Separability by piecewise testable languages and downward closures beyond subwords

Georg Zetzsche
IRIF (Université Paris-Diderot & CNRS)
France
zetzsche@irif.fr

Abstract

We introduce a flexible class of well-quasi-orderings (WQOs) on words that generalizes the ordering of (not necessarily contiguous) subwords. Each such WQO induces a class of piecewise testable languages (PTLs) as Boolean combinations of upward closed sets. In this way, a range of regular language classes arises as PTLs. Moreover, each of the WQOs guarantees regularity of all downward closed sets. We consider two problems. First, we study which (perhaps non-regular) language classes allow to decide whether two given languages are separable by a PTL with respect to a given WQO. Second, we want to effectively compute downward closures with respect to these WQOs. Our first main result is that for each of the WQOs, under mild assumptions, both problems reduce to the simultaneous unboundedness problem (SUP) and are thus solvable for many powerful system models. In the second main result, we apply the framework to show decidability of separability of regular languages by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$, a fragment of first-order logic with modular predicates.

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

Keywords separability, piecewise testable languages, downward closures, well-quasi-orderings

ACM Reference Format:
Georg Zetzsche. 2018. Separability by piecewise testable languages and downward closures beyond subwords. In *LICS '18: 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, July 9–12, 2018, Oxford, United Kingdom.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3209108.3209201

1 Introduction

In the verification of infinite-state systems, it is often useful to construct finite-state abstractions. This is because finite-state systems are much more amenable to analysis. For example, if a pertinent property of our system is reflected in a finite-state abstraction, then we can work with the abstraction instead of the infinite-state system itself. Another example is that the abstraction acts as a certificate for correctness: A violation-free overapproximation of the set of behaviors of a system certifies absence of violations in the system itself. Here, we study two types of such abstractions: *downward closures*, which are overapproximations of individual languages and *separators* as certificates of disjointness.

Downward closures A particularly appealing abstraction is the *downward closure*, the set of all (not necessarily contiguous) subwords of the members of a language. What makes this abstraction interesting is that since the subword ordering is a well-quasi-ordering (WQO), the downward closure of *any* language is regular [16, 17]. Recently, there has been progress on when the downward closure is not only regular but can also be effectively computed. It is known that downward closures are computable for *context-free languages* [7, 28], *Petri net languages* [14], and *stacked counter automata* [30]. Moreover, recently, a general sufficient condition for computability was presented in [29]. Using the latter, downward closures were then shown to be computable for *higher-order pushdown automata* [15] and *higher-order recursion schemes* [6]. Hence, downward closures are computable for very powerful models.

If we want to use downward closures to prove absence of violations, then using the downward closure in this way has the disadvantage that it is not obvious how to refine it, i.e. systematically construct a more precise overapproximation in case the current one does not certify absence of violations. Therefore, we wish to find abstractions that are refinable in a flexible way and still guarantee regularity and computability.

Separability Another type of finite-state abstractions is that of separators. Since safety properties of multi-threaded programs can often be formulated as the disjointness of two languages, one approach to this task is to use regular languages to certify disjointness [2, 4, 22]. A *separator* of two languages K and L is a set S such that $K \subseteq S$ and $L \cap S = \emptyset$. Therefore, especially in cases where disjointness of languages is undecidable or hard, it would be useful to have a decision procedure for the *separability problem*: Given two languages, it asks whether they are separable by a language from a particular class of separators. In particular, if we want to apply such algorithms to infinite-state systems, it would be desirable to find large classes of separators (and systems) for which the separability problem is decidable.

It has long been known that separability of context-free languages is undecidable already for very simple classes of regular languages [18, 27] and this stifled hope that separability would be decidable for any interesting classes of infinite-state systems and classes of separators. However, the subword ordering turned out again to have excellent decidability properties: It was shown recently that for a wide range of language classes, it is decidable whether two given languages are separable by a piecewise testable language (PTL) [9]. A PTL is a finite Boolean combination of upward closures (with respect to the subword ordering) of single words. In fact, in turned out that (under mild closure assumptions)

LICS '18, July 9–12, 2018, Oxford, United Kingdom

ACM ISBN 978-1-4503-5583-4/18/07…$15.00
https://doi.org/10.1145/3209108.3209201

separability by PTL is decidable if and only if downward closures are computable [10].

However, while this separability result applies to very expressive models of infinite-state systems, it is still limited in terms of the separators: The small class of PTL will not always suffice as disjointness certificates.

Contribution This work makes two contributions, a conceptual one and a technical one. The conceptual contribution is the introduction of a fairly flexible class of WQOs on words. These are refinable and provide generalizations of the subword ordering. These orders are parameterized by transducers, counter automata or other objects and can be chosen to reflect various properties of words. Moreover, the classes of corresponding PTLs are a surprisingly rich collection of classes of regular languages.

Furthermore, it is shown that all these orders have the same pleasant properties in terms of downward closure computation and decidability of PTL-separability as the subword ordering. Specifically, it is shown that (under mild assumptions) decidability of the abovementioned unboundedness problem again characterizes (i) those language classes for which downward closures are computable and (ii) those classes where separability by PTL is decidable.

In addition, it turns out that this framework can also be used to obtain decidable separability of regular languages by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$, a fragment of first-order logic with modular predicates. This is technically relatively involved and generalizes the fact that definability of regular languages in $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$ is decidable [5].

2 Preliminaries

If Σ is an alphabet, Σ^* denotes the set of words over Σ. The empty word is denoted $\varepsilon \in \Sigma^*$. A *quasi-ordering* is an ordering that is reflexive and transitive. An ordering $(X, \leq)$ is called a *well-quasi-ordering (WQO)* if for every sequence $x_1, x_2, \ldots \in X$, there are indices $i < j$ with $x_i \leq x_j$. This is equivalent to requiring that every sequence $x_1, x_2, \ldots \in X$ contains an infinite subsequence $x'_1, x'_2, \ldots \in X$ that is *ascending*, meaning $x'_i \leq x'_j$ for $i \leq j$. For a subset $L \subseteq X$, we define $\downarrow_{\leq} L = \{x \in X \mid \exists y \in L \colon x \leq y\}$ and $\uparrow_{\leq} L = \{x \in X \mid \exists y \in L \colon y \leq x\}$. These are called the *downward closure* and *upward closure* of L, respectively. If the ordering $\leq$ is clear from the context, we sometimes just write $\downarrow L$ or $\uparrow L$. A set $L \subseteq X$ is called *downward closed* (resp. *upward closed*) if $\downarrow_{\leq} L = L$ (resp. $\uparrow_{\leq} L = L$). A (defining) property of well-quasi-orderings is that for every non-empty upward-closed set U, there are finitely many elements $x_1, \ldots, x_n \in U$ such that $U = \uparrow_{\leq}\{x_1, \ldots, x_n\}$. See [20] for an introduction. An ordering $(\Sigma^*, \leq)$ on words is called *multiplicative* if $u_1 \leq v_1$ and $u_2 \leq v_2$ implies $u_1u_2 \leq v_1v_2$. For $u, v \in \Sigma^*$, we write $u \preccurlyeq v$ if $u = u_1 \cdots u_n$ and $v = v_0u_1v_1 \cdots u_nv_n$ for some $u_1, \ldots, u_n, v_0, \ldots, v_n \in \Sigma^*$. This ordering is called the *subword ordering* and it is well-known to be a WQO [17].

A well-studied class of regular languages is that of the piecewise testable languages. Classically, a language $L \subseteq \Sigma^*$ is a *piecewise testable language (PTL)* [25] if it is a finite Boolean combination of sets of the form $\uparrow_{\preccurlyeq} w$ for $w \in \Sigma^*$. However, this notion makes sense for any WQO $(X, \leq)$ [13] and we call a set $L \subseteq X$ *piecewise testable* if it is a finite Boolean combination of sets $\uparrow_{\leq} x$ for $x \in X$.

3 Parameterized WQOs and examples

In this section, we introduce the parameterized WQOs on words, state the main results of this work, and present some applications. We define the class of parameterized WQOs inductively using rules (Rules 1 to 3). The simplest example is Higman's subword ordering.

Rule 1. *For each Σ, $(\Sigma^*, \preccurlyeq)$ is a parameterized WQO.*

Orderings defined by transducers To make things more interesting, we have a type of WQOs that are defined by functions. Suppose X and Y are sets and we have a function $f \colon X \to Y$. A general way of constructing a WQO on X is to take a WQO $(Y, \leq)$ and set $x \leq_f x'$ if and only if $f(x) \leq f(x')$. It is immediate from the definition that then $\leq_f$ is a WQO on X. We apply this idea to transducers.

A *finite-state transducer* over Σ and Γ is a tuple $\mathcal{T} = (Q, \Sigma, \Gamma, E, I, F)$, where Q is a finite set of states, $E \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times (\Gamma \cup \{\varepsilon\}) \times Q$ is its set of *edges*, $I \subseteq Q$ is the set of *initial states*, and $F \subseteq Q$ is the set of *final states*. Transducers accept sets of pairs of words. A *run* of $\mathcal{T}$ is a sequence $(q_0, u_1, v_1, q_1)(q_1, u_2, v_2, q_2) \cdots (q_{n-1}, u_n, v_n, q_n)$ of edges such that $q_0 \in I$, $q_n \in F$. The pair *read by the run* is $(u_1 \cdots u_n, v_1 \cdots v_n)$. Then, $\mathcal{T}$ *realizes* the relation

$$T(\mathcal{T}) = \{(u, v) \in \Sigma^* \times \Gamma^* \mid (u, v) \text{ is read by a run of } \mathcal{T}\}.$$

Relations of this form are called *rational transductions*. A transduction is *functional* if for every $u \in \Sigma^*$, there is *exactly* one $v \in \Gamma^*$ with $(u, v) \in T(\mathcal{T})$. In other words, $T(\mathcal{T})$ is a function $T(\mathcal{T}) \colon \Sigma^* \to \Gamma^*$ and we can use it to define a WQO.

Rule 2. *Let $f \colon \Sigma^* \to \Gamma^*$ be a functional rational transduction. If $(\Gamma^*, \leq)$ is a parameterized WQO, then so is $(\Sigma^*, \leq_f)$.*

Conjunctions Another way to build a WQO on a set is to combine two existing WQOs. Suppose $(X, \leq_1)$ and $(X, \leq_2)$ are WQOs. Their *conjunction* is the ordering $(X, \leq)$ with $x \leq x'$ if and only if $x \leq_1 x'$ and $x \leq_2 x'$. Then $(X, \leq)$ is a WQO via the characterization using ascending subsequences.

Rule 3. *If $(\Sigma^*, \leq_1)$ and $(\Sigma^*, \leq_2)$ are parameterized WQOs, then so is their conjunction $(\Sigma^*, \leq)$.*

Using the three building blocks in Rules 1 to 3, we can construct a wealth of WQOs on words. Let us mention a few examples, including the accompanying classes of PTL.

Labeling transductions Our first class of examples concerns orderings whose PTLs are fragments of first-order logic with additional predicates. A *labeling transduction* is a functional transduction $f \colon \Sigma^* \to (\Sigma \times \Lambda)^*$ for some alphabet Λ labels such that for each $w = a_1 \cdots a_n \in \Sigma^*$, $a_1, \ldots, a_n \in \Sigma$, we have $f(w) = (a_1, \ell_1) \cdots (a_n, \ell_n)$ for some $\ell_1, \ldots, \ell_n \in \Lambda$.

In this case, we can interpret $\preccurlyeq_f$-PTLs logically. To each word $w = a_1 \cdots a_n$, $a_1, \ldots, a_n \in \Sigma$, we associate a finite relational structure $\mathfrak{M}_{f,w}$ as follows. Its domain is $D = \{1, \ldots, n\}$ and as predicates, it has the binary $<$, unary letter predicates P_a for $a \in \Sigma$, and for each $\ell \in \Lambda$, we have a unary predicate π_ℓ. While the predicates $<$ and P_a are interpreted as expected, we have to explain π_ℓ. If $f(w) = (a_1, \ell_1) \cdots (a_n, \ell_n)$, then $\pi_\ell(i)$ expresses that $\ell_i = \ell$. Hence, the π_ℓ give access to the labels produced by f. We denote the $\mathcal{B}\Sigma_1$-fragment (Boolean combinations of Σ_1-formulas) as $\mathcal{B}\Sigma_1[<, f]$.

Suppose $\mathfrak{M}_1$ and $\mathfrak{M}_2$ are relational structures over the same signature. An *embedding* of $\mathfrak{M}_1$ in $\mathfrak{M}_2$ is an injective mapping from the domain of $\mathfrak{M}_1$ to the domain of $\mathfrak{M}_2$ such that each predicate holds for a tuple in $\mathfrak{M}_1$ if and only the predicate holds for the image of that tuple. This defines a quasi-ordering: We write $\mathfrak{M}_1 \hookrightarrow \mathfrak{M}_2$ if $\mathfrak{M}_1$ can be embedded into $\mathfrak{M}_2$. Observe that for $u, v \in \Sigma^*$, we have $u \preccurlyeq_f v$ if and only if $\mathfrak{M}_{f,u} \hookrightarrow \mathfrak{M}_{f,v}$.

It was observed by Goubault-Larrecq and Schmitz [13] that if the embedding order is a WQO on a set of structures, then the $\mathcal{B}\Sigma_1$-fragment (i.e. Boolean combinations of Σ_1 formulas) can express precisely the PTL with respect to $\hookrightarrow$. This implies that the languages definable in $\mathcal{B}\Sigma_1[<, f]$ are precisely the $\preccurlyeq_f$-PTL.

To illustrate the utility of the fragments $\mathcal{B}\Sigma_1[<, f]$, suppose we are given regular languages W_i, P_i, S_i, for $i \in [1, n]$. Suppose we have for each $i \in [1, n]$ a 0-ary predicate w_i that expresses that our whole word belongs to W_i. For each $i \in [1, n]$ we also have unary predicates pre_i and suf_i, which express that the prefix and suffix, respectively, corresponding to the current position, belongs to P_i and S_i, respectively. Then the corresponding fragment

$$\mathcal{B}\Sigma_1[<, (\mathsf{w}_i)_{i\in[1,n]}, (\mathsf{pre}_i)_{i\in[1,n]}, (\mathsf{suf}_i)_{i\in[1,n]}]$$

can clearly be realized as $\mathcal{B}\Sigma_1[<, f]$.

Of course, we can capture many other predicates by labeling transducers. For example, it is easy to realize a predicate for "the distance to the closest position to the left with an a is congruent k modulo d" (for some fixed d).

Finally, let us observe in passing that instead of enriching $\mathcal{B}\Sigma_1[<]$, we could also construct fragments that do not have access to letters. Suppose we perform the construction of $\mathfrak{M}_{f,w}$ for length-preserving transductions f that only produce labels and do not reproduce the output, meaning $f : \Sigma^* \to \Lambda^*$. Then, one can choose f so that $\preccurlyeq_f$-PTLs correspond to a logic where, for example, we can only express whether "this position is even *and* carries an a".

Orderings defined by finite automata Our second example slightly specializes the first example. The reason we make it explicit is that we shall present explicit ideal representations that will be applied to decide separability of regular languages by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$. The example still generalizes the subword order. While in the latter, a smaller word is obtained by deleting arbitrary infixes, these orders use an automaton to restrict the permitted deletions.

A *finite automaton* is a tuple $\mathcal{A} = (Q, \Sigma, E, I, F)$, where Q is a finite set of *states*, Σ is the *input alphabet*, $E \subseteq Q \times \Sigma \times Q$ is the set of *edges*, $I \subseteq Q$ is the set of *initial states*, and $F \subseteq Q$ is the set of *final states*. The language $L(\mathcal{A})$ is defined in the usual way. Here, we use automata as a means to assign a labeling to an input word. A labeling is defined by a run. A *run* of $\mathcal{A}$ on $w = a_1 \cdots a_n$, $a_1, \ldots, a_n \in \Sigma$, is a sequence

$$(q_0, a_1, q_1)(q_1, a_2, q_2) \cdots (q_{n-1}, a_n, q_n) \in E^*$$

with $q_0 \in I$ and $q_n \in F$. By $\mathsf{Runs}(\mathcal{A})$, denote the set of runs of $\mathcal{A}$. Since we want $\mathcal{A}$ to label every word from Σ^*, we call an automaton $\mathcal{A}$ a *labeling automaton* if for each word $w \in \Sigma^*$, $\mathcal{A}$ has exactly one run on w. In this case, we write $\mathcal{A}(w)$ for the run of $\mathcal{A}$ on w. Moreover, we define $\sigma_{\mathcal{A}}(w) = (p, q)$, where p and q are the first and last state, respectively, visited during w's run. Hence, such an automaton defines a map $\mathcal{A} : \Sigma^* \to E^*$.

Let $u \leq_{\mathcal{A}} v$ if and only if v is obtained from u by "inserting loops of $\mathcal{A}$". In other words, v can be written as $v = u_0 v_1 u_1 \cdots v_n u_n$ with $u = u_0 \cdots u_n$ such that the run of $\mathcal{A}$ on v occupies the same state before reading v_i and after reading v_i. Equivalently, we have $u \leq_{\mathcal{A}} v$ if and only if $\sigma_{\mathcal{A}}(u) = \sigma_{\mathcal{A}}(v)$ and $\mathcal{A}(u) \preccurlyeq \mathcal{A}(v)$. The ordering $\leq_{\mathcal{A}}$ is a parameterized WQO: The ordering $\leq$ with $u \leq v$ if and only if $\sigma_{\mathcal{A}}(u) = \sigma_{\mathcal{A}}(v)$ is parameterized because we can use a functional transduction f that maps u to the length-1 word $\sigma_{\mathcal{A}}(u)$ in $(Q \times Q)^*$. Moreover, with with a functional transduction g that maps a word w to its run $\mathcal{A}(w)$, the ordering $\leq_{\mathcal{A}}$ is the conjunction of $\preccurlyeq_f$ and $\preccurlyeq_g$.

- If $\mathcal{A}$ consists of just one state and a loop for every $a \in \Sigma$, then $\leq_{\mathcal{A}}$ is the ordinary subword ordering.
- Suppose $\mathcal{B}$ is a complete deterministic automaton accepting a regular language $L \subseteq \Sigma^*$. Then L is simultaneously upward closed and downward closed with respect to $\leq_{\mathcal{A}}$, where $\mathcal{A}$ is obtained from $\mathcal{B}$ by making all states final. In particular, every regular language can occur as an upward closure and as a downward closure with respect to some $\leq_{\mathcal{A}}$.

As for labeling transducers, we can consider logical fragments where $\leq_{\mathcal{A}}$ is the embedding order. Again, our signature consists of $<$, P_a for $a \in \Sigma$. Furthermore, for each $q \in Q$, we have the 0-ary predicates ι_q and τ_q and unary predicates λ_q and ρ_q. Let $(q_0, a_1, q_1) \cdots (q_{n-1}, a_n, q_n)$ be the run of $\mathcal{A}$ on w. Then $\lambda_q(i)$ is true iff $q_{i-1} = q$. Moreover, $\rho_q(i)$ holds iff $q_i = q$. Hence, λ_q and ρ_q give access to the state occupied by $\mathcal{A}$ to the left and to the right of each position, respectively. Accordingly, ι_q and τ_q concern the first and the last state: ι_q is satisfied iff $q_0 = q$ and τ_q is true iff $q_n = q$.

As an example, let $\mathcal{M}_d$ be the automaton that consists of a single cycle of length d so that on each input letter, $\mathcal{M}_d$ moves one step forward in the cycle. This is equivalent to having a predicate for each $k \in [1, d]$ that express that the current position is congruent k modulo d. Moreover, we have a predicate for each $k \in [1, d]$ to express that the length of the word is k modulo d. This is sometimes denoted $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$. If these predicates are available for *every* d, the resulting class is denoted $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$ [5] and will be the subject of Theorem 4.5.

Regularity and multiplicative well-partial-orderings Ehrenfeucht et al. [11] have characterized the regular languages as those sets of words that are upward upward closed with respect to some multiplicative WQO. To show necessity, they provide the syntactic congruence, which, as a finite-index equivalence, is a WQO. Since finite-index equivalences are somewhat pathological examples of WQOs, this raises the question of whether the same holds for well-*partial*-orders, i.e. WQOs that are also antisymmetric. It does, and we exhibit a natural way to construct such well-partial-orders. Suppose M is a finite monoid and $\theta : \Sigma^* \to M$ is a morphism that recognizes the language $L \subseteq \Sigma^*$, i.e. $L = \theta^{-1}(\theta(L))$. Let $f : \Sigma^* \to (M^2 \times \Sigma \times M^2)^*$ be the functional transduction such that for $w = a_1 \cdots a_n$, $a_1, \ldots, a_n \in \Sigma$:

$$f(w) = (\ell_0, r_0, a_1, \ell_1, r_1) \cdots (\ell_{n-1}, r_{n-1}, a_n, \ell_n, r_n),$$

where $\ell_i = \theta(a_1 \cdots a_i)$ and $r_i = \theta(a_{i+1} \cdots a_n)$. Then we have $u \preccurlyeq_f v$ if and only if v can be written as $v = u_0 v_1 u_1 \cdots v_n u_n$ such that $u = u_0 \cdots u_n$ and $\theta(u_0 \cdots u_{i-1} v_i) = \theta(u_0 \cdots u_{i-1})$ and $\theta(v_i u_i \cdots u_n) = \theta(u_i \cdots u_n)$ for every $i \in [1, n]$. In this case, we write $\leq_\theta$ for $\preccurlyeq_f$. Note that $\leq_\theta$ is multiplicative and L is $\leq_\theta$-upward closed. Thus, the ordering $\leq_\theta$ is a natural example that shows: A language is regular if and only if it is upward closed with respect to some multiplicative well-partial order.

Remark 3.1. Another source of WQOs on words a work of Bucher et. al. [3], in which they study a class of multiplicative orderings on words arising from rewriting systems. They show that all WQOs considered there can be represented by finite monoids equipped with a multiplicative quasi-order. For such a monoid $(M, \leq)$ and a morphism $\theta : \Sigma^* \to M$, they set $u \sqsubseteq_\theta v$ if and only if $u = u_1 \cdots u_n$, $u_1, \ldots, u_n \in \Sigma$, and $v = v_1 \cdots v_n$ such that $\theta(u_i) \leq \theta(v_i)$. However, they leave open for which monoids $(M, \leq)$ the ordering $\sqsubseteq_\theta$ is a WQO. In the case that θ above is a morphism into a finite group (whose order is the equality), the ordering $\leq_\theta$ coincides with $\sqsubseteq_\theta$. However, while the orderings considered by Bucher et al. are always multiplicative, this is not always the case for parameterized WQOs.

Orderings defined by counter automata In addition, we can use automata with counters to obtain parameterized WQOs. A *counter automaton* is a tuple $\mathcal{A} = (Q, \Sigma, C, E, I, F)$, where Q is a finite set of *states*, Σ is the *input alphabet*, C is a set of *counters*, $E \subseteq Q\times(A\cup\{\varepsilon\})\times\mathbb{N}^C\times Q$ is the finite set of *edges*, $I \subseteq Q$ is the set of *initial states*, and $F \subseteq Q$ is the set of *final states*. A *configuration of* $\mathcal{A}$ is a tuple (q, w, μ), where $q \in Q$, $w \in A^*$, $\mu \in \mathbb{N}^C$. The step relation is defined as follows. Let $(q, w, \mu) \to_{\mathcal{A}} (q', w', \mu')$ iff there is an edge $(q, v, \nu, q') \in E$ such that $w' = wv$ and $\mu' = \mu + \nu$. A *run (arriving at* μ*)* on an input word w is a sequence $(q_0, w_0, \mu_0), \ldots, (q_n, w_n, \mu_n)$ where $(q_{i-1}, w_{i-1}, \mu_{i-1}) \to_{\mathcal{A}} (q_i, w_i, \mu_i)$ for $i \in [1, n]$, $q_0 \in I$, $w_0 = \varepsilon$, $\mu_0 = 0$, $q_n \in F$, $w_n = w$, and $\mu_n = \mu$.

We use counter automata not primarily as accepting devices, but rather to define maps or to specify unboundedness properties. We call $\mathcal{A}$ a *counting automaton* if it has exactly one run for every word $w \in \Sigma^*$. In this case, it defines a function $\mathcal{A}\colon \Sigma^* \to \mathbb{N}^C$: We have $\mathcal{A}(w) = \mu$ iff $\mathcal{A}$ has a run on w arriving at μ.

This gives rise to an ordering: Let $\mathcal{A}$ be a counting automaton. Then, given $u, v \in \Sigma^*$, let $u \leq_{\mathcal{A}} v$ if and only if $\mathcal{A}(u) \leq \mathcal{A}(v)$. This is a parameterized WQO for the following reason. For each $c \in C$, we can build a functional transduction $f_c\colon \Sigma^* \to \{c\}^*$ that operates like $\mathcal{A}$, but instead of incrementing c, it outputs a c. Then, $\leq_{\mathcal{A}}$ is the conjunction of all the WQOs $\preccurlyeq_{f_c}$ for $c \in C$.

Let $k \in \mathbb{N}$ and $C_k = \{a_u, b_u, c_u \mid u \in \Sigma^{\leq k}\}$. We say that a word u *occurs at position* ℓ *in* v if $v = xuy$ with $|x| = \ell - 1$. It is easy to construct a counting automaton $\mathcal{P}_k$ with counter set C_k that satisfies $\mathcal{P}_k(w) = \mu$ iff for each $u \in \Sigma^{\leq k}$,

- if u is a prefix of w, then $\mu(a_u) = 1$, otherwise $\mu(a_u) = 0$,
- if u is a suffix of w, then $\mu(b_u) = 1$, otherwise $\mu(b_u) = 0$,
- $\mu(c_u)$ is the number of positions in w where u occurs.

Using this counting automaton, we can realize another class of regular languages. Let $k \in \mathbb{N}$. A *k-locally threshold testable language* is a finite Boolean combination of sets of the form

- $u\Sigma^*$ for some $u \in \Sigma^{\leq k}$,
- Σ^*u for some $u \in \Sigma^{\leq k}$, or
- $\{w \in \Sigma^* \mid u \text{ occurs at} \geq \ell \text{ positions in } w\}$ for some $u \in \Sigma^{\leq k}$ and $\ell \in \mathbb{N}$.

The class of k-locally threshold testable languages is denoted LTT_k. Observe that the $\leq_{\mathcal{P}_k}$-PTL are precisely the k-locally threshold testable languages. Indeed, each of the basic building blocks of k-locally threshold testable languages is $\leq_{\mathcal{P}_k}$-upward closed and hence a $\leq_{\mathcal{P}_k}$-PTL. Conversely, for each $w \in \Sigma^*$, the upward closure of w with respect to $\leq_{\mathcal{P}_k}$ is clearly in LTT_k.

Conjunctions Let us illustrate the utility of conjunctions. Let S be a finite collection of WQOs on Σ^*. We call a language $L \subseteq \Sigma^*$ an *S-PTL* if it is a finite Boolean combination of sets of the form $\uparrow_{\leq} w$, where $\leq$ belongs to S and $w \in \Sigma^*$. Our framework also applies to S-PTLs for the following reason.

Observation 3.2. *Let $\leq$ be the conjunction of the WQOs in S. Then a language is an S-PTL iff it is a $\leq$-PTL.*

Proof. Suppose S consists of the WQOs $\leq_i$ for $i \in [1, n]$. Every $\leq$-PTL is an S-PTL, because the set $\uparrow_{\leq}\{w\}$ can be written as the intersection $\bigcap_{i\in[1,n]} \uparrow_{\leq_i}\{w\}$. Conversely, an S-PTL is a Boolean combination of sets of the form $\uparrow_{\leq_i} w$ with $w \in \Sigma^*$. Clearly, $\uparrow_{\leq_i} w$ is upward closed also with respect to $\leq$ and can thus be written as $\uparrow_{\leq}\{w_1, \ldots, w_m\}$ for some $w_1, \ldots, w_m \in \Sigma^*$, which is a $\leq$-PTL. □

As an example, suppose we have subsets $\Sigma_1, \ldots, \Sigma_n \subseteq \Sigma$ and the functional transductions π_i, $i \in [1, n]$, such that $\pi_i\colon \Sigma^* \to \Sigma_i^*$ is the projection onto Σ_i, meaning $\pi_i(a) = a$ for $a \in \Sigma_i$ and $\pi_i(a) = \varepsilon$ for $a \notin \Sigma_i$. If S consists of the $\preccurlyeq_{\pi_i}$ for $i \in [1, n]$, then the S-PTL are precisely those languages that are Boolean combinations of sets $\uparrow_{\preccurlyeq} w$ for $w \in \Sigma_1^* \cup \cdots \cup \Sigma_i^*$. Hence, we obtain a subclass of the classical PTL. Of course, there are many other examples. One can, for example, combine WQOs for logical fragments with WQOs defined by counting automata and thus obtain logics that refer to positions as well as counter values, etc.

4 Main results

In this section, we present the main results of this work.

Computing downward closures The first problem we will study is that of computing downward closures. As in the case of the subword ordering, we will see that for all parameterized WQOs, every downward closed language is regular. While mere regularity is often easy to see, it is not obvious how, given a language $L \subseteq \Sigma^*$, to compute a finite automaton for $\downarrow_{\leq} L$. We are insterested in when this can be done algorithmically. If $\leq$ is a WQO on words, we say that *$\leq$-downward closures are computable* for a language class $\mathcal{C}$ if there is an algorithm that, given a language $L \subseteq \Sigma^*$ from $\mathcal{C}$, computes a finite automaton for $\downarrow_{\leq} L$. This is especially interesting when $\mathcal{C}$ is a class of languages of infinite-state systems.

Until now, downward closure computation has focused mainly on the case where $\leq$ is the subword ordering. In that case, there is a charaterization for when downward closures are computable [29]. For a rational transduction $T \subseteq \Sigma^* \times \Gamma^*$ and a language $L \subseteq \Sigma^*$, let $TL = \{v \in \Gamma^* \mid \exists u \in L\colon (u, v) \in T\}$. When we talk about *language classes*, we always assume that there is a way of representing their languages such as by automata or grammars. We call a language class $\mathcal{C}$ a *full trio* if it is effectively closed under rational transductions, i.e. given a representation of L from $\mathcal{C}$, we can compute a representation of TL in $\mathcal{C}$. The *simultaneous unboundedness problem (SUP)* for $\mathcal{C}$ is the following decision problem.

Given A language $L \subseteq a_1^* \cdots a_n^*$ from $\mathcal{C}$.
Question Does $a_1^* \cdots a_n^* \subseteq \downarrow_{\preccurlyeq} L$ hold?

The aforementioned characterization now states that downward closures for the subword ordering are computable for a full trio $\mathcal{C}$ if and only if the SUP is decidable. The SUP is decidable for many important and very powerful infinite-state systems. It is known to be decidable for Petri net languages [10, 14, 29] and matrix languages [29]. Moreover, it was shown to be decidable for second-order pushdown automata [29], which was generalized to higher-order pushdown automata [15] and then further to higher-order recursion schemes [6].

Separability by PTL We also consider separability problems. We say that two languages $K \subseteq \Sigma^*$ and $L \subseteq \Sigma^*$ are *separated* by a language $R \subseteq \Sigma^*$ if $K \subseteq R$ and $L \cap R = \emptyset$. If two languages are separated by a regular language, we can regard this regular language as a finite-state abstraction of the two languages. We therefore want to decide when two given languages can be separated by a language from some class of separators. More precisely, we say that for a language class C and a class of separators $\mathcal{S}$, *separability by $\mathcal{S}$ is decidable* if given language K and L from C, it is decidable whether there is an R in $\mathcal{S}$ that separates K and L. In the case where $\mathcal{S}$ is the class (subword) PTL, it is known when separability is decidable: In [10], it was shown that in a full trio, separability by PTL is decidable if and only if the SUP is decidable (the "if" direction had been obtained in [9]).

We are now ready to state the first main result.

Theorem 4.1. *For every full trio C, the following are equivalent:*

1. *The SUP is decidable for C.*
2. *For every parameterized WQO $\leq$, $\leq$-downward closures are computable for C.*
3. *For every parameterized WQO $\leq$, separability by $\leq$-PTL is decidable for C.*

This generalizes the two aforementioned results on downward closures and PTL separability. In addition, Theorem 4.1 applies to all the examples of $\leq$-PTL described above.

Recall that for each regular language R, there is a labeling automaton $\mathcal{A}$ such that R is $\leq_{\mathcal{A}}$-upward closed and thus a $\leq_{\mathcal{A}}$-PTL. Thus, for languages K and L, the following are equivalent: (i) There *exists* a labeling automaton $\mathcal{A}$ such that K and L are separable by a $\leq_{\mathcal{A}}$-PTL and (ii) K and L are separable by a regular language. Already for one-counter languages, separability by regular languages is undecidable [8] (for context-free languages, this was shown in [18, 27]). However, Theorem 4.1 tells us that for each fixed $\mathcal{A}$, separability by $\leq_{\mathcal{A}}$-PTL is decidable. We make a few applications explicit.

Corollary 4.2. *Let C be a full trio with decidable SUP. For each $d \in \mathbb{N}$, separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$ is decidable for C.*

Note that since a language $L \subseteq \Sigma^*$ is separable from its complement $\Sigma^* \setminus L$ by some $\leq$-PTL if and only if L is a $\leq$-PTL itself, Theorem 4.1 implies the following.

Corollary 4.3. *Let $\leq$ be a parameterized WQO. Given a regular language L, it is decidable whether L is an $\leq$-PTL.*

It was shown by Place et al. [24] that for context-free languages, separability by LTT_k is decidable for each $k \in \mathbb{N}$. Their algorithm uses semilinearity of context-free languages and Presburger arithmetic. Since models like Petri nets and higher-order pushdown automata do not have semilinear Parikh images, their proof method does not apply to them. Here, we extend this result to all full trios with a decidable SUP.

Corollary 4.4. *Let C be a full trio with decidable SUP. For each $k \in \mathbb{N}$, separability by LTT_k is decidable for C.*

Separability beyond PTLs Our framework can also be applied to separators that do not arise as PTLs for a particular WQO. This is because we can sometimes apply the developed ideal representations to separator classes that are infinite unions of invidual classes of PTLs. For example, consider the fragment $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$ of first-order logic on words with modular predicates. In terms of expressible languages, it is the union over all fragments $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$ with $d \in \mathbb{N}$. Using a non-trivial algebraic proof, it was shown by Chaubard, Pin, and Straubing [5] that it is decidable whether a regular language is definable in $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$. Here, we show the following generalization using a purely combinatorial proof.

Theorem 4.5. *Given two regular languages, it is decidable whether they are separable by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$.*

Of course, this raises the question of whether separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$ reduces to the SUP, as it is the case of separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$ for fixed d. However, this is not the case, as is shown here as well.

Theorem 4.6. *Separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$ is undecidable for second-order pushdown languages.*

Since the second-order pushdown languages constitute a full trio [1, 23] and have a decidable SUP [29], this means separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$ does not reduce to the SUP.

5 Computing closures and deciding separability

In this section, we present the algorithms used in Theorem 4.1. We apply the abstract framework for computing downward closures and deciding separability by PTL by Goubault-Larrecq and Schmitz [13], which is applicable to WQOs with particular effectiveness assumptions. In this section, we explain how these assumptions translate to our setting. In section 6, we will then show that all parameterized WQO indeed satisfy these properties.

Our algorithms for computing downward closures and deciding separability rely heavily on the concept of ideals, which have recently attracted attention [12, 13, 21]. Observe that, when deciding separability of languages from a class C by a recursively enumerable class of regular languages, it is usually easy to devise a semi-algorithm for the separability case: We can just enumerate separators. Verifying them is possible as soon as C has decidable emptiness of intersections with regular sets. The non-trivial part is to show that inseparability can be witnessed and in our case, this role will be played by the concept of ideals.

Let us define ideals. Suppose $(X, \leq)$ is a WQO. An *$\leq$-ascending chain* is a sequence $x_1, x_2, \ldots$ with $x_i \leq x_{i+1}$ for every $i \in \mathbb{N}$. A subset $Y \subseteq X$ is called *($\leq$-)directed* if for any $x, y \in Y$, there is a $z \in Y$ with $x \leq z$ and $y \leq z$. An *($\leq$-)ideal* is a non-empty subset $I \subseteq X$ that is $\leq$-downward closed and $\leq$-directed. Equivalently, a non-empty subset $I \subseteq X$ is an $\leq$-ideal if I is $\leq$-downward closed and for any two $\leq$-downward closed sets $Y, Z \subseteq X$ with $I \subseteq Y \cup Z$, we have $I \subseteq Y$ or $I \subseteq Z$. It is well-known that every downward closed set can be written as a finite union of ideals.

In order to explain how ideals can witness inseparability, we need the notion of adherences. For a set $L \subseteq X$, its *adherence* $\mathsf{Adh}_{\leq}(L)$ is defined as the set of those ideals I of X for which there exists a directed set $D \subseteq L$ with $I = \downarrow_{\leq} D$. Equivalently, $I \in \mathsf{Adh}_{\leq}(L)$ if and only if $I \subseteq \downarrow_{\leq}(L \cap I)$ [13, 21]. In this work, we also use a slightly modified version of adherences in order to describe ideals of conjunctions of WQOs. Let $(\leq_s)_{s \in S}$ be a family of well-quasi-orderings on a common set X. Moreover, let $\leq$ denote the conjunction of $(\leq_s)_{s \in S}$. For $L \subseteq X$, $\mathsf{Adh}_S(L)$ is the set of all families $(I_s)_{s \in S}$ of ideals for which there exists a $\leq$-directed set $D \subseteq L$ such that $I_s = \downarrow_{\leq_s} D$ for each $s \in S$.

Unboundedness reductions We use counter automata (that are not necessarily count*ing* automata) to specify unboundedness properties. Let $\mathcal{A}$ be a counter automaton with counter set C. Let $\mathbb{N}_\omega = \mathbb{N} \cup \{\omega\}$ and extend $\leq$ to $\mathbb{N}_\omega$ by setting $n < \omega$ for all $n \in \mathbb{N}$.

We define a function $\bar{\mathcal{A}}\colon \Sigma^* \to \mathbb{N}_\omega$ by

$$\bar{\mathcal{A}}(w) = \sup\left\{\inf_{c\in C}\mu(c)\,\middle|\, \mathcal{A} \text{ has a run on } w \text{ arriving at } \mu \in \mathbb{N}^C\right\}$$

We say that a counter automaton $\mathcal{A}$ is *unbounded on* $L \subseteq \Sigma^*$ if for every $k \in \mathbb{N}$, there is a $w \in L$ with $\bar{\mathcal{A}}(w) \geq k$. In other words, iff for every $\nu \in \mathbb{N}^C$, there is a $w \in L$ such that $\mathcal{A}$ has a run on w arriving at some $\mu \geq \nu$.

We say that a language $L \subseteq \Sigma^*$ has the *diagonal property* if for every $k \in \mathbb{N}$, there is a $w \in L$ that contains every letter from Σ at least k times. The *diagonal problem for a language class* C [9, 10] is to decide whether a given language from C has the diagonal property. Via computing downward closures, it is known that in full trios, the diagonal problem reduces to the SUP [29]. Since unboundedness of a counter automata easily reduces to the diagonal problem, we have the following.

Lemma 5.1. *Let C be a full trio with decidable SUP. Then, given a counter automaton $\mathcal{A}$ and a language L from C, it is decidable whether $\mathcal{A}$ is unbounded on L.*

We are now ready to state the effectiveness assumptions on which our algorithms rely. Let Σ be an alphabet and $(\Sigma^*, \preceq)$ be a WQO. We say that $(\Sigma^*, \preceq)$ is an *effective WQO with an unboundedness reduction (EWUR)* if the following are satisfied:

(a) For each $w \in \Sigma^*$, the set $\uparrow_\preceq w$ is effectively regular.
(b) The set of ideals of $(\Sigma^*, \preceq)$ is a recursively enumerable set of regular languages.
(c) Given an ideal $I \subseteq \Sigma^*$, one can effectively construct a counter automaton $\mathcal{A}_I$ such that for every $L \subseteq \Sigma^*$, $\mathcal{A}_I$ is unbounded on L if and only if I belongs to $\mathsf{Adh}_\preceq(L)$.

In order to decide separability by $\preceq$-PTL and compute downward closures, it would have sufficed to require decidability of adherence membership in full trios with decidable SUP. The reason why we require the stronger condition (c) is that in order to show that all parameterized WQOs satisfy these conditions, we want the latter to be passed on to conjunctions and to WQOs $\preceq_f$.

The conditions imply that every upward closed language (hence every downward closed language) is regular: If U is upward closed, then we can write $U = \uparrow_\preceq\{w_1, \ldots, w_n\} = \bigcup_{i=1}^n \uparrow_\preceq\{w_i\}$, which is regular because each $\uparrow_\preceq\{w_i\}$ is regular. Moreover, we may conclude that given a regular language $R \subseteq \Sigma^*$ it is decidable whether R is an ideal: If R is an ideal, we find it in an enumeration; if it is not an ideal, we find words that violate directedness or downward closedness.

According to the definition of EWUR, given an ideal I, we can construct a counter automaton $\mathcal{A}$ such that I belongs to the adherence of L if and only if $\mathcal{A}$ is unbounded on L. Hence, Lemma 5.1 implies the following.

Proposition 5.2. *Let $(\Sigma^*, \preceq)$ be an EWUR and let C be a full trio with decidable SUP. Then, given an $\preceq$-ideal $I \subseteq \Sigma^*$ and $L \in C$, it is decidable whether $I \in \mathsf{Adh}_\preceq(L)$.*

In section 6, we develop ideal representations for all parameterized WQOs and thus show that they are EWUR.

Proof sketch for Theorem 4.1 Let us now outline how to show Theorem 4.1 assuming that every parameterized WQO is an EWUR. The implication "2⟹1" holds because computing downward closures clearly allows deciding the SUP. This was shown in [29]. The implication "3⟹1" follows from [10], which presents a reduction of the SUP to separability by PTL. Thus, it remains to prove that downward closures are computable and PTL-separability is decidable for EWUR and full trios with decidable SUP. For the former, we can use an algorithm for downward closure computation from [13], which reduces the computation to adherence membership.

Proposition 5.3. *Let C be a full trio with decidable SUP and let $\preceq$ be an EWUR. Then $\preceq$-downward closures of languages in C are computable.*

We continue with the decidability of separability by $\preceq$-PTL for EWUR $\preceq$. We employ the following characterization of separability in terms of adherences [13] for reducing the separability problem to adherence membership.

Proposition 5.4. *Let $(X, \preceq)$ be a WQO. Then, $K \subseteq X$ and $L \subseteq X$ are separable by a $\preceq$-PTL iff $\mathsf{Adh}_\preceq(K) \cap \mathsf{Adh}_\preceq(L) = \emptyset$.*

We can now use the algorithm from [13] for deciding separability of languages K and L in our setting. By Proposition 5.4, we can use two semi-decision procedures. On the one hand, we enumerate potential separators S and check whether $K \subseteq S$ and $L \cap S = \emptyset$. On the other hand, we enumerate $\preceq$-ideals I and check if I belongs to $\mathsf{Adh}_\preceq(K) \cap \mathsf{Adh}_\preceq(L)$.

Proposition 5.5. *Let C be a full trio with decidable SUP and $\preceq$ be an EWUR. Then separability by $\preceq$-PTL is decidable for C.*

6 Ideal representations

In this section, we show that every parameterized WQO is an EWUR. The fact that the subword ordering is an EWUR follows using ideal representations for subwords [19] and arguments from [10, 29].

Proposition 6.1. *The subword ordering $(\Sigma^*, \preccurlyeq)$ is an EWUR.*

The next step is to show that whenever $(\Gamma^*, \preceq)$ is an EWUR and $f\colon \Sigma^* \to \Gamma^*$ is a functional rational transduction, then $(\Sigma^*, \preceq_f)$ is an EWUR. We begin with some general observations about ideals in WQOs of the shape $\preceq_f$, where $f\colon X \to Y$ is an arbitrary function and $(Y, \preceq)$ is a WQO. First, we describe ideals of $(X, \preceq_f)$ in terms of ideals of $(Y, \preceq)$.

It is easy to see that every ideal of $(X, \preceq_f)$ is of the form $f^{-1}(J)$ for some ideal J of $(Y, \preceq)$. However, a set $f^{-1}(J)$ is not always an ideal of $(X, \preceq_f)$. For example, suppose $f\colon \Sigma^* \to \mathbb{N} \times \mathbb{N}$ has $f(w) = (|w|, 0)$ if $|w|$ is even and $f(w) = (0, |w|)$ if $|w|$ is odd. Then $f^{-1}(\mathbb{N} \times \mathbb{N})$ is not upward directed although $\mathbb{N} \times \mathbb{N}$ is an ideal.

Lemma 6.2. *$I \subseteq X$ is an ideal of $(X, \preceq_f)$ if and only if $I = f^{-1}(J)$ for some ideal J of $(Y, \preceq)$ such that $\downarrow f(f^{-1}(J)) = J$.*

Proof. If $I \subseteq X$ is an ideal, then the set $J := \downarrow f(I)$ is downward closed by definition and upward directed because I is. Hence, J is an ideal. Moreover, $I = f^{-1}(J)$, because $I \subseteq f^{-1}(J)$ is immediate and $f^{-1}(J) \subseteq I$ holds because I is downward closed. This also implies $\downarrow f(f^{-1}(J)) = \downarrow f(I) = J$.

Conversely, let $I = f^{-1}(J)$ for an ideal $J \subseteq Y$ that satisfies $\downarrow f(f^{-1}(J)) = J$. First, $I = f^{-1}(J)$ is downward closed because J is. Moreover, we have $\downarrow f(I) = J$, which means given $x, y \in I$, we can find a common upper bound $z \in J$ for $f(x) \in J$ and $f(y) \in J$ and then a $z' \in f(I)$ with $z \preceq z'$. Then $z' = f(w)$ for some $w \in I$ and hence $x \preceq_f w$ and $y \preceq_f w$. Thus I is upward directed. ◻

Lemma 6.2 tells us how to represent ideals of $(X, \preceq_f)$ when we have a way of representing ideals of $(Y, \preceq)$. Hence, if the set of ideals of $(\Gamma^*, \preceq)$ is recursively enumerable, then so is the set of ideals of $(\Sigma^*, \preceq_f)$: As an ideal, J is regular, meaning that $\downarrow f(f^{-1}(J))$ is effectively regular and can be compared with J. We also need to transfer membership in adherences from $(Y, \preceq)$ to $(X, \preceq_f)$.

Lemma 6.3. *If $J \subseteq Y$ is an ideal of $(Y, \preceq)$ with $\downarrow f(f^{-1}(J)) = J$, then $f^{-1}(J) \in \mathsf{Adh}(L)$ if and only if $J \in \mathsf{Adh}(f(L))$.*

Proof. Let $f^{-1}(J) \in \mathsf{Adh}(L)$, equivalently, $f^{-1}(J) \subseteq \downarrow(L \cap f^{-1}(J))$. We prove that $J \subseteq \downarrow(f(L) \cap J)$. For $y \in J$, we can find $y' \in f(f^{-1}(J))$ with $y \preceq y'$. Say $y' = f(x')$ with $x' \in f^{-1}(J)$. Thus, there exists an $x'' \in L \cap f^{-1}(J)$ with $x' \preceq_f x''$. Since $y \preceq y' = f(x') \preceq f(x'')$ and $f(x'') \in f(L) \cap J$, we have shown $J \subseteq \downarrow(f(L) \cap J)$.

Conversely, let $J \in \mathsf{Adh}(f(L))$, i.e. $J \subseteq \downarrow(f(L) \cap J)$. This means, for $x \in f^{-1}(J)$, we can find $x' \in L$ with $f(x) \preceq f(x')$ and $f(x') \in J$. Thus, $f^{-1}(J) \subseteq \downarrow(L \cap f^{-1}(J))$ and hence $f^{-1}(J) \in \mathsf{Adh}(L)$. □

Equipped with Lemmas 6.2 and 6.3, it is now straightforward to show that $(\Sigma^*, \preceq_f)$ is an EWUR.

Proposition 6.4. *If $(\Gamma^*, \preceq)$ is an EWUR and $f\colon \Sigma^* \to \Gamma^*$ is a functional rational transduction, then $(\Sigma^*, \preceq_f)$ is an EWUR.*

It remains to be shown that being an EWUR is preserved by taking a conjunction. Our first step is to characterize which sets are ideals of a conjunction. Once the statement is found, the proof is relatively straightforward.

Proposition 6.5. *Let $S = (\preceq_s)_{s \in S}$ be a finite family of WQOs over X and let $(X, \preceq)$ be the conjunction of S. Then $I \subseteq X$ is an ideal of $(X, \preceq)$ iff it can be written as $I = \bigcap_{s \in S} I_s$, where each I_s is an ideal of $(X, \preceq_s)$ and $(I_s)_{s \in S}$ belongs to $\mathsf{Adh}_S(I)$.*

The next step describes how to reduce the adherence membership problem for conjunctions to the adherence membership problem for the participating orderings.

Proposition 6.6. *Let $S = (\preceq_s)_{s \in S}$ be a finite family of WQOs over X and let $(X, \preceq)$ be the conjunction of S. Suppose I_s is an $\preceq_s$-ideal for each $s \in S$ and $I = \bigcap_{s \in S} I_s$ and that $(I_s)_{s \in S}$ belongs to $\mathsf{Adh}_S(I)$. Then I belongs to $\mathsf{Adh}_{\preceq}(L)$ iff $(I_s)_{s \in S}$ belongs to $\mathsf{Adh}_S(L)$.*

As expected, a product construction allows us to characterize the adherence membership for conjunction.

Lemma 6.7. *Suppose $(\Sigma^*, \preceq_i)$ is an EWUR for $i = 1, 2$. Given ideals I_1 and I_2 for $\preceq_1$ and $\preceq_2$, respectively, we can construct a counter automaton $\mathcal{A}$ such that for every language $L \subseteq \Sigma^*$, $\mathcal{A}$ is unbounded on L iff (I_1, I_2) belongs to $\mathsf{Adh}_{\preceq_1, \preceq_2}(L)$.*

The following is now a consequence of the previous steps.

Proposition 6.8. *If $\preceq_1$ and $\preceq_2$ are EWUR, then their conjunction is an EWUR as well.*

Orderings defined by labeling automata The preceding results already show that every parameterized WQO is an EWUR. However, since we will study separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$, it will be crucial to have an explicit, i.e. syntactic representation of ideals of a particular type of parameterized WQOs, namely those defined by labeling automata. Here, we develop such a syntax.

Let $\mathcal{A}$ be a labeling automaton over Σ^*, $u_0, \ldots, u_n \in \Sigma^*$, and $v_1, \ldots, v_n \in \Sigma^*$. The word $w = u_0 v_1 u_1 \cdots v_n u_n$ (more precisely: this particular decomposition) is a *loop pattern (for $\mathcal{A}$)* if the run of $\mathcal{A}$ on w loops at each v_i, $i \in [1, n]$. In other words, $\mathcal{A}$ is in the same state before and after reading v_i.

Theorem 6.9. *Let $\mathcal{A}$ be a labeling automaton. The $\preceq_{\mathcal{A}}$-ideals are precisely the sets $\downarrow_{\preceq_{\mathcal{A}}} u_0 v_1^* u_1 \cdots v_n^* u_n$, where $u_0 v_1 u_1 \cdots v_n u_n$ is a loop pattern for $\mathcal{A}$.*

By standard arguments, it suffices to show that those sets are ideals and that every downward closed set is a finite union of such sets.

7 Separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$

In this section, we prove Theorem 4.5 and Theorem 4.6. The latter will be shown in section 7.1 and the former is an immediate consequence of the following.

Proposition 7.1. *Let $\mathcal{A}_1, \mathcal{A}_2$ be finite automata with $\leq m$ states. $L(\mathcal{A}_1)$ and $L(\mathcal{A}_2)$ are separable by $\mathcal{B}\Sigma_1[<, \mathsf{mod}]$ if and only if they are separable by $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$, where $d = 2m^3!$.*

Recall that $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$ are the $\preceq_{\mathcal{M}_d}$-PTL, where $\mathcal{M}_d$ is the labeling automaton defined on page 3. From now on, we write $\preceq_d$ for $\preceq_{\mathcal{M}_d}$. Proposition 7.1 follows from:

Proposition 7.2. *Let $\mathcal{A}_i$ be a finite automaton for $i = 1, 2$ with $\leq m$ states and let d be a multiple of $2m^3!$. If*

$$\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_1)) \cap \mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_2)) \neq \emptyset,$$

then

$$\mathsf{Adh}_{\preceq_{\ell \cdot d}}(L(\mathcal{A}_1)) \cap \mathsf{Adh}_{\preceq_{\ell \cdot d}}(L(\mathcal{A}_2)) \neq \emptyset$$

for every $\ell \geq 1$.

The "if" direction of Proposition 7.1 is trivial and the "only if" can be derived from Proposition 7.2 as follows. If $L(\mathcal{A}_1)$ and $L(\mathcal{A}_2)$ are separable by $\mathcal{B}\Sigma_1[<, \mathsf{mod}_\ell]$ for some $\ell \in \mathbb{N}$, then this separator is also expressible in $\mathcal{B}\Sigma_1[<, \mathsf{mod}_{\ell \cdot d}]$. Moreover, together with Proposition 5.4, Proposition 7.2 tells us that separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}_{\ell \cdot d}]$ implies separability by $\mathcal{B}\Sigma_1[<, \mathsf{mod}_d]$.

The rest of this section outlines the proof of Proposition 7.1. Note that according to Theorem 6.9, the ideals for $\preceq_d$ are the sets of the form $I = \downarrow_{\preceq_d} u_0 v_1^* u_1 \cdots v_n^* u_n$ where $v_i \in (\Sigma^d)^*$. The ideal I belongs to $\mathsf{Adh}_{\preceq_d}(L)$ if and only if for each $k \in \mathbb{N}$, there is a word $w \in L$ such that $u_0 v_1^k u_1 \cdots v_n^k u_n \preceq_d w$ and $w \in I$. We call such words w *witness words*.

It is tempting to think that Proposition 7.2 just requires a simple pumping argument: Given witness words for adherence membership of an $\preceq_d$-ideal $\downarrow_{\preceq_d} u_0 v_1^* u_1 \cdots v_n^* u_n$, we pump the gaps in between embedded letters from the word $u_0 v_1^{\ell \cdot k} u_1 \cdots v_n^{\ell \cdot k} u_n$. These gaps, after all, always have length divisible by d. For a d with sufficiently many divisors, we would be able to pump the gaps up to a length divisible by $\ell \cdot d$ so that we can embed $u_0 (v_1^\ell)^k u_1 \cdots (v_n^\ell)^k u_n$ via $\preceq_{\ell \cdot d}$ and conclude membership of the ideal $\preceq_{\ell \cdot d}$-ideal

$$I' = \downarrow_{\preceq_{\ell \cdot d}} u_0 (v_1^\ell)^* u_1 \cdots (v_n^\ell)^* u_n.$$

However, in order to show that I' contained in the $\preceq_{\ell \cdot d}$-adherence, we also have to make sure that resulting witness words are *members of I'*. This makes the proof challenging.

Part I: Small periods Our proof of Proposition 7.2 consists of three parts. In the first part, we show that if two regular languages share an ideal in their adherences, then there exists one in which all loops (the words v_i) are, in a certain sense, highly periodic. Let $\mathcal{P}(\Sigma)$ denote the power set of Σ and let $\mathcal{P}(\Sigma)^{[1,d]}$ denote the set of mappings $\mu\colon [1, d] \to \mathcal{P}(\Sigma)$. For each word $w \in \Sigma^*$ and $d \in \mathbb{N}$, let $\kappa_d(w) \in \mathcal{P}(\Sigma)^{[1,d]}$ be defined as follows. For $i \in [1, d]$, we set

$$\kappa_d(w)(i) = \{a \in \Sigma \mid a \text{ occurs in } w \text{ at a position } p \equiv i \bmod d\}.$$

The function κ_d lets us characterize inclusion among simple ideals.

Lemma 7.3. *Suppose $v, w \in (\Sigma^d)^*$. Then $\downarrow_{\leq_d} v^* \subseteq \downarrow_{\leq_d} w^*$ if and only if $\kappa_d(v) \subseteq \kappa_d(w)$.*

For each word $w \in \Sigma^*$, let $\rho(w)$ be obtained from rotating w by one position to the right. Hence, for $v \in \Sigma^*$ and $a \in \Sigma$ we have $\rho(va) = av$, and $\rho(\varepsilon) = \varepsilon$. Let λ be the inverse map of ρ, i.e. rotation to the left. For $v \in \Sigma^*$ and $d \in \mathbb{N}$, let $\pi_d(v) \in [1, d]$ be the smallest $t \in [1, d]$ that divides d such that $\kappa_d(v)(i + t) = \kappa_d(v)(i)$ for all $i \in [1, d - t]$. Thus, t can be thought of as a period of $\kappa_d(v)$. An automaton $\mathcal{A} = (Q, \Sigma, E, I, F)$ is *cyclic* if $I = F$ and $|I| = 1$. The first step towards ideals with high periodicity is to achieve high periodicity in single-loop ideals in cyclic automata:

Lemma 7.4. *Let $\mathcal{A}_i$ be a cyclic automaton with $\leq m$ states for each $i = 1, 2$ and let d be a multiple of $m^2!$. If $\downarrow_{\leq_d} v^*$ belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$, then there is a $w \in (\Sigma^d)^*$ such that (i) $\downarrow_{\leq_d} v^* \subseteq \downarrow_{\leq_d} w^*$, (ii) $\downarrow_{\leq_d} w^*$ also belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$, and (iii) $\pi_d(w) \leq m^2$.*

Proof of Lemma 7.4. Write $v = v_1 \cdots v_n$, $v_1, \ldots, v_n \in \Sigma$. Since the ideal $\downarrow_{\leq_d} v^*$ belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$, we can find $v \in \downarrow_{\leq_d}(L(\mathcal{A}_i) \cap \downarrow_{\leq_d} v^*)$ for $i = 1, 2$. This means there are witnesses

$$\bar{v}_i = \bar{u}_{i,0} v_1 \bar{u}_{i,1} \cdots v_n \bar{u}_{i,n} \in L(\mathcal{A}_i) \cap \downarrow_{\leq_d} v^*$$

such that $\bar{u}_{i,j} \in (\Sigma^d)^*$ for $j \in [1, n]$ and $i = 1, 2$. Note that since $\bar{v}_i \in \downarrow_{\leq_d} v^*$ and $v \leq_d \bar{v}_i$, we have $\downarrow_{\leq_d} \bar{v}_i^* = \downarrow_{\leq_d} v^*$ and thus we have $\kappa_d(\bar{v}_i) = \kappa_d(v)$ according to Lemma 7.3.

In the run of $\mathcal{A}_i$ for $\bar{u}_{i,0} v_1 \bar{u}_{i,1} \cdots v_n \bar{u}_{i,n}$, let $q_{i,j}$ be the state occupied after reading $\bar{u}_{i,j}$, for $j \in [0, n]$ and $i = 1, 2$. Since $m^2!$ divides d, which in turn divides n, we have $n+1 > m^2! \geq m^2$. Therefore, there are $j, k \in [0, n]$, $j < k$, with $(q_{1,j}, q_{2,j}) = (q_{1,k}, q_{2,k})$. Moreover, they can be chosen so that $t := k - j < m^2$. Since $m^2!$ divides d, we know that $t < m^2$ divides d and may define $r = d/t$. Let $x_i = \bar{u}_{i,0} v_1 \bar{u}_{i,1} \cdots v_j \bar{u}_{i,j}$, $y_i = v_{j+1} \bar{u}_{i,j+1} \cdots v_k \bar{u}_{i,k}$, $z_i = v_{k+1} \bar{u}_{i,k+1} \cdots v_n \bar{u}_{i,n}$. Then, by the choice of j, k, we have $(x_i y_i^* z_i)^* \subseteq L(\mathcal{A}_i)$. In particular, the word

$$w_i = \prod_{\ell=0}^{r-1} x_i y_i y_i^{\ell} z_i x_i y_i y_i^{r-\ell} z_i$$

belongs to $L(\mathcal{A}_i)$. Moreover, since $|y_i| \equiv t \bmod d$, we can conclude $|w_i| \equiv r(2|x_i y_i z_i| + rt) \equiv 0 \bmod d$. We claim that

$$\kappa_d(w_i) = \bigcup_{\ell=0}^{r-1} \kappa_d(\rho^{\ell t}(\bar{v}_i)).$$

We begin with the inclusion "$\supseteq$". For each $\ell \in [0, r-1]$ and $i \in \{1, 2\}$,

- the word x_i occurs in w_i at a position $p \equiv |x_i y_i z_i| + \ell t \pmod d$ and hence $p \equiv \ell t \pmod d$,
- the word y_i occurs in w_i at a position $p \equiv |x_i| + \ell t \pmod d$,
- the word z_i occurs in w_i at a position $p \equiv |x_i y_i| + \ell t \pmod d$.

Hence, for each position p in $\bar{v}_i$ and each $\ell \in [0, r - 1]$, there is a position $p' \equiv p + \ell t \pmod d$ with $\kappa_d(\bar{v}_i)(p) \subseteq \kappa_d(w_i)(p')$. This prove the inclusion "$\supseteq$".

On the other hand, every factor x_i, y_i, and z_i that occurs in the definition of w_i at a position $p \in [1, |w_i|]$ also occurs in $\bar{v}_i$ at a position $p' \in [1, n]$ with $p' \equiv p - \ell t \pmod d$ for some $\ell \in [0, r - 1]$. Therefore, we also have the inclusion "$\subseteq$".

The identity $\kappa_d(w_i) = \bigcup_{\ell=0}^{r-1} \kappa_d(\rho^{\ell \cdot t}(\bar{v}_i))$ clearly implies that $\pi_d(w_i) \leq t$ and also $\downarrow_{\leq_d}(\bar{v}_i)^* \subseteq \downarrow_{\leq_d} w_i^*$, which in turn yields $\downarrow_{\leq_d} v^* \subseteq \downarrow_{\leq_d} w_i^*$. Moreover, since $(x_i y_i^* z_i)^* \subseteq L(\mathcal{A}_i)$, we have $w_i^* \subseteq L(\mathcal{A}_i)$ and in particular $\downarrow_{\leq_d} w_i^* \subseteq \downarrow_{\leq_d} L(\mathcal{A}_i)$. This clearly implies that $\downarrow_{\leq_d} w_i^*$ belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$. Hence, if we can show $\downarrow_{\leq_d} w_1^* = \downarrow_{\leq_d} w_2^*$, the proof is complete. We use ρ also as a rotation map on $\mathcal{P}(\Sigma)^{[1,d]}$: For $\mu \in \mathcal{P}(\Sigma)^{[1,d]}$ and $i \in [1, d]$, let $\rho(\mu)(i) = \mu(i')$, where $i' \in [1, d]$ is chosen so that $i' \equiv i - 1 \bmod d$. Note that $\kappa_d(\rho(z)) = \rho(\kappa_d(z))$ for every $z \in \Sigma^*$. Observe that since $\kappa_d(\bar{v}_i) = \kappa_d(v)$ for $i \in \{1, 2\}$, we have

$$\kappa_d(w_i) = \bigcup_{\ell=0}^{r-1} \kappa_d(\rho^{\ell t}(\bar{v}_i)) = \bigcup_{\ell=0}^{r-1} \rho^{\ell t}(\kappa_d(\bar{v}_i)) = \bigcup_{\ell=0}^{r-1} \rho^{\ell t}(\kappa_d(v)),$$

and thus $\kappa_d(w_1) = \kappa_d(w_2)$, which, according to Lemma 7.3, implies $\downarrow_{\leq_d} w_1^* = \downarrow_{\leq_d} w_2^*$. □

Associated patterns In order to extend this to general ideals and automata, we need more guarantees on how words $u_0 v_1^k u_1 \cdots v_n^k u_n$ embed into witness words.

Let $u_0 v_1 u_1 \cdots v_n u_n$ be a loop pattern for $\mathcal{M}_d$ and let $L \subseteq \Sigma^*$. We say that the loop pattern is *associated to* L if for every $k \geq 0$, there is a word $\bar{u}_0 \bar{v}_1 \bar{u}_1 \cdots \bar{v}_n \bar{u}_n \in L$ such that $v_i^k \leq_d \bar{v}_i \in \downarrow_{\leq_d} v_i^*$ for every $i \in [1, n]$ and $u_i \leq_d \bar{u}_i \in \downarrow_{\leq_d} v_i^* u_i v_{i+1}^*$ for $i \in [1, n-1]$ and $u_0 \leq_d \bar{u}_0 \in \downarrow_{\leq_d} u_0 v_1^*$ and $u_n \leq_d \bar{u}_n \in \downarrow_{\leq_d} v_n^* u_n$.

Of course, if the pattern $u_0 v_1 u_1 \cdots v_n u_n$ is associated to L, then the ideal $I = \downarrow_{\leq_d} u_0 v_1^* u_1 \cdots v_n^* u_n$ belongs to $\mathsf{Adh}_{\leq_d}(L)$. However, the converse is not true. Consider, for example, the case $d = 2$ and the loop pattern $\varepsilon \cdot (aa) \cdot \varepsilon \cdot (abba) \cdot \varepsilon$, where aa and $abba$ are loops and the constant parts are all empty. The resulting ideal $\downarrow_{\leq_2}(aa)^*(abba)^*$ belongs to $\mathsf{Adh}_{\leq_2}((abba)^*)$ because of the identity $\downarrow_{\leq_2}(aa)^*(abba)^* = \downarrow_{\leq_2}(abba)^*$: Both sets contain precisely the words in $\{a, b\}^*$ of even length. Note that the pattern $\varepsilon \cdot (aa) \cdot \varepsilon \cdot (abba) \cdot \varepsilon$ is not associated to $(abba)^*$, because no word in the latter contains $(aa)^2$ as an infix, let alone arbitrarily high powers of aa.

However, we will see that every ideal admits a representation by a loop pattern so that membership in the adherence implies association of the loop pattern. A loop pattern $u_0 v_1 u_1 \cdots v_n u_n$ for $\mathcal{M}_d$ is *irreducible* if removing any loop would induce a strictly smaller ideal. This means, for each $i \in [1, n]$, the loop pattern $u_0(v_1)u_1 \cdots (v_{i-1})u_{i-1} u_i \cdots (v_n)u_n$ induces a strictly smaller ideal than $u_0 v_1 u_1 \cdots v_n u_n$. Every ideal is induced by some irreducible loop pattern: Just pick one with a minimal number of loops.

We shall prove that for an irreducible loop pattern, membership in the adherence of a language L implies association to L (Lemma 7.6). To that end, we prove first that if the loop pattern $u_0 v_1 u_1 \cdots v_n u_n$ is irreducible, then for each $k \in \mathbb{N}$, any embedding of $u_0 v_1^{x_1} u_1 \cdots v_n^{x_n} u_n$ into $u_0 v_1^{y_1} u_1 \cdots v_n^{y_n} u_n$ for sufficiently large x_i forces at least k copies of each v_i to be embedded into $v_i^{y_i}$.

Let us make this precise. Suppose $x, y \in \Sigma^*$, $x = x_1 \cdots x_r$, and $y = y_1 \cdots y_s$, where $x_1, \ldots, x_r, y_1, \ldots, y_r$ are letters in Σ. A strictly monotone map $\alpha\colon \{1, \ldots, r\} \to \{1, \ldots, s\}$ is a *d-embedding of x in y* if $r \equiv s \pmod d$, $x_i = y_{\alpha(i)}$ for $i \in [1, r]$, and for each $i \in [1, r]$, we have $\alpha(i) \equiv i \pmod d$. Clearly, we have $x \leq_d y$ if and only if there is a d-embedding of x in y. Now let $u_0 v_1 u_1 \cdots v_n u_n$ be a loop pattern for $\mathcal{M}_d$ and $x = u_0 v_1^{x_1} u_1 \cdots v_n^{x_n} u_n$ and $y = u_0 v_1^{y_1} u_1 \cdots v_n^{y_n} u_n$. A d-embedding of x in y is called *k-normal* if for each $i \in [1, n]$, α maps at least k-many factors v_i in x to $v_i^{y_i}$. Clearly, if $k \leq x_i \leq y_i$

for all $i \in [1, n]$, then there exists a normal d-embedding of x in y. However, not every d-embedding has to be k-normal.

Lemma 7.5. *Let $u_0v_1u_1 \cdots v_nu_n$ be an irreducible loop pattern for $\mathcal{M}_d$. For each $k \in \mathbb{N}$, there is a constant $\ell \in \mathbb{N}$ such that if α is a d-embedding of $u_0v_1^{x_1}u_1 \cdots v_n^{x_n}u_n$ in $u_0v_1^{y_1}u_1 \cdots v_n^{y_n}u_n$ and $x_i \geq \ell$ for $i \in [1, n]$, then α is k-normal.*

Here, the idea is the following. If there were a k such that for simultaneously unbounded vectors $(x_1, \ldots, x_n)$, we can embed $u_0v_1^{x_1}u_1 \cdots v_n^{x_n}u_n$ into $u_0v_1^{y_1}u_1 \cdots v_n^{y_n}u_n$ while sending at most k copies of v_i to $v_i^{y_i}$ for some $i \in [1, n]$, then an unbounded amount of copies of v_i has to be placed either to the left or to the right of $v_i^{y_i}$. From that, one can deduce that the loop v_i in the pattern can be removed without shrinking the generated ideal.

Lemma 7.6. *Let $u_0v_1u_1 \cdots v_nu_n$ be an irreducible loop pattern for $\mathcal{M}_d$. Then $\downarrow_{\leq_d} u_0v_1^*u_1 \cdots v_n^*u_n$ belongs to $\mathsf{Adh}_{\leq_d}(L)$ if and only if $u_0v_1u_1 \cdots v_nu_n$ is associated to L.*

Using Lemma 7.6, we can complete the first proof part:

Lemma 7.7. *Let $\mathcal{A}_i$ be a finite automaton with $\leq m$ states for each $i = 1, 2$ and let d be a multiple of $m^2!$. If $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_1)) \cap \mathsf{Adh}_{\leq_d}(L(\mathcal{A}_2)) \neq \emptyset$, then there is a loop pattern $u_0v_1u_1 \cdots v_nu_n$ for $\mathcal{M}_d$ such that $\downarrow_{\leq_d} u_0v_1^*u_1 \cdots v_n^*u_n$ belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$ and $\pi_d(v_j) \leq m^2$ for $j \in [1, n]$.*

Part II: Almost perfect witnesses In the second part, we place further restrictions on the structure of ideals that witness inseparability. In return, we get stronger guarantees on the shape of witness words. Using Lemma 7.7, proving Proposition 7.2 would not be difficult if for a loop pattern $u_0v_1u_1 \cdots v_nu_n$, we could guarantee witness words of the shape $u_0\bar{v}_1u_1 \cdots \bar{v}_nu_n$ with $\bar{v}_i \in \downarrow_{\leq_d} v_i^*$. Let us call such witnesses *perfect*. Unfortunately, even for irreducible loop patterns, we cannot guarantee perfect witnesses: Consider the ideal $I = \downarrow_{\leq_2} a(abba)^*$. The loop pattern $a \cdot (abba) \cdot \varepsilon$ (with the loop $abba$) is clearly irreducible. Moreover, I belongs to $\mathsf{Adh}_{\leq_2}(L)$ for $L = b\{a, b\}^*$: For $k \in \mathbb{N}$, the word $b(abba)^{k+1} \in L$ satisfies $a(abba)^k \leq_2 b(abba)^{k+1} \leq_2 a(abba)^{k+2}$, which proves $I \subseteq \downarrow_{\leq_2}(L \cap I)$. Here, the witness words $b(abba)^{k+1}$ are not perfect because they start in b instead of a.

We shall see later that, with an extended syntax for loop patterns and an adapted notion of irreducibility, we can guarantee almost perfect witnesses. An *extended loop pattern (for $\mathcal{M}_d$)* is an expression of the form $u_0v_1^{[r_1]}u_1 \cdots v_n^{[r_n]}u_n$, in which $u_0v_1u_1 \cdots v_nu_n$ is a loop pattern for $\mathcal{M}_d$ (i.e. $v_i \in (\Sigma^d)^*$ for $i \in [1, n]$) and where $r_1, \ldots, r_n \in [0, d-1]$. The *ideal generated by the extended loop pattern* is defined as $\downarrow_{\leq_d} u_0v_1^*w_1u_1 \cdots v_n^*w_nu_n$, where w_i is the length-r_i prefix of v_i for $i \in [1, n]$. Slightly abusing notation, we use $\downarrow_{\leq_d} u_0v_1^{[r_1]}u_1 \cdots v_n^{[r_n]}u_n$ to denote the generated ideal. When we use such an expression with $r_i > d$, this stands for the pattern $u_1v_1^{[s_1]}u_1 \cdots v_n^{[s_n]}u_n$, where $s_i \in [0, d-1]$ and $s_i \equiv r_i \pmod{d}$.

Let us now introduce our notion of "almost perfect witnesses". Consider an extended loop pattern $u_0v_1^{[r_1]}u_1 \cdots v_n^{[r_n]}u_n$ for $\mathcal{M}_d$ and let w_i be the length-r_i prefix of v_i for $i \in [1, n]$. The pattern is said to be *associated* to a language L if for every $k \in \mathbb{N}$, there is a word $\bar{u}_0\bar{v}_1\bar{u}_1 \cdots \bar{v}_n\bar{u}_n \in L$ so that for every $i \in [1, n]$, we have $v_i^kw_i \leq_d \bar{v}_i$ and $\bar{v}_i \in \downarrow_{\leq_d} v_i^{[r_i]}$. Moreover, $\bar{u}_0 = u_0$, $\bar{u}_n = u_n$, and for each $i \in [1, n-1]$: (i) if u_i is not empty, then $\bar{u}_i = u_i$ and (ii) if u_i is empty, then $\bar{u}_i \in \downarrow_{\leq_d} \lambda^{r_i}(v_i)^*v_{i+1}^*$. Let us call such witnesses *almost perfect*.

As in Lemma 7.6, we have a notion of irreducible extended loop patterns. Consider an extended loop pattern $u_0v_1^{[r_1]}u_1 \cdots v_n^{[r_n]}u_n$ and let w_i be the length-r prefix of v_i for $i \in [1, n]$. We say that this extended loop pattern is *irreducible* if

1. the corresponding loop pattern $u_0(v_1)w_1u_1 \cdots (v_n)w_nu_n$ is irreducible and
2. for each $i \in [0, n-1]$, u_i is either empty or the last letter of u_i is not contained in $\kappa_d(v_{i+1})(d)$ and
3. for each $i \in [1, n]$, u_i is either empty or the first letter of u_i is not contained in $\kappa_d(v_i)(r_i + 1)$.

Irreducible extended loop pattern admit almost perfect witnesses:

Lemma 7.8. *The ideal generated by an irreducible extended loop pattern p for $\mathcal{M}_d$ belongs to $\mathsf{Adh}_{\leq_d}(L)$ iff p is associated to L.*

However, irreducible extended loop patterns do not guarantee perfect witnesses. In other words, we cannot guarantee $\bar{u}_i = u_i$ if $u_i = \varepsilon$ but have to allow $\bar{u}_i \in \downarrow_{\leq_d} \lambda^{r_i}(v_i)^*v_{i+1}^*$. Consider, for example, the extended loop pattern $(ab)^{[0]}(cd)^{[0]}$. It is irreducible and its ideal $I = \downarrow_{\leq_2}(ab)^*(cd)^*$ belongs to $\mathsf{Adh}_{\leq_2}((ab)^*ad(cd)^*)$, but the witness words $(ab)^kad(cd)^k \in I$ always contain a factor $ad \in \downarrow_{\leq_2}(ab)^*(cd)^*$.

To complete the second part, we need to show that every ideal can be represented by an irreducible extended loop pattern. Moreover, the construction of the pattern should not destroy the previously established upper bound on $\pi_d(v_i)$. The following can be shown straighforwardly by choosing an extended loop pattern with a suitable minimality condition.

Lemma 7.9. *Let $x_0y_1^{[s_1]} \cdots y_\ell^{[s_\ell]}x_\ell$ be an extended loop pattern for $\mathcal{M}_d$ for which $\pi_d(y_i) \leq B$ for every $i \in [1, \ell]$. Then there is an irreducible extended loop pattern $u_0v_1^{[r_1]}u_1 \cdots v_n^{[r_n]}u_n$ for $\mathcal{M}_d$ generating the same ideal where also $\pi_d(v_i) \leq B$ for every $i \in [1, n]$.*

Part III: Pumping up The final part of the proof of Proposition 7.2 is to construct $\leq_{\ell \cdot d}$-ideals using pumping.

Lemma 7.10. *Let $\mathcal{A}$ be a finite automaton with $\leq m$ states and let d be a multiple of $2m^3!$. If $u_0v_1^{[r_1]}u_1 \cdots v_n^{[r_n]}u_n$ is an irreducible extended loop pattern with $\pi_d(v_i) \leq m^2$ such that its ideal belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}))$, then for each $\ell \in \mathbb{N}$, the ideal*

$$\downarrow_{\leq_{\ell \cdot d}} u_0(v_1^\ell)^{[r_1]}u_1 \cdots (v_n^\ell)^{[r_n]}u_n \tag{1}$$

belongs to $\mathsf{Adh}_{\leq_{\ell \cdot d}}(L(\mathcal{A}))$.

Here, the strong guarantees of associated extended loop patterns allow us to focus on two types of factors in which we must pump: factors $\bar{v}_i$ and factors $\bar{u}_i$ for empty u_i. One can show that repeating subfactors thereof whose length is divisible by a particular $\pi_d(v_i)$ will not lead out of the $\leq_{\ell \cdot d}$-ideal. Moreover, since we established in the first part that each period $\pi_d(v_i)$ is small ($\leq m^2$), we can always find a subfactor of length divisible by $\pi_d(v_i)$ that is pumpable.

7.1 Undecidability

In this section, we prove Theorem 4.6. Second-order pushdown languages are those accepted by second-order pushdown automata [23]. In order to prove that separability of second-order pushdown languages by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$ is undecidable, we need no detailed definition of second-order pushdown automata. All we need is that their languages form a full trio (shown by Aho [1] for the equivalent indexed grammars) and that we can construct automata for two particular types of languages. Let us describe these languages. For $w \in \{1,2\}^*$, let $\nu(w)$ be the number obtained by interpreting the word as a reverse 2-adic representation. Thus, for $w \in \{1,2\}^*$, let

$$\nu(\varepsilon) = 0, \qquad \nu(1w) = 2 \cdot \nu(w) + 1, \qquad \nu(2w) = 2 \cdot \nu(w) + 2.$$

Note that $\nu\colon \{1,2\}^* \to \mathbb{N}$ is a bijection. In the full version of [29], it was shown[1] that given two morphisms $\alpha, \beta\colon \Sigma^* \to \{1,2\}^*$, one can construct in polynomial time an indexed grammar generating $\{a^{\nu(\alpha(w))}b^{\nu(\beta(w))} \mid w \in \Sigma^+\}$. Applying a simple transduction yields

$$L_{\alpha,\beta} = \{a^{\nu(\alpha(w))}cb^{\nu(\beta(w))} \mid w \in \Sigma^+\}$$

and hence an indexed grammar for $L_{\alpha,\beta}$. The context-free language $E = \{a^n cb^n \mid n \in \mathbb{N}\}$ is also a second-order pushdown language. We apply a technique introduced by Hunt [18] and simplified by Czerwiński and Lasota [8]. The idea is to show that *every* decidable problem reduces to our problem in polynomial time:

Proposition 7.11. *For each decidable $D \subseteq \Gamma^*$, there is a polynomial-time algorithm that, given $u \in \Gamma^*$, computes morphisms α, β such that $L_{\alpha,\beta}$ is inseparable from E by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$ if and only if $u \in D$.*

Thus, decidability of separability by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$ would contradict the time hierarchy theorem (see, e.g. [26, Thm 9.10]).

Let us prove Proposition 7.11. Recall that the *Post Correspondence Problem* asks, given two morphisms $\alpha, \beta\colon \Sigma^* \to \{1,2\}^*$, whether there is a word $w \in \Sigma^+$ such that $\alpha(w) = \beta(w)$. The standard undecidability proof [26] constructs, given a Turing machine M, morphisms α, β such that for $w \in \Sigma^*$, any common prefix of $\alpha(w)$ and $\beta(w)$ encodes a prefix of a computation history of M. Thus, if M is terminating, there is a bound on the length of common prefixes of $\alpha(w)$ and $\beta(w)$ for $w \in \Sigma^*$. For our decidable set D, there exists a fixed terminating Turing machine, so we can proceed as follows. Given a word $u \in \Gamma^*$, we apply the construction to compute in polynomial time morphisms $\alpha, \beta\colon \Sigma^* \to \{1,2\}^*$ such that

(i) $u \in D$ iff there is a $w \in \Sigma^+$ with $\alpha(w) = \beta(w)$ and
(ii) there exists $k \in \mathbb{N}$ so that for every $w \in \Sigma^*$, the words $\alpha(w)$ and $\beta(w)$ have no common prefix longer than k.

We claim that $u \in D$ if and only if $L_{\alpha,\beta}$ and E are inseparable by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$. Clearly, if $u \in D$, then the languages $L_{\alpha,\beta}$ and E intersect and cannot be separable. Suppose $u \notin D$. Then (ii) implies that $L_{\alpha,\beta}$ is included in

$$S_k = \{a^r cb^s \mid r \not\equiv s \bmod 2^{k+1}\}$$
$$\cup \{a^r cb^s \mid \min(r,s) < 2^{k+1} - 1, r \neq s\}$$

because $x, y \in \{1,2\}^*$, $|x|, |y| > k$, have a common prefix of length $> k$ iff $\nu(x) \equiv \nu(y) \bmod 2^{k+1}$. Moreover, for $x \in \{1,2\}^*$, we have $|x| \le k$ iff $\nu(x) < 2^{k+1} - 1$. Since S_k is clearly definable in $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$ and disjoint from E, this shows that $L_{\alpha,\beta}$ and E are separable by $\mathcal{B}\Sigma_1[<, \mathrm{mod}]$.

Acknowledgments

The author is grateful to Wojciech Czerwiński, Sylvain Schmitz, and Marc Zeitoun for discussions that yielded important insights.

This work is supported by a fellowship from the Fondation Sciences Mathématiques de Paris (FSMP) and partially funded by the DeLTA project (ANR-16-CE40-0007). The results were partially obtained when the author was affiliated with LSV (ENS Paris-Saclay) and supported by a fellowship within the Postdoc-Program of the German Academic Exchange Service (DAAD) and by Labex DigiCosme, Université Paris-Saclay, project VERICONISS.

[1] To be precise, this was shown for the unreversed 2-adic representation, but the reversed case follows by just reversing the images of the morphisms.

References

[1] Alfred V Aho. 1968. Indexed grammars—an extension of context-free grammars. *J. ACM* 15, 4 (1968), 647–671.
[2] Ahmed Bouajjani, Javier Esparza, and Tayssir Touili. 2003. A generic approach to the static analysis of concurrent programs with procedures. *Int. J. Found. Comput. S.* 14, 04 (2003), 551–582.
[3] Walter Bucher, Andrzej Ehrenfeucht, and David Haussler. 1985. On total regulators generated by derivation relations. *Theoretical Computer Science* 40 (1985), 131–148.
[4] Sagar Chaki, Edmund M. Clarke, Nicholas A. Kidd, Thomas W. Reps, and Tayssir Touili. 2006. *Verifying Concurrent Message-Passing C Programs with Recursive Calls.* Springer-Verlag, Berlin Heidelberg, 334–349.
[5] Laura Chaubard, Jean Éric Pin, and Howard Straubing. 2006. First Order Formulas with Modular Predicates. In *LICS 2006.* 211–220.
[6] Lorenzo Clemente, Paweł Parys, Sylvain Salvati, and Igor Walukiewicz. 2016. The Diagonal Problem for Higher-Order Recursion Schemes is Decidable. In *LICS 2016.* ACM, New York, NY, USA, 96–105.
[7] Bruno Courcelle. 1991. On constructing obstruction sets of words. *Bulletin of the EATCS* 44 (1991), 178–186.
[8] Wojciech Czerwinski and Slawomir Lasota. 2017. Regular separability of one counter automata. In *LICS 2017.* 1–12.
[9] Wojciech Czerwiński, Wim Martens, Lorijn van Rooijen, and Marc Zeitoun. 2015. A Note on Decidable Separability by Piecewise Testable Languages. In *FCT 2015.* Springer-Verlag, Berlin Heidelberg, 173–185.
[10] Wojciech Czerwiński, Wim Martens, Lorijn van Rooijen, Marc Zeitoun, and Georg Zetzsche. 2017. A Characterization for Decidable Separability by Piecewise Testable Languages. (2017). To appear in Discrete Mathematics & Theoretical Computer Science.
[11] Andrzej Ehrenfeucht, David Haussler, and Grzegorz Rozenberg. 1983. On regularity of context-free languages. *Theor. Comput. Sci.* 27, 3 (1983), 311–332.
[12] Alain Finkel and Jean Goubault-Larrecq. 2009. Forward Analysis for WSTS, Part I: Completions. In *STACS 2009*, Vol. 3. 433–444.
[13] Jean Goubault-Larrecq and Sylvain Schmitz. 2016. Deciding Piecewise Testable Separability for Regular Tree Languages. In *ICALP 2016.*
[14] Peter Habermehl, Roland Meyer, and Harro Wimmel. 2010. The Downward-Closure of Petri Net Languages. In *ICALP 2010.*
[15] Matthew Hague, Jonathan Kochems, and C.-H. Luke Ong. 2016. Unboundedness and Downward Closures of Higher-order Pushdown Automata. In *POPL 2016.* ACM, New York, NY, USA, 151–163.
[16] Leonard H. Haines. 1969. On free monoids partially ordered by embedding. *Journal of Combinatorial Theory* 6, 1 (1969), 94–98.
[17] Graham Higman. 1952. Ordering by divisibility in abstract algebras. *P. Lond. Math. Soc.* 2 (1952), 326–336.
[18] Harry B. Hunt III. 1982. On the decidability of grammar problems. *J. ACM* 29, 2 (1982), 429–447.
[19] Pierre Jullien. 1969. *Contribution à létude des types d'ordres dispersés.* Ph.D. Dissertation. Université de Marseille.
[20] Joseph B. Kruskal. 1972. The theory of well-quasi-ordering: A frequently discovered concept. *J. Comb. Theory A* 13, 3 (1972), 297–305.
[21] Jérôme Leroux and Sylvain Schmitz. 2015. Demystifying Reachability in Vector Addition Systems. In *LICS 2015.* 56–67.
[22] Zhenyue Long, Georgel Calin, Rupak Majumdar, and Roland Meyer. 2012. Language-Theoretic Abstraction Refinement. In *FASE 2012 (Lecture Notes in Computer Science)*, Vol. 7212. Springer-Verlag, 362–376.
[23] A. N. Maslov. 1976. Multilevel stack automata. *Problems of Information Transmission* 12, 1 (1976), 38–42.
[24] Thomas Place, Lorijn van Rooijen, and Marc Zeitoun. 2013. Separating Regular Languages by Locally Testable and Locally Threshold Testable Languages. In *FSTTCS 2013*, Vol. 24. 363–375.
[25] Imre Simon. 1975. Piecewise Testable Events. In *Proceedings of the 2nd GI Conference on Automata Theory and Formal Languages.* Springer-Verlag, Berlin Heidelberg, 214–222.
[26] Michael Sipser. 2013. *Introduction to the Theory of Computation.* Cengage Learning.
[27] Thomas G. Szymanski and John H. Williams. 1976. Noncanonical extensions of bottom-up parsing techniques. *SIAM J. Comput.* 5, 2 (1976), 231–250.
[28] Jan van Leeuwen. 1978. Effective constructions in well-partially-ordered free monoids. *Discrete Mathematics* 21, 3 (1978), 237–252.
[29] Georg Zetzsche. 2015. An Approach to Computing Downward Closures. In *ICALP 2015.* Full version available at http://arxiv.org/abs/1503.01068.
[30] Georg Zetzsche. 2015. Computing downward closures for stacked counter automata. In *STACS 2015*, Vol. 30. 743–756.

A Proof of Lemma 5.1

Proof. Let $\mathcal{A} = (Q, \Sigma, C, E, q_0, F)$. We regard C as an alphabet. Consider the transducer $T = (Q, \Sigma, C, E', q_0, F)$, where E' is obtained by adding, for each edge $(q, x, \mu, q') \in E$, an edge (q, x, u, q'), where $u \in C^*$ is a word with $|u|_c = \mu(c)$ for each $c \in C$. Then by definition, $\mathcal{A}$ is unbounded on L if and only if for each $n \in \mathbb{N}$, there is a $w \in TL$ with $|w|_c \geq n$ for each $c \in C$. The latter is an instance of the *diagonal problem* [9, 10], which, given a language $K \subseteq \Sigma^*$, asks whether for every $n \in \mathbb{N}$, there is a $w \in K$ with $|w|_a \geq n$ for all $a \in \Sigma$. As mentioned in [29], for full trios, decidability of the SUP implies decidability of the diagonal problem, because the former implies computability of downward closures (with respect to the subword ordering). ◻

B Proof of Proposition 5.3

The following was shown in [21]. We include a proof for convenience.

Lemma B.1. *Let $(X, \leq)$ be a WQO and $I_1, \ldots, I_n$ be ideals such that $L \subseteq I_1 \cup \cdots \cup I_n$ and $I_i \nsubseteq I_j$ for $i \neq j$. Then $I_i \subseteq {\downarrow}L$ if and only if $I_i \in \mathsf{Adh}(L)$.*

Proof. Clearly, $I_i \in \mathsf{Adh}(L)$ implies $I_i \subseteq {\downarrow}L$. Conversely, suppose $I_1 \subseteq {\downarrow}L$ and $I_1 \notin \mathsf{Adh}(L)$. Then there is an $x \in I_1$ with $x \notin {\downarrow}(L \cap I_1)$, which means $x \in I_2 \cup \cdots \cup I_n$. We claim that then $I_1 \subseteq I_2 \cup \cdots \cup I_n$. Let $y \in I_1$. There is a $z \in I_1$ with $x \leq z$ and $y \leq z$. Since $x \leq z$, we have $z \notin {\downarrow}(L \cap I_1)$ and hence $z \in L_2 \cup \cdots \cup L_n$, which implies $y \in L_2 \cup \cdots \cup L_n$. This means $I_1 \subseteq I_2 \cup \cdots \cup I_n$ and since $I_1, \ldots, I_n$ are ideals, we have $I_1 \subseteq I_j$ for some $j \in [2, n]$, contrary to our assumption. ◻

Proof of Proposition 5.3. Given L in $\mathcal{C}$, we enumerate $\leq$-downward closed languages. Since every downward closed set is a finite union of ideals, we enumerate finite unions $I_1 \cup \cdots \cup I_n$ of $\leq$-ideals $I_1, \ldots, I_n$, which is possible because the set of ideals is a recursively enumerable set of regular languages. Clearly, we only need to enumerate unions where for any $i, j \in [1, n]$ with $i \neq j$, we have $I_i \nsubseteq I_j$.

It remains to check whether ${\downarrow_{\leq}}L = I_1 \cup \cdots \cup I_n$. Note that ${\downarrow_{\leq}}L \subseteq I_1 \cup \cdots \cup I_n$ if and only if $L \subseteq I_1 \cup \cdots \cup I_n$, so that we can check whether $L \cap (\Sigma^* \setminus (I_1 \cup \cdots \cup I_n)) = \emptyset$. The latter is decidable because the decidability of the SUP implies the decidability of the emptiness problem and $\mathcal{C}$ is effectively closed under intersection with regular languages.

The other inclusion is more interesting. Suppose we have already established ${\downarrow_{\leq}}L \subseteq I_1 \cup \cdots \cup I_n$. Then, according to Lemma B.1, we have $I_i \subseteq {\downarrow_{\leq}}L$ if and only if $I_i \in \mathsf{Adh}_{\leq}(L)$. We can therefore apply Proposition 5.2 to check whether the latter holds. ◻

C Proof of Proposition 5.5

Proof. Suppose we are given languages K and L. We decide separability by combining two semi-algorithms. One enumerates $\leq$-PTL and for each such language R, decides whether $K \subseteq R$ and $L \cap R = \emptyset$. If such an R is found, the languages are reported separable. The other semi-algorithm enumerates ideals I of $(\Sigma^*, \leq)$ and then, via Proposition 5.2, decides whether $I \in \mathsf{Adh}_{\leq}(K)$ and $I \in \mathsf{Adh}_{\leq}(L)$. If such an ideal I is found, the languages are reported inseparable. The correctness and termination of this algorithm is guaranteed by Proposition 5.4. ◻

D Proof of Proposition 6.1

Proof. Of course, for every $w \in \Sigma^*$, ${\uparrow_{\preccurlyeq}} w$ is effectively regular. Moreover, it is well-known that the ideals of $(\Sigma^*, \preccurlyeq)$ are exactly the languages of the form $\{a_0, \varepsilon\}\Gamma_1^*\{a_1, \varepsilon\} \cdots \Gamma_n^*\{a_n, \varepsilon\}$, where $a_0, \ldots, a_n \in \Sigma$ and $\Gamma_1, \ldots, \Gamma_n \subseteq \Sigma$ [19]. Lastly, if $I = \{a_0, \varepsilon\}\Gamma_1^*\{a_1, \varepsilon\} \cdots \Gamma_n^*\{a_n, \varepsilon\}$, we build $\mathcal{A}_I$ as follows. For each $i \in [1, n]$, choose a word $w_i \in \Gamma_i^*$ that contains each letter of Γ_i exactly once. Then, it is easy to construct $\mathcal{A}_I$ so that $\bar{\mathcal{A}}_I(w) \geq k$ if and only if $w \in I$ and $a_0 w_1^k a_1 \cdots w_n^k a_n \preccurlyeq w$. Then clearly $\mathcal{A}_I$ is unbounded on L if and only if we have $I \subseteq {\downarrow_{\preccurlyeq}}(L \cap I)$. The latter is equivalent to $I \in \mathsf{Adh}_{\preccurlyeq}(L)$. ◻

E Proof of Proposition 6.4

Proof. First, for every $w \in \Sigma^*$, we have ${\uparrow_{\leq_f}} w = f^{-1}({\uparrow_{\leq}} f(w))$, which is effectively regular because ${\uparrow_{\leq}} f(w)$ is.

Second, Lemma 6.2 tells us that the ideals of $(\Sigma^*, \leq_f)$ are precisely the sets of the form $f^{-1}(I)$ where $I \subseteq \Gamma^*$ is an ideal of $(\Gamma^*, \leq)$ and for which ${\downarrow_{\leq}} f(f^{-1}(I)) = I$. Therefore, the set of ideals of $(\Sigma^*, \leq_f)$ is recursively enumerable: Enumerate the ideals I of $(\Gamma^*, \leq)$ and check whether ${\downarrow_{\leq}} f(f^{-1}(I)) = I$. The latter is possible because $f(f^{-1}(I)) \subseteq \Gamma^*$ is effectively regular (regular languages are closed under rational transductions) and because for the EWUR $(\Gamma^*, \leq)$, we can effectively compute a finite automaton for the downward closure ${\downarrow_{\leq}} f(f^{-1}(I))$: The regular languages constitute a full trio with decidable SUP. Thus, we can compare the regular languages ${\downarrow_{\leq}} f(f^{-1}(I))$ and I.

Third, given an ideal $J \subseteq \Sigma^*$ (represented as a finite automaton), we can find an ideal $I \subseteq \Gamma^*$ with $J = f^{-1}(I)$. Since $(\Gamma^*, \leq)$ is an EWUR, we can compute a counter automaton $\mathcal{A}_I$ such that $\mathcal{A}_I$ is unbounded on a language $L \subseteq \Gamma^*$ if and only if $I \in \mathsf{Adh}_{\leq}(L)$. According to Lemma 6.3, we know that $J \in \mathsf{Adh}_{\leq_f}(K)$ if and only if $I \in \mathsf{Adh}_{\leq}(f(K))$, which in turn is equivalent to $\mathcal{A}_I$ being unbounded on $f(K)$. We can thus construct $\mathcal{A}_J$ as a product of $\mathcal{A}_I$ and the transducer for f so that $\mathcal{A}_J(w) = \mathcal{A}_I(f(w))$ for every $w \in \Sigma^*$. Clearly, $\mathcal{A}_J$ is unbounded on K if and only if $\mathcal{A}_I$ is unbounded on $f(K)$. ◻

F Proof of Proposition 6.5

Proof. Let $I \subseteq X$ be an ideal of $(X, \leq)$. Then I is directed with respect to $\leq_s$ for each $s \in S$. Hence, $I_s = {\downarrow_{\leq_s}} I$ is an ideal for each $s \in S$. We claim that $I = \bigcap_{s \in S} I_s$. Clearly, $I \subseteq {\downarrow_{\leq_s}} I = I_s$, hence $I \subseteq \bigcap_{s \in S} I_s$. On the other hand, if $x \in \bigcap_{s \in S} I_s$, then for each $s \in S$, there is a $x_s \in I$ with $x \leq_s x_s$. Since I is directed, we find a $y \in I$ with $x_s \leq y$ for each $s \in S$. Hence, in particular $x \leq_s y$. This implies $x \leq y$ and thus $x \in I$. This proves $I = \bigcap_{s \in S} I_s$. Finally, as a $\leq$-directed set, I itself witnesses that $(I_s)_{s \in S}$ belongs to $\mathsf{Adh}_S(I)$.

Conversely, suppose $I = \bigcap_{s \in S} I_s$ and that $(I_s)_{s \in S}$ belongs to $\mathsf{Adh}_S(I)$. The latter means that there is a $\leq$-directed set $D \subseteq I$ such that for each $s \in S$, we have $I_s = {\downarrow_{\leq_s}} D$. We claim that $I = {\downarrow_{\leq}} D$. If $x \in I$, then for each $s \in S$, there is an $x_s \in D$ with $x \leq_s x_s$. Since S is finite and D is $\leq$-directed, we find a $y \in D$ with $x_s \leq y$ for all $s \in S$. Then for each $s \in S$, we have $x \leq_s x_s \leq_s y$ and thus $x \leq y$. Hence, $I \subseteq {\downarrow_{\leq}} D$. On the other hand, if $x \leq y$ for $y \in D$, then clearly $x \leq_s y$ for each $s \in S$ and thus $x \in \bigcap_{s \in S} I_s = I$. ◻

G Proof of Proposition 6.6

Proof. Let $D \subseteq I$ be a $\preceq$-directed set with $I_s = \downarrow_{\preceq_s} D$ for every $s \in S$. Suppose $I \in \mathsf{Adh}_{\preceq}(L)$. Then there is a $\preceq$-directed set $D' \subseteq L$ with $I = \downarrow_{\preceq} D'$. We claim that $I_s = \downarrow_{\preceq_s} D'$. For $x \in I_s$, there is a $y \in D$ with $x \preceq_s y$. Since $y \in I$, there is a $z \in D'$ with $y \preceq z$. In particular, we have $x \preceq_s z \in D'$. This proves "$\subseteq$". On the other hand, we know $D' \subseteq I \subseteq I_s$, which implies $\downarrow_{\preceq_s} D' \subseteq I_s$, since I_s is $\preceq_s$-downard closed.

Conversely, suppose that $(I_s)_{s\in S}$ belongs to $\mathsf{Adh}_S(I)$ with a directed set $D' \subseteq L \cap I$ such that $I_s = \downarrow_{\preceq_s} D'$. We claim that $I = \downarrow_{\preceq} D'$. Of course, we have the inclusion "$\supseteq$" because $D' \subseteq I$, so assume $x \in I$. Since $I_s = \downarrow_{\preceq_s} D'$ and $I = \bigcap_{s\in S} I_s$, for each $s \in S$, there is a $y_s \in D'$ with $x \preceq_s y_s$. The $\preceq$-directedness of D' yields a $y \in D'$ with $y_s \preceq y$ for every $s \in S$. Then in particular $x \preceq y$ and hence $x \in \downarrow_{\preceq} D'$. ◻

H Proof of Lemma 6.7

Proof. Let $\mathcal{A}_i = (Q_i, \Sigma, C_i, E_i, q_0^i, F_i)$ be a counter automaton that characterizes adherence membership of I_i with respect to $\preceq_i$ for $i = 1, 2$. We construct a product automaton $\mathcal{A}$ so that $\mathcal{A}$ has states $Q_1 \times Q_2$, counters $C_1 \cup C_2$, and satisfies $(q_0^1, q_0^2, \varepsilon, 0) \xrightarrow{*}_{\mathcal{A}} (q^1, q^2, w, \mu)$ if and only if $(q_0^i, \varepsilon, 0) \xrightarrow{*} (q^i, w, \mu|_{C_i})$ for $i = 1, 2$. Moreover, $\mathcal{A}$ has final states $F_1 \times F_2$.

We claim that $\mathcal{A}$ is unbounded on L if and only if (I_1, I_2) belongs to $\mathsf{Adh}_{\preceq_1,\preceq_2}(L)$. We will use the fact that when a counter automaton $\mathcal{B}$ is unbounded on $K \cup L$, then it is unbounded on K or on L. Suppose $\mathcal{A}$ is unbounded on L. By construction, unboundedness of $\mathcal{A}$ implies unboundedness of $\mathcal{A}_1$ and of $\mathcal{A}_2$. Therefore, $\mathcal{A}$ must be unbounded on $L \cap I_1$: Otherwise, $\mathcal{A}$, and thus $\mathcal{A}_1$, would be unbounded on $L \setminus I_1$, which is impossible by definition of $\mathcal{A}_1$. By the same argument, $\mathcal{A}$ must be unbounded on $L \cap I_1 \cap I_2$. Then, $\mathcal{A}$ is also unbounded on some sequence $w_1, w_2, \ldots \in L \cap I_1 \cap I_2$ and since $\preceq$ is a WQO, we may assume that this sequence is a $\preceq$-chain. Therefore, the $\preceq$-directed set $D = \{w_i \mid i \geq 1\}$ satisfies $D \subseteq I_1 \cap I_2$ and $I_i \subseteq \downarrow_{\preceq_i} D$ for $i = 1, 2$. This proves $(I_1, I_2) \in \mathsf{Adh}_{\preceq_1,\preceq_2}(L)$.

Conversely, suppose $(I_1, I_2) \in \mathsf{Adh}_{\preceq_1,\preceq_2}(L)$. Then there is a $\preceq$-directed set $D \subseteq L$ with $I_i = \downarrow_{\preceq_i} D$. This implies that $\mathcal{A}_1$ and $\mathcal{A}_2$ are unbounded on D. Hence, there are sequences $u_1, u_2, \ldots \in D$ and $v_1, v_2, \ldots \in D$ such that $\mathcal{A}_1$ is unbounded on $u_1, u_2, \ldots$ and $\mathcal{A}_2$ is unbounded on $v_1, v_2, \ldots$. Thus, we have $I_1 \subseteq \downarrow_{\preceq_1}\{u_i \mid i \geq 1\}$ and $I_2 \subseteq \downarrow_{\preceq_2}\{v_i \mid i \geq 1\}$. Since D is $\preceq$-directed, we can successively find elements $w_1, w_2, \ldots \in D$ such that $u_i \preceq w_i$ and $v_i \preceq w_i$ and $w_i \preceq w_{i+1}$. Then we have $I_i \subseteq \downarrow_{\preceq_i}\{w_k \mid k \geq 1\}$ for $i = 1, 2$ and since $D \subseteq I_1 \cap I_2$, we have $\downarrow_{\preceq_i}\{w_k \mid k \geq 1\} = I_i$.

Hence, $\mathcal{A}_1$ and $\mathcal{A}_2$ are both unbounded on $w_1, w_2, \ldots$. We can therefore pick a subsequence $w_1', w_2', \ldots$ such that $\bar{\mathcal{A}}_1(w_k') \geq k$ for $k \geq 1$. As an infinite subsequence of $w_1, w_2, \ldots$, this sequence will still satisfy $\downarrow_{\preceq_2}\{w_k' \mid k \geq 1\} = I_2$ and in particular, $\mathcal{A}_2$ is unbounded on $w_1', w_2', \ldots$. We can therefore find another subsequence $w_1'', w_2'', \ldots$ such that $\bar{\mathcal{A}}_i(w_k'') \geq k$ for every $k \geq 1$ and $i \in \{1, 2\}$. Thus, $\mathcal{A}$ is unbounded on $w_1'', w_2'', \ldots$ and hence on L. ◻

I Proof of Proposition 6.8

Proof. Let $\preceq$ be the conjunction of $\preceq_1$ and $\preceq_2$. First, for $w \in \Sigma^*$, we have $\uparrow_{\preceq} w = \uparrow_{\preceq_1} w \cap \uparrow_{\preceq_2} w$, so that $\uparrow_{\preceq} w$ inherits effective regularity from $\uparrow_{\preceq_1} w$ and $\uparrow_{\preceq_1} w$.

According to Proposition 6.5, we can represent an ideal I of $\preceq$ by a pair (I_1, I_2) such that I_i is an ideal for $\preceq_i$, $I = I_1 \cap I_2$, and $(I_1, I_2) \in \mathsf{Adh}_{\preceq_1,\preceq_2}(I)$. Hence, in order to show that the set of ideals of $\preceq$ is a recursively enumerable set of regular languages, we need to show that it is decidable whether $(I_1, I_2) \in \mathsf{Adh}_{\preceq_1,\preceq_2}(I)$. To this end, we use Lemma 6.7 to construct a counter automaton $\mathcal{A}$ that is unbounded on L if and only if $(I_1, I_2) \in \mathsf{Adh}_{\preceq_1,\preceq_2}(L)$. Since $I = I_1 \cap I_2$ is effectively regular, we can decide whether $\mathcal{A}$ is unbounded on I using Lemma 5.1. ◻

J Proof of Theorem 6.9

Note that every unambiguous automaton $\mathcal{A}$ defines an ordering $\preceq_{\mathcal{A}}$ on $L(\mathcal{A})$ in the same way labeling automata define an ordering on Σ^*. We will now also use $\preceq_{\mathcal{A}}$ to denote this order. We say that an unambiguous automaton $\mathcal{B}$ is a *subautomaton* of $\mathcal{A}$ if $\mathcal{B}$ is obtained from $\mathcal{A}$ by deleting some edges. The following can be shown, roughly speaking, by decomposing $\mathcal{B}$ into strongly connected components and dividing $L(\mathcal{B})$ according to which path through the resulting graph a word takes.

Lemma J.1. *For a subautomaton $\mathcal{B}$ of an unambiguous automaton $\mathcal{A}$, $L(\mathcal{B})$ is a finite union of sets of the form*

$$\downarrow_{\preceq_{\mathcal{A}}} u_0 v_1^* u_1 \cdots v_n^* u_n,$$

where $u_0 v_1 u_1 \cdots v_n u_n$ is a loop pattern for $\mathcal{A}$.

Proof. We decompose $\mathcal{B}$ into its directed acyclic graph G of strongly connected components and notice that this graph has only finitely many paths. Moreover, for each strongly connected component C and and states p and q in C, there are only finitely many simple paths from p to q. Every run through C from p to q can be reduced to one of these simple paths by deleting loops. Therefore, we can divide the set $L(\mathcal{B})$ according to which paths in G they a word follows and to which simple paths in each component it reduces. This yields a decomposition of $L(\mathcal{B})$ as a finite union of sets of the form $u_0 L_1 u_1 \cdots L_n u_n$ such that there are states $q_0, \ldots, q_n$ so that

- q_0 is initial and q_n is final,
- for $i \in [0, n]$, either (q_i, u_i, q_{i+1}) is an edge in $\mathcal{B}$, or $u_i = \varepsilon$ and $q_{i+1} = q_i$,
- for $i \in [1, n]$, L_i is the set of words read on a cycle from q_i to q_i.

For each $i \in [1, n]$, consider the strongly connected component of $\mathcal{B}$ that contains q_i and let E_i be the set of edges of $\mathcal{B}$ in this component.

There exists a word $v_i \in L_i$ whose run from q_i to q_i (note that there is at most one such run because $\mathcal{A}$ is a labeling automaton) uses every edge from E_i at least once: For each $e \in E_i$, take a run from q_i to q_i that uses e. Then take v_i to be the word read on the concatenation of all these runs.

We claim that $u_0 L_1 u_1 \cdots L_n u_n = \downarrow_{\preceq_{\mathcal{A}}} u_0 v_1^* u_1 \cdots v_n^* u_n$. Since $u_0 L_1 u_1 \cdots L_n u_n$ is clearly downward closed with respect to $\preceq_{\mathcal{A}}$ and contains $u_0 v_1^* u_1 \cdots v_n^* u_n$, the inclusion "$\supseteq$" holds. Conversely, suppose $w_i \in L_i$ for $i \in [1, n]$. Consider a particular $i \in [1, n]$ and let $r = e_1 \cdots e_k \in E^*$ be the run of $\mathcal{B}$ when reading w_i from q_i to q_i. Each e_j occurs in the run $s \in E^*$ of v_i, so that the run s^k of v_i^k contains $e_1 \cdots e_k$ as a subsequence and we can write $s^k = t_0 e_1 t_1 \cdots e_k t_k$ for some $t_0, \ldots, t_k \in E^*$. Since e_i ends in the state where e_{i+1} starts and r and s^k are both cycles from q_i to q_i, every run t_i is a cycle.

This implies that $u_0w_1u_1\cdots w_nu_n \preceq_{\mathcal{A}} u_0v_1^{|w_1|}u_1\cdots v_n^{|w_n|}u_n$. This proves the inclusion "$\subseteq$". □

We shall prove that the ideals of $(\Sigma^*, \preceq_{\mathcal{A}})$ are precisely those sets of the form $\downarrow_{\preceq_{\mathcal{A}}} u_0v_1^*u_1\cdots v_n^*u_n$. The first step in proving that is to show that every downward closed language is a finite union of such sets. Here, we will use the fact that ideals of the subword ordering are precisely the languages $\{a_0,\varepsilon\}\Gamma_1^*\{a_1,\varepsilon\}\cdots\Gamma_n^*\{a_n,\varepsilon\}$, where $a_0,\ldots,a_n\in\Sigma$ and $\Gamma_1,\ldots,\Gamma_n\subseteq\Sigma$ [19].

Proposition J.2. *Let $\mathcal{A}$ be a labeling automaton and $L\subseteq\Sigma^*$. The set $\downarrow_{\preceq_{\mathcal{A}}}L$ is a finite union of sets of the form*

$$\downarrow_{\preceq_{\mathcal{A}}} u_0v_1^*u_1\cdots v_n^*u_n,$$

where $u_0v_1u_1\cdots v_nu_n$ is a loop pattern for $\mathcal{A}$.

Proof. Let $\mathcal{A}=(Q,\Sigma,E,I,F)$. For each $p,q\in Q$, we define $K_{p,q}=\{w\in L\mid \sigma_{\mathcal{A}}(w)=(p,q)\}$. Then we have

$$\downarrow_{\preceq_{\mathcal{A}}}L=\bigcup_{p,q\in Q}\downarrow_{\preceq_{\mathcal{A}}}K_{p,q}.$$

Therefore, it suffices to consider the case that there are fixed $p,q\in Q$ such that for every $u,v\in L$, we have $\sigma_{\mathcal{A}}(u)=(p,q)$. Note that then $u\preceq_{\mathcal{A}} v$ if and only if $\mathcal{A}(u)\preccurlyeq\mathcal{A}(v)$ for $u,v\in L$. Let $\mathsf{Runs}_{p,q}(\mathcal{A})$ denote the set of all runs of $\mathcal{A}$ that start in p and end in q. Let $\pi\colon E^*\to\Sigma^*$ be the projection onto labels of edges. Observe that $\downarrow_{\preceq_{\mathcal{A}}}L=\pi((\downarrow\mathcal{A}(L))\cap\mathsf{Runs}_{p,q}(\mathcal{A}))$. (Here, $\downarrow\mathcal{A}(L)$ denotes the downward closure with respect to the subword ordering.)

Observe that the language $\downarrow\mathcal{A}(L)$ is a finite union of sets of the form $e_0E_1^*e_1\cdots E_n^*e_n$, where $E_i\subseteq E$ and $e_i\in E\cup\{\varepsilon\}$. Hence, we would like to prove the proposition for sets of the form

$$\pi(e_0E_1^*e_1\cdots E_n^*e_n\cap\mathsf{Runs}_{p,q}(\mathcal{A})).$$

However, these are not necessarily downward closed. Therefore, we prove that

$$\downarrow_{\preceq_{\mathcal{A}}}\pi(e_0E_1^*e_1\cdots E_n^*e_n\cap\mathsf{Runs}_{p,q}(\mathcal{A}))$$

can be written as a finite union of sets $\downarrow_{\preceq_{\mathcal{A}}}u_0v_1^*u_1\cdots v_n^*u_n$.

The set $e_0E_1^*e_1\cdots E_n^*e_n\cap\mathsf{Runs}_{p,q}(\mathcal{A})$ is a finite union of sets of the form $e_0S_1e_1\cdots S_ne_n$ such that there are states $q_0,\ldots,q_{n+1}$ so that

- for $i\in[0,n]$, either $e_i=\varepsilon$ and $q_{i+1}=q_i$, or e_i is an edge from q_i to q_{i+1},
- for $i\in[1,n]$, $S_i\subseteq E_i^*$ is the set of runs of $\mathcal{A}$ from q_i to q_{i+1} that only use edges in E_i.

Therefore, it suffices to show that $\downarrow_{\preceq_{\mathcal{A}}}\pi(e_0S_1e_1\cdots S_ne_n)$ can be written as a finite union as desired. Let $\mathcal{A}_i$ be the unambiguous automaton obtained from $\mathcal{A}$ by making q_i the only initial state and q_{i+1} the only final state. Moreover, let $\mathcal{B}_i$ be obtained from $\mathcal{A}_i$ be removing all edges outside of E_i. Then, we have have $\pi(S_i)=L(\mathcal{B}_i)$. According to Lemma J.1, $S_i=L(\mathcal{B}_i)$ is a finite union of sets of the form $\downarrow_{\preceq_{\mathcal{A}_i}}u_0v_1^*u_1\cdots v_k^*u_k$, where $u_0v_1u_1\cdots v_ku_k$ is a loop pattern for $\mathcal{A}_i$. Therefore, our set $\downarrow_{\preceq_{\mathcal{A}}}\pi(e_0S_1e_1\cdots S_ne_n)$ is a finite union of sets of the form

$$\downarrow_{\preceq_{\mathcal{A}}}\left(\pi(e_0)(\downarrow_{\mathcal{A}_1}I_1)\pi(e_1)\cdots(\downarrow_{\preceq_{\mathcal{A}_n}}I_n)\pi(e_n)\right), \qquad (2)$$

where $I_i=u_{i,0}v_{i,1}^*u_{i,1}\cdots v_{i,k_i}^*u_{i,k_i}$ for $i\in[1,n]$. The definition of $\preceq_{\mathcal{A}}$ implies immediately that eq. (2) equals

$$\downarrow_{\preceq_{\mathcal{A}}}\big(\pi(e_0)\big(u_{1,0}v_{1,1}^*u_{1,1}\cdots v_{1,k_1}^*u_{1,k_1}\big)\pi(e_1)\cdots\big(u_{n,0}v_{n,1}^*u_{n,1}\cdots v_{n,k_n}^*u_{n,k_n}\big)\pi(e_n)\big).$$

Moreover,

$$\pi(e_0)u_{1,0}v_{1,1}u_{1,1}\cdots v_{1,k_1}u_{1,k_1}\pi(e_1)\cdots u_{n,0}v_{n,1}u_{n,1}\cdots v_{n,k_n}u_{n,k_n}\pi(e_n) \qquad (3)$$

is clearly a loop pattern for $\mathcal{A}$ (where the $v_{i,j}$ play the role of the v_i). □

We are now ready to prove Theorem 6.9.

Proof of Theorem 6.9. Let us show that the language

$$I=\downarrow_{\preceq_{\mathcal{A}}}u_0v_1^*u_1\cdots v_n^*u_n$$

is in fact an $\preceq_{\mathcal{A}}$-ideal. It is clearly $\preceq_{\mathcal{A}}$-downward closed. Consider the word $w_k=u_0v_1^ku_1\cdots v_n^ku_n$ for each $k\in\mathbb{N}$. Then we have $w_0\preceq_{\mathcal{A}}w_1\preceq_{\mathcal{A}}\cdots$, so that the set $D=\{w_k\mid k\in\mathbb{N}\}$ is $\preceq_{\mathcal{A}}$-directed. Moreover, $I=\downarrow_{\preceq_{\mathcal{A}}}D$, which proves that I is the $\preceq_{\mathcal{A}}$-downward closure of a $\preceq_{\mathcal{A}}$-directed set and hence an $\preceq_{\mathcal{A}}$-ideal.

It remains to be shown that every ideal is of the above form. Let I be an ideal of $\preceq_{\mathcal{A}}$. In Proposition J.2 we have seen that every downward closed is a finite union of sets of the above form. In particular, we can write $I=I_1\cup\cdots\cup I_k$, where each I_k is of the above form. However, since I is an ideal and the I_i are downward closed, this implies that for some $i\in[1,n]$, we have $I\subseteq I_i$ and thus $I=I_i$. □

K Proofs for section 7

K.1 Auxiliary lemmas, including Lemma 7.3

Lemma K.1. *Suppose $v\in(\Sigma^d)^*$. Then $\downarrow_{\preceq_d}v^*=\{w\in(\Sigma^d)^*\mid \kappa_d(w)\subseteq\kappa_d(v)\}$.*

Proof. Let $w\in\downarrow_{\preceq_d}v^*$, say $w\preceq_d v^k$. Then clearly $w\in(\Sigma^d)^*$. Moreover, if $a\in\Sigma$ occurs at a position p in w with $p\equiv i \pmod d$, then a occurs at some position $p+d\mathbb{N}$ in v. Hence, $\kappa_d(w)\subseteq\kappa_d(v)$.

Suppose $w\in(\Sigma^d)^*$ and $\kappa_d(w)\subseteq\kappa_d(v)$. Write $w=a_1\cdots a_n$, $a_1,\ldots,a_n\in\Sigma$. Since $a_i\in\kappa_d(w)(i)\subseteq\kappa_d(v)(i)$, each a_i occurs at some position p in v with $p\equiv i \pmod d$. Hence, we can write $v=x_ia_iy_i$ with $|x_i|\equiv i-1 \pmod d$ and therefore $|y_i|\equiv|v|-|x_i|-1\equiv d-i \pmod d$. In particular, $|y_ix_{i+1}|\equiv(d-i)+i=d$. Moreover, $|x_1|\equiv 0 \mod d$ and $y_n\equiv d-n\equiv 0 \pmod d$. Therefore,

$$w=a_1\cdots a_n\preceq_d \overline{x_1}a_1\overline{y_1x_2}a_2\overline{y_2x_3}\cdots\overline{y_{n-1}x_n}a_n\overline{y_n}=v^n$$

where $\overline{u}$ expresses that $u\in(\Sigma^d)^*$. Thus $w\in\downarrow_{\preceq_d}v^*$. □

Proof of Lemma 7.3. If $\downarrow_{\preceq_d}v^*\subseteq\downarrow_{\preceq_d}w^*$, then in particular $v\in\downarrow_{\preceq_d}w^*$ and thus $\kappa_d(v)\subseteq\kappa_d(w)$ by Lemma K.1.

Suppose $\kappa_d(v)\subseteq\kappa_d(w)$. Since $v\in(\Sigma^d)^*$, we have $\kappa_d(v^n)=\kappa_d(v)$ for any $n\in\mathbb{N}$ and hence $v^n\in\downarrow_{\preceq_d}w^*$ by Lemma K.1. This implies $\downarrow_{\preceq_d}v^*\subseteq\downarrow_{\preceq_d}w^*$. □

Lemma K.2. *If $\kappa_d(xyz)\subseteq\kappa_d(v)$ and $\pi_d(v)$ divides $|y|$, then we have $\kappa_d(xyyz)\subseteq\kappa_d(v)$.*

Proof. Let $i \in [1,d]$. We will show that $\kappa_d(xyyz)(i) \subseteq \kappa_d(v)(i)$. Hence, let $a \in \kappa_d(xyyz)(i)$. Then there is a position $p \in [1,|xyyz|]$ with $p \equiv i \pmod{d}$ such that the p-th position of $xyyz$ reads a.

If $p \in [1,|xy|]$, we are done, so assume $p \in [|xy|+1,|xyyz|]$. Then, a also occurs at position $q = p - |y|$ in xyz. This means, if $j \equiv q \pmod{d}$, then $a \in \kappa_d(xyz)(j) \subseteq \kappa_d(v)(j)$. Observe that $i \equiv p \pmod{d}$ implies $i \equiv p \pmod{\pi_d(v)}$ and thus $i \equiv p = q+|y| \equiv q \equiv j \pmod{\pi_d(v)}$. Therefore, we have $a \in \kappa_d(v)(j) = \kappa_d(v)(i)$. ◻

Lemma K.3. *Suppose $v \in (\Sigma^d)^*$. Then for every $r \in [0,d-1]$:*

$$\downarrow_{\preceq_d} v^{[r]} = \{u \in \Sigma^* \mid |u| \equiv r \bmod d,\ \kappa_d(u) \subseteq \kappa_d(v)\}.$$

Proof. Let w be the length-r prefix of v. Let $u \in \downarrow_{\preceq_d} v^{[r]}$, say $u \preceq_d v^k w$. Then clearly $|u| \equiv r \bmod d$. Moreover, if $a \in \Sigma$ occurs at a position p in u with $p \equiv i \pmod{d}$, then a occurs at some position $p + d\mathbb{N}$ in v. Hence, $\kappa_d(u) \subseteq \kappa_d(v)$.

Suppose $u \in \Sigma^*$ with $|u| \equiv r \bmod d$ and $\kappa_d(u) \subseteq \kappa_d(v)$. Write $u = a_1 \cdots a_n$, $a_1,\ldots,a_n \in \Sigma$. Since $a_i \in \kappa_d(u)(i) \subseteq \kappa_d(v)(i)$, each a_i occurs at some position p in v with $p \equiv i \bmod d$. Hence, we can write $v = x_i a_i y_i$ with $|x_i| \equiv i-1 \bmod d$ and therefore $|y_i| \equiv |v|-|x_i|-1 \equiv d-i \bmod d$. In particular, $|y_i x_{i+1}| \equiv (d-i)+i = d \bmod d$. Moreover, $|x_1| \equiv 0 \bmod d$ and $|y_n w| \equiv d-n+r \equiv 0 \bmod d$. Therefore,

$$u = a_1 \cdots a_n \preceq_d \overline{x_1} a_1 \overline{y_1 x_2} a_2 \overline{y_2 x_3} \cdots \overline{y_{n-1} x_n} a_n \overline{y_n w} = v^n w$$

where $\overline{z}$ expresses that $z \in (\Sigma^d)^*$. Thus $u \in \downarrow_{\preceq_d} v^{[r]}$. ◻

Lemma K.4. *Suppose $\pi_d(v)$ divides $|y|$ and $|y|$ divides d. If $xyz \in \downarrow_{\preceq_d} v^{[r]}$, then for every $\ell \in \mathbb{N}$, $xy^{1+\ell\cdot d/|y|}z \in \downarrow_{\preceq_d} v^{[r]}$.*

Proof. Let $w = xy^{1+\ell\cdot d/|y|}z$. Since d divides $|xyz|$, it also divides $|w| = |xyz| + (\ell \cdot d/|y|)\cdot|y|$. According to Lemma K.3, we have $\kappa_d(xyz) \subseteq \kappa_d(v)$. An $(\ell\cdot d/|y|)$-fold application of Lemma K.2 tells us that $\kappa_d(xy^{1+\ell\cdot d/|y|}z) \subseteq \kappa_d(v)$. Now, Lemma K.3 states that $xy^{1+\ell\cdot d/|y|}z \in \downarrow_{\preceq_d} v^*$. ◻

K.2 Proof of Lemma 7.5 and Lemma 7.6

Proof of Lemma 7.5. Let us call a d-embedding *(k,i)-normal* if it maps at least k-many factors v_i in x into the factor $v_i^{y_i}$ in y. To simplify notation, we will always write x and y for the words $x = u_0 v_1^{x_1} u_1 \cdots v_n^{x_n} u_n$ and $y = u_0 v_1^{y_1} u_1 \cdots v_n^{y_n} u_n$.

Suppose the contrary. Then there is a $k \in \mathbb{N}$ such that for every $\ell \in \mathbb{N}$, there are $x_1,\ldots,x_n \in \mathbb{N}$ and $y_1,\ldots,y_n \in \mathbb{N}$ with $x_i \geq \ell$ for $i \in [1,n]$ such that there is a d-embedding of x in y that is not (k,j)-normal for some $j \in [1,n]$. Among the j for which this occurs, one has to occur infinitely often. Hence, there is a $k \in \mathbb{N}$ and a $j \in [1,n]$ such that for every $\ell \in \mathbb{N}$, there are $x_1,\ldots,x_n \in \mathbb{N}$ and $y_1,\ldots,y_n \in \mathbb{N}$ with $x_i \geq \ell$ for $i \in [1,n]$ such that there is a d-embedding of x in y that is not (k,j)-normal.

If a d-embedding is not (k,j)-normal, then all but at most $(k-1)+2$ factors v_j must be mapped either to the factor $u_0 v_1^{y_1} u_1 \cdots v_{j-1}^{y_{j-1}} u_{j-1}$ or to the factor $u_{j+1} v_{j+2}^{y_{j+2}} u_{j+2} \cdots v_n^{y_n} u_n$: At most $k-1$ factors are mapped to $v_j^{y_j}$ and at most two further factors are partially mapped to $v_j^{y_j}$. Therefore, we have at least one of the following cases:

1. for each $\ell \in \mathbb{N}$, there are $x_1,\ldots,x_n$ and $y_1,\ldots,y_n$ with $x_i \geq k$ for $i \in [1,n]$ such that there is a d-embedding of x in y that maps at least ℓ factors v_j to $u_0 v_1^{y_1} u_1 \cdots v_{j-1}^{y_{j-1}} u_{j-1}$.
2. for each $\ell \in \mathbb{N}$, there are $x_1,\ldots,x_n$ and $y_1,\ldots,y_n$ with $x_i \geq k$ for $i \in [1,n]$ such that there is a d-embedding of x in y that maps at least ℓ factors v_j to $u_{j+1} v_{j+2}^{y_{j+2}} u_{j+2} \cdots v_n^{y_n} u_n$.

Let us consider the first case (the second can be treated the same way). We claim that this implies

$$\downarrow_{\preceq_d} u_0 v_1^* u_1 \cdots v_n^* u_n = \downarrow_{\preceq_d} u_0 v_1^* u_1 \cdots v_{j-1}^* u_{j-1} u_j \cdots v_n^* u_n. \quad (4)$$

The inclusion "$\supseteq$" clearly holds. For the other direction, consider $u_0 v_1^{z_1} u_1 \cdots v_n^{z_n} u_n$. Then there are $x_1,\ldots,x_n,y_1,\ldots,y_n \in \mathbb{N}$ such that $x_i \geq z_i$ and there exists a d-embedding of x into y that maps at least z_j factors v_j into $u_0 v_1^{y_1} u_1 \cdots v_{j-1}^{y_{j-1}} u_{j-1}$. This means we have

$$u_0 v_1^{z_1} u_1 \cdots v_{j-1}^{z_{j-1}} u_{j-1} v_j^{z_j} \preceq_d u_0 v_1^{y_1} u_1 \cdots v_{j-1}^{y_{j-1}} u_{j-1}$$

and hence

$$u_0 v_1^{z_1} u_1 \cdots v_n^{z_n} u_n \preceq_d u_0 v_1^{y_1} u_1 \cdots v_{j-1}^{y_{j-1}} u_{j-1} u_j v_{j+1}^{z_{j+1}} \cdots v_n^{z_n} u_n.$$

since clearly $u_j v_{j+1}^{z_{j+1}} \cdots v_n^{z_n} u_n \preceq_d u_j v_{j+1}^{z_{j+1}} \cdots v_n^{z_n} u_n$ and $\preceq_d$ is multiplicative. This implies the inclusion "$\subseteq$" of eq. (4). Finally, note that eq. (4) contradicts the assumed irreducibility. ◻

Proof of Lemma 7.6. Clearly, if a loop pattern is associated with a language, then its induced ideal belongs to the adherence of the language. Conversely, suppose the ideal $I = \downarrow_{\preceq_d} u_0 v_1^* u_1 \cdots v_n^* u_n$ belongs to $\mathsf{Adh}_{\preceq_d}(L)$. Let $k \in \mathbb{N}$ and $x_1,\ldots,x_n \geq k$ and let $\ell \in \mathbb{N}$ be the constant provided by Lemma 7.5. Without loss of generality, we may assume that $\ell \geq k$.

Since I belongs to $\mathsf{Adh}_{\preceq_d}(L)$, there is a word $w \in L$ such that $u_0 v_1^{\ell} u_1 \cdots v_n^{\ell} u_n \preceq_d w \preceq_d u_0 v_1^{y_1} u_1 \cdots v_n^{y_n} u_n$ for some $y_1,\ldots,y_n \in \mathbb{N}$. This means in particular that there is a d-embedding α of the word $u_0 v_1^{\ell} u_1 \cdots v_n^{\ell} u_n$ into w and a d-embedding β of w into the word $u_0 v_1^{y_1} u_1 \cdots v_n^{y_n} u_n$. By composing these two d-embeddings, we therefore obtain a d-embedding γ of $u_0 v_1^{\ell} u_1 \cdots v_n^{\ell} u_n$ into the word $u_0 v_1^{y_1} u_1 \cdots v_n^{y_n} u_n$. By the choice of ℓ, γ has to be k-normal. This means that γ maps at least k copies of v_i to $v_i^{y_i}$ for each $i \in [1,n]$. We can therefore decompose $w = \bar{u}_0 \bar{v}_1 \bar{u}_1 \cdots \bar{v}_n \bar{u}_n$ so that these k copies of v_i that γ maps to $v_i^{y_i}$ are mapped by α to $\bar{v}_i$ and $|\bar{v}_i|$ is divisible by d.

Since β maps $\bar{v}_i$ to $v_i^{y_i}$, we have $\bar{v}_i \in \downarrow_{\preceq_d} v_i^*$. This also implies that β maps $\bar{u}_0$ to $u_0 v_1^{y_1}$, and β maps $\bar{u}_i$ to $v_i^{y_i} u_i v_{i+1}^{y_{i+1}}$, and β maps $\bar{u}_n$ to $v_n^{y_n} u_n$. Moreover, α maps u_i to $\bar{u}_i$ for each $i \in [0,n]$. In other words, we have $v_i^k \preceq_d \bar{v}_i \in \downarrow_{\preceq_d} v_i^*$ for every $i \in [1,n]$ and $u_i \preceq_d \bar{u}_i \in \downarrow_{\preceq_d} v_i^* u_i v_{i+1}^*$ for $i \in [1,n-1]$ and $u_0 \preceq_d \bar{u}_0 \in \downarrow_{\preceq_d} u_0 v_1^*$ and $u_n \preceq_d \bar{u}_n \in \downarrow_{\preceq_d} v_n^* u_n$. Thus, I is associated to L. ◻

K.3 Proof of Lemma 7.7

Proof. Suppose I belongs to $\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_i))$ for $i = 1,2$. Let

$$u_0 v_1 u_1 \cdots v_n u_n$$

be an irreducible loop pattern for $\mathcal{M}_d$ such that we have $I = \downarrow_{\preceq_d} u_0 v_1^* u_1 \cdots v_n^* u_n$. According to Lemma 7.6, the loop pattern $u_0 v_1 u_1 \cdots v_n u_n$ is associated to $L(\mathcal{A}_i)$ for $i = 1,2$.

In particular, there is a word $\bar{u}_{i,0} \bar{v}_{i,1} \bar{u}_{i,1} \cdots \bar{v}_{i,n} \bar{u}_{i,n} \in L(\mathcal{A}_i)$ such that $v_j^m \preceq_d \bar{v}_{i,j} \in \downarrow_{\preceq_d} v_j^*$ for $j \in [1,n]$ and $i = 1,2$ and $u_j \preceq_d \bar{u}_{i,j} \in \downarrow_{\preceq_d} v_j^* u_j v_{j+1}^*$ for $j \in [1,n-1]$ and $u_0 \preceq_d \bar{u}_{i,0} \in \downarrow_{\preceq_d} u_0 v_1^*$ and $u_n \preceq_d \bar{u}_{i,n} \in \downarrow_{\preceq_d} v_n^* u_n$.

We can therefore write $\bar{v}_{i,j} = t_{i,j,1} \cdots t_{i,j,m}$ with $v_j \preceq_d t_{i,j,\ell} \in \downarrow_{\preceq_d} v_j^*$. Consider the run of $\mathcal{A}_i$ on the word

$$\bar{u}_{i,0}\bar{v}_{i,1}\bar{u}_{i,1} \cdots \bar{v}_{i,n}\bar{u}_{i,n}.$$

Since $\mathcal{A}_i$ has $\leq m$ states, for each $j \in [1,n]$, this run must occupy the same state before and after reading some infix $t_{i,j,\ell} \cdots t_{i,j,k}$. Let $q_{i,j}$ be this state and let $\bar{v}_{i,j} = x_{i,j}y_{i,j}z_{i,j}$ be the decomposition so that $y_{i,j} = t_{i,j,\ell} \cdots t_{i,j,k}$. Then we have $v_j \preceq_d y_{i,j} \in \downarrow_{\preceq_d} v_j^*$ and also $x_{i,j}, z_{i,j} \in \downarrow_{\preceq_d} v_j^*$. The former implies that $\downarrow_{\preceq_d} y_{i,j}^* = \downarrow_{\preceq_d} v_j^*$.

Let $\mathcal{A}_{i,j}$ be the automaton obtained from $\mathcal{A}_i$ by making $q_{i,j}$ the only initial and final state. Then $\mathcal{A}_{i,j}$ is cyclic and we have $y_{i,j}^* \subseteq L(\mathcal{A}_{i,j})$. In particular, the ideal $\downarrow_{\preceq_d} v_j^* = \downarrow_{\preceq_d} y_{i,j}^*$ belongs to $\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_{i,j}))$. Now Lemma 7.4 yields a $w_j \in (\Sigma^d)^*$ such that

- $\downarrow_{\preceq_d} v_j^* \subseteq \downarrow_{\preceq_d} w_j^*$,
- $\downarrow_{\preceq_d} w_j^*$ belongs to $\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_{i,j}))$,
- $\pi_d(w_j) \leq m^2$.

We claim that $u_0 w_1 u_1 \cdots w_n u_n$ is a loop pattern as desired in the lemma. It remains to show that $\downarrow_{\preceq_d} u_0 w_1^* u_1 \cdots w_n^* u_n$ belongs to $\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$.

Let $k \in \mathbb{N}$. Since $\downarrow_{\preceq_d} w_j^*$ belongs to $\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_{i,j}))$ for $i \in \{1,2\}$ and $j \in [1,n]$, there is a word $w'_{i,j} \in L(\mathcal{A}_i)$ such that $w_j^k \preceq_d w'_{i,j} \in \downarrow_{\preceq_d} w_j^*$. Define

$$t = \bar{u}_{i,0} x_{i,1} w'_{i,1} z_{i,1} \bar{u}_{i,1} \cdots x_{i,n} w'_{i,n} z_{i,n} \bar{u}_{i,n}.$$

Then we have $u_0 \bar{w}_1^k u_1 \cdots \bar{w}_n^k u_n \preceq_d t \in L(\mathcal{A}_i)$. Moreover, since $x_{i,j}, z_{i,j} \in \downarrow_{\preceq_d} v_j^* \subseteq \downarrow_{\preceq_d} \bar{w}_j^*$ and by the choice of the $\bar{u}_{i,j}$, the word t is contained in $\downarrow_{\preceq_d} u_0 \bar{w}_1^* u_1 \cdots \bar{w}_n^* u_n$. This proves that the ideal $\downarrow_{\preceq_d} u_0 \bar{w}_1^* u_1 \cdots \bar{w}_n^* u_n$ belongs to $\mathsf{Adh}_{\preceq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$ and hence completes the lemma. ◻

K.4 Proof of Lemma 7.8

Proof of Lemma 7.8. Clearly, if the ideal generated by p is associated to L, then it belongs to $\mathsf{Adh}_{\preceq_d}(L)$.

Conversely, let $p = u_0 v_1^{[r_1]} u_1 \cdots v_n^{[r_n]} u_n$ be an irreducible extended loop pattern for $\mathcal{M}_d$ and suppose its generated ideal I belongs to $\mathsf{Adh}_{\preceq_d}(L)$. Let w_i be the length-r_i prefix of v_i for $i \in [1,n]$. Since the loop pattern $u_0(v_1)w_1u_1 \cdots (v_n)w_nu_n$ (the loop parts are in brackets) is irreducible, it is associated to L according to Lemma 7.6.

Thus, for given $k \in \mathbb{N}$, we find a word

$$w = \tilde{u}_0\tilde{v}_1\tilde{u}_1 \cdots \tilde{v}_n\tilde{u}_n \in L \tag{5}$$

such that $v_i^{k+1} \preceq_d \tilde{v}_i \in \downarrow_{\preceq_d} v_i^*$ for every $i \in [1,n]$ and $w_iu_i \preceq_d \tilde{u}_i \in \downarrow_{\preceq_d} v_i^* w_i u_i v_{i+1}^*$ for $i \in [1, n-1]$ and $u_0 \preceq_d \tilde{u}_0 \in \downarrow_{\preceq_d} u_0 v_1^*$ and $w_n u_n \preceq_d \tilde{u}_n \in \downarrow_{\preceq_d} v_n^* w_n u_n$.

In the first step, we modify the decomposition eq. (5) of w by moving, for each $i \in [1,n]$, the last $d - r_i$ letters of $\tilde{v}_i$ to its right neighbor $\tilde{u}_i$. Let the resulting decomposition be

$$w = \hat{u}_0\hat{v}_1\hat{u}_1 \cdots \hat{v}_n\hat{u}_n.$$

Since $v_i^{k+1} \preceq_d \tilde{v}_i \in \downarrow_{\preceq_d} v_i^*$ and $w_iu_i \preceq_d \tilde{u}_i \in \downarrow_{\preceq_d} v_i^* w_i u_i v_{i+1}^*$, we now have

1. $v_i^k w_i \preceq_d \hat{v}_i \in \downarrow_{\preceq_d} v_i^{[r_i]}$ for each $i \in [1,n]$,
2. $u_i \preceq_d \hat{u}_i \in \downarrow_{\preceq_d} \lambda^{r_i}(v_i)^* u_i v_{i+1}^*$ for $i \in [1, n-1]$,
3. $u_0 \preceq_d \hat{u}_0 \in \downarrow_{\preceq_d} u_0 v_1^*$, and
4. $u_n \preceq_d \hat{u}_n \in \downarrow_{\preceq_d} \lambda^{r_i}(v_n)^* u_n$.

We claim that for each $i \in [0,n]$, there are words x_i, y_i so that

1. for each $i \in [1, n-1]$ for which u_i is non-empty, $\hat{u}_i = x_i u_i y_i$ with $x_i \in \downarrow_{\preceq_d} \lambda^{r_i}(v_i)^*$, $y_i \in \downarrow_{\preceq_d} v_{i+1}^*$,
2. $\hat{u}_0 = u_0 y_0$ and $y_0 \in \downarrow_{\preceq_d} v_1^*$,
3. $\hat{u}_n = x_n u_n$ and $x_0 \in \downarrow_{\preceq_d} \lambda^{r_i}(v_n)^*$.

Note that is establishes the lemma: We can then again modify the decomposition as follows. We move y_0 from $\hat{u}_0$ to $\hat{v}_1$ and we move x_n from $\hat{u}_n$ to $\hat{v}_n$. Moreover, for each non-empty u_i, we move x_i from $\hat{u}_i$ to $\hat{v}_i$ and we move y_i from $\hat{u}_i$ to $\hat{v}_{i+1}$. Each $\hat{u}_i$ where u_i is empty is left unchanged. The resulting decomposition $w = \bar{u}_0\bar{v}_1\bar{u}_1 \cdots \bar{v}_n\bar{u}_n$ is then as desired.

First, note that if some u_i is empty (whether $i \in [1, n-1]$ or $i \in \{0, n\}$), then we need not construct any x_i and y_i. We show how to construct x_i and y_i for $i \in [1, n-1]$ where u_i is non-empty. The proof for y_0 and x_n is then analogous.

Recall that $u_i \preceq_d \hat{u}_i \in \downarrow_{\preceq_d} \lambda^{r_i}(v_i)^* u_i v_{i+1}^*$. This means there is some ℓ so that $\hat{u}_i \preceq_d \lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$. Consider the d-embedding α of u_i into $\hat{u}_i$ and the d-embedding β of $\hat{u}_i$ into $\lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$. The composition γ of α and β is a d-embedding of u_i into $\lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$.

We now use the fact that our extended loop pattern is irreducible. The d-embedding γ cannot send the left-most letter of u_i to a position in $\lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$ left of u_i, because that would mean that this letter is contained in $\kappa_d(v_i)(r_i + 1)$. Moreover, γ cannot send the right-most letter of u_i to a position in $\lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$ to the right of u_i, because that would mean that this letter is contained in $\kappa_d(v_{i+1})(d)$. This implies that γ sends u_i exactly to the factor u_i of $\lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$. Thus, $\hat{u}_i$ has a factor u_i that is sent by β to u_i of $\lambda^{r_i}(v_i)^\ell u_i v_{i+1}^\ell$. Let $\hat{u}_i = x_i u_i y_i$ be the corresponding decomposition. Then β has to map x_i into $\lambda^{r_i}(v_i)^\ell$ and y_i into v_{i+1}^ℓ. In particular, we have $x_i \in \downarrow_{\preceq_d} \lambda^{r_i}(v_i)^*$ and $y_i \in \downarrow_{\preceq_d} v_{i+1}^*$. This completes the proof of the claim and hence the lemma. ◻

K.5 Proof of Lemma 7.9

Proof of Lemma 7.9. We define the *length* of an extended loop pattern $u_0 v_1^{[r_1]} u_1 \cdots v_n^{[r_n]} u_n$ to be $|u_0| + \cdots |u_n| + n \cdot d$. In other words, each loop v_i contributes d to the length.

Let I be the ideal $\downarrow_{\preceq_d} x_0 y_1^{[s_1]} x_1 \cdots y_\ell^{[s_\ell]} x_\ell$. Furthermore, let

$$u_0 v_1^{[r_1]} u_1 \cdots v_n^{[r_n]} u_n$$

be an extended loop pattern of minimal length N among all extended loop patterns that generate I and for which $\pi_d(v_i) \leq B$ for every $i \in [1,n]$. Let w_i be the length-r_i prefix of v_i for $i \in [1,n]$.

By minimality, the loop pattern $u_0(v_1)w_1u_1 \cdots (v_n)w_nu_n$ has to be irreducible: Otherwise, there would be a loop v_i such that

$$I = \downarrow_{\preceq_d} u_0 v_1^* w_1 u_1 \cdots v_{i-1}^* w_{i-1} u_{i-1} w_i u_i \cdots v_n^* w_n u_n$$

and hence the extended loop pattern

$$u_0 v_1^{[r_i]} u_1 \cdots v_{i-1}^{[r_{i-1}]} u_{i-1} w_i u_i \cdots v_n^{[r_n]} u_n$$

would generate I and have length $N - d + r_i < N$.

Now consider some non-empty u_i and suppose its first letter is contained in $\kappa_d(v_i)(r_i + 1)$. In other words, $u_i = a\bar{u}_i$ with $a \in \kappa_d(v_i)(r_i + 1)$. Consider the extended loop pattern obtained by replacing the term $v_i^{[r_i]} u_i$ with $v_i^{[r_i+1]} \bar{u}_i$. It clearly generates the same ideal. Moreover, the condition on π_d for loops would still be met. Finally, this extended loop pattern would have length $N - 1$, in contradiction to minimality.

Now consider some non-empty u_i and suppose its last letter is contained in $\kappa_d(v_{i+1})(d)$. In other words, $u_i = \bar{u}_i a$ with $a \in \kappa_d(v_{i+1})(d)$. Consider the extended loop pattern obtained by replacing the term $u_i v_{i+1}^{[r_{i+1}]}$ with $\bar{u}_i \lambda(v_{i+1})^{[r_{i+1}+1]}$. It is easy to see that it generates the same ideal. Moreover, since $\pi_d(\lambda(v_{i+1})) = \pi_d(v_{i+1}) \leq B$, the condition on π_d for loops would still be met. Finally, this extended loop pattern would have length $N-1$, contradicting minimality. □

K.6 Proof of Lemma 7.10

Lemma K.5. *Let $\mathcal{A}$ be an automaton with $\leq m$ states and let d be a multiple of $m^3!$. Moreover, let $\pi_d(v) \leq m^2$ and let $u \in \downarrow_{\leq_d} v^{[r]}$ be accepted by $\mathcal{A}$ such that $m \cdot \pi_d(v) \leq |u| \leq d$. Then there is a $u' \in \downarrow_{\leq_{\ell \cdot d}} (v^\ell)^{[r']}$ in $L(\mathcal{A})$ such that $r' = |u'| = r + (\ell-1)d$.*

Proof. Since $|u| \geq m \cdot \pi_d(v)$, u begins with at least $|u|/\pi_d(v) \geq m$ factors of length $\pi_d(v)$. Consider the run of $\mathcal{A}$ on u. Since $\mathcal{A}$ has at most m states, we can decompose $u = fgh$ such that g is a contiguous block of $k \leq m$ factors of length $\pi_d(v)$ and g is read on a cycle. Since $|g| = k \cdot \pi_d(v) \leq m^3$, $|g|$ divides d. Let $u' = fg^{1+(\ell-1)d/|g|}h$. Then according to Lemma K.4, we have $u' \in \downarrow_{\leq_d} v^{[r]}$. Therefore, $\kappa_d(u') \subseteq \kappa_d(v)$. This implies

$$\kappa_{\ell \cdot d}(u') \subseteq \kappa_{\ell \cdot d}(v) \subseteq \kappa_{\ell \cdot d}(v^\ell).$$

Finally, note that $|u'| = |u| + (\ell-1)d \leq \ell \cdot d$. Therefore, if we set $r' = |u'|$, then $u' \in \downarrow_{\leq_d} (v^\ell)^{[r']}$ by Lemma K.3. □

Lemma K.6. *Let $\mathcal{A}$ be an automaton with $\leq m$ states and let d be a multiple of $m^3!$. Moreover, let $v \in (\Sigma^d)^*$ with $\pi_d(v) \leq m^2$. If $u \in L(\mathcal{A})$ with $w \preceq_d u \in \downarrow_{\leq_d} v^{[r]}$, then there is a $u' \in L(\mathcal{A})$ with $w \preceq_{\ell \cdot d} u' \in \downarrow_{\leq_{\ell \cdot d}} (v^\ell)^{[r]}$*

Proof. Since $w \preceq_d u$, we can write $u = u_0 w_1 u_1 \cdots w_n u_n$, where $w = w_1 \cdots w_n$ and $w_1, \ldots, w_n \in \Sigma$, and $u_i \in (\Sigma^d)^*$. Since $u \in \downarrow_{\leq_d} v^{[r]}$, we have $\kappa_d(u) \subseteq \kappa_d(v)$ and hence $u_i \in \downarrow_{\leq_d} \lambda^i(v)^*$ for $i \in [0,n]$.

For each $i \in [0,n]$, we construct u_i' as follows. Consider the run of $\mathcal{A}$ on u and suppose it reads u_i from state p_i to state q_i.

- If u_i is empty, then $u_i' = u_i$. Note that then of course $u_i' \in \downarrow_{\leq_{\ell \cdot d}} \lambda^i(v^\ell)^*$.
- If u_i is non-empty, then we split u_i in $|u_i|/d$ factors of length d and apply to each factor Lemma K.5. This yields a word u_i' such that $u_i' \in \downarrow_{\leq_{\ell \cdot d}} (\lambda^i(v)^\ell)^*$ and so that u_i' can be read from state p_i to q_i. Moreover, we have $|u_i'|$ is a multiple of $\ell \cdot d$. Since $\lambda^i(v)^\ell = \lambda^i(v^\ell)$, we have $u_i' \in \downarrow_{\leq_{\ell \cdot d}} \lambda^i(v^\ell)^*$.

Therefore, the word $u' = u_0' w_1 u_1' \cdots w_n u_n'$ is accepted by $\mathcal{A}$, belongs to $\downarrow_{\leq_{\ell \cdot d}} (v^\ell)^{[r]}$ and satisfies $w \preceq_{\ell \cdot d} u'$. □

Lemma K.7. *Let $\mathcal{A}$ be an automaton with $\leq m$ states and and let d be a multiple of $2m^3!$. Moreover, let $v_i \in (\Sigma^d)^*$ with $\pi_d(v_i) \leq m^2$ for $i = 1,2$. If $u \in L(\mathcal{A})$ with $u \in \downarrow_{\leq_d} v_1^* v_2^*$, then there is a $u' \in L(\mathcal{A})$ with $u' \in \downarrow_{\leq_{\ell \cdot d}} (v_1^\ell)^* (v_2^\ell)^*$*

Proof. Let $K = \downarrow_{\leq_d} v_1^* v_2^*$. Observe that K consists precisely of the words of the form $u = x_1 \cdots x_p s t y_1 \cdots y_q$, where for some $r \in [0, d-1]$,

- $x_i \in \downarrow_{\leq_d} v_1^*$ and $x_i \in \Sigma^d$ for $i \in [1,p]$,
- $s \in \downarrow_{\leq_d} v_1^{[r]}$ and $|s| = r$,
- $t \in \downarrow_{\leq_d} \lambda^r(v_2)^{[d-r]}$, and $|t| = d - r$, and
- $y_i \in \downarrow_{\leq_d} v_2^*$ and $y_i \in \Sigma^d$ for $i \in [1,q]$.

On the one hand, all such words belong to $\downarrow_{\leq_d} v_1^* v_2^*$: The parts s and t arise when dropping length-d blocks on the border between v_1^* and v_2^*. On the other hand, by induction on the number of deleted length-d blocks, it follows that any word in $\downarrow_{\leq_d} v_1^* v_2^*$ is of that shape.

Since $|s| + |t| = d$, we have either $|s| \geq d/2$ or $|t| \geq d/2$. We treat the case that $|s| \geq d/2$, the other case is analogous.

We apply Lemma K.5 to each factor $x_1, \ldots, x_p, s, y_1, \ldots, y_q$. Note that this is possible because each of these words has length either exactly d or $\geq d/2$ and we have $\geq d/2 \geq m^3! \geq m^3 \geq m \cdot \pi_d(v_i)$ for $i = 1,2$. This yields words $x_1', \ldots, x_p', s', y_1', \ldots, y_q'$ such that

- $x_i' \in \downarrow_{\leq_{\ell \cdot d}} (v_1^\ell)^{[0]}$ for $i \in [1,p]$,
- $s' \in \downarrow_{\leq_{\ell \cdot d}} (v_1^\ell)^{[r']}$, where $r' = r + (\ell-1)d$,
- $y_i' \in \downarrow_{\leq_{\ell \cdot d}} (v_2^\ell)^{[0]}$ for $i \in [1,q]$,
- $\mathcal{A}$ accepts $u' = x_1' \cdots x_p' s' t y_1' \cdots y_q'$.

Recall that $t \in \downarrow_{\leq_d} \lambda^r(v_2)^{[d-r]}$. This means $\kappa_d(t) \subseteq \kappa_d(\lambda^r(v_2))$ and hence

$$\kappa_d(t) \subseteq \kappa_d(\lambda^r(v_2)^\ell) = \kappa_d(\lambda^r(v_2^\ell))$$

(recall that $\lambda^r(w)^\ell = \lambda^r(w^\ell)$ for every word w). Therefore, we also have

$$\kappa_{\ell \cdot d}(t) \subseteq \kappa_{\ell \cdot d}(\lambda^r(v_2^\ell)). \tag{6}$$

Note that since $\pi_{\ell \cdot d}(v_2^\ell) = \pi_d(v_2)$ divides d, we can rotate the word v_2^ℓ by a multiple of d without changing its image under $\kappa_{\ell \cdot d}(\cdot)$. Hence

$$\kappa_{\ell \cdot d}(\lambda^r(v_2^\ell)) = \kappa_{\ell \cdot d}(\lambda^{r+(\ell-1)d}(v_2^\ell))$$

Together with eq. (6), we may conclude that t belongs to the ideal $\downarrow_{\leq_{\ell \cdot d}} (\lambda^{r+(\ell-1)d}(v_2^\ell))^{[d-r]}$ according to Lemma K.3. Therefore, the above characterization of K, adapted to $\downarrow_{\leq_{\ell \cdot d}} (v_1^\ell)^* (v_2^\ell)^*$, is satisfied for the word u' and hence $u' \in \downarrow_{\leq_{\ell \cdot d}} (v_1^\ell)^* (v_2^\ell)^*$. □

Proof of Lemma 7.10. Since $u_0 v_1^{[r_1]} u_1 \cdots v_n^{[r_n]} u_n$ is irreducible and its ideal belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}))$, we know from Lemma 7.8 that the extended loop pattern is associated to $L(\mathcal{A})$.

Let I be the ideal in eq. (1). Let w_i be the length-r_i prefix of v_i for every $i \in [1,n]$.

In order to show that I belongs to $\mathsf{Adh}_{\leq_{\ell \cdot d}}(L(\mathcal{A}))$, we have to exhibit for each $k \in \mathbb{N}$ a word $w \in L(\mathcal{A})$ so that

$$u_0 (v_1^\ell)^k w_1 u_1 \cdots (v_n^\ell)^k w_n u_n \preceq_{\ell \cdot d} w$$

and $w \in I$.

Therefore, let $k \in \mathbb{N}$. Because of association, there is a word $\bar{w} = \bar{u}_0 \bar{v}_1 \bar{u}_1 \cdots \bar{v}_n \bar{u}_n \in L(\mathcal{A})$ such that for every $i \in [1,n]$, we have $v_i^{k \cdot \ell} w_i \preceq_d \bar{v}_i$ and $\bar{v}_i \in \downarrow_{\leq_d} v_i^{[r_i]}$. Moreover, $\bar{u}_0 = u_0$, $\bar{u}_n = u_n$, and for each $i \in [1, n-1]$:

- If u_i is not empty, then $\bar{u}_i = u_i$.
- If u_i is empty, then $\bar{u}_i \in \downarrow_{\leq_d} \lambda^{r_i}(v_i)^* v_{i+1}^*$.

Consider the run of $\mathcal{A}$ on $\bar{w}$. Using Lemma K.6, we can choose $\bar{v}_i'$ such that $v_i^{k \cdot \ell} w_i \preceq_{\ell \cdot d} \bar{v}_i'$ and $\bar{v}_i' \in \downarrow_{\leq_{\ell \cdot \ell}} (v_i^\ell)^{[r_i]}$ and so that it has a run parallel to $\bar{v}_i$ in $\mathcal{A}$. Now consider $\bar{u}_i$ for $i \in [0,n]$.

- If $\bar{u}_i = u_i$, then choose $\bar{u}_i' = \bar{u}_i = u_i$.
- If $\bar{u}_i \neq u_i$, then u_i is empty and $\bar{u}_i \in \downarrow_{\leq_d} \lambda^{r_i}(v_i)^* v_{i+1}^*$. Then we use Lemma K.7 to choose $\bar{u}_i'$ such that $\bar{u}_i'$ has a run parallel to $\bar{u}_i$ in $\mathcal{A}$ and $\bar{u}_i' \in \downarrow_{\leq_{\ell \cdot d}} (\lambda^{r_i}(v_i^\ell))^* (v_{i+1}^\ell)^*$.

Now the resulting word $w' = \bar{u}'_0\bar{v}'_1\bar{u}'_1 \cdots \bar{v}'_n\bar{u}'_n$ is accepted by the automaton $\mathcal{A}$. This shows that the extended loop pattern

$$u_0(v_1^\ell)^{[r_1]}u_1 \cdots (v_n^\ell)^{[r_n]}u_n$$

is associated to $L(\mathcal{A})$ and hence the ideal I belongs to $\mathsf{Adh}_{\leq_{\ell \cdot d}}(L(\mathcal{A}))$. □

K.7 Proof of Proposition 7.2

Proof. Suppose there is an ideal in the adherence $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$. By Lemma 7.7, there is a loop pattern $u_0v_1u_1 \cdots v_nu_n$ for $\mathcal{M}_d$ such that the ideal $I = \downarrow_{\leq_d} u_0v_1^*u_1 \cdots v_n^*u_n$ belongs to $\mathsf{Adh}_{\leq_d}(L(\mathcal{A}_i))$ for $i = 1, 2$ and $\pi_d(v_i) \leq m^2$ for every $i \in [1, n]$. Using Lemma 7.9, we can construct an irreducible extended loop pattern

$$\bar{u}_0\bar{v}_1^{[r_1]}\bar{u}_1 \cdots \bar{v}_n^{[r_n]}\bar{u}_n$$

that induces I and satisfies $\pi_d(\bar{v}_i) \leq m^2$ for $i \in [1, n]$. Now Lemma 7.10 tells us that the ideal

$$\downarrow_{\leq_{\ell \cdot d}} \bar{u}_0(\bar{v}_1^\ell)^{[r_1]}\bar{u}_1 \cdots (\bar{v}_n^\ell)^{[r_n]}\bar{u}_n$$

belongs to $\mathsf{Adh}_{\leq_{\ell \cdot d}}(L(\mathcal{A}_i))$ for $i = 1, 2$. □