_i \alpha_i$, where α_i is a first-order formula.

Let Φ be defined as $\{\alpha_i \mid i = 1, \dots, n\} \cup \{Id, Neg, Conj\}$, where

$$\begin{aligned} Id(Y, X) &:= \forall t Y(t) \leftrightarrow X(t) \\ Neg(Y, X) &:= \forall t Y(t) \leftrightarrow \neg X(t) \\ Conj(Y, X_1, X_2) &:= \forall t Y(t) \leftrightarrow (X_1(t) \wedge X_2(t)) \end{aligned}$$

To every formula $\varphi \in TL$ we assign a Φ-conjunctive formula $Tr(\varphi)$ with a free variable Y and additional free variables in a list $\overline{Z}$ such that the following condition holds.

(*) $\{a \mid \langle \mathcal{M}, a \rangle \models \varphi\}$ is the unique predicate that satisfies $\exists \overline{Z}\, Tr(\varphi)$ in $\mathcal{M}$.

Below we use $\psi\{X/Y\}$ for the formula obtained from ψ when Y is replaced by X.

atomic formulas: $Tr(P_j) := Id(Y, P_j)$.

negation: $Tr(\neg\varphi) := Neg(Y, Z) \wedge Tr(\varphi)\{Z/Y\}$, where Z is a fresh variable.

conjunction: $Tr(\varphi_1 \wedge \varphi_2) := Conj(Y, Z_1, Z_2) \wedge Tr(\varphi_1)\{Z_1/Y\} \wedge Tr(\varphi_2)\{Z_2/Y\}$, where Z_1, Z_2 are fresh variables.

modality: Assume that O is an m-place modality with an $\exists$-MSO generalized truth table $\exists \overline{W} \alpha(Y, X_1, \dots . X_m, \overline{W})$. Define $Tr(O(\varphi_1, \dots, \varphi_m))$ as $\alpha(Y, Z_1, \dots, Z_m, \overline{W}) \wedge \bigwedge_{i=1}^m Tr(\varphi_i)\{Z_i/Y\}$, where Z_i are fresh variables. By the inductive hypothesis for every $i \le m$ there is a list of variables $\overline{U^i}$ such that $\{a \mid \langle \mathcal{M}, a \rangle \models \varphi_i\}$ is the unique predicate that satisfies $\exists \overline{U^i}\, Tr(\varphi_i)$ in $\mathcal{M}$. Without restriction of generality we can assume that $\overline{U^i}$ are disjoint lists of variables and that they are disjoint from $Y, Z_1, \dots, Z_m$. It is easy to see that $\{a \mid \langle \mathcal{M}, a \rangle \models \varphi\}$ is the unique predicate that satisfies $\exists Z_1 \dots \exists Z_m \exists \overline{U^1} \dots \exists \overline{U^m}\, Tr(\varphi)$ in $\mathcal{M}$.

It is clear that Tr is computable in linear time. □

4. Elements of the Composition Method

Our proofs make use of a technique known as the composition method [8, 21, 11, 23]. To fix notations and to aid a reader unfamiliar with this technique, we briefly review the required definitions and results.

4.1. Hintikka formulas and n-types

Let $\mathcal{M}$ and $\mathcal{M}'$ be structures over a relational signature Σ. For $n \in \mathbb{N}$, the structures $\mathcal{M}$ and $\mathcal{M}'$ are said to be $\equiv^n$-equivalent if no first-order sentence of quantifier depth $\le n$ distinguishes between $\mathcal{M}$ and $\mathcal{M}'$; i.e., for every φ of quantifier depth $\le n$:

$$\mathcal{M} \models \varphi \text{ iff } \mathcal{M}' \models \varphi.$$

Lemma 4.1 (Hintikka Lemma). *For $n \in \mathbb{N}$ and a finite relational signature Σ we can compute a* finite *set $Hin^n := Hin^n(\Sigma)$ of sentences of quantifier depth $\le n$ such that:*

1. *For every $\equiv^n$-equivalence class E there is a* unique *$\tau \in Hin^n$ such that for every Σ-structure $\mathcal{M}$: $\mathcal{M} \in E$ if and only if $\mathcal{M} \models \tau$.*
2. *Every sentence with $\mathrm{qd}(\varphi) \le n$ is equivalent to a (finite) disjunction of sentences from Hin^n. There is an algorithm which for every sentence φ computes a finite set $G_\varphi \subseteq Hin^{\mathrm{qd}(\varphi)}$ such that φ is equivalent to the disjunction of all the sentences from G_φ. Moreover, if $\tau \in Hin^{\mathrm{qd}(\varphi)}$ and τ is satisfiable, then $\tau \in G_\varphi$ iff $\tau \to \varphi$ is valid.*

(Note that the algorithm in Lemma 4.1(2) is not efficient in the sense of complexity theory, because its complexity is non-elementary.)

We call any member of Hin^n an *n-Hintikka sentence*. We use τ, τ_i, τ' to range over the Hintikka sentences.

Definition 4.2 (n-Type). *For $n \in \mathbb{N}$ and a Σ-structure $\mathcal{M}$, we denote by $type^n(\mathcal{M})$ the unique member of Hin^n satisfied in $\mathcal{M}$.*

4.2. The ordered sum of chains and of n-types

A (labeled) chain $\mathcal{M}$ is a linear order expanded by monadic predicates; if $\overline{P}$ is a set of monadic predicate names, and the signature of $\mathcal{M}$ is $\{<, \overline{P}\}$, we say $\mathcal{M}$ is a $\overline{P}$-chain. The *concatenation* or *ordered sum* of chains is defined as follows:

Definition 4.3 (Sum of Chains). *Let $\mathcal{I} := (I, <^{\mathcal{I}})$ be a linear order, $l \in \mathbb{N}$, and $\mathfrak{S} := (\mathcal{M}_\alpha \mid \alpha \in I)$ be a sequence of chains, where $\mathcal{M}_\alpha := (A_\alpha, <^\alpha, {P_1}^\alpha, \dots, {P_l}^\alpha)$. Assume that $A_\alpha \cap A_\beta = \emptyset$ whenever $\alpha \neq \beta$ are in I. The* ordered sum *of $\mathfrak{S}$ is the chain*

$$\sum_{\alpha\in\mathcal{I}} \mathcal{M}_\alpha := (\bigcup_{\alpha\in I} A_\alpha, <^{\mathcal{I},\mathfrak{S}}, \bigcup\nolimits_{\alpha\in I} {P_1}^\alpha, \dots, \bigcup_{\alpha\in I} {P_l}^\alpha),$$

where:

If $\alpha, \beta \in I$, $a \in A_\alpha$, $b \in A_\beta$, then $b <^{\mathcal{I},\mathfrak{S}} a$ iff $\beta <^{\mathcal{I}} \alpha$ or $\beta = \alpha$ and $b <^\alpha a$.

If the domains of the $\mathcal{M}_\alpha$'s are not disjoint, replace them with isomorphic chains that have disjoint domains, and proceed as before.

If $\mathcal{I} = (\{0,1\}, <)$ and $\mathfrak{S} = (\mathcal{M}_0, \mathcal{M}_1)$, we denote $\sum_{\alpha\in\mathcal{I}} \mathcal{M}_\alpha$ by $\mathcal{M}_0 + \mathcal{M}_1$.

If $\mathcal{M}_\alpha$ is isomorphic to $\mathcal{M}$ for every $\alpha \in I$, we denote $\sum_{\alpha\in\mathcal{I}} \mathcal{M}_\alpha$ by $\mathcal{M} \times \mathcal{I}$.

The next proposition states that taking ordered sums preserves $\equiv^n$-equivalence.

Lemma 4.4. *Let $n \in \mathbb{N}$. Assume:*

1. *$(I, <^{\mathcal{I}})$ is a linear order,*
2. *$\left(\mathcal{M}^0_\alpha \mid \alpha \in I\right)$ and $\left(\mathcal{M}^1_\alpha \mid \alpha \in I\right)$ are sequences of chains (in the same signature), and*
3. *for every $\alpha \in I$, $\mathcal{M}^0_\alpha \equiv^n \mathcal{M}^1_\alpha$.*

Then, $\sum_{\alpha\in I} \mathcal{M}^0_\alpha \equiv^n \sum_{\alpha\in I} \mathcal{M}^1_\alpha$.

This allows us to define the sum of formulas in $Hin^n(<, P_1, \dots P_l)$ with respect to any linear order.

In particular, this theorem justifies the notation $\tau_0 + \tau_1$ for the n-type of a chain which is the ordered sum of two chains of n-types τ_0 and τ_1, respectively. Similarly, we write $\tau \times \omega$ for the n-type of a sum $\Sigma_{i\in\omega}\mathcal{M}_i$ where all $\mathcal{M}_i$ are of n-type τ; the n-type $\tau \times \omega^{-1}$ is defined similarly, where ω^{-1} is the order type of negative integers.

Another important operation on chains and on n-types is **shuffle**.

Let $\mathfrak{S} := (\mathcal{M}_\alpha \mid \alpha \in \mathbb{Q})$ be a sequence of chains indexed by the rationals. Let $Q_1, \dots, Q_k \subseteq \mathbb{Q}$ be a partition of $\mathbb{Q}$ into k everywhere dense sets. Let $\mathcal{N}_1, \dots, \mathcal{N}_k$ be chains. If for $i = 1, \dots, k$ and $q \in Q_i$, $\mathcal{M}_q$ is isomorphic to $\mathcal{N}_i$, we denote $\sum_{\alpha\in\mathbb{Q}} \mathcal{M}_\alpha$ by *shuffle*$(\mathcal{N}_1, \dots, \mathcal{N}_k)$. Note that different partitions of $\mathbb{Q}$ into k everywhere dense sets are isomorphic; hence, the shuffle is well defined. The corresponding operation on n-types will be also denoted by *shuffle*.

4.3. Additive coloring and uniform labeling

The definition and results of this section will be used in Section 10.

Definition 4.5. 1. *A coloring of a chain $\mathcal{L}$ is a function* $\mathrm{col} : [\mathcal{L}]^2 \to T$ *where* $[\mathcal{L}]^2$ *is the set of unordered pairs of distinct elements of* $\mathcal{L}$ *and* T *is a finite set (the set of colors).*

2. *The coloring* f *is additive if for every* $x_1 < y_1 < z_1$ *and* $x_2 < y_2 < z_2$ *in* $\mathcal{L}$ *the following condition holds:*

$$\mathrm{col}(x_1, y_1) = \mathrm{col}(x_2, y_2) \text{ and } \mathrm{col}(y_1, z_1) = \mathrm{col}(y_2, z_2) \quad \text{implies} \quad \mathrm{col}(x_1, z_1) = \mathrm{col}(x_2, z_2).$$

In this case a partial operation + is well defined on T: $t_1 + t_2 = t$ *iff there are* $x < y < z$ *such that* $\mathrm{col}(x, y) = t_1$, $\mathrm{col}(y, z) = t_2$ *and* $\mathrm{col}(x, z) = t$.

3. *A sub-chain* $D \subseteq \mathcal{L}$ *is homogeneous (for* col*) if there exists* $t_0 \in T$ *such that for every* $x, y \in D$, $\mathrm{col}(x, y) = t_0$.

Let $\mathcal{L}$ be a chain. For $k \in \mathbb{N}$ define $\mathrm{col}_k(x, y)$ as the k-type of the restriction of $\mathcal{L}$ on the interval $[x, y)$. This is an additive coloring of $\mathcal{L}$.

The following theorem is an instance of Theorem 1.1 in [21].

Theorem 4.6 (Ramsey theorem for additive colorings). *Let* $\mathrm{col} : [\mathcal{L}]^2 \to T$ *be an additive coloring where* $\mathcal{L}$ *is order-isomorphic to a limit ordinal, and* T *is finite. Then there is* $H \subseteq \mathcal{L}$*, cofinal and homogeneous for* col.

Definition 4.7. *A labeling of a chain* $\mathcal{L}$ *is a function lab from* $\mathcal{L}$ *into a finite set. An interval* I *of* $\mathcal{L}$ *is uniform for a labeling* $lab : \mathcal{L} \to \Delta$ *if for every* $\delta \in \Delta$*, the set* $\{x \in I \mid lab(x) = \delta\}$ *is either empty or dense in* I*. For* $\Delta' \subseteq \Delta$*, we say that* I *is* Δ'*-uniform if* I *is uniform for lab and* $\Delta' = \{\delta \in \Delta \mid \exists x \in I(lab(x) = \delta)\}$.

Lemma 4.8. *If lab is a labeling of a dense chain* $\mathcal{L}$*, then there is an open non-empty interval* J *which is uniform for lab.*

5. Recursively Defined Classes of Structures

In this section we introduce recursively defined classes of structures. We prove some simple properties of such classes. In the next section we show that the satisfiability problem of conjunctive formulas over recursively defined classes of structures is in EXPTIME.

Let Δ be a signature and $k \in \mathbb{N}$. A k-ary Δ-*operator* is a function F which assigns to every k-tuple of Δ-structures a Δ-structure. A *finite-set* Δ-*operator* is a function F which assigns to every finite set of Δ-structures a Δ-structure. A Δ-*operator* is a k-ary ($k \in \mathbb{N}$) or a finite-set Δ-operator.

Let $\mathcal{C}$ be a set of Δ-structures. $\mathcal{C}$ is closed under a Δ-operator F if the result of application of F to structures from $\mathcal{C}$ is in $\mathcal{C}$.

Let $\mathcal{C}$ be a set of Δ-structures and $\mathfrak{F}$ be a family of Δ-operators. The *closure* of $\mathcal{C}$ under $\mathfrak{F}$ is the minimal class $\mathcal{C}'$ of Δ-structure which contains $\mathcal{C}$ and is closed under $\mathfrak{F}$. We denote this class by $Cl(\mathcal{C}, \mathfrak{F})$. It is said to be *recursively defined* from $\mathcal{C}$ by $\mathfrak{F}$.

Let $Cl^0(\mathcal{C}, \mathfrak{F}) := C$ and for $i \in \mathbb{N}$ define $Cl^{i+1}(\mathcal{C}, \mathfrak{F}) := Cl^i(\mathcal{C}, \mathfrak{F}) \cup \{\mathcal{M} \mid \mathcal{M} = F(\mathcal{M}_1, \ldots, \mathcal{M}_k)$ for k-ary $F \in \mathfrak{F}$ and $\mathcal{M}_j \in Cl^i(\mathcal{C}, \mathfrak{F})\} \cup \{\mathcal{M} \mid \mathcal{M} = F(\mathcal{A})$ for finite-set operator $F \in \mathfrak{F}$ and $\mathcal{A} \subseteq Cl^i(\mathcal{C}, \mathfrak{F})\}$. Define $Cl^*(\mathcal{C}, \mathfrak{F}) := \cup_{i \in \mathbb{N}} Cl^i(\mathcal{C}, \mathfrak{F})$. Note that $Cl^*(\mathcal{C}, \mathfrak{F}) = Cl(\mathcal{C}, \mathfrak{F})$.

Let $\sim$ be an equivalence on Δ-structures. The *index* of $\sim$ is the cardinality of the set of $\sim$-equivalence classes; $\sim$ has a *finite index* if there are only finitely many $\sim$-equivalence classes.

A k-ary Δ-operator F *respects* $\sim$ if for Δ-structures $\mathcal{M}_1, \ldots, \mathcal{M}_k, \mathcal{N}_1, \ldots, \mathcal{N}_k$

$$F(\mathcal{M}_1, \ldots, \mathcal{M}_k) \sim F(\mathcal{N}_1, \ldots, \mathcal{N}_k)$$

whenever $\mathcal{M}_i \sim \mathcal{N}_i$ $(i = 1, \ldots, k)$.

If F respects $\sim$, then it induces a k-ary operation on the $\sim$-equivalence classes. We denote this operation by F as it will always be clear from the context whether we use an operator on Δ-structures or the corresponding operation on the $\sim$-equivalence classes.

If $\mathcal{A}$ and $\mathcal{B}$ are sets of Δ-structures, we say that $\mathcal{A}$ is $\sim$-equivalent to $\mathcal{B}$ if $\forall \mathcal{M} \in \mathcal{A}\ \exists \mathcal{N} \in \mathcal{B}(\mathcal{M} \sim \mathcal{N})$ and $\forall \mathcal{M} \in \mathcal{B}\ \exists \mathcal{N} \in \mathcal{A}(\mathcal{M} \sim \mathcal{N})$.

A finite-set Δ-operator *respects* $\sim$ if $F(\mathcal{A}) \sim F(\mathcal{B})$ whenever $\mathcal{A} \sim \mathcal{B}$.

If a finite-set operator F respects $\sim$, then it induces an operation which assigns a $\sim$-equivalence class to every finite subset of $\sim$-equivalence classes.

A family $\mathfrak{F}$ of Δ-operators *respects* $\sim$ if every operator in $\mathfrak{F}$ respects $\sim$.

Lemma 5.1. *Assume that $\sim$ is an equivalence of finite index l, and $\mathfrak{F}$ respects $\sim$. Then for every $\mathcal{M} \in Cl(\mathcal{C}, \mathfrak{F})$ there is $\mathcal{N} \in Cl^l(\mathcal{C}, \mathfrak{F})$ such that $\mathcal{M} \sim \mathcal{N}$.*

Proof. Let E_n be the set of $\sim$-equivalence classes of structures from $Cl^n(\mathcal{C}, \mathfrak{F})$. Then, $\forall n E_n \subseteq E_{n+1}$. Hence, there is $i \leq l$ such that $E_i = E_{i+1}$. This implies that $\forall j E_i = E_{i+j}$. In particular, $\forall j E_l \supseteq E_j$, therefore, the lemma holds. □

For every n the set of operators $\{+, \times\omega, \times\omega^{-1}, \mathit{shuffle}\}$ respects $\equiv^n$.

Strictly speaking, these are polymorphic operators. For every set $\overline{P}$ of monadic predicate names, there is a corresponding binary operator $+$ on $\overline{P}$-labeled chains.

Recall that for a Δ-structure $\mathcal{M}$ and $\Delta' \subseteq \Delta$ the Δ' reduct of $\mathcal{M}$ on Δ' is a Δ'-structure which has the same domain as $\mathcal{M}$ and the same interpretation of symbols from Δ'. We denote by $\mathcal{M}|\Delta'$ the reduct of $\mathcal{M}$ on Δ'.

The reduct distributes over the sum in the following sense:

The reduct distributes over +

Let $\overline{P'} \subseteq \overline{P}$ be sets of monadic predicate names, let $\mathcal{M}$ and $\mathcal{N}$ be $\overline{P}$-chains. Then $(\mathcal{M}+\mathcal{N})|\{<, \overline{P'}\}$ and $(\mathcal{M}|\{<, \overline{P'}\}) + (\mathcal{N}|\{<, \overline{P'}\})$ are isomorphic.

The reduct also distributes over $\{\times\omega, \times\omega^{-1}, \mathit{shuffle}\}$.

Let $\overline{P}$ be a set of monadic predicate names, let $\overline{P}_1, \dots, \overline{P}_k \subseteq \overline{P}$ be a sequence of subsets of $\overline{P}$, and let $\mathcal{M}$ be a $\overline{P}$-chain. Define $\mathit{ptype}^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k))$, the *product n-type* of $\mathcal{M}$ with respect to $\overline{P}_1, \dots, \overline{P}_k$, as

$$\mathit{ptype}^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k)) := (\tau_1, \dots, \tau_k)\,,$$

where $\tau_i = \mathit{type}^n(\mathcal{M}|\{<, \overline{P}_i\})$ are the n-types of the reducts.

For a class $\mathcal{C}$ of $\overline{P}$-chains,

$$\mathit{ptype}^n(\mathcal{C}; (\overline{P}_1, \dots, \overline{P}_k)) := \{\mathit{ptype}^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k)) \mid \mathcal{M} \in \mathcal{C}\}.$$

Lemma 5.2.

1. *If $\mathit{ptype}^n(\mathcal{M}^i; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1^i, \dots, \tau_k^i)$ for $i \in \{0,1\}$, then*

$$\mathit{ptype}^n(\mathcal{M}^0 + \mathcal{M}^1; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1^0 + \tau_1^1, \dots, \tau_k^0 + \tau_k^1)$$

2. *If $\mathit{ptype}^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1, \dots, \tau_k)$, then*

$$\mathit{ptype}^n(\mathcal{M} \times \omega; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1 \times \omega, \dots, \tau_k \times \omega)$$

$$\mathit{ptype}^n(\mathcal{M} \times \omega^{-1}; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1 \times \omega^{-1}, \dots, \tau_k \times \omega^{-1})$$

3. *If $\mathcal{A}$ is a finite set of structures and for $j = 1, \dots k$, and $U_j = \{\tau_j \mid \mathit{ptype}^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1, \dots, \tau_j, \dots, \tau_k) \wedge \mathcal{M} \in \mathcal{A}\}$, then $\mathit{ptype}^n(\mathit{shuffle}(\mathcal{A}); (\overline{P}_1, \dots, \overline{P}_k)) = (\mathit{shuffle}(U_1), \dots, \mathit{shuffle}(U_k))$.*

6. EXPTIME Algorithm

In this section we present an EXPTIME algorithm for the satisfiability of conjunctive formulas over recursively defined classes of structures. Then we apply a theorem of Läuchli and Leonard [13] (see Theorem 6.5) to derive that the satisfiability of conjunctive formulas over the class of all linear orders is in EXPTIME.

Let Φ be a finite set of formulas of quantifier depth $\leq n$ in the first-order monadic logic over $\{<\}$ with free variables among $X_1, \dots, X_m$.

Let $\psi = \varphi_1(\overline{P_1}) \wedge \dots \wedge \varphi_k(\overline{P_k})$ be a Φ-conjunctive formula. Let $\mathfrak{F} := \{+, \times\omega, \times\omega^{-1}, \mathit{shuffle}\}$. Let $\mathcal{C}$ be a set of structures over signature $\{<, \cup_{i=1}^{k} \overline{P_i}\}$. Recall that $\mathfrak{F}$ respects $\equiv^n$, therefore, by Lemma 5.1, ψ is satisfiable over $Cl(\mathcal{C}, \mathfrak{F})$ if it is satisfiable over $Cl^l(\mathcal{C}, \mathfrak{F})$, where $l := |Hin^n(<, \cup_{i=1}^{k} \overline{P_i})|$ is the cardinality of the set $Hin^n(<, \cup_{i=1}^{k} \overline{P_i})$ of Hintikka formulas. This l grows like the n-time iterated exponential function $\exp(n, k)$ ($\exp(1, x) := 2^x$ and $\exp(i+1, x) := 2^{\exp(i,x)}$). We replace this bound by a bound exponential in k and derive an exponential time algorithm for the satisfiability of Φ-conjunctive formulas over $Cl(\mathcal{C}, \mathfrak{F})$. Our arguments are valid not only for this recursively defined class, but for any recursive class which is definable by a finite set of operators that respect $\equiv^n$-equivalence and satisfy an analog of Lemma 5.2.

Lemma 6.1. *Let Φ be a finite set of formulas of quantifier depth $\leq n$ in the first-order monadic logic over $\{<\}$ with free variables among $X_1, \dots, X_m$. A Φ-conjunctive formula $\varphi_1(\overline{P_1}) \wedge \dots \wedge \varphi_k(\overline{P_k})$ is satisfiable in $\mathcal{M}$ if and only if $ptype^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k)) = (\tau_1, \dots, \tau_k)$ and $\tau_i(\overline{P_i}) \to \varphi_i(\overline{P_i})$ is valid for $i = 1, \dots, k$.*

Define the equivalence $\sim^n_{(\overline{P}_1,\dots,\overline{P}_k)}$ on chains over the signature $\{<, \cup_{i=1}^{k} \overline{P_i}\}$ as $\mathcal{M} \sim^n_{(\overline{P}_1,\dots,\overline{P}_k)} \mathcal{N}$ iff $ptype^n(\mathcal{M}; (\overline{P}_1, \dots, \overline{P}_k)) = ptype^n(\mathcal{N}; (\overline{P}_1, \dots, \overline{P}_k))$. The number of $\sim^n_{(\overline{P}_1,\dots,\overline{P}_k)}$-equivalence classes is $\leq |Hin^n(<, P_1, \dots, P_m)|^k$; hence, it is at most exponential in k. $\mathfrak{F}$ respects $\sim^n_{(\overline{P}_1,\dots,\overline{P}_k)}$. Therefore, by Lemma 5.1, we obtain:

Lemma 6.2. *For every finite set Φ of first-order formulas there is c_Φ such that a Φ-conjunctive formula $\psi = \varphi_1(\overline{P_1}) \wedge \dots \wedge \varphi_k(\overline{P_k})$ is satisfiable in $Cl(\mathcal{C}, \mathfrak{F})$ iff it is satisfiable in $Cl^{{c_\Phi}^k}(\mathcal{C}, \mathfrak{F})$.*

Consider the following problem.

Membership Problem for fixed $n, m \in \mathbb{N}$; all tuples $\overline{P}_i$ are of length $\leq m$.

Input: $\overline{\tau} = (\tau_1 \dots \tau_k) \in Hin^n(<, \overline{P}_1) \times \dots \times Hin^n(<, \overline{P}_k)$ and an oracle I for membership in $ptype^n(\mathcal{C}; (\overline{P}_1, \dots, \overline{P}_k))$.

Question: Is $\overline{\tau}$ in $ptype^n(Cl(\mathcal{C}, \mathfrak{F}); (\overline{P}_1, \dots, \overline{P}_k))$?

Lemma 6.3. *The membership problem is in EXPTIME^I.*

PROOF. Our algorithm is presented below.

Algorithm 1 Membership Problem is in EXPTIME^I

$R \leftarrow I$ { i.e., for every $\overline{\tau}$ if $\overline{\tau} \in I$ then add $\overline{\tau}$ to R.}
Updated $\leftarrow$ True.
while Updated **do**
1. Updated $\leftarrow$ False;
2. Compute $R' = Cl^1(R, +)$; **If** $R' \neq R$ **then** Updated $\leftarrow$ True;
3. $R \leftarrow R'$; Compute $R' = Cl^1(R, \times\omega)$; **If** $R' \neq R$ **then** Updated $\leftarrow$ True;
4. $R \leftarrow R'$; Compute $R' = Cl^1(R, \times\omega^{-1})$; **If** $R' \neq R$ **then** Updated $\leftarrow$ True;
5. $R \leftarrow R'$; Compute $R' = Cl^1(R, \mathit{shuffle})$; **If** $R' \neq R$ **then** Updated $\leftarrow$ True;

end while
if $\overline{\tau} \in R$ **return** True.

Let $N_0 = |Hin^n(<, X_1, \ldots, X_m)|$. The number of iterations of the loop is bounded by N_0^k.

$R' = Cl^1(R, +)$ can be computed in time $O(N_0^{2k})$ as follows. Let $R' \leftarrow R$. For each pair $\overline{\tau} = (\tau_1, \ldots, \tau_k), \overline{\tau'} = (\tau'_1, \ldots, \tau'_k) \in R$ add $(\tau_1 + \tau'_1, \ldots, \tau_k + \tau'_k)$ to R'. Hence, Step 2 can be implemented in time $O(N_0^{2k})$.

Steps 3 and 4 can be implemented in $O(N_0^k)$.

The computation of $R' = Cl^1(R, shuffle)$ is more subtle. Indeed, a naive approach can try to compute *shuffle* for every subset of R. However, the number of such subsets is $2^{{N_0}^k}$ and it is double-exponential. Algorithm 2 computes $R' = Cl^1(R, shuffle)$ in EXPTIME.

Algorithm 2 Computation of $Cl^1(R, shuffle)$

Let $H_i := \mathcal{P}(Hin^n(<, \overline{P_i}))$ be the set of subsets of $Hin^n(<, \overline{P_i})$.
for every $U = (U_1, \ldots, U_k) \in H_1 \times \cdots \times H_k$ **do**
{ *Check if there is a sequence* $(\tau_1^1, \ldots, \tau_k^1), \ldots, (\tau_1^m, \ldots, \tau_k^m) \in R$ *such that* $U_i = \{\tau_i^j \mid j \leq m\}$ *and update* R' *as follows:* }

1. $(B_1, \ldots, B_k) \leftarrow (U_1, \ldots, U_k)$;
2. **for** every $\overline{\tau} = (\tau_1, \ldots \tau_k) \in R$ **if** $\wedge_i \tau_i \in U_i$ **then** $B_i \leftarrow B_i \setminus \{\tau_i\}$;
3. **If** $\wedge_{i \leq k}(B_i = \emptyset)$ **then** {such a sequence exists, and we have to update R'}
   $R' \leftarrow R' \cup \{(shuffle(U_1), \ldots, shuffle(U_k))\}$;

end for

The number of iterations of the external loop of Algorithm 2 is $2^{N_0 k}$ and the number of iterations of the internal loop is bounded by N_0^k. Hence, Step 5 can be implemented in time $O(2^{N_0 k} \times N_0^k)$.

Since every step can be implemented in EXPTIME and the number of iterations is exponential, we obtain that the membership problem is in EXPTIME with the oracle I. □

Let *One* be the class of one-element chains. It is clear that we can decide in EXPTIME, whether $\tau \in ptype^n(One; (\overline{P}_1, \ldots, \overline{P}_k))$. Hence, as a consequence of Lemma 6.3, we obtain:

Proposition 6.4. *The satisfiability problem for Φ-conjunctive formulas over the class $Cl(One, \mathfrak{F})$ is in EXPTIME.*

PROOF (OF PROPOSITION 6.4). For every $\varphi \in \Phi$ we can pre-compute the set $H_\varphi := \{\tau \in Hin^n(<, X_1, \ldots, X_m) \mid \tau \rightarrow \varphi\}$ (this depends only on Φ and is independent from the input).

Let $\psi = \varphi_1(\overline{P_1}) \wedge \cdots \wedge \varphi_k(\overline{P_k})$ be a Φ-conjunctive formula. First compute the set S of all $\overline{\tau}$ in $ptype^n(Cl(One, \mathfrak{F}); (\overline{P}_1, \ldots, \overline{P}_k))$. The cardinality of S is at most exponential. By the previous lemma, S can be computed in EXPTIME. Then, by Lemma 6.1, it is enough to check whether there is $(\tau_1, \ldots, \tau_k) \in S$ such that $\tau_i(\overline{P_i}) \rightarrow \varphi_i(\overline{P_i})$ for $i = 1, \ldots, k$. This can be done in EXPTIME using the pre-computed sets H_φ. □

Läuchli and Leonard [13] proved the following theorem:

Theorem 6.5. *A first-order formula is satisfiable over a linear order if it is satisfiable over $Cl(One, \mathfrak{F})$.*

Actually, in [13] the logic with the order relation only was considered. However, its proof can be adapted easily to the first-order monadic logic over chains [20, 5].

As a consequence of Theorem 6.5 and Propositions 6.4 and 3.1 we obtain:

Theorem 6.6. *Let TL be a temporal logic with a finite set of ∃-MSO definable modalities. The satisfiability problem for TL over the class of chains is in* EXPTIME.

In the next section we will show that this EXPTIME upper bound can be replaced by a PSPACE upper bound.

Let us conclude this section by a remark on optimality of our algorithm. The only properties of operators $\{+, \times\omega, \times\omega^{-1}, shuffle\}$ which were used in our EXPTIME algorithm are (1) they respect $\equiv^n$ and (2) the reduct distributes over these operators. If $\mathfrak{F}$ is any set of operators with these properties, then the membership problem for $Cl(One, \mathfrak{F})$ is in EXPTIME.

Below we will show that for such $\mathfrak{F}$ in general, the EXPTIME bound cannot be improved.

Let $\Delta_2 = \{<, Left, Right\}$ be a signature, where $<$ is a binary predicate and $Left, Right$ are unary predicates. We will interpret Δ_2 over the binary trees, where $<$ is the ancestor relation and *Left* (respectively, *Right*) are interpreted as the set of left (respectively, right) children. Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be binary trees expanded by unary predicates $P_1, \dots, P_k$, and let R be a one-element chain in the signature $\{<, P_1, \dots, P_k\}$. We assume that the domains of $\mathcal{M}_1, \mathcal{M}_2$ and R are disjoint and define a ternary operation $\boxplus(\mathcal{M}_1, R, \mathcal{M}_2)$ as follows. $\boxplus(\mathcal{M}_1, R, \mathcal{M}_2)$ is a binary tree; its domain is the union of the domains of $\mathcal{M}_1$, R and $\mathcal{M}_2$; the unique node r of R is the root of this tree. The left and right subtrees of r are $\mathcal{M}_1$ and $\mathcal{M}_2$ respectively. Predicate name P_i is interpreted as the union of its interpretations in $\mathcal{M}_1, R$ and $\mathcal{M}_2$.

The operation $\boxplus$ has properties (1) and (2). The closure of *One* under $\boxplus$ is the set of all finite binary trees. As a consequence, we can derive that the satisfiability problem for any temporal logics with a finite set of $\exists$-MSO definable modalities over the class of finite binary trees is in EXPTIME. Note that CTL can be described as a temporal logic with a finite set of modalities definable in $\exists$-MSO and the satisfiability problem for CTL over the class of finite binary trees is EXPTIME hard. Hence, in general our EXPTIME upper bound for the satisfiability problem over recursively definable classes is optimal.

7. PSPACE Algorithm

In this section we assign a rank to every structure in a recursively definable class. We modify the EXPTIME algorithm from Sect. 6 and show that for every polynomial p the problem whether a Φ-conjunctive formula φ is satisfiable over the structures of rank $p(|\varphi|)$ is in PSPACE. The main effort to show that the satisfiability problem for a recursively definable class is in PSPACE is to establish the *small rank property*: if a conjunctive formula is satisfiable, then it is satisfiable over a structure of a polynomial rank in the size of the formula. Lemma 7.2 states the small rank property; however its proof will be given in Sect. 9.

Let $\mathfrak{F} = \{+, \times\omega, \times\omega^{-1}, shuffle\}$. To every chain in $Cl(One, \mathfrak{F})$ we assign a natural number - the rank of the chain. Define sets $\mathcal{C}^{\leq i} \subseteq Cl(One, \mathfrak{F})$ as follows:

1. $\mathcal{C}^{\leq 0}$ is the set of finite chains.
2. $\mathcal{C}^{\leq i+1}$ is the closure under $+$ of the union of $\mathcal{C}^{\leq i}$, $\{\mathcal{M} \times \omega \mid \mathcal{M} \in \mathcal{C}^{\leq i}\}$, $\{\mathcal{M} \times \omega^{-1} \mid \mathcal{M} \in \mathcal{C}^{\leq i}\}$ and $\{shuffle(\mathcal{A}) \mid \mathcal{A}$ is a finite subset of $\mathcal{C}^{\leq i}\}$.

A chain $\mathcal{M}$ has *rank* $i+1$ if $\mathcal{M} \in \mathcal{C}^{\leq i+1} \wedge \mathcal{M} \notin \mathcal{C}^{\leq i}$.

Every chain of a finite rank can be described by its finite construction tree. Let $\overline{P}$ be a set of monadic predicate names. A construction tree T for $\overline{P}$-chains is a labeled tree which has the following properties: the leaves of T are labeled by one-element $\overline{P}$-chains; the internal nodes are labeled by $+, \times\omega, \times\omega^{-1}$ and *shuffle*; a node labeled by $\times\omega$ or by $\times\omega^{-1}$ has one child; a node labeled by $+$ has at least two children and these children are linearly ordered; a node labeled by *shuffle* has at least one child.

Let T be a construction tree. A chain $[|T|]$, assigned to T, is defined as follows:

1. if T is a one-element tree then $[|T|]$ is the one-element chain which is the label of its only node.
2. If the root of T is labeled by $\times\omega$ (or by $\times\omega^{-1}$), then $[|T|]$ is $[|T_1|] \times \omega$ (respectively, $[|T_1|] \times \omega^{-1}$) where T_1 is the subtree of T rooted at the child of its root.
3. If the root of T is labeled by $+$ and its children (ordered from younger to older) are trees $T_1, \dots, T_m$ then $[|T|] := [|T_1|] + \dots + [|T_m|]$.
4. If the root of T is labeled by *shuffle* and its children are trees $T_1, \dots, T_m$ then $[|T|] := shuffle([|T_1|], \dots, [|T_m|])$.

Lemma 7.1. *If a chain $\mathcal{M}$ has rank $\leq i$, then there is a chain construction tree T such that $\mathcal{M} = [|T|]$ and the height of T is bounded by $2i+1$.*

PROOF. A chain $\mathcal{M}$ has rank $\leq i$ if there is a tree T such that $\mathcal{M} = [|T|]$ and the number of nodes labeled by $\times\omega$, $\times\omega^{-1}$ and *shuffle* on any path from the root to a leaf is bounded by i (we do not count nodes labeled by $+$). For every tree T there is a tree T' such that $[|T'|] = [|T|]$ and no $+$ node has a child labeled by $+$. Indeed, if a $+$ node v of T has as a child a $+$ node u we can remove u and make its children to be children of v (between the left and the right brothers of u). Hence, if a chain $\mathcal{M}$ has rank $\leq i$ then there is a tree T such that $\mathcal{M} = [|T|]$ and the height of T is bounded by $2i{+}1$. □

We are going to present a PSPACE algorithm for the satisfiability problem for Φ-conjunctive formulas. Its correctness and complexity analyses are based on the following Lemma which refines Lemma 6.2 and will be proved in Sect. 9.

Lemma 7.2 (small rank property). *For every finite set Φ of first-order formulas there is r_Φ such that every Φ-conjunctive formula $\psi = \varphi_1(\overline{P_1}) \wedge \cdots \wedge \varphi_k(\overline{P_k})$ is satisfiable in $Cl(One, \mathfrak{F})$ iff it is satisfiable in a chain of rank $\leq k \times r_\Phi$.*

By Theorem 6.5, Lemmas 7.1, and 7.2, $\varphi_1(\overline{P_1}) \wedge \cdots \wedge \varphi_k(\overline{P_k})$ is satisfiable iff

(A) there is a chain construction tree T of height $\leq 2k \times r_\Phi + 1$ such that $ptype^n([|T|]; (\overline{P}_1, \ldots, \overline{P}_k)) = (\tau_1, \ldots, \tau_k)$ and

(B) $\tau_i \rightarrow \varphi_i$ for $i = 1, \ldots, k$.

Now, we are ready to improve our EXPTIME bound of Theorem 6.6 to PSPACE.

Theorem 7.3. *Let TL be a temporal logic with a finite set of $\exists$-MSO definable modalities. The satisfiability problem for TL over the class of chains is in* PSPACE.

By proposition 3.1 it is sufficient to provide a PSPACE algorithm for the satisfiability of Φ-conjunctive formulas. Let $\psi = \varphi_1(\overline{P_1}) \wedge \cdots \wedge \varphi_k(\overline{P_k})$ be such a formula. Our algorithm guesses $(\tau_1, \ldots, \tau_k)$ and checks in linear time condition (B). Then the non-deterministic algorithm SAT, defined below, checks (A). SAT works in polynomial space in k, assuming that the last argument is polynomial in k which is the case with $N = 2k \times r_\Phi + 1$. Fig. 1 contains the definition of the algorithm SAT (some details are omitted).

Input 1. $(\tau_1, \ldots, \tau_k)$, where $\tau_i \in Hin^n(<, \overline{P}_i)$ and $\overline{P}_i \subseteq \overline{P}$ are sets of l predicate names (note that n and l are fixed and are not part of the input).
2. $N \in \mathbb{N}$.

Output True, if there is a construction tree T of height $\leq N$ such that $ptype^n([|T|]; (\overline{P}_1, \ldots, \overline{P}_k)) = (\tau_1, \ldots, \tau_k)$.

- If $N = 0$ and there is a one-element chain $\mathcal{M}$ such that $ptype^n(\mathcal{M}; (\overline{P}_1, \ldots, \overline{P}_k)) = (\tau_1, \ldots, \tau_k)$ then return True;
- Go non-deterministically to 1-5.

  (1.) Return SAT$((\tau_1, \ldots, \tau_k), N-1)$.
  (2.) Guess $(\tau'_1, \ldots, \tau'_k)$ such that SAT$((\tau'_1, \ldots, \tau'_k), N-1)$ returns True and $\tau_i = \tau'_i \times \omega$ for $0 < i \leq k$.
  (3.) Guess $(\tau'_1, \ldots, \tau'_k)$ such that SAT$((\tau'_1, \ldots, \tau'_k), N-1)$ returns True and $\tau_i = \tau'_i \times \omega^{-1}$ for $0 < i \leq k$.
  (4.) Guess on-the-fly a sequence

  $$(\tau_1^1, \ldots, \tau_k^1), (\tau_1^2, \ldots, \tau_k^2), \ldots, (\tau_1^m, \ldots, \tau_k^m)$$

  such that
  (4.1) for $0 < i \leq m$, SAT$((\tau_1^i, \ldots, \tau_k^i), N-1)$ returns True,
  (4.2) for $0 < j \leq k$, $\tau_j = \tau_j^1 + \ldots + \tau_j^m$.

(5.) Guess $(U_1, \dots, U_k)$, where $U_i \subseteq \mathit{Hin}^n(<, \overline{P_i})$ such that

(5.1) for $0 < j \leq k$, $\tau_j = \mathit{shuffle}(U_j)$

and guess on-the-fly a sequence

$$\left(\tau_1^1, \dots, \tau_k^1\right), \left(\tau_1^2, \dots, \tau_k^2\right), \dots, \left(\tau_1^m, \dots, \tau_k^m\right)$$

such that

(5.2) for $0 < i \leq m$, $\mathrm{SAT}(\left(\tau_1^i, \dots, \tau_k^i\right), N-1)$ returns True,

(5.3) for $0 < j \leq k$, $U_j = \{\tau_j^i \mid i \leq m\}$.

Figure 1: Algorithm SAT

Since $+$ is associative, to verify condition (4.2) we need to keep in the memory at every stage p only two tuples: the tuple of the partial sums $\left(\sum_{s=1}^{s<p} \tau_1^s, \dots, \sum_{s=1}^{s<p} \tau_k^s\right)$ and the current guess $(\tau_1^p, \dots, \tau_k^p)$. The tuple of the partial sums can be easily updated. We can assume that all partial sums are different; hence, m is bounded by the number of possible $\mathit{ptype}^n(\mathcal{M}; \left(\overline{P}_1, \dots, \overline{P}_k\right))$ which is bounded by $|\mathit{Hin}^n(< .X_1, \dots, X_l)|^k$ and the counter for m can be saved in space linear in k.

To verify condition (5.3) we need to keep in memory at every stage p only two tuples: the tuple $U_i^p = \{\tau_i^s \mid s < p\}$ (for $i = 1, \dots, k$) and the current guess $(\tau_1^p, \dots, \tau_k^p)$. We have to verify that $(\tau_1^p, \dots, \tau_k^p)$ is in $(U_1, \dots, U_k)$, i.e., $\tau_i^p \in U_i$ and update the tuple $(U_1^p, \dots, U_k^p)$. In (5.) we can assume that no tuple occurs twice; hence, m is bounded by the number of possible $\mathit{ptype}^n(\mathcal{M}; \left(\overline{P}_1, \dots, \overline{P}_k\right))$ and the counter for m can be saved in space linear in k.

The depth of recursion is bounded by N. Hence, SAT works in non-deterministic space $O(kN)$.

In order to check (A) we call SAT with $N = 2r_\Phi \times k + 1$. Therefore, our procedure works in non-deterministic polynomial space and by Savitch's theorem it can be implemented by a deterministic PSPACE algorithm.

The next two sections are geared towards the proof of Lemma 7.2.

8. Automata on Linear Orders

Büchi [3] used finite automata over ω-words to prove that monadic second-order logic is decidable over ω. In order to prove the decidability of monadic second-order logic over countable ordinals, Büchi introduced finite automata on words of ordinal length [4]. Büchi's model extends traditional finite automata using limit transitions to handle positions with no predecessor. He proved that over countable ordinals these automata are equivalent to monadic second-order logic.

These automata were extended to finite automata on linear orderings by Bruyère and Carton [2]. This model further extends traditional finite automata using limit transitions to handle positions with no successor or no predecessor. In [19] it was shown that these automata can be complemented over countable scattered linear orderings and are equivalent to monadic second-order logic over the countable scattered linear orderings. However, this equivalence fails over dense orders and over uncountable orders [1].

We first recall some basic definitions about linear orders. Then, we introduce finite base automata which have the same expressive power as finite state automata of [2]. The finite base automata play a crucial role in our proof of the small rank property. The equivalence between finite state and finite base automata is proved in the Appendix.

In order to define the runs of an automaton, we use the notion of cut. A *cut* of a linear order J is a partition (L, U) of J such that $a < b$ for any $a \in L$ and $b \in U$. A cut (L, U) is a *gap* if neither L has a maximal element, nor U has a minimal element and $L \neq \emptyset \neq U$. An order is *Dedekind-complete* if it does not have gaps. We denote by $\widehat{J}$ the set of cuts of J. This set is equipped with the order defined by $(L_1, U_1) < (L_2, U_2)$ if $L_1 \subsetneq L_2$. This ordering on $\widehat{J}$ can be extended to $J \cup \widehat{J}$ in a natural way: $(L, U) < a$ if $a \in U$. The order $\widehat{J}$ is Dedekind-complete. Its minimal (maximal) element is $\widehat{J}_{\min} = (\emptyset, J)$ (respectively, $\widehat{J}_{\max} = (J, \emptyset)$). For any element a of J, there are two successive cuts: $a^- := (\{b \in J \mid b < a\}, \{b \in J \mid b \geq a\})$

and $a^+ := (\{b \in J \mid b \leq a\}, \{b \in J \mid b > a\})$. Note that if $a < b$ are consecutive elements of J then a^+ and b^- denote the same cut.

Given an alphabet Σ and any linear order J, a Σ-word of length J is a sequence $(\sigma_a \mid a \in J)$ of elements of Σ indexed by J.

In [7] we introduced simple ordinal automata which work over words of ordinal length. We extend this definition to finite base automata working on words over arbitrary linear orders.

Definition 8.1 (finite base automata). *A finite base automaton $\mathfrak{A}$ is a tuple of the form $(B, Q, \Sigma, \delta_{next}, \delta_{\lim}, Q_{init}, Q_{fin})$ such that*

- *B is a finite set (the basis of $\mathfrak{A}$),*
- *$Q \subseteq \mathcal{P}(B)$ (the set of states),*
- *$Q_{init}, Q_{fin} \subseteq Q$ (the sets of initial states and final states),*
- *Σ is a finite alphabet,*
- *$\delta_{next} \subseteq Q \times \Sigma \times Q$ is the next-step transition relation,*
- *$\delta_{\lim} \subseteq (\mathcal{P}(B) \times Q) \bigcup (Q \times \mathcal{P}(B))$ is the limit transition relation.*

For $(q, \sigma, q') \in \delta_{next}$, we sometimes write $q \xrightarrow{\sigma} q'$; for $(D, q) \in \delta_{\lim}$ (respectively, $(q, D) \in \delta_{\lim}$), we write $D \to q$ (respectively, $q \leftarrow D$) and say that this is a *left* (respectively, *right*) *limit transition.*

Let f be a function from a set I into $\mathcal{P}(B)$. Define

$$\text{always}(f) := \{\mathfrak{b} \in B \mid \forall c \in I \quad \mathfrak{b} \in f(c)\}.$$

If I is a linear order, we define the left and right base-limit sets of f at $c \in I$ as the sets of base elements that appear in every state arbitrarily close to c (respectively, to its left and to its right). Formally, $Base_{\underrightarrow{\lim}}(c, f)$ is defined as

$$Base_{\underrightarrow{\lim}}(c, f) := \{\mathfrak{b} \in B \mid \forall a < c \exists d (a < d < c) \wedge \mathfrak{b} \in \text{always}(f\lfloor(d, c))\},$$

where $f\lfloor(d, c)$ is the restriction of f to the interval (d, c).

$Base_{\underleftarrow{\lim}}(c, f)$ is defined similarly.

Given a finite base automaton $\mathfrak{A}$, a *run* of $\mathfrak{A}$ on Σ-word s over a linear order $\mathcal{I}$ is a function $\rho : \widehat{\mathcal{I}} \to Q$ such that

- For each $c \in \mathcal{I}$, $\rho(c^-) \xrightarrow{s(c)} \rho(c^+)$,
- if $c \in \widehat{\mathcal{I}} \setminus \widehat{\mathcal{I}}_{\min}$ has no predecessor, $(Base_{\underrightarrow{\lim}}(c, \rho), \rho(c)) \in \delta_{\lim}$, and
- if $c \in \widehat{\mathcal{I}} \setminus \widehat{\mathcal{I}}_{\max}$ has no successor, $(\rho(c), Base_{\underleftarrow{\lim}}(c, \rho)) \in \delta_{\lim}$.

An $\mathfrak{A}$-run ρ is *accepting* if $\rho(\widehat{\mathcal{I}}_{\min}) \in Q_{init}$ and $\rho(\widehat{\mathcal{I}}_{\max}) \in Q_{fin}$. $\mathfrak{A}$ *accepts* a word s if there is an accepting run on s.

Let $\mathfrak{A}_1, \dots, \mathfrak{A}_m$ be finite base automata. One can easily construct an automaton $\mathfrak{A}$ that accepts the intersection of the languages accepted by these automata. The number of states in $\mathfrak{A}$ is the product of the numbers of states of $\mathfrak{A}_i$ and this grows exponentially in m; however, the base size of $\mathfrak{A}$ is the sum of the base sizes of $\mathfrak{A}_i$.

Lemma 8.2 (intersection of finite base automata). *Let $\mathfrak{A}_1$ and $\mathfrak{A}_2$ be finite base automata. Assume that the base size of $\mathfrak{A}_1$ and $\mathfrak{A}_2$ are n_1 and n_2. There is a finite base automaton $\mathfrak{A}$ such that the base size of $\mathfrak{A}$ is $n_1 + n_2$ and a word s is accepted by $\mathfrak{A}$ iff it is accepted by $\mathfrak{A}_1$ and by $\mathfrak{A}_2$.*

A word $s := (\sigma_a \mid a \in J)$ indexed by J over an alphabet $\{0,1\}^k$ can be identified with a chain $(J, <, P_1, \dots, P_k)$ over J where $P_i = \{a \in J \mid$ the i-th bit of $\sigma_a = 1\}$. This is a bijection between the $\{0,1\}^k$-words over J and the chains with k monadic predicates over J.

An automaton is said to be equivalent to a formula $\varphi(P_1, \dots, P_k)$ over a class $\mathcal{C}$ of linear orders if for every linear order $J \in \mathcal{C}$ and every word s indexed by J, $\mathfrak{A}$ accepts s if and only if the corresponding chain satisfies φ.

Cristau [6] proved a very unexpected result: every formula of the first-order fragment of the monadic logic is equivalent (over the class of all linear orders) to a finite state automaton. In the appendix we describe finite state automata and prove that they are equivalent to finite base automata. Hence,

Theorem 8.3. *For every first-order formula φ there is a finite base automaton $\mathfrak{A}_\varphi$ equivalent to φ over the class of all linear orders.*

9. Small Rank Property

In this section we introduce automata types, state a small rank property for the automata type and prove Lemma 7.2 (the small rank property for the conjunctive formulas) which played a crucial role in the complexity analysis of our PSPACE algorithm.

Let $\mathfrak{A}$ be a finite base automaton, $\mathcal{L}$ a chain and $\rho : \widehat{\mathcal{L}} \to Q$ be a run of $\mathfrak{A}$ on $\mathcal{L}$. Define the $\mathfrak{A}$-type of ρ as $type_{\mathfrak{A}}(\rho) := (q, D, q')$, where $\rho(\widehat{\mathcal{L}}_{\min}) = q$, $\rho(\widehat{\mathcal{L}}_{\max}) = q'$ and $D := \mathrm{always}(\rho)$.

If $type_{\mathfrak{A}}(\rho) := (q, D, q')$ we sometimes write $\rho : q \xrightarrow{D} q'$; we write $\rho :\xrightarrow{D}$ if $type_{\mathfrak{A}}(\rho) := (q, D, q')$ for some q and q'.

Define an equivalence relation $\sim_{\mathfrak{A}}$ on $\mathfrak{A}$-runs:

$$\rho_1 \sim_{\mathfrak{A}} \rho_2 \text{ if and only if } type_{\mathfrak{A}}(\rho_1) = type_{\mathfrak{A}}(\rho_2)$$

Weight. Let D be a subset of the base B of $\mathfrak{A}$. The weight of D is defined as the cardinality of $B \setminus D$. The weight of a transition of $\mathfrak{A}$ is defined as follows. The weight of a successor transition is 0; the weight of limit transitions $(D, q) \in \delta_{\mathrm{lim}}$ and $(q, D) \in \delta_{\mathrm{lim}}$ is the weight of D. The weight of a run ρ is defined as the maximum of the weights of transitions that appear in ρ. We denote the weight of ρ by $\mathrm{weight}(\rho)$; the weight is always between 0 and the cardinality of the base of $\mathfrak{A}$. The following lemma is proved in the next section.

Lemma 9.1 (Main). *Assume that ρ is a run on a countable chain of a finite base automaton $\mathfrak{A}$.*

1. *If $\rho :\xrightarrow{D}$ and* $\mathrm{weight}(\rho) = \mathrm{weight}(D) = w$*, then there is a run on a chain of rank $\leq 2w+1$, which is equivalent to ρ.*
2. *Any run ρ of weight $\leq w$ is equivalent to a run on a chain of rank $\leq 2w+2$.*

As a consequence, we obtain the following small rank property:

Proposition 9.2 (small rank property). *Let $\mathfrak{A}$ be a finite base automaton with base of size $n_{\mathfrak{A}}$. Every run of $\mathfrak{A}$ is equivalent to a run on a chain of rank $\leq 2n_{\mathfrak{A}} + 2$. In particular, if $\mathfrak{A}$ has an accepting run, then it accepts a chain of rank $\leq 2n_{\mathfrak{A}} + 2$.*

The complexity analysis of our PSPACE algorithm was based on Lemma 7.2. Now we are ready to prove it.

PROOF (OF LEMMA 7.2). Let Φ be a finite set of first-order formulas. By Theorem 8.3, every formula in $\varphi \in \Phi$ is equivalent to a finite base automaton $\mathfrak{A}_\varphi$. Let n_Φ be an upper bound on the base size of $\mathfrak{A}_\varphi$ for $\varphi \in \Phi$.

By Lemma 8.2, ψ is equivalent to a finite base automata with the base of size $\leq k \times n_\Phi$. By Proposition 9.2, if ψ is satisfiable in a countable chain, then it is satisfiable in a chain of rank $\leq k(2n_\Phi + 2)$. Hence, we can define r_Φ as $(2n_\Phi + 2)$. □

It is instructive to compare the small rank property of finite base automata with the short run property of simple ordinal automata from [7]. A simple ordinal automaton is a finite base automaton with $\delta_{\lim} \subseteq \mathcal{P}(B) \times Q$. Hence, the domain of every run ρ of a simple ordinal automaton is order-isomorphic to an ordinal, and if ρ is a run on $\mathcal{M}$ then $\mathcal{M}$ is a chain over an ordinal. An ordinal α has rank $i \geq 1$ iff $\alpha < \omega^{i+1}$. Lemma 6 in [7] states that every run of a simple ordinal automaton $\mathfrak{A}$ is equivalent to an $\mathfrak{A}$-run on an ordinal $< \omega^{n_{\mathfrak{A}}+1}$, where $n_{\mathfrak{A}}$ is the size of the base of $\mathfrak{A}$.

10. Proof of Lemma 9.1

In the next subsection we develop elements of the composition method for automata. We define the sums of runs and of automata types. In subsection 10.2 we adopt the technique used by Läuchli and Leonard for automata types and illustrate it by proving a proposition which implies that if an automaton accepts a countable chain, then it accepts a chain of finite rank. This technique will be used with more subtle inductive assertions in subsections 10.3-10.4 to prove Lemma 9.1. Finally, a stronger version of Lemma 9.1 is stated in subsection 10.5.

10.1. Sum of runs

This section develops elements of the composition method for automata. We define the sums of runs and of automata types. Unlike the sum of chains, the sum of runs might be the empty set, a singleton run or it might contain many runs. Unlike the sum of first-order types, the sum of automata types might be the empty set, a singleton automaton type, or it might contain many automaton types. All lemmas stated here follow easily from the definitions.

Assume that ρ is a run on $\mathcal{L}$. For an interval I of $\widehat{\mathcal{L}}$, we denote by $\rho\lfloor I$ the restriction of ρ on I; we also denote by $\rho\lfloor \leq c$, the restriction of ρ on $\{a \in \widehat{\mathcal{L}} \mid a \leq c\}$. Note that if I is a closed interval with at least two points, then $\rho\lfloor I$ is a run on $\{a \in \mathcal{L} \mid a^-, a^+ \in I\}$.

Let $\mathcal{J} := (J, <^{\mathcal{J}})$ be a linear order and $\mathfrak{S} := (\mathcal{L}_\alpha \mid \alpha \in J)$ a sequence of chains.

To a cut (L_α, U_α) of $\mathcal{L}_\alpha$ corresponds the cut $\left(L_\alpha \cup \bigcup_{\beta<\alpha} \mathcal{L}_\beta, \bigcup_{\beta>\alpha} \mathcal{L}_\beta \cup U_\alpha\right)$ of $\sum_{\beta\in\mathcal{J}} \mathcal{L}_\beta$.

If $\alpha < \alpha'$ are consecutive elements of $\mathcal{J}$ then to the cuts $(\mathcal{L}_\alpha, \emptyset)$ and $(\emptyset, \mathcal{L}_{\alpha'})$ of $\mathcal{L}_\alpha$ and of $\mathcal{L}_{\alpha'}$ corresponds the same cut of $\sum_{\beta\in\mathcal{J}} \mathcal{L}_\beta$.

Usually we will not distinguish between a cut c in $\mathcal{L}_\alpha$ and the corresponding cut in $\sum_{\beta\in\mathcal{J}} \mathcal{L}_\beta$ which will be also denoted by c. We also denote by $\widehat{\mathcal{L}_\alpha}$ the set of cuts of $\mathcal{L}_\alpha$ and the corresponding sets of cuts of $\sum_{\beta\in\mathcal{J}} \mathcal{L}_\beta$.

Note that if (J_1, J_2) is a gap in $\mathcal{J}$, then the cut $(\cup_{\alpha\in J_1}\mathcal{L}_\alpha, \cup_{\alpha\in J_2}\mathcal{L}_\alpha)$ of $\sum_{\beta\in\mathcal{J}} \mathcal{L}_\beta$ does not correspond to any cut in the summands.

If the index structure is order-isomorphic to the rationals $(\mathbb{Q}, <)$, then $\sum_{\beta\in\mathbb{Q}} \mathcal{L}_\beta$ has the minimal and maximal cuts and the following set of cuts:

irrational cuts For every real $x \in \mathbb{R} \setminus \mathbb{Q}$ corresponds cut $(\cup_{\alpha\in\mathbb{Q}_{<x}}\mathcal{L}_\alpha, \cup_{\alpha\in\mathbb{Q}_{>x}}\mathcal{L}_\alpha)$, where $\mathbb{Q}_{<x} := \{\alpha \in \mathbb{Q} \mid \alpha < x\}$ and $\mathbb{Q}_{>x} := \{\alpha \in \mathbb{Q} \mid \alpha > x\}$.

cuts of the summands For every $\alpha \in \mathbb{Q}$ and a cut (L_α, U_α) of $\mathcal{L}_\alpha$ corresponds cut $(\cup_{\beta<\alpha}\mathcal{L}_\beta \cup L_\alpha, \cup_{\beta>\alpha}\mathcal{L}_\beta \cup U_\alpha)$

Definition 10.1 (sum of runs). *Let $\mathcal{J} := (J, <^{\mathcal{J}})$ be a linear order and $\mathfrak{S} := (\mathcal{L}_\alpha \mid \alpha \in J)$ a sequence of chains. Assume that the domains of $\mathcal{L}_\alpha$ and $\mathcal{L}_\beta$ are disjoint whenever $\alpha \neq \beta$ are in J. Assume that ρ is a run on $\sum_{\beta\in\mathcal{J}} \mathcal{L}_\beta$ and ρ_α are runs on $\mathcal{L}_\alpha$ for $\alpha \in J$. If $\rho\lfloor\widehat{\mathcal{L}_\alpha} = \rho_\alpha$ for $\alpha \in J$, then ρ is said to be in the sum $\sum_{\beta\in\mathcal{J}} \rho_\beta$.*

Recall that the sum of chains is unique up to isomorphism. Unlike the sum of chains, the sum of runs might be empty, singleton set or it might contain many elements.

If $\mathcal{J} = (\{0,1\}, <)$, we denote $\sum_{\alpha\in\mathcal{J}} \rho_\alpha$ by $\rho_0 + \rho_1$, and this is a singleton set iff the last state of ρ_0 is the same as the first state of ρ_1; otherwise $\rho_0 + \rho_1$ is empty.

If ρ_α is isomorphic to ρ for every $\alpha \in J$, we denote $\sum_{\alpha\in\mathcal{J}} \rho_\alpha$ by $\rho \times \mathcal{J}$.

Note that $\rho \times \omega$ is non-empty iff $type_{\mathfrak{A}}(\rho) = (q, D, q)$ and there is a limit transition $D \to q'$ in $\mathfrak{A}$.

Lemma 10.2 ($\sim_{\mathfrak{A}}$ is a congruence wrt sums). *Let $\mathfrak{A}$ be an automaton. Assume:*

1. *$\mathcal{J} := (J, <^{\mathcal{J}})$ is a linear order,*
2. *$\left(\rho^0_\alpha \mid \alpha \in J\right)$ and $\left(\rho^1_\alpha \mid \alpha \in J\right)$ are sequences of $\mathfrak{A}$-runs, and*
3. *for every $\alpha \in J$, $\rho^0_\alpha \sim_{\mathfrak{A}} \rho^1_\alpha$, and*
4. *$\rho^0 \in \sum_{\alpha\in J} \rho^0_\alpha$.*

Then, there is $\rho^1 \in \sum_{\alpha\in J} \rho^1_\alpha$ such that $\rho^0 \sim_{\mathfrak{A}} \rho^1$.

A *formal* $\mathfrak{A}$*-type* is a tuple (q, D, q'), where q, q' are states of $\mathfrak{A}$ and D is a subset of $q \cap q'$ (recall that the states of $\mathfrak{A}$ are subsets of the base of $\mathfrak{A}$).

Lemma 10.2 implies that the sum of $\mathfrak{A}$-types over any linear order is well defined. Below we provide explicit definitions for three important instances of the sum of automata types: binary sum, multiplication by an ordinal and shuffle.

Definition 10.3 (formal binary sum of types). *Let $\tau_1 = (q_1, D_1, p_1)$ and $\tau_2 = (q_2, D_2, p_2)$ be formal $\mathfrak{A}$-types. If $p_1 \neq q_2$, then $\tau_1 + \tau_2$ is defined to be the empty set; otherwise it is defined to be $\{(q_1, D_1 \cap D_2, p_2)\}$.*

Lemma 10.4. *Let ρ_1 and ρ_2 be $\mathfrak{A}$-runs. Then $\{type_{\mathfrak{A}}(\rho') \mid \rho' \in \rho_1 + \rho_2\}$ is equal to $\tau_1 + \tau_2$.*

Definition 10.5 (formal multiplication by a limit ordinal). *Let $\tau = (q, D, q)$ be a formal $\mathfrak{A}$-type.*

1. *$\tau \times \omega$ is defined as the set $\{(q, D \cap q', q') \mid$ there is a limit transition $D \to q'$ in $\mathfrak{A}\}$.*
2. *Let α be a limit ordinal greater than ω. If there is a limit transition $D \to q$ in $\mathfrak{A}$, then $\tau \times \alpha$ is defined as $\tau \times \omega$; otherwise it is defined to be the empty set.*

Note that $\tau \times \alpha$ depends on an automaton $\mathfrak{A}$, and in order to be precise we need to use $\tau \times_{\mathfrak{A}} \alpha$; however $\mathfrak{A}$ will be always clear from the context.

Lemma 10.6. *Let α be a limit ordinal and let ρ be an $\mathfrak{A}$-run. Then $\{type_{\mathfrak{A}}(\rho') \mid \rho' \in \rho \times \alpha\}$ is equal to $\tau \times \alpha$.*

Given a linear ordering J, we denote by J^{-1} the backwards linear ordering obtained by reversing the ordering relation. Formal multiplication by a reverse limit ordinal is defined in a similar way to Definition 10.5, and an analogue of Lemma 10.6 holds for this multiplication.

Let $\mathfrak{S} := (\mathcal{L}_\alpha \mid \alpha \in \mathbb{Q})$ be a sequence of chains indexed by the rationals. Let $Q_1, \dots, Q_k \subseteq \mathbb{Q}$ be a partition of $\mathbb{Q}$ into k everywhere dense sets. Let $R_1, \dots, R_l$ be a partition of $\mathbb{R} \setminus \mathbb{Q}$ into l everywhere dense sets. Let $\rho_1, \dots, \rho_k$ be $\mathfrak{A}$-runs. and $p_1, \dots, p_l$ be states of $\mathfrak{A}$. A run ρ on $\sum_{\alpha\in\mathbb{Q}} \mathcal{L}_\alpha$ is in the *shuffle* of $\rho_1, \dots, \rho_k$ and $p_1, \dots, p_l$ if $\rho\lfloor\widehat{\mathcal{L}_\alpha}$ is isomorphic to ρ_i for $\alpha \in Q_i$ and $\rho(x) = p_j$ for the irrational cuts $x \in R_j$. We denote by *shuffle*$_{\mathfrak{A}}(\rho_1, \dots, \rho_k, p_1, \dots, p_l)$ the set $\{\rho \mid \rho$ is in the shuffle of $\rho_1, \dots, \rho_k$ and $p_1, \dots, p_l\}$.

Recall that the shuffle of chains $\mathcal{L}_1, \dots, \mathcal{L}_k$ is unique up to isomorphism, yet *shuffle*$_{\mathfrak{A}}(\rho_1, \dots, \rho_k, p_1, \dots, p_l)$ contains many non-isomorphic runs, because there are uncountable many non-isomorphic partitions of irrationals into l everywhere dense sets. However, the set of $\mathfrak{A}$-types of all runs in *shuffle*$_{\mathfrak{A}}(\rho_1, \dots, \rho_k, p_1, \dots, p_l)$ is computable from $type_{\mathfrak{A}}(\rho_1), \dots, type_{\mathfrak{A}}(\rho_k)$.

Definition 10.7 (formal shuffle of types). *Assume that $\tau_j = \left(q_j, D_j, q'_j\right)$ for $j = 1, \dots, k$ are formal $\mathfrak{A}$-types and $p_1, \dots, p_l$ are states of $\mathfrak{A}$ for $k, l \geq 1$. Let D be the set of base elements which belongs to every p_i and every D_j. Let $S := \{p_i \mid i \leq l\} \cup \{q'_j \mid j \leq k\}$ and $E := \{p_i \mid i \leq l\} \cup \{q_j \mid j \leq k\}$. If $p_i \leftarrow D$, $q'_j \leftarrow D$, $D \to p_i$ and $D \to q_j$ are $\mathfrak{A}$-limit transitions, then shuffle$_{\mathfrak{A}}(\tau_1, \dots, \tau_k, p_1, \dots, p_l)$ is defined as $\{(s, D, s') \mid$ there are limit transitions $s \leftarrow D$ and $D \to s'$ in $\mathfrak{A}\}$; otherwise shuffle$_{\mathfrak{A}}(\tau_1, \dots, \tau_k, p_1, \dots, p_l)$ is defined to be the empty set.*

Lemma 10.8 (Shuffle). *Assume that ρ_j are $\mathfrak{A}$-runs and $type_{\mathfrak{A}}(\rho_j) = \tau_j$ for $j = 1, \dots, k$ and $p_1, \dots, p_l$ are states of $\mathfrak{A}$. Then $\{type_{\mathfrak{A}}(\rho) \mid \rho \in shuffle_{\mathfrak{A}}(\rho_1, \dots, \rho_k, p_1, \dots, p_l)\}$ is equal to $shuffle_{\mathfrak{A}}(\tau_1, \dots, \tau_k, p_1, \dots, p_l)$.*

Often we will use $shuffle_{\mathfrak{A}}(\rho_1, \dots, \rho_k)$ for the union of $shuffle_{\mathfrak{A}}(\rho_1, \dots, \rho_k, p_1, \dots, p_l)$ over all sequences $p_1, \dots, p_l$. Whenever $\mathfrak{A}$ is clear from a context we will drop the subscript $\mathfrak{A}$ and use "$shuffle(\rho_1, \dots, \rho_k)$" for "$shuffle_{\mathfrak{A}}(\rho_1, \dots, \rho_k)$". Similar notations and conventions will be used for the shuffles of formal types.

10.2. Reduction to regular runs

Let $\mathcal{C}_{reg}$ be the closure of one-element chains under $\{+, \times\omega, \times\omega^{-1}, shuffle\}$. Let $\mathfrak{A}$ be an automaton, $One^{\mathfrak{A}}$ be the set of $\mathfrak{A}$-runs over the one-element chains, and let $\mathcal{R}^{\mathfrak{A}}_{reg}$ be the closure of $One^{\mathfrak{A}}$ under $\{+, \times\omega, \times\omega^{-1}, shuffle_{\mathfrak{A}}\}$. The runs in $\mathcal{R}^{\mathfrak{A}}_{reg}$ are called $\mathfrak{A}$-regular runs. Note that every run in $\mathcal{R}^{\mathfrak{A}}_{reg}$ is a run of $\mathfrak{A}$ on a chain in $\mathcal{C}_{reg}$.

In this subsection we are going to prove the following proposition.

Proposition 10.9. *Let $\mathfrak{A}$ be an automaton and ρ a run of $\mathfrak{A}$ on a countable chain. There is a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$ which is $\sim_{\mathfrak{A}}$-equivalent to ρ.*

The technique used in this proof was introduced by Läuchli and Leonard [13] to prove that the first-order theory of linear order is decidable. It was extended by Shelah [21] to the monadic second-order logic over labeled linear orders. We are going to use it for the automata types several times. In this subsection it will be used with a simple inductive assertion while in sections 10.3-10.5 inductive assertions will be more subtle.

Proof. Let ρ be a run. Define $\sim_\rho$ on the domain of ρ:

$x \sim_\rho y$ iff $x = y$ or for all $z < v \in [\min(x, y), \max(y, x)]$ the run $\rho\lfloor[z, v]$ is equivalent to a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$.

This is a *convex* equivalence relation, i.e., its equivalence classes are intervals. An equivalence class is called *degenerate* if it is a singleton.

Let I be a non-degenerate equivalence class and let $c^{\sup}_I, c^{\inf}_I$ be the supremum and infimum of its elements. We claim that $I = [c^{\inf}_I, c^{\sup}_I]$, i.e.,

(a) Each $\sim_\rho$-equivalence class I is a closed interval.

Proof (of (a)). Toward a contradiction assume that I has no maximal element. Then there is an increasing ω-sequence $y_0 < y_1 \dots$ in I converging to the supremum $c^{\sup}_I$ of I. Let $\mathrm{col}(y_i, y_j)$ be the $\mathfrak{A}$-type of $\rho\lfloor[y_i, y_j]$. This is an additive coloring. By the Ramsey theorem (Theorem 4.6) there is a homogeneous cofinal subsequence z_i.

First, let us show that $\rho\lfloor[z_i, c^{\sup}_I]$ is equivalent to a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$. Since $z_i \sim_\rho z_{i+1}$ there is a run ρ' in $\mathcal{R}^{\mathfrak{A}}_{reg}$, which is $\sim_{\mathfrak{A}}$-equivalent to $\rho\lfloor[z_i, z_{i+1}]$. By homogeneity, we know that $type_{\mathfrak{A}}(\rho') = (q, D, q)$ for some q and D, and $Base_{\underrightarrow{\lim}}(c^{\sup}_I, \rho) = D = D \cap q$. Let $q' := \rho(c^{\sup}_I)$. The left limit transition in ρ at $c^{\sup}_I$ is $D \to q'$. Hence, there is a limit transition $D \to q'$ in $\mathfrak{A}$ and the $type_{\mathfrak{A}}$ of $\rho\lfloor[z_i, c^{\sup}_I]$ is $(q, D \cap q', q')$. By Lemma 10.6, there is a run $\rho_1 \in \rho' \times \omega \subseteq \mathcal{R}^{\mathfrak{A}}_{reg}$ such that its type is also $(q, D \cap q', q')$.

Let $y \in I$ and let $z_i > y$ (such z_i exists by cofinality of the sequence z_i). Since $y \sim_\rho z_i$ there is a run ρ_0 in $\mathcal{R}^{\mathfrak{A}}_{reg}$ which is $\sim_{\mathfrak{A}}$-equivalent to $\rho\lfloor[y, z_i]$. Therefore, $\rho\lfloor[y, c^{\sup}_I]$ is $\sim_{\mathfrak{A}}$-equivalent to $\rho_0 + \rho_1 \in \mathcal{R}^{\mathfrak{A}}_{reg}$.

We proved that for every $y \in I$ there is a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$ which is $\sim_{\mathfrak{A}}$-equivalent to $\rho\lfloor[y, c^{\sup}_I]$. This together with the definition of I and of $\sim_\rho$ implies that every $y \in I$ is $\sim_\rho$-equivalent to $c^{\sup}_I$. Therefore, $c^{\sup}_I \in I$ and this contradicts that I does not have a maximal element.

A proof that $c^{\inf}_I \in I$ is similar. □

Let $L_{\sim_\rho}$ be the chain of $\sim_\rho$-equivalence classes. An equivalence class I_1 is less than an equivalence class I_2 if $\forall x \in I_1 \forall y \in I_2 (x < y)$. We are going to show:

(b) $L_{\sim_\rho}$ does not contain consecutive elements.

(c) $L_{\sim_\rho}$ is not dense.

From (b) and (c) it follows that $L_{\sim_\rho}$ has only one element. Therefore, there is only one $\sim_\rho$-equivalence class and ρ is $\sim_{\mathfrak{A}}$-equivalent to a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$.

It remains to prove (b) and (c).

Proof (of (b)). Toward a contradiction assume that $I_1 < I_2$ are consecutive equivalence classes. I_1 has a maximal element x and I_2 has a minimal element y. Therefore, x and y are consecutive elements and in ρ there is a δ_{next} transition between x and y. In this case $x \sim_\rho y$ and this contradicts that x and y are in different equivalence classes. □

Proof (of (c)). Toward a contradiction assume that $L_{\sim_\rho}$ is dense. Label $I \in L_{\sim_\rho}$ as follows: If I is a degenerate equivalence class and its only element is x, then $lab(I) = \rho(x)$; otherwise $lab(I) = type_{\mathfrak{A}}(\rho\lfloor I)$.

By Lemma 4.8, there is an open non-empty interval $J \subseteq L_{\sim_\rho}$ which is uniform for lab.

Let $\Delta_1 := \{p \mid p = \rho(x) \text{ and } \{x\} \in J \text{ is a degenerate } \sim_\rho \text{ -equivalence class }\}$ and $\Delta_2 := \{type_{\mathfrak{A}}(\rho\lfloor I) \mid I \in J \text{ and } I \text{ is not degenerate}\}$. Note that the set of non-degenerate $\sim_\rho$-equivalence classes in J is countable and order-isomorphic to $\mathbb{Q}$. The set of degenerate $\sim_\rho$-equivalence classes in J is uncountable and hence $\Delta_1 \neq \emptyset$. Let $p_1, \dots, p_l$ be an enumeration of elements from Δ_1, and let $(q_1, D, q'_1), \dots, (q_k, D, q'_k)$ be an enumeration of elements from Δ_2. For every $\big(q_j, D, q'_j\big) \in \Delta_2$ choose $I \in J$ such that $type_{\mathfrak{A}}(\rho\lfloor I) = \big(q_j, D, q'_j\big)$ and let $\rho_j := \rho\lfloor I$. Since I is a $\sim_\rho$-equivalence class it follows that ρ_j is an $\mathfrak{A}$-equivalent to a run $\rho'_j \in \mathcal{R}^{\mathfrak{A}}_{reg}$. Let D be the set of base elements which belong to every p_i and every D_j. Then for every $i \leq l$ and $j \leq k$ the limit transitions $p_i \leftarrow D$, $q'_j \leftarrow D$, $D \rightarrow p_i$ and $D \rightarrow q_j$ occur in ρ and hence they are $\mathfrak{A}$-limit transition.

This together with Lemma 10.8 implies that if $I_1 < I_2$ are in J and x is the last element of I_1 and y is the first element of I_2 then $\rho\lfloor[x, y]$ is $\sim_{\mathfrak{A}}$-equivalent to a run $\rho_{I_1,I_2} \in shuffle(\rho_1, \dots, \rho_k) \subseteq \mathcal{R}^{\mathfrak{A}}_{reg}$.

Now let $x' < y'$ be in the union of the $\sim_\rho$-equivalence classes in J. We are going to show that

(d) there is a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$ which is $\sim_{\mathfrak{A}}$-equivalent to $\rho\lfloor[x', y']$.

Hence, all elements in the union of the $\sim_\rho$-equivalence classes in J are $\sim_\rho$ equivalent and this contradicts that J contains more than one $\sim_\rho$-equivalence class.

In order to prove (d) observe that if x', y' are in the same equivalence class, then such a run exists by the definition of $\sim_\rho$.

Let $x' \in I_1$ and $y' \in I_2$ and let x be the last element of I_1 and y be the first element of I_2. Then there are runs $\rho_1, \rho_3 \in \mathcal{R}^{\mathfrak{A}}_{reg}$ which are $\sim_{\mathfrak{A}}$-equivalent to $\rho\lfloor[x', x]$ and $\rho\lfloor[y, y']$.

Hence, $\rho\lfloor[x', y']$ is $\sim_{\mathfrak{A}}$-equivalent to $\rho_1 + \rho_{I_1,I_2} + \rho_3$ and this is a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$. □

Remark 10.10. *This remark sketches how Proposition 10.9 can be extended to the class of all linear orders. For this extension we need an additional shuffle operation on runs and on automata types. In subsection 10.1 the shuffle of $\mathbb{Q}$ indexed family of runs was defined. $\mathbb{Q}$ is not Dedekind-complete. We need to define the shuffle indexed by Dedekind-complete dense linear orders. The shuffle operation indexed by the reals is denoted by $\mathbb{R}-shuffle(r_1, \dots, r_n)$ and is defined in a natural way. The corresponding definition of $\mathbb{R}-shuffle(\tau_1, \dots, \tau_n)$ for automata types is similar to Definition 10.7, however no states p_i appear as parameters in the shuffle, i.e., $l = 0$. Let $R^{\mathfrak{A}}$ be the closure of $One^{\mathfrak{A}}$ under $\{+, \times\omega, \times\omega^{-1}, shuffle, \mathbb{R} - shuffle\}$. A proof similar to the proof of Proposition 10.9 shows that every $\mathfrak{A}$-run is $\sim_{\mathfrak{A}}$-equivalent to a run in $R^{\mathfrak{A}}$. The only difference in these proofs is that for (a) the cardinality of any sequence converging to the supremum $c^{\sup}_I$ of I might be uncountable. We need to use instead of Theorem 4.6, the Shelah theorem (corollary 1.2 [21]) for an additive coloring of ordinals of uncountable cofinality.*

10.3. Proof of Lemma 9.1(1)

Terminology. Throughout this and the next subsection we often will use the following terminology. Let ρ be a run of a finite base automaton. If c is a left limit point (in the domain of ρ), and there is a D transition to c from the left, we say that c is D^- cut. If c is a right limit point, and there is a D transition to c from the right, we say that c is D^+ cut.

We are going to prove Lemma 9.1 by induction on w.

The *inductive base* is trivial.

For the inductive step we first assume that Lemma 9.1(1) and Lemma 9.1(2) hold for w and prove Lemma 9.1(1) for $w+1$. Then, we assume that Lemma 9.1(2) holds for w and Lemma 9.1(1) holds for $w+1$ and prove Lemma 9.1(2) for $w+1$.

For the inductive step $w \mapsto w+1$ of Lemma 9.1(1) we consider several cases.

Case 1. The leftmost and the rightmost transitions are limit transitions over D.

Subcase 1. There is a D^- cut c and $y < c$ such that no D^+ or D^- cut appears in ρ on (y, c).

Let $y_1 < y_2 < \cdots < y_n < \ldots$ be an ω-sequence converging to c. For $i < j$ let $\mathrm{col}(y_i, y_j) := type_{\mathfrak{A}}(\rho\lfloor[y_i, y_j])$.

By the Ramsey theorem (Theorem 4.6) there is a cofinal homogeneous subsequence z_i of y_i, i.e., $\mathrm{col}(z_i, z_j)$ is the same for all pairs $i < j$. Let $q \xrightarrow{D'} q'$ be this color. Then from homogeneity it follows that $q = q'$ and $D' = D' \cap q$. Because this sequence converges to c and at c there is a D transition, we obtain that $D' = D$.

Let ρ' be $\rho\lfloor[z_1, z_2]$. ρ' contains no D^- or D^+ transition. Hence its weight is $\leq w$. Therefore by the inductive assumption ρ' is equivalent to a run ρ_1 on $\mathcal{L}$ of rank $\leq 2w+2$. Let ρ_2 be a run on $\mathcal{L} \times (\omega^{-1} + \omega)$ which is isomorphic to ρ_1 on each summand. The leftmost and the rightmost transitions of ρ_2 are limit transitions on D. Hence, replacing the last and the first states of ρ_2 by the last and first state of ρ we obtain a run ρ_3 on $\mathcal{L} \times (\omega^{-1} + \omega)$, which is equivalent to ρ. Since $\mathrm{rank}(\mathcal{L} \times (\omega^{-1} + \omega)) \leq 2(w+1)+1$, we have proved subcase 1.

Subcase 2. There is a D^+ cut c and $y > c$ such that no D^+ or D^- cut appears in ρ on (c, y).

This subcase is similar to subcase 1.

Subcase 3. neither subcase 1 nor subcase 2 holds.

Define an equivalence $\sim$ on the domain of ρ as: $c_1 \sim c_2$ if no D transition occurs in ρ on $[\min(c_1, c_2), \max(c_1, c_2)]$. This is a convex equivalence relation, i.e., its equivalence classes are intervals. No D transition occurs inside any $\sim$-equivalence class.

Every equivalence class has a minimal element. Indeed, if E has no minimal element, let $\mathrm{llim}(E) := \{\mathfrak{b} \in B \mid \exists y \in E \forall y' \in E (y' < y) \rightarrow \mathfrak{b} \in \rho(y')\}$. Then $\mathrm{llim}(E) \supseteq D$. If $\mathrm{llim}(E) = D$ then subcase 2 holds. If $\mathrm{llim}(E) = D_1 \supsetneq D$, let c be the infimum of elements in E. Then c is equivalent to every element in E and this contradicts that $c \notin E$.

Similarly, every equivalence class has a maximal element.

Hence, the restriction of ρ to every equivalence class is a run of weight $< \mathrm{weight}(D) = w+1$, and, by the inductive assumption, it is equivalent to a run on a chain of rank $\leq 2w+2$.

There is no consecutive equivalence classes because a D transition must occur between them, but they have minimal and maximal elements. Therefore, the chain of equivalence classes is either singleton or dense. If there is only one equivalence class we are done.

Assume that the chain of the equivalence classes is dense.

Let b be a maximal element of an equivalence class. We claim that there is a limit transition $q \leftarrow D$ at b in ρ. Indeed, there should be a limit transition $q \leftarrow D_1$ at b in ρ (otherwise there is a consecutive element which could be added to the equivalence class of b). $D_1 \supseteq D$ because D holds everywhere in ρ. If $D_1 \supsetneq D$ then there is $c > b$ such that D_1 holds in ρ on $[b, c]$, Therefore, $c \sim b$ and this contradicts the assumption that b is a maximal element in its class. Hence, $q \leftarrow D$ is the transition at b in ρ.

Similarly, if c is a minimal element in an equivalence class, then there is a transition $D \rightarrow q'$ at c in ρ.

Call an equivalence class $\mathcal{L}$-class if it contains the cuts a^- and a^+ for an element $a \in \mathcal{L}$. The chain $\mathcal{L}_\sim$ of $\mathcal{L}$-equivalence classes is dense and countable, hence, it is isomorphic to $\mathbb{Q}$. Label $q \in \mathbb{Q}$ by the type of ρ on the corresponding $\mathcal{L}_\sim$-class. Since there are only finitely many labels, by Lemma 4.8, there is an open non-empty interval $I \subseteq \mathbb{Q}$ and types $\tau_1, \ldots \tau_p$ such that the points labeled by τ_i are dense everywhere in I and the predicates $P_{\tau_l} := \{q \in I \mid \text{the corresponding interval } \mathcal{L}_q \text{ is of type } \tau\}$ partition I.

Let $I_1 < I_2$ be two equivalence classes in I and let a_1 be the last element of I_1 and a_2 be the first element of I_2.

First note that $\rho\lfloor[a_1, a_2]$ is equivalent to a run on the shuffle of chains where each chain is of rank $\leq 2w+2$.

Let ρ' be obtained by changing the first and the last states of $\rho\lfloor[a_1, a_2]$ to the first and the last states of ρ. ρ' is a run because the first and the last transitions of ρ were limit transitions over D. $type_{\mathfrak{A}}(\rho') = type_{\mathfrak{A}}(\rho)$

and ρ' is equivalent to a run on the shuffle of chains of rank $\leq 2w+2$. Hence, it is equivalent to a run on a chain of rank $\leq 2w+3 = 2(w+1)+1$. This completes the proof of Case 1.

Case 2. There is no D^- transition in ρ.

If ρ contains no D^+ cut then the $weight(\rho) \leq w$ and by the inductive hypothesis it is equivalent to a run on chain of rank $2w+2$.

If ρ contains a D^+ cut, then there is a maximal D^+ cut. Otherwise let the left transition at the supremum c' of such cut be on D'. Then $D' \subseteq D$, because it is the supremum of D transitions. On the other hand $D' \supseteq D$ because D holds everywhere. Therefore, $D' = D$ and this contradicts that there is no D^- transition.

Let e be the maximal D^+ cut and let b be the infimum of D^+ cuts.

Let $\rho_1 := \rho\lfloor\leq b$ and $\rho_3 := \rho\lfloor\geq e$. Since ρ_1 does not contain any cut of weight $\geq w$, it is equivalent to a run ρ_1' on a chain $\mathcal{L}_1$ of rank $\leq 2w+2$. By arguments similar to the subcase 1 of Case 1 we can show that ρ_3 is equivalent to a run ρ_3' on a chain $\mathcal{L}_3$ of rank $\leq 2w+3$.

Indeed let $y_1 > y_2 > \dots > y_n > \dots$ be an ω-sequence converging to e. For $i < j$ let $\mathrm{col}(y_i, y_j) := type_{\mathfrak{A}}(\rho\lfloor[y_j, y_i])$. By the Ramsey theorem there is a cofinal homogeneous subsequence z_i of y_i, i.e., $\mathrm{col}(z_i, z_j)$ is the same for all $i < j$. Let $q \xrightarrow{D'} q'$ be this color. Then from homogeneity it follows that $q = q'$ and $D' = D' \cap q$. Because this sequence converges to e and at e there is a D transition, we obtain that $D' = D$. Let ρ' be $\rho\lfloor[z_2, z_1]$. ρ' contains no D^- or D^+ transition. Hence its weight is $\leq w$. Therefore by the inductive assumption ρ' is equivalent to a run ρ_4 on $\mathcal{L}$ of rank $\leq 2w+2$. Let ρ_5 be a run on $\mathcal{L}\times\omega^{-1}$ which is isomorphic to ρ_4 on each summand. The leftmost transition of ρ_5 is a limit transition on D. Hence, replacing the first state of ρ_5 by the first state of ρ_3 we obtain a run ρ_5' on $\mathcal{L}_5 := \mathcal{L}\times\omega^{-1}$, which is equivalent to $\rho_3\lfloor\leq z_1$. Since $\rho_3\lfloor\geq z_1$ does not contain any transition of weight $\geq w$, it is equivalent to a run ρ_2 on $\mathcal{L}_2$ of rank $\leq 2w+2$. Therefore, ρ_3 is equivalent to the run $\rho_3' := \rho_5 + \rho_2$ on $\mathcal{L}_3 := \mathcal{L}_5 + \mathcal{L}_2$.

The transition on the right of b in ρ is a limit transition on D^+. The transition on the right of e in ρ_3' is a limit transition on D^+. Therefore by changing the first state of ρ_3' to $\rho(b)$ we obtain a run on $\mathcal{L}_3$. Hence $\rho_1 + \rho_3$ is well defined and is a run on $\mathcal{L}_1 + \mathcal{L}_3$ which has the rank $\leq 2(w+1)+1$.

This run is equivalent to ρ because they have the same state at the beginning and the end and the set of base elements true everywhere in these runs is D.

Case 3. There is no D^+ transition in ρ. This case is similar to Case 2.

Case 4. (1)-(3) fails. Hence, ρ contains D^+ and D^- cuts.

Let $c_+^{\inf}$ be infimum of D^+ cuts and $c_-^{\sup}$ be the supremum of D^- cuts.

There is a D^+ transition at $c_+^{\inf}$ and D^- transition at $c_-^{\sup}$.

$\rho\lfloor\leq c_+^{\inf}$ does not contain D^+ cuts, hence by case 3 it is equivalent to a run on $\mathcal{L}_1$ of rank $\leq 2w+3$.

$\rho\lfloor\geq c_-^{\sup}$ does not contain D^- cuts, hence by case 2 it is equivalent to a run on $\mathcal{L}_2$ of rank $\leq 2w+3$.

If $c_-^{\sup} = c_+^{\inf}$, then ρ is equivalent to a run on $\mathcal{L}_1 + \mathcal{L}_2$ of rank $\leq 2w+3$.

If $c_-^{\sup} < c_+^{\inf}$, then $\rho\lfloor[c_-^{\sup} < c_+^{\inf}]$ does not contain any D cut and therefore its weight is $\leq w$ and it is equivalent to a run on $\mathcal{L}_3$ of rank $\leq 2w+2$. Hence, ρ is equivalent to a run on $\mathcal{L}_1 + \mathcal{L}_3 + \mathcal{L}_2$ of rank $\leq 2w+3$.

If $c_-^{\sup} > c_+^{\inf}$, then by case (1), $\rho\lfloor[c_+^{\inf}, c_-^{\sup}]$ is equivalent to a run on $\mathcal{L}_3$ of rank $\leq 2w+3$. Hence, ρ is equivalent to a run on $\mathcal{L}_1 + \mathcal{L}_3 + \mathcal{L}_2$ of rank $\leq 2w+3$.

10.4. Proof of Lemma 9.1(2)

We are going to prove Lemma 9.1(2) by induction on w. By Proposition 10.9, it is sufficient to consider the runs on the countable chains.

The *inductive base* is trivial.

Inductive step. $w \rightarrowtail w+1$ (we assume that Lemma 9.1(2) holds for w and Lemma 9.1(1) holds for $w+1$ and prove Lemma 9.1(2) for $w+1$).

Let ρ be a run. Define $\sim_\rho$ on the domain of ρ:

$x \sim_\rho y$ iff $x = y$ or for all $z < v \in [\min(x,y), \max(y,x)]$ the run $\rho\lfloor[z,v]$ is equivalent to a run on $\mathcal{L}$ of rank $\leq 2w+3$.

This is a convex equivalence relation. An equivalence class is *degenerate* if it singleton.

Let I be a non-degenerate equivalence class and let $c_I^{\sup}, c_I^{\inf}$ be the supremum and infimum of its elements. We claim that $\rho_I := \rho\lfloor[c_I^{\inf}, c_I^{\sup}]$ is equivalent to a run on $\mathcal{L}$ of rank $\leq 2w+4$.

Indeed, take any $x \in I$. If I has a maximal element then $\rho_I\lfloor \geq x$ is of rank $\leq 2w+3$.

If I has no maximal element, let $y_0 < y_1 \dots$ be an ω-sequence in I converging to the supremum $c_I^{\sup}$ of I. Let $\mathrm{col}(y_i, y_j)$ be the $\mathfrak{A}$-type of $\rho\lfloor[y_i, y_j]$. By the Ramsey theorem there is a homogeneous infinite subsequence z_i.

Since $z_0 \sim_\rho z_1$ there is a run ρ_1 on $\mathcal{L}_1$ equivalent to $\rho\lfloor[z_0, z_1]$, where $\mathrm{rank}(\mathcal{L}_1) \leq 2w+3$. Therefore, $\rho_\lfloor[z_0, c_I^{\sup}]$ is equivalent to a run on $\mathcal{L}_1 \times \omega$. Since $x \sim_\rho z_0$, we have that $\rho\lfloor[x, z_0]$ is equivalent to a run on $\mathcal{L}_0$ of rank $\leq 2w+3$. Therefore, $\rho_I\lfloor \geq x$ is equivalent to a run on $\mathcal{L}_0 + \mathcal{L}_1 \times \omega$ of rank $\leq 2w+4$.

Similar arguments show that $\rho_I\lfloor \leq x$ is equivalent to a run on a chain of rank $\leq 2w+4$. Hence ρ_I is equivalent to a run on a chain of rank $\leq 2w+4$.

Let $L_{\sim_\rho}$ be the chain of $\sim_\rho$-equivalence classes. We are going to show:

(a) $L_{\sim_\rho}$ does not contain consecutive elements.

(b) $L_{\sim_\rho}$ has no limit element.

From (a) and (b) it follows that $L_{\sim_\rho}$ has only one element. Therefore, there is only one equivalence class and ρ is equivalent to a run on a chain of rank $\leq 2w+4 = 2(w+1)+2$.

It remains to prove (a) and (b).

Proof (of *(a)*). Assume that $I_1 < I_2$ are consecutive equivalence classes.

Case 1. I_1 has a maximal element x and I_2 has a minimal element y. In this case $x \sim_\rho y$ and this contradicts that x and y are in different equivalence classes.

Case 2. I_1 has no maximal element and I_2 has a minimal element x. Let $D = \{\mathfrak{b} \in B \mid \exists z \in I_1 \forall c (c \in I_1\lfloor \geq z \rightarrow \mathfrak{b} \in \rho(c))\}$. The weight of D is at most $w+1$. There is $c_D \in I_1$ such that $\forall c \in I_1\lfloor \geq c_D(\rho(c) \supseteq D)$. Then for all $c \in [c_D, x]$: $\rho\lfloor[c, x]$ satisfies the assumptions of Lemma 9.1(1) and therefore it is equivalent to a run on $\mathcal{L}$ of rank $2(w+1)+1$. Then for all $c < c_2 \in [c_D, x]$: $\rho\lfloor[c, c_2]$ is equivalent to a run on a chain of rank $2(w+1)+1 = 2w+3$. Hence $c_D \sim_\rho x$. Contradiction.

Case 3. The case when I_1 has a maximal element and I_2 has no minimal element leads similarly to a contradiction.

Case 4. I_1 has no maximal element and I_2 has no minimal element. This is impossible, because the domain of ρ is a Dedekind-complete chain. □

Proof (of *(b)*). Let I be a limit $\sim_\rho$-equivalence class. W.l.o.g. assume that it is a right limit of elements in $L_{\sim_\rho}$. Since the domain of ρ is Dedekind-complete, it follows that I has a maximal element x. Assume that the right limit transition at x in ρ is over a set $D \subseteq B$. Let $y > x$ be such that $D \subseteq \rho(z)$ for all $z \in (x, y)$ and $\mathfrak{b} \notin D \rightarrow \exists z \in (x, y) \wedge \mathfrak{b} \notin \rho(z)$. For $z < v \in (x, y)$, $\rho\lfloor[z, v]$ either is a run of weight $\leq w$ or it satisfies the assumption of Lemma 9.1(1). Hence, it is equivalent to a run on $\mathcal{L}$ of rank $\leq 2w+3$. Hence, $z \sim_\rho v$ and this contradicts that between x and y there are infinitely many $\sim_\rho$-equivalence classes. □

10.5. A stronger version of Lemma 9.1

We state here a lemma which is slightly stronger than Lemma 9.1.

Let $\mathfrak{A}$ be an automaton. Similar to the definition of the rank of chain we assign a rank to the runs in $\mathcal{R}_{reg} := \mathcal{R}_{reg}^{\mathfrak{A}}$.

Define sets $\mathcal{R}^{\leq i} \subseteq \mathcal{R}_{reg}$ as follows:

1. $\mathcal{R}^{\leq 0}$ is the set of finite runs.
2. $\mathcal{R}^{\leq i+1}$ is the closure under $+$ of the union of the following sets
   (a) $\mathcal{R}^{\leq i}$,
   (b) $\rho \times \omega$ and $\rho \times \omega^{-1}$ for every $\rho \in \mathcal{R}^{\leq i}$ and
   (c) *shuffle*$(\mathcal{A})$ for every finite subset $\mathcal{A}$ of $\mathcal{R}^{\leq i}$.

A run ρ has rank $i+1$ if $\rho \in \mathcal{R}^{\leq i+1} \wedge \rho \notin \mathcal{R}^{\leq i}$.

It is clear that every run in $\mathcal{R}_{reg}$ has a finite rank. It is also clear that if ρ is a run of rank w then it is a run on a chain of rank $\leq w$. Observe also that there are runs on chains of rank two which are not in $\mathcal{R}_{reg}$.

If we replace everywhere in the proof of Lemma 9.1. "a run on a chain of rank w" by "a run of rank w" we obtain a proof of the following lemma.

Lemma 10.11. *Assume that ρ is a run on a countable chain of a finite base automaton $\mathfrak{A}$.*

1. *If $\rho :\xrightarrow{D}$ and* $\text{weight}(\rho) = \text{weight}(D) = w$*, then there is a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$ of rank $\leq 2w+1$ which is equivalent to ρ.*
2. *Any run ρ of weight $\leq w$ is equivalent to a run in $\mathcal{R}^{\mathfrak{A}}_{reg}$ of rank $\leq 2w+2$.*

Similar to the construction trees for the chains of finite rank we define construction trees for runs of finite rank. An important difference is that every construction tree for chains describes a unique chain. However, a construction tree for runs describes a set of runs.

Let $\mathfrak{A}$ be an automaton. A construction tree T for $\mathfrak{A}$-runs is a labeled tree which has the following properties: the leaves of T are labeled by pairs (s, s') where $\mathfrak{A}$ has a next state transition from s to s'; the internal nodes are labeled by $+, \times\omega, \times\omega^{-1}$ and *shuffle*; a node labeled by $\times\omega$ or by $\times\omega^{-1}$ has one child; a node labeled by $+$ has at least two children and these children are linearly ordered; a node labeled by *shuffle* has at least one child.

Let T be a construction tree. The set $[|T|]$ of $\mathfrak{A}$-runs described by T is defined as follows:

1. if T is a one-element tree then $\rho \in [|T|]$ if ρ is the label of the unique node of T.
2. Assume that the root of T is labeled by $\times\omega$ (or by $\times\omega^{-1}$) and T_1 is the subtree of T rooted at the child of its root, then $\rho \in [|T|]$ if $\rho \in \rho_1 \times \omega$ (respectively, $\rho \in \rho_1 \times \omega^{-1}$) for some $\rho_1 \in [|T_1|]$.
3. Assume that the root of T is labeled by $+$ and its children (ordered from younger to older) are trees $T_1, \dots, T_m$. Then $\rho \in [|T|]$ if there are $\rho_i \in [|T_i|]$ such that $\rho \in \rho_1 + \dots + \rho_m$.
4. Assume that the root of T is labeled by *shuffle* and its children are trees $T_1, \dots, T_m$. Then $\rho \in [|T|]$ if $\rho \in \textit{shuffle}(\rho_1, \dots, \rho_m)$ for $\rho_i \in [|T_i|]$.

Now similar to Lemma 7.1, we obtain

Lemma 10.12. *If an $\mathfrak{A}$-run ρ has rank $\leq i$, then there is a run construction tree T such that $\rho \in [|T|]$ and the height of T is bounded by $2i+1$.*

11. Temporal logics with automata definable modalities

In this section we provide an extension of our results to temporal logics with modalities having generalized truth tables definable by automata.

Let us first compare the expressive power of automata and of $\exists$-MSO. Theorem 8.3 implies that for every $\exists$-MSO formula φ there is an automaton $\mathfrak{A}_\varphi$ such that $\mathcal{M} \models \varphi$ iff $\mathfrak{A}_\varphi$ accepts $\mathcal{M}$. Hence, if a language is definable by an $\exists$-MSO formula, then it is definable by an automaton. However, there are languages definable by automata which are not definable by $\exists$-MSO. In particular,

Proposition 11.1.

1. *There is an automaton which accepts a linear order iff it is Dedekind-complete.*
2. *There is no $\exists$-MSO sentence which is satisfiable in a linear order iff it is Dedekind-complete.*

Proof. (1) Consider an automaton $\mathfrak{A}$ over a unary alphabet defined as follows. It has three states q_l, q_r and q_n. We define the transition relation of $\mathfrak{A}$ in such a way that in every $\mathfrak{A}$-run ρ if a cut $c = (L, U)$ and $L \neq \emptyset$ has no maximal element, then it will be labeled by q_l; if $U \neq \emptyset$ and has no minimal element it will be labeled by q_r; other non-gap cuts will be labeled by q_n.

Define the basis as $B := \{0, 1\}$. It is big enough to assign a different subsets of B to q_l, q_r and q_n.

δ_{next} has three transitions (q_l, q_n), (q_l, q_r) and (q_n, q_n).

The limit transition relation $\delta_{\lim}$ contains (q_r, D) and (D, q_l) for every $D \subseteq B$.

$Q_{init} := \{q_n, q_r\}$ and $Q_{fin} := \{q_n, q_l\}$.

It is easy to check that no $\mathfrak{A}$-run can assign a state to a cut which corresponds to a gap, and a linear order is accepted by $\mathfrak{A}$ iff it is Dedekind-complete.

(2) The standard, but lengthy, Ehrenfeucht-Fraïssé game arguments show that if φ is an $\exists$-MSO sentence and $\mathbb{Z} \models \varphi$, then $\mathbb{Z} + \mathbb{Z} \models \varphi$. However, $\mathbb{Z}$ is Dedekind-complete while $\mathbb{Z} + \mathbb{Z}$ is not. This proves (2). □

It is an open problem whether a language definable by an automaton is always definable by an MSO formula. A natural formalization of a run of automaton talks about sets of cuts of a linear order. The cuts are represented by the downward closed subset of linear order. Hence, such a formalization refers to a set of sets which is a third-order object.

Let $\mathfrak{A}$ be an automaton over an alphabet $\{0,1\}^n$ and let $\mathcal{L} := \langle A, < \rangle$ be linear order. A relation $R_{\mathfrak{A}} \subseteq \mathcal{P}(A)^n$ is said to be definable by $\mathfrak{A}$ in $\mathcal{L}$ if $R_{\mathfrak{A}} = \{(P_1, \dots, P_n) \in \mathcal{P}(A)^n \mid \mathfrak{A} \text{ accepts } (A, <, P_1, \dots, P_n)\}$.

An operator $F : \mathcal{P}(A)^{n-1} \to \mathcal{P}(A)$ is definable by $\mathfrak{A}$ in $\mathcal{L}$ if $R_{\mathfrak{A}} = \{(P_1, \dots, P_n) \in \mathcal{P}(A)^n \mid P_1 = F(P_2, \dots, P_n)\}$. A modality O is said to be definable by $\mathfrak{A}$ in $\mathcal{L}$ if the operator assigned to O in $\mathcal{L}$ is definable by $\mathfrak{A}$.

A modality O is said to be definable by $\mathfrak{A}$ if O is definable by $\mathfrak{A}$ in $\mathcal{L}$ for every $\mathcal{L}$.

Proposition 11.1 implies that the set of $\exists$-MSO definable modalities is a proper subset of the set of automata definable modalities.

The next theorem is the main result of this section.

Theorem 11.2. *Let TL be a temporal logic with a finite set of modalities and every modality of TL is definable by an automaton. Then the satisfiability problem for TL over the class of countable chains is in* PSPACE*.*

Remark 11.3 (On countability). *There are automata which accept a non-empty language, while do not accept any countable chain. For example, one can define an automaton which accepts only Dedekind-complete dense chains. This automaton is the product of the automaton for Dedekind-complete chains described in the proof of Proposition 11.1 and an automaton which accepts only dense chains (such an automaton can be easily described, also its existence follows from the first-order definability of the class of dense linear orders and Theorem 8.3). This automaton accepts the chain of reals, but does not accept any countable chain. Remark 10.10 implies that if an automaton accepts a chain, then it accepts a chain of cardinality at most continuum.*

The proof of Theorem 11.2 is similar to the proof of Theorem 7.3 and we outline it in this section.

Let Φ be a finite set of automata. A Φ-conjunctive formula is an expression φ of the form $\mathfrak{A}_1(\overline{P}_1) \wedge \mathfrak{A}_2(\overline{P}_2) \wedge \dots \wedge \mathfrak{A}_k(\overline{P}_k)$, where $\mathfrak{A}_i \in \Phi$ and $\overline{P}_i$ is an n_i-tuple of predicate names, whenever the alphabet of $\mathfrak{A}_i$ is $\{0,1\}^{n_i}$. The *size* of φ is defined to be k and is denoted by $|\varphi|$.

A chain $\mathcal{L} = (A, <, \overline{P})$ satisfies (or is accepted by) φ if $\mathfrak{A}_i$ accepts the reduct of $\mathcal{L}$ on $\{<, \overline{P}_i\}$ for every i.

The next proposition is similar to Proposition 3.1 and reduces (in linear time) the satisfiability problem for *TL* to the satisfiability problem for Φ-conjunctive formulas.

Proposition 11.4. *Let TL be a temporal logic with a finite set of modalities. Assume that every modality of TL is definable by an automaton. Then there is a finite set Φ of automata and a linear time algorithm which for every formula $\psi \in TL$ computes a Φ-conjunctive formula φ such that φ is satisfiable iff ψ is satisfiable.*

The proof of this proposition is based on a simple unnesting procedure similar to the proof of Proposition 3.1.

Let $\varphi := \mathfrak{A}_1(\overline{P}_1) \wedge \mathfrak{A}_2(\overline{P}_2) \wedge \dots \wedge \mathfrak{A}_k(\overline{P}_k)$ be a Φ-conjunctive formula. It is clear that for every conjunct $\mathfrak{A}_i(\overline{P}_i)$ there is an automaton which accepts $\mathcal{L}$ iff $\mathcal{L}$ satisfies $\mathfrak{A}_i(\overline{P}_i)$. The only difference between this automaton and $\mathfrak{A}_i$ is in the next transition relation which takes into an account the order of names in the tuple $\overline{P}_i$. We denote this automaton by $\mathfrak{A}_i(\overline{P}_i) := (B_i, Q_i, \Sigma_i, \delta_i^{next}, \delta_i^{\lim}, Q_i^{init}, Q_i^{fin})$. W.l.o.g we can assume that B_i are disjoint for $i = 1, \dots, k$ (otherwise we can take isomorphic copies). Let us define $\mathfrak{A}_\varphi := (B, Q, \Sigma, \delta_{next}, \delta_{\lim}, Q_{init}, Q_{fin})$ equivalent to φ as the product of $\mathfrak{A}_i(\overline{P}_i)$:

1. $B = \cup B_i$.
2. $s \in \mathcal{P}(B)$ is a state of $\mathfrak{A}_\varphi$ if $s_i := s \cap B_i \in Q_i$; such s will be denoted by $(s_1, \dots, s_k)$.
3. $(s_1, \dots, s_k)$ in Q_{init} (respectively, in Q_{fin}) iff $s_i \in Q_i^{init}$ (respectively, $s_i \in Q_i^{fin}$) for $i \leq k$.
4. $D \to (s_1, \dots, s_k)$ is a left limit transition of $\mathfrak{A}_\varphi$, if $D \cap B_i \to s_i$ is a left limit transition of $\mathfrak{A}_i(\overline{P}_i)$.
5. The right limit transition relation is defined similarly. The next state transition relation is defined in the standard way.

The following lemmas are immediate.

Lemma 11.5 (Equivalence of φ and $\mathfrak{A}_\varphi$). *$\mathcal{L}$ satisfies φ iff $\mathfrak{A}_\varphi$ accepts $\mathcal{L}$.*

Lemma 11.6. *For every finite set Φ of automata there is n_Φ such that for every Φ-conjunctive formula φ the size of the base of $\mathfrak{A}_\varphi$ is at most $|\varphi| \times n_\Phi$.*

By Lemmas 10.11, 10.12 and 11.6, we obtain the following lemma:

Lemma 11.7. *For every finite set Φ of automata there is r_Φ such that for every Φ-conjunctive formula $\varphi = \mathfrak{A}_1(\overline{P_1}) \wedge \dots \wedge \mathfrak{A}_k(\overline{P_k})$ and every $\mathfrak{A}_\varphi$-type τ, there is an $\mathfrak{A}_\varphi$-run ρ on a countable chain such that $\tau = type_{\mathfrak{A}_\varphi}(\rho)$ iff there is a run construction tree T of height $\leq |\varphi| \times r_\Phi$ and a run $\rho' \in [|T|]$ such that $\tau = type_{\mathfrak{A}_\varphi}(\rho')$.*

Let $\varphi := \mathfrak{A}_1(\overline{P_1}) \wedge \dots \wedge \mathfrak{A}_k(\overline{P_k})$. For $i = 1, \dots, k$, let $\tau_i = (s_i, D_i, s'_i)$ be a formal $\mathfrak{A}_i(\overline{P}_i)$ type. We denote by $(\tau_1, \dots, \tau_k)$ a formal $\mathfrak{A}_\varphi$-type $((s_1, \dots, s_k), \cup D_i, (s'_1, \dots, s'_k))$. Observe that for every $\mathfrak{A}_\varphi$-type $\tau := (s, D, s')$ there is a unique tuple $(\tau_1, \dots, \tau_k)$ such that $\tau = (\tau_1, \dots, \tau_k)$; moreover $\tau_j = (s \cap B_j, D \cap B_j, s' \cap B_j)$ for $j = 1, \dots, k$ are computable from τ in linear time.

Hence, $\mathfrak{A}_\varphi$ accepts a countable chain iff

(A) there is an $\mathfrak{A}_\varphi$-type $\tau = ((s_1, D_1, s'_1), \dots, (s_k, D_k, s'_k))$, a run construction tree T of height $\leq |\varphi| \times r_\Phi$ and a run $\rho \in [|T|]$ such that $type_{\mathfrak{A}_\varphi}(\rho) = ((s_1, D_1, s'_1), \dots, (s_k, D_k, s'_k))$ and

(B) $(s_1, \dots, s_k) \in Q_{init}$ and $(s'_1, \dots, s'_k) \in Q_{fin}$.

Condition (B) can be checked in linear time. Hence, by Lemmas 11.4 and 11.5 we obtain

Lemma 11.8. *The satisfiability problem for TL over the class of countable chains is in* PSPACE *if (A) can be checked in* PSPACE.

Recall that in Section 10.1 we defined operations $+, \times\omega, \times\omega^{-1}$ and *shuffle* on runs and on automata types. Unlike similar operations on the chains and on first-order types, these operations return sets of runs and sets of automata types. The next lemma is a version of Lemma 5.2 for automata runs and types.

Lemma 11.9. *Let $\varphi := \mathfrak{A}_1(\overline{P}_1) \wedge \mathfrak{A}_2(\overline{P}_2) \wedge \dots \wedge \mathfrak{A}_k(\overline{P}_k)$ be a Φ-conjunctive formula.*

1. *Assume that ρ_1 and ρ_2 are $\mathfrak{A}_\varphi$-runs, $type_{\mathfrak{A}_\varphi}(\rho_1) = (\tau_1^1, \dots, \tau_k^1)$ and $type_{\mathfrak{A}_\varphi}(\rho_2) = (\tau_1^2, \dots, \tau_k^2)$ Then*
$$\{type_{\mathfrak{A}_\varphi}(\rho) \mid \rho \in \rho_1 + \rho_2\} = \{(\tau_1, \dots, \tau_k) \mid \tau_j \in \tau_j^1 + \tau_j^2\}$$
2. *Assume that ρ' is an $\mathfrak{A}_\varphi$-run and $type_{\mathfrak{A}_\varphi}(\rho') = (\tau'_1, \dots, \tau'_k)$. Then*
$$\{type_{\mathfrak{A}_\varphi}(\rho) \mid \rho \in \rho' \times \omega\} = \{(\tau_1, \dots, \tau_k) \mid \tau_j \in \tau'_j \times \omega\}$$
.
3. *Similar to (2) but for multiplication by ω^{-1}.*
4. *Assume that $\rho_1, \dots, \rho_m$ are $\mathfrak{A}_\varphi$-runs, $type_{\mathfrak{A}_\varphi}(\rho_i) = (\tau_1^i, \dots, \tau_k^i)$ for $i = 1, \dots, k$ and $s_1 = (s_1^1, \dots, s_k^1) \dots, s_l = (s_1^l, \dots, s_k^l)$ are states of $\mathfrak{A}_\varphi$. Let $U_j = \{\tau_j^i \mid i \leq m\}$ for $0 < j \leq k$ and, $S_j = \{s_j^i \mid i \leq l\}$ for $0 < j \leq l$. Then*
$$\{type_{\mathfrak{A}_\varphi}(\rho) \mid \rho \in shuffle_{\mathfrak{A}_\varphi}(\rho_1, \dots, \rho_m, s_1 \dots, s_l)\} = \{(\tau_1, \dots, \tau_k) \mid \tau_j \in shuffle_{\mathfrak{A}_j(\overline{P}_j)}(U_j, S_j)\}.$$

Fig. 2 contains an algorithm SAT for the following problem:

Problem for a finite set Φ of automata

Input A Φ-conjunctive formula φ, an $\mathfrak{A}_\varphi$-type τ and $N \in \mathbb{N}$.

Output True, if there a run construction tree T of height $\leq N$ and an $\mathfrak{A}_\varphi$-run $\rho \in [|T|]$ such that $type_{\mathfrak{A}_\varphi}(\rho) = \tau$.

The only difference between this algorithm and the algorithm in Fig. 1 is that it uses automata types instead of first-order types.

The correctness of SAT easily follows from Lemma 11.9 and the definition of run construction trees and their semantics. Arguments similar to the argument for the algorithm presented in Fig. 1 (Sect. 7) show that SAT is in NSPACE($|\varphi| \times N$). Hence, SAT works in polynomial space in $|\varphi|$, assuming that N is polynomial in $|\varphi|$ which is the case with $N = |\varphi| \times r_\Phi$. Hence, (A) can be verified in PSPACE, and by Lemma 11.8, we obtain that the satisfiability problem for *TL* is in PSPACE.

This completes our proof of Theorem 11.2.

Remark 11.10. *Theorem 11.2 provides a* PSPACE *algorithm for the satisfiability problem for any temporal logic with a finite set of automata definable modalities over the class of countable linear orders. This theorem can be extended to the class of all linear orders. A version of Lemma 10.11 holds for arbitrary runs of automata (see Remark 10.10). The algorithm in Fig. 2 can be modified to include a clause for* $\mathbb{R} - shuffle$. *Thus we obtain the desirable* PSPACE *algorithm.*

12. Extension to subclasses of linear orders

Let *TL* be any temporal logic with a finite set of $\exists$-MSO definable modalities. We proved that the satisfiability problem for *TL* over the class of all linear orders can be solved in PSPACE. This improves the Cristau result [6] that the satisfiability problem over this class for the temporal logic having the four modalities Until, Since, Until_{Stavi} and Since_{Stavi} is in double exponential space.

In the rest of this section we explain how the PSPACE bound can be extended uniformly to many interesting classes of linear orders.

Definition 12.1. *Let ψ be an $\exists$-MSO sentence. A set $\mathcal{C}$ of chains is said to be* definable by ψ, *if $\mathcal{C} = \{\mathcal{M} \mid \mathcal{M} \models \psi\}$. A set $\mathcal{C}$ of chains is said to be* definable by ψ relatively to a class $\mathcal{C}'$, *if $\mathcal{C} = \{\mathcal{M} \in \mathcal{C}' \mid \mathcal{M} \models \psi\}$.*

Theorem 7.3 immediately implies

Corollary 12.2. *Let TL be a temporal logic with a finite set of $\exists$-MSO definable modalities, and let ψ be an $\exists$-MSO sentence. If the satisfiability problem for TL over $\mathcal{C}'$ is in* PSPACE, *then the satisfiability problem for TL over the class of chains definable by ψ relatively to $\mathcal{C}'$ is in* PSPACE. *In particular, the satisfiability problem for TL over the class of chains definable by ψ is in* PSPACE.

A linear order is called *unbounded* if it has neither a minimum nor a maximum. Note that an $\exists$-MSO formula φ is satisfiable in $\mathbb{Q}$ iff it is satisfiable in an unbounded dense order. There are first-order sentences *Unbound* and *Dense* that express that an order is unbounded, respectively, dense. Therefore, φ is satisfiable in $\mathbb{Q}$ iff $Unbound \wedge Dense \wedge \varphi$ is satisfiable over a linear order. Hence, there is a PSPACE algorithm for satisfiability in $\mathbb{Q}$.

Recall that a cut (L, U) of a linear order $\mathcal{L}$ is a gap if neither L has a maximal element, nor U has a minimal element and $L \neq \emptyset \neq U$. A chain is Dedekind-complete if its underlining order does not have gaps. The class of non-Dedekind-complete chains can be easily definable by an $\exists$-MSO sentence. Hence, there is a PSPACE algorithm for the satisfiability over the class of non-Dedekind-complete chains. By Proposition 11.1, the class of Dedekind-complete chains is not definable by an $\exists$-MSO sentence. However, we will show (Theorem 12.8) that there is a PSPACE algorithm for the satisfiability over the class of Dedekind-complete chains.

Input 1. $(\tau_1, \ldots, \tau_k)$, where $\tau_i = (s_i, D_i, s'_i)$ is an $\mathfrak{A}_i(\overline{P}_i)$ type.
2. $N \in \mathbb{N}$.

Output True, if there is a run construction tree T of height N and a run $\rho \in [|T|]$ such that $type_{\mathfrak{A}_\varphi}(\rho) = (\tau_1, \ldots, \tau_k)$.

- If $N = 0$ and there is a next state transition in $\mathfrak{A}_\varphi$ from $(s_1, \ldots, s_k)$ to $(s'_1, \ldots, s'_k)$ then return True;
- Go non-deterministically to 1-5.

(1.) Return SAT$((\tau_1, \ldots, \tau_k), N-1)$.
(2.) Guess $(\tau'_1, \ldots, \tau'_k)$ such that SAT$((\tau'_1, \ldots, \tau'_k), N-1)$ returns True and $\tau_i \in \tau'_i \times \omega$ for $0 < i \leq k$.
(3.) Guess $(\tau'_1, \ldots, \tau'_k)$ such that SAT$((\tau'_1, \ldots, \tau'_k), N-1)$ returns True and $\tau_i \in \tau'_i \times \omega^{-1}$ for $0 < i \leq k$.
(4.) Guess on-the-fly a sequence

$$\left(\tau_1^1, \ldots, \tau_k^1\right), \left(\tau_1^2, \ldots, \tau_k^2\right), \ldots, \left(\tau_1^m, \ldots, \tau_k^m\right)$$

such that

(4.1) for $0 < i \leq m$, SAT$(\left(\tau_1^i, \ldots, \tau_k^i\right), N-1)$ returns True,
(4.2) for $0 < j \leq k$, $\tau_j \in \tau_j^1 + \ldots + \tau_j^m$.

(5.) Guess $(U_1, \ldots, U_k)$, and $(S_1, \ldots, S_k)$ where U_i is a set of $\mathfrak{A}_i(\overline{P}_i)$ types and S_i is a set of $\mathfrak{A}_i(\overline{P}_i)$ states such that

(5.1) for $0 < j \leq k$, $\tau_j \in shuffle_{\mathfrak{A}_j(\overline{P}_j)}(U_j, S_j)$,
(5.2) for $0 < j \leq k$, and every $s_j \in S_j$ check that $D_j \to s_j$ and $s_j \leftarrow D_j$ are limit transitions of $\mathfrak{A}_j(\overline{P}_j)$, where $D_j := (\bigcap_{s \in S_j} s) \cap (\bigcap_{(q,D,q') \in U_j} D)$.

Guess on-the-fly a sequence of $\mathfrak{A}_\varphi$ types and a non-empty sequence of $\mathfrak{A}_\varphi$ states:

$$\left(\tau_1^1, \ldots, \tau_k^1\right), \left(\tau_1^2, \ldots, \tau_k^2\right), \ldots, \left(\tau_1^m, \ldots, \tau_k^m\right) \text{ and } \left(s_1^1, \ldots, s_k^1\right) \ldots, \left(s_1^l, \ldots, s_k^l\right)$$

such that

(5.3) for $0 < i \leq m$, SAT$(\left(\tau_1^i, \ldots, \tau_k^i\right), N-1)$ returns True,
(5.4) for $0 < j \leq k$ and $U_j = \{\tau_j^i \mid i \leq m\}$,
(5.5) for $0 < j \leq l$ and $S_j = \{s_j^i \mid i \leq l\}$.

Figure 2: Algorithm SAT

Let OP be a subset of $\{\omega, \omega^{-1}, \mathit{shuffle}\}$. We can prove a version of Lemma 9.1, where "a run on $\mathcal{L}$ of rank m" is replaced by "a run on $\mathcal{L} \in \mathit{Cl}(\mathit{One}, OP \cup \{+\})$ of rank m". The proof of this lemma is exactly like the proof of Lemma 9.1. The only additional property of the class $\mathit{Cl}(\mathit{One}, OP \cup \{+\})$ needed for this proof is: if $\mathcal{L} \in \mathit{Cl}(\mathit{One}, OP \cup \{+\})$ has rank m, then for every interval I the sub-chain of $\mathcal{L}$ over I is in $\mathit{Cl}(\mathit{One}, OP \cup \{+\})$ and its rank is at most m. As a consequence we obtain the following variant of the small rank property (Lemma 7.2).

Lemma 12.3. *For every finite set Φ of first-order formulas and every $OP \subseteq \{\omega, \omega^{-1}, \mathit{shuffle}\}$ there is $N_{\Phi,OP} \in \mathbb{N}$ such that every Φ-conjunctive formula ψ is satisfiable in $\mathit{Cl}(\mathit{One}, OP \cup \{+\})$ iff it is satisfiable in a chain $\mathcal{M} \in \mathit{Cl}(\mathit{One}, OP \cup \{+\})$ of rank $\leq |\psi| \times N_{\Phi,OP}$.*

Hence, the satisfiability problem for any temporal logic with a finite set of $\exists$-MSO definable modalities over $\mathit{Cl}(\mathit{One}, OP \cup \{+\})$ is in PSPACE.

Recall that a linear order is scattered if it does not contain a dense sub-order (i.e., a substructure order-isomorphic to $\mathbb{Q}$). An $\exists$-MSO formula is satisfiable in a chain over an ordinal (respectively, over a scattered order) iff it is satisfiable in $\mathit{Cl}(\mathit{One}, \{\omega, +\})$ (respectively, in $\mathit{Cl}(\mathit{One}, \{\omega, \omega^{-1}, +\})$) [13, 20]. Hence, we obtain:

Theorem 12.4. *Let TL be a temporal logic with a finite set of modalities definable in the existential fragment of MSO.*

1. *The satisfiability problem for TL in the class of chains over ordinals is in* PSPACE *[7].*
2. *The satisfiability problem for TL in the class of scattered chains is in* PSPACE.

A linear order is *continuous* if it is dense and Dedekind-complete; it is separable if it has a countable dense subset. Any unbounded separable continuous order is order-isomorphic to the reals.

Burgess and Gurevich [5] proved that $\mathit{TL}(\mathsf{Until}, \mathsf{Since})$ is decidable over the reals. They introduced the following class of chains.

Definition 12.5. *Let $\mathcal{C}$ be the minimal class of chains that contains all one-element chains and has the following properties:*

1. *If $\mathcal{M}$ and $\mathcal{N}$ are in $\mathcal{C}$ and $\mathcal{M}$ has a maximum or $\mathcal{N}$ has a minimum, then $\mathcal{M} + \mathcal{N} \in \mathcal{C}$.*
2. *If $\mathcal{M} \in \mathcal{C}$ and $\mathcal{M}$ has either a minimum or a maximum, then $\mathcal{M} \times \omega^{-1}$ and $\mathcal{M} \times \omega$ are in $\mathcal{C}$.*
3. *If $\mathcal{A} \subseteq \mathcal{C}$ is finite and each $\mathcal{M} \in \mathcal{A}$ has both a minimum and a maximum, and some $\mathcal{N} \in \mathcal{A}$ are one-element chains, then $\mathit{shuffle}(\mathcal{A}) \in \mathcal{C}$.*

The next theorem was a key step in their decidability proof.

Theorem 12.6. *Let φ be an $\exists$-MSO formula. The following are equivalent:*

1. *φ is satisfiable over the class of Dedekind-complete separable chains.*
2. *φ is satisfiable over the class of Dedekind-complete chains.*
3. *φ is satisfiable in $\mathcal{C}$.*

As a consequence, they obtained a (non-elementary) algorithm for the decidability of $\mathit{TL}(\mathsf{Until}, \mathsf{Since})$ over the reals.

The definition of $\mathcal{C}$ is slightly more general than the definition of a recursively defined class of structures. However, our definition is easily extended to the (mutual) recursive definition of a finite number of classes.

One can easily rephrase Definition 12.5 as a mutual recursive definition of three classes: $\mathcal{C}$, $\mathcal{C}_{\max}$ and $\mathcal{C}_{\min}$, where $\mathcal{C}_{\max}$ (respectively, $\mathcal{C}_{\min}$) is the set of chains in $\mathcal{C}$ with a maximal, (respectively, minimal) element. (Note that $\mathcal{C}_{\max}$ and $\mathcal{C}_{\min}$ are $\exists$-MSO definable relatively to $\mathcal{C}$.)

Our results and proofs are easily extended to these classes. We need modified versions of Lemma 9.1. In the version of Lemma 9.1 for $\mathcal{C}$ (respectively, for $\mathcal{C}_{\max}$ and $\mathcal{C}_{\min}$) "a run on $\mathcal{L}$ of rank m" is replaced by "a run on $\mathcal{L} \in \mathcal{C}$ (respectively in $\mathcal{C}_{\max}$ and $\mathcal{C}_{\min}$) of rank m". The proof of these lemmas is exactly like the proof of Lemma 9.1. The only additional property needed in these proofs is: if $\mathcal{L} \in \mathcal{C}$ (respectively, in $\mathcal{C}_{\max}$

and $\mathcal{C}_{\min}$) has rank m, then for every interval (respectively, right closed interval, left closed interval) I the sub-chain of $\mathcal{L}$ over I is in $\mathcal{C}$ (respectively, in $\mathcal{C}_{\max}$ and $\mathcal{C}_{\min}$) and its rank is at most m.

As a consequence we obtain that for every Φ there is r_Φ such that a Φ-conjunctive formula ψ is satisfiable in $\mathcal{C}$ iff it is satisfiable in $\mathcal{M} \in \mathcal{C}$ of rank $\leq r_\Phi \times |\psi|$. Hence,

Lemma 12.7. *Let TL be a temporal logic with a finite set of modalities definable in ∃-MSO. The satisfiability problem for TL in $\mathcal{C}$ is in* PSPACE.

As a consequence, we obtain:

Theorem 12.8. *Let TL be a temporal logic with a finite set of modalities definable in the existential fragment of MSO.*

1. *The satisfiability problem for TL over the class of Dedekind-complete separable chains is in* PSPACE.
2. *The satisfiability problem for TL over the class of Dedekind-complete chains is in* PSPACE.
3. *The satisfiability problem for TL in the class of chains over the reals is in* PSPACE.
4. *The satisfiability problem for TL over the class of continuous chains is in* PSPACE.

Proof. (1) and (2) follow from Theorem 12.6 and Lemma 12.7.

Let *Unbound* and *Dense* be first-order formulas that express that an order is unbounded and dense. By Theorem 12.6, $\varphi \in TL$ is satisfiable over the reals iff $\varphi \wedge Dense \wedge Unbound$ is satisfiable in $\mathcal{C}$. Therefore, (3) follows by Lemma 12.7.

$\varphi \in TL$ is satisfiable over the class of continuous chains iff $\varphi \wedge Dense$ is satisfiable in $\mathcal{C}$. Therefore, (4) follows by Lemma 12.7. □

The class of non-scattered chains can be easily definable by an ∃-MSO formula. Hence, by Theorem 12.8, there is a PSPACE algorithm for satisfiability over the class of non-scattered Dedekind-complete chains. The class of scattered chains is not definable by automata [1].

In order to prove PSPACE bound for the class of scattered Dedekind-complete chains, we need a characterization similar to Definition 12.5 and Theorem 12.6.

Definition 12.9. *Let $\mathcal{C}'$ be the minimal class of chains that contains all one-element chains and has the following properties:*

1. *If $\mathcal{M}$ and $\mathcal{N}$ are in $\mathcal{C}'$ and $\mathcal{M}$ has a maximum or $\mathcal{N}$ has a minimum, then $\mathcal{M} + \mathcal{N} \in \mathcal{C}'$.*
2. *If $\mathcal{M} \in \mathcal{C}'$ and $\mathcal{M}$ has either a minimum or a maximum, then $\mathcal{M} \times \omega^{-1}$ and $\mathcal{M} \times \omega$ are in $\mathcal{C}'$.*

Similar to Theorem 12.6 we have the following characterization:

Proposition 12.10. *Let φ be in the existential fragment of MSO. The following are equivalent:*

1. *φ is satisfiable over the class of Dedekind-complete scattered chains.*
2. *φ is satisfiable in $\mathcal{C}'$.*

One can easily rephrase Definition 12.9 as a mutual recursive definition of three classes: $\mathcal{C}'$, $\mathcal{C}'_{\max}$ and $\mathcal{C}'_{\min}$, where $\mathcal{C}'_{\max}$ (respectively, $\mathcal{C}'_{\min}$) is the set of chains in $\mathcal{C}'$ with a maximal, (respectively, minimal) element. (Note that $\mathcal{C}'_{\max}$ and $\mathcal{C}'_{\min}$ are ∃-MSO definable relatively to $\mathcal{C}'$.)

Theorem 12.11. *Let TL be a temporal logic with a finite set of modalities definable in the existential fragment of MSO. The satisfiability problem for TL over the class of Dedekind-complete scattered chains is in* PSPACE.

Proof. Arguments are similar to the proof of Theorem 12.8(1), but use the class $\mathcal{C}'$ instead of $\mathcal{C}$. □

Let us state one more theorem.

Theorem 12.12. *Let TL be a temporal logic with a finite set of automata definable modalities. The satisfiability problem for TL over the class of Dedekind-complete countable chains is in* PSPACE.

Proof. Follows immediately from Theorem 11.2 and Proposition 11.1 which states that the class of Dedekind-complete linear orders is definable by an automaton. □

Let us summarize the principles and methods used in our proofs to establish the PSPACE upper bound for the satisfiability problem for temporal logics.

Our first unnesting reduction shows that for every temporal logic $\mathcal{L}$ with a finite set of $\exists$-MSO definable modalities there is a finite set Φ of first-order formulas and a linear time algorithm that reduces the satisfiability problem for $\mathcal{L}$ to the satisfiability problem for Φ-conjunctive formulas. This reduction is valid over every class of structures. Technically, it is a very simple reduction; however, it frees us from temporal logics and allows to apply a rich variety of methods developed for first-order logic.

We introduced recursively definable classes of structures. Let $\mathcal{C} = Cl(One, \mathfrak{F})$ be a recursively definable class, where (1) the operators from $\mathfrak{F}$ respect $\equiv^n$ and (2) the reduct distributes over these operators.

The satisfiability problem over the structures of rank $\leq k$ in $\mathcal{C}$ is in PSPACE$(|\varphi| \times r)$. This is a general result and its proof is based on composition method.

To establish the PSPACE upper bound for the satisfiability problem of Φ-conjunctive formula over $\mathcal{C}$, it is sufficient to prove the small rank property for $\mathcal{C}$: if a conjunctive formula φ is satisfiable over a chain in $\mathcal{C}$, then it is satisfiable over a chain in $\mathcal{C}$ of rank polynomial in φ. The proof of the small rank property for the class $Cl(One, \{+, \times\omega, \times\omega^{-1}, \mathit{shuffle}\})$ was based on the technique introduced by Läuchli and Leonard [13] to prove the decidability of the first-order logic over the linear orders. We (1) adopted this technique to automata types and (2) strengthened its inductive assertions and using finite base automata established a polynomial bound which ensures the small rank property. We sketched proofs of the small rank property for several other recursively definable classes. These proofs are almost the same as the proof for $Cl(One, \{+, \times\omega, \times\omega^{-1}, \mathit{shuffle}\})$. However, the small rank property is not valid for a general recursively definable class of chains. In the cases where we succeeded to prove the small rank property for a recursively defined class $\mathcal{C}$, we first proved that if $\mathcal{L} \in \mathcal{C}$ has rank m, then for every interval I if the sub-chain of $\mathcal{L}$ over I is in $\mathcal{C}$, then its rank is at most m.

The following standard general principles allow to transfer an upper bound for the satisfiability problem over one class to another class.

Definability arguments If $\mathcal{C}$ is definable relatively to $\mathcal{C}'$, then the satisfiability problem for $\mathcal{C}$ has at most the same complexity as for $\mathcal{C}'$.

Density arguments If $\mathcal{C}'$ is dense in $\mathcal{C}$ (i.e., a formula φ is satisfiable in $\mathcal{C}$ iff it is satisfiable in $\mathcal{C}$), then the satisfiability problem for $\mathcal{C}$ is the same (and hence has the same complexity) as for $\mathcal{C}'$.

Proofs that one class is dense in another class often rely on the composition method.

13. Discussion of Reynolds' results

Recall that a temporal logic TL is expressively complete for first-order monadic logic of order (FOMLO) over a class $\mathcal{C}$ of structures if for every $\varphi \in TL$ there is a formula $\psi(t) \in$ FOMLO which is equivalent to φ over $\mathcal{C}$, and for every formula $\psi(t) \in$ FOMLO with at most one free variable there is a formula $\varphi \in TL$ which is equivalent to ψ over $\mathcal{C}$.

A major result concerning linear-time temporal logics is Kamp's theorem [12, 10, 9] which states that $TL(\mathsf{Until}, \mathsf{Since})$, the temporal logic having Until and Since as the only modalities, is expressively complete for first-order monadic logic of order over the class of Dedekind-complete chains.

Stavi introduced two modalities Until_{Stavi} and Since_{Stavi} and proved that the temporal logic having the four modalities Until, Since, Until_{Stavi} and Since_{Stavi} is expressively complete for first-order monadic logic of order over the class of all chains [9].

Reynolds [17] considered the complexity of the satisfiability problem for $TL(\mathsf{Until}, \mathsf{Since})$ over the reals and proved the following theorem.

Theorem 13.1. *The satisfiability problem for* $TL(\mathsf{Until}, \mathsf{Since})$ *over the reals is in* PSPACE.

This is an instance of Theorem 12.8(3). Due to Kamp's theorem, Reynolds' theorem implies that the satisfiability problem over the reals for any temporal logic with a finite set of first-order definable modalities is in PSPACE. Reynolds' proof relies on particular properties of Until and Since and uses temporal mosaics. The proof in [17] is very non-trivial and difficult to grasp, probably because it has been developed from scratch.

We do not fully understand the details of Reynolds' proof; however, there are some elements which are similar to our proof of Theorem 12.8(3). He considers operations on mosaics which correspond to sum, multiplication by ω and by ω^{-1} and shuffle of chains. He decides whether a finite set of small pieces is sufficient to be used to build a real-number model of a given formula. This is also equivalent to the existence of a winning strategy for player one in a two-player game played with mosaics. The search for a winning strategy is arranged into a search through a tree of mosaics. By establishing limits on the depth of the tree (polynomial in terms of the length of the formula) he constructs a PSPACE algorithm. There is an analogy between such mosaic trees and construction trees for chains of finite rank.

Recently, Reynolds [18] proved the following instance of Theorem 7.3.

Theorem 13.2. *The satisfiability problem for the temporal logic having the four modalities* Until*,* Since*,* Until_{Stavi} *and* Since_{Stavi} *over the class of chains is in* PSPACE*.*

He also established a PSPACE upper bound for the satisfiability problem over several interesting classes including dense chains, discrete chains, finite chains, $\mathbb{Q}$, $\mathbb{N}$ and $\mathbb{Z}$ [18]. All these proofs provide a reduction to Theorem 13.1.

An advantage of this approach is that many results are reduced to the satisfiability problem over the reals. A disadvantage is that a direct proof of a PSPACE upper bound for these classes is simpler than his proof of a PSPACE bound for the reals.

In [18], the correctness proof of the reductions proceeds by a quite lengthy case analysis.

In the rest of this section we simplify these reductions. First, we provide a simple reduction of Theorem 7.3 (and hence of Theorem 13.2) to Theorem 13.1. This proof is based on general arguments and the only result of our paper used here is Proposition 3.1. We need the following generalization of Definition 12.1.

Definition 13.3. *Let $\psi(X)$ be a formula and let $\mathcal{M}$ be a structure. The set of structures definable by ψ in $\mathcal{M}$ is the set of substructures of $\mathcal{M}$ over the non-empty subsets that satisfy ψ. This set is denoted by $\mathcal{M}^\psi$. For a set $\mathcal{C}$ of structures we denote by $\mathcal{C}^\psi$ the set $\cup_{\mathcal{M}\in\mathcal{C}}\mathcal{M}^\psi$.*

Note that if $\mathcal{C}'$ is definable by an $\exists$-MSO sentence φ relatively to a class $\mathcal{C}$, then $\mathcal{C}' = \mathcal{C}^\psi$ for an $\exists$-MSO formula $\psi(X) := \varphi \wedge \forall t X(t)$.

Lemma 13.4. *Let $\psi(X)$ be an $\exists$-MSO formula and $\mathcal{C}$ a set of structures. For every finite set Φ of first-order formulas there is a finite set Φ' of first-order formulas and a linear time algorithm which reduces the satisfiability problem for Φ-conjunctive formulas over $\mathcal{C}^\psi$ to the satisfiability problem for Φ'-conjunctive formulas over $\mathcal{C}$.*

PROOF. Let φ be a formula and X be a monadic variable that does not appear in φ. We denote by φ^X the formula obtained from φ by relativizing all first-order quantifiers over X. Note that the free variables of φ^X are the free variables of φ and X. For every structure $\mathcal{M}$ and a non-empty subset P of its domain: φ holds in the substructure of $\mathcal{M}$ over P iff φ^X holds in $\mathcal{M}$ when X is interpreted as P. If φ_1 is an instance of φ, then φ_1^X is an instance of φ^X.

Therefore, a Φ-conjunctive formula $\varphi_1 \wedge \cdots \wedge \varphi_k$ is satisfiable in $\mathcal{C}^\psi$ iff $\psi(X) \wedge \varphi_1^X \wedge \cdots \wedge \varphi_k^X$ is satisfiable in $\mathcal{C}$. Hence, if ψ is $\exists \overline{Y} \chi$, where χ is a first-order formula, then Φ' defined as $\{\chi\} \cup \{\varphi^X \mid \varphi \in \Phi\}$ satisfies the conclusion of the Lemma. □

Lemma 13.5. *Let TL be a temporal logic which is expressively complete for FOMLO over a class $\mathcal{C}$. For every finite set Φ of first-order formulas there is a linear time algorithm which reduces the satisfiability problem for Φ-conjunctive formulas over $\mathcal{C}$ to the satisfiability problem for TL over $\mathcal{C}$.*

Proof. By the expressive completeness of TL we know that for every formula $\varphi(\overline{X}) \in \Phi$ there is a TL formula $\widehat{\varphi}(\overline{X})$ which is equivalent to φ over $\mathcal{C}$. Hence, a Φ-conjunctive formula $\varphi_1(\overline{P_1}) \wedge \cdots \wedge \varphi_k(\overline{P_k})$ is satisfiable in $\mathcal{C}$ iff a TL formula $\widehat{\varphi}_1(\overline{P_1}) \wedge \cdots \wedge \widehat{\varphi}_k(\overline{P_k})$ is satisfiable in $\mathcal{C}$. Note that for every finite Φ we can pre-compute the set $\{\widehat{\varphi} \mid \varphi \in \Phi\}$. Hence, this is a linear time reduction. □

As an immediate consequence of Proposition 3.1, Lemmas 13.4 and 13.5 we obtain the following corollary.

Corollary 13.6. *Let $\psi(X)$ be an $\exists$-MSO formula and TL be a temporal logic which is expressively complete for FOMLO over a class $\mathcal{C}$, and let TL' be a temporal logic with a finite set of $\exists$-MSO definable modalities. There is a linear time algorithm which reduces the satisfiability problem for TL' over $\mathcal{C}^{\psi}$ to the satisfiability problem for TL over $\mathcal{C}$.*

Now, we can derive Theorem 7.3 from the PSPACE upper bound of the satisfiability of $TL(\mathsf{Until}, \mathsf{Since})$ over $\mathbb{R}$ as follows. An $\exists$-MSO formula is satisfiable iff it is satisfiable over a countable linear order iff it is satisfiable over a sub-order of $\mathbb{Q}$ and hence iff it is satisfiable over a sub-order of $\mathbb{R}$ iff it is satisfiable in $\mathbb{R}^{\psi}$, where $\psi(X)$ is a formula equivalent to true.

By Kamp's theorem $TL(\mathsf{Until}, \mathsf{Since})$ is expressively complete over $\mathbb{R}$. Therefore, by Corollary 13.6 and Theorem 13.1 we obtain that the satisfiability problem for any temporal logic with a finite set of $\exists$-MSO definable modalities. over the class of chains is in PSPACE.

Next, let us derive that the satisfiability problems over the dense orders, discrete orders, finite orders, $\mathbb{Q}$, $\mathbb{N}$ and $\mathbb{Z}$ are in PSPACE. This bound was also proved in [18]. These results can be easily derived from Theorems 13.1 and 13.2 by Corollaries 12.2 and 13.6. Dense orders are first-order definable, discrete orders are the orders such that every element has the next and the previous element and this class is first-order definable. $\mathbb{Q}$ is first-order definable relatively to the class of countable linear orders. There is a first-order formula $\mathrm{Nat}(X)$ (respectively, $\mathrm{Integer}(X)$ and $\mathrm{Fin}(X)$) such that a subset of $\mathbb{R}$ satisfies it iff it is order-isomorphic to ω (respectively, to $\mathbb{Z}$ and finite order).

The PSPACE upper bound for the satisfiability problem over the ordinals was proved in [7] using automata theoretical techniques. However, one can also apply the above arguments to derive this from Reynolds' PSPACE upper bound for the reals. Indeed, for every countable ordinal there is a sub-chain of $\mathbb{R}$ which is order-isomorphic to it, and it is easy to write a first-order formula $\mathrm{Ord}(X)$ such that a subset P of $\mathbb{R}$ satisfies it iff P is order-isomorphic to a countable ordinal. $\mathrm{Ord}(X)$ says that "X has a minimal element" and "for every $r \in \mathbb{R}$ if there is an element of X greater than r, then there is a minimal such element."

We do not know whether there is an $\exists$-MSO formula $\psi(X)$ such that $\mathbb{R}^{\psi}$ is the set of countable scattered orders.

14. Conclusion and Further Results

We provided an EXPTIME algorithm for the satisfiability problem for any temporal or modal logic with a finite set of $\exists$-MSO definable modalities over a recursively defined class of structures, and proved that EXPTIME bound is optimal in the worst case. This algorithm works also in other frameworks. For example, the same algorithm works for the temporal logics with a finite set of MSO-definable modalities. However, if a recursive class is defined as the closure of $\mathcal{C}$ under $\mathfrak{F}$, then now we need to require that operators in $\mathfrak{F}$ respect $\equiv^n_{MSO}$, where structures are $\equiv^n_{MSO}$-equivalent if they are not distinguishable by the MSO sentences of quantifier depth $\leq n$. It is interesting to find a characterization of recursively definable classes when there is a polynomial $p(n)$ such that a conjunctive formula φ is satisfiable in $Cl(\mathcal{C}, \mathfrak{F})$ iff it is satisfiable in $Cl^{p(|\varphi|)}(\mathcal{C}, \mathfrak{F})$. For such classes the satisfiability problem can be solved in PSPACE.

We provided a PSPACE algorithm for the satisfiability problem for any temporal logic with a finite set of $\exists$-MSO definable modalities over the class of linear orders. We applied the same techniques in a "plug-and-play" manner to show that the problem is in PSPACE over many interesting classes of linear orders.

Theorem 11.2 provides a PSPACE algorithm for the satisfiability problem for any temporal logic with a finite set of automata definable modalities over the class of countable linear orders. As explained in Remark 11.10, this theorem can be extended to the class of all linear orders.

The constants hidden in the complexity analysis of our PSPACE algorithm are huge. Indeed to check the satisfiability of a formula φ in a temporal logic TL with a set of modalities B, we first translated the (generalized) truth tables for every modality m in B to an equivalent finite base automaton $\mathfrak{A}_m$. Then we proved that φ is satisfiable iff it is satisfiable on a chain of rank $O(n_B|\varphi|)$, where n_B is an upper bound on the base size of the automata in $\{\mathfrak{A}_m : m \in B\}$. For every temporal logic $TL(B)$ this constant n_B is fixed; however n_B cannot be bounded by an elementary function in the size of the $\exists$-MSO truth tables for the modalities of $TL(B)$. It is interesting to find more practical PSPACE algorithms.

Acknowledgments

We are grateful to Moshe Vardi for pointing out reference [18]. We thank anonymous referees for their insightful suggestions.

References

[1] N. Bedon, A. Bes, O. Carton, C. Rispal: Logic and Rational Languages of Words Indexed by Linear Orderings. CSR 2008, LNCS 5010, pp. 76-85, 2008.
[2] V. Bruyère and O. Carton. Automata on linear orderings. MFCS 2001, LNCS 2136, pp. 236-247, 2001.
[3] J. R. Büchi. On a decision method in restricted second order arithmetic. In: Logic, Methodology and Philosophy of Science, Stanford University Press, (1962), 1-11.
[4] J. R. Büchi and D. Siefkes, The Monadic Second-order Theory of all Countable Ordinals, Springer Lecture Notes 328, 1973.
[5] J. P. Burgess and Y. Gurevich. The decision problem for linear temporal logic. Notre Dame J. Formal Logic, 26(2):115-128, 1985.
[6] J. Cristau. Automata and temporal logic over arbitrary linear time. FSTTCS 2009, LIPIcs 4, pp. 133-144, 2009.
[7] S. Demri and A. Rabinovich. The Complexity of Temporal Logic with Until and Since over Ordinals. LPAR 2007: 531-545.
[8] S. Feferman and R. L. Vaught. The first-order properties of products of algebraic systems. *Fundamenta Mathematicae* 47:57–103, 1959.
[9] D. M. Gabbay, I. Hodkinson, M. Reynolds. Temporal Logics volume 1. Clarendon Press, Oxford 1994.
[10] D. M. Gabbay, A. Pnueli, S. Shelah, J. Stavi. On the Temporal Analysis of Fairness. 7th POPL, pp. 163-173, 1980.
[11] Y. Gurevich. Monadic second-order theories. In *Model-Theoretic Logics*, (J. Barwise and S. Feferman, eds.), 479-506, Springer-Verlag, 1985.
[12] H. Kamp. Tense Logic and the Theory of Linear Order. Ph.D. thesis, University of California L.A., 1968.
[13] H. Läuchli and J. Leonard. On the elementary theory of linear order. Fundamenta Mathematicae, 59:109-116, 1966.
[14] M. O. Rabin. Decidability of second-order theories and automata on infinite trees. Transactions of the American Mathematical Society, vol. 141:1-35, 1969.
[15] A. Rabinovich. Temporal logics over linear time domains are in PSPACE. LNCS 6227, 29-50, 2010.
[16] M. Reynolds. The complexity of the temporal logic with until over general linear time. J. Comput. Syst. Sci., 66:393-426, 2003.
[17] M. Reynolds. The Complexity of Temporal Logic over the Reals. Annals of Pure and Applied Logic, 161(8): 1063-1096, 2010.
[18] M. Reynolds. The Complexity of Decision Problems for Linear Temporal Logics. Journal of Studies in Logic, 3(1) (2010):19-50, 2010.
[19] C. Rispal, O. Carton. Complementation of rational sets on countable scattered linear orderings. Int. J. Found. Comput. Sci. 16(4): 767-786, 2005.
[20] J. G. Rosenstein. *Linear ordering.* Academic Press, New York, 1982.
[21] S. Shelah. The monadic theory of order. *Ann. of Math.*, **102**, pp 349-419, 1975.
[22] A. P. Sistla and E. M. Clarke. The Complexity of Propositional Linear Temporal Logics J. ACM 32(3): 733-749, 1985.
[23] W. Thomas. Ehrenfeucht games, the composition method, and the monadic theory of ordinal words. In *Structures in Logic and Computer Science: A Selection of Essays in Honor of A. Ehrenfeucht*, LNCS 1261, pp. 118-143, 1997.

Appendix A. Finite state automata over linear orders

Cristau [6] proved that every formula of the first-order fragment of the monadic logic is equivalent (over the class of all linear orders) to a *finite state automaton.* Throughout our paper we used finite base automata instead of finite state automata. In this appendix we recall the definition of finite state automata over linear orders [2] and prove their equivalence to finite base automata.

The following remark explains why we used finite base automata.

Remark (On advantages of finite base over finite state automata) *If $\mathfrak{A}_i$ $(i = 1, \ldots, n)$ are finite state automata, then the number of states in the finite state automaton which accepts the intersection of the languages definable by $\mathfrak{A}_i$ is the product of the numbers of states in $\mathfrak{A}_i$ and is exponential in n. An advantage of finite base automata over finite state automata is that taking the intersection of the former is easy. The number of states in the finite base automaton which accepts the intersection of the languages accepted by $\mathfrak{A}_i$ is also exponential in n; however, the base of the automaton for the intersection grows linearly in n.*

A finite state automaton $\mathfrak{A}$ is a tuple of the form $(Q, \Sigma, \delta_{next}, \delta_{\lim}, Q_{init}, Q_{fin})$ where

- Q is a finite set of states, $Q_{init}, Q_{fin} \subseteq Q$ are sets of initial and final states,
- Σ is finite alphabet,
- $\delta \subseteq Q \times \Sigma \times Q$ is the next-step transition relation,
- $\delta_{\lim} \subseteq \mathcal{P}(Q) \times Q \cup Q \times \mathcal{P}(Q)$ is the limit transition relation.

The only difference between the finite base automata and the finite state automata are (1) the set of states of a finite base automata are subset of the base and (2) the type of the limit transition relation.

We write

- $q \xrightarrow{a} q'$ if $(q, a, q') \in \delta$ (successor transition),
- $P \to q$ if $(P, q) \in \delta_{\lim}$ (left limit transition)
- $q \leftarrow P$ if $(q, P) \in \delta_{\lim}$ (right limit transition)

Let $\mathcal{I} = (I, <)$ be a linear order, V a set (of labels), and $f : I \to V$ be a function. We define the left and right limit sets of f at $c \in I$ as the sets of labels that appear arbitrarily close to c (respectively to its left and to its right). Formally:

$$\overrightarrow{\lim}(f, c) := \{v \in V \mid \forall a < c \exists b(a < b < c) \wedge v = f(b)\}$$

$$\overleftarrow{\lim}(f, c) := \{v \in V \mid \forall a > c \exists b(c < b < a) \wedge v = f(b)\}$$

Given an automaton $\mathfrak{A}$, a run of $\mathfrak{A}$ on Σ-word s over a linear order $\mathcal{I}$ is a function ρ from the set of cuts $\widehat{\mathcal{I}}$ of $\mathcal{I}$ into Q such that

- For each $c \in \mathcal{I}$, $c^- \xrightarrow{s(c)} c^+$
- if $c \in \widehat{\mathcal{I}} \setminus \widehat{\mathcal{I}}_{\min}$ has no predecessor, and $P = \overrightarrow{\lim}(\rho, c)$, then $P \to \rho(c)$;
- if $c \in \widehat{\mathcal{I}} \setminus \widehat{\mathcal{I}}_{\max}$ has no successor, and $P = \overleftarrow{\lim}(\rho, c)$, then $\rho(c) \leftarrow P$.

An $\mathfrak{A}$ run ρ is accepting if $\rho(\widehat{\mathcal{I}}_{\min}) \in Q_{init}$ and $\rho(\widehat{\mathcal{I}}_{\max}) \in Q_{fin}$. $\mathfrak{A}$ accepts a word s if there is an accepting run on s.

We are going to show that for every finite state automaton there is an equivalent finite base automaton.

Let $\mathfrak{A} := (Q, \Sigma, \delta_{next}, \delta_{\lim}, Q_{init}, Q_{fin})$ be a finite state automaton.

Let $B := \mathcal{P}(Q)$ be the set of subsets of Q. For every $q \in Q$ let $\widehat{q} \in \mathcal{P}(B)$ be defined as $\{b \in B \mid q \in b\}$.

Let $\widehat{Q} := \{\widehat{q} \mid q \in Q\}$. Define a finite base automaton $\mathfrak{B}$ as follows:

- The basis of $\mathfrak{B}$ is the set of subsets of Q,
- the set of states is $\widehat{Q}$, the set of initial (respectively final) states is $\{\widehat{q} \mid q \in Q_{init}\}$ (respectively, $\{\widehat{q} \mid q \in Q_{fin}\}$),

- the next transition relation is defined as: $\widehat{q_1} \xrightarrow{a} \widehat{q_2}$ if there is a transition $q_1 \xrightarrow{a} q_2$ in $\mathfrak{A}$.
- $D \to \widehat{q}$ if there is $P \subseteq Q$ such that $P \to q$ in $\mathfrak{A}$ and $D = \{b \in B \mid b \supseteq P\}$.
- $\widehat{q} \leftarrow D$ is defined dually.

It is easy to verify that $\mathfrak{A}$ and $\mathfrak{B}$ accept the same chains.

Now let us show that for every finite base automaton $\mathfrak{A} := (B, Q, \Sigma, q_i, \Delta_{next}, \delta_{\lim}, Q_{init}, Q_{fin})$ there is an equivalent finite state automaton $\mathfrak{B}$.

Define

- The set of states of $\mathfrak{B}$ is the same as the set of states of $\mathfrak{A}$. The initial (respectively, final) states of $\mathfrak{B}$ are the initial (respectively, final) states of $\mathfrak{A}$.
- The next transition relation of $\mathfrak{B}$ is the same as the next transition relation of $\mathfrak{A}$.
- $P \to q$ is a left limit transition of $\mathfrak{B}$ if $(\bigcap_{p \in P} p) \to q$ is a left limit transition of $\mathfrak{A}$.
- the right limit transitions are defined dually.

It is easy to verify that $\mathfrak{A}$ and $\mathfrak{B}$ accept the same chains.