The Word Problem for Omega-Terms over the Trotter-Weil Hierarchy

Manfred Kufleitner · **Jan Philipp Wächter**

Abstract For two given ω-terms α and β, the word problem for ω-terms over a variety $\mathbf{V}$ asks whether $\alpha = \beta$ in all monoids in $\mathbf{V}$. We show that the word problem for ω-terms over each level of the Trotter-Weil Hierarchy is decidable. More precisely, for every fixed variety in the Trotter-Weil Hierarchy, our approach yields an algorithm in nondeterministic logarithmic space (NL). In addition, we provide deterministic polynomial time algorithms which are more efficient than straightforward translations of the NL-algorithms. As an application of our results, we show that separability by the so-called corners of the Trotter-Weil Hierarchy is witnessed by ω-terms (this property is also known as ω-reducibility). In particular, the separation problem for the corners of the Trotter-Weil Hierarchy is decidable.

1 Introduction

Algebraic characterizations of classes of regular languages are interesting as they often allow to decide the class's membership problem. For example, by Schützenberger's famous theorem [23], one can decide whether a given regular language is star-free by computing its syntactic monoid M and checking its aperiodicity. The latter can be achieved by verifying $x^{|M|!} = x^{|M|!}x$ for all $x \in M$. This equation is also stated as $x^\omega = x^\omega x$ since this notation is independent of the monoid's size. More formally, we can see the equation as a pair of ω-terms: these are finite words built using letters, which are interpreted as variables, concatenation and an additional formal ω-power. Checking an equation $\alpha = \beta$ in a finite monoid is easy: one can simply substitute each variable

Institut für Formale Methoden der Informatik
University of Stuttgart, Germany
{kufleitner,jan-philipp.waechter}@fmi.uni-stuttgart.de

by all elements of the monoid. For each substitution, this yields a monoid element on the left hand side and one on the right hand side. The equation holds if and only if they are always equal.

Often, the question whether an equation holds is not only interesting for a single finite monoid but for a (possibly infinite) class of such monoids. For example, one may ask whether all monoids in a certain class are aperiodic. This is trivially decidable if the class is finite. But what if it is infinite? If the class forms a variety (of finite monoids, sometimes also referred to as a pseudo-variety), i. e. a class of finite monoids closed under (possibly empty) direct products, submonoids and homomorphic images, then this problem is called the variety's *word problem for ω-terms*. Usually, the study of a variety's word problem for ω-terms also gives more insight into the variety's structure, which is interesting in its own right. McCammond showed that the word problem for ω-terms of the variety **A** of aperiodic finite monoids is decidable [17]. The problem was shown to be decidable in linear time for **J**, the class of $\mathcal{J}$-trivial finite monoids, by Almeida [3] and for **R**, the class of $\mathcal{R}$-trivial monoids, by Almeida and Zeitoun [4]. For the variety **DA**, Moura adapted and expanded those ideas to show decidability in time $\mathcal{O}((nk)^5)$ where n is the length of the input ω-terms and k is the maximal nesting depth of the ω-power (which can be linear in n) [18]. Remember that **DA** is the class of finite monoids whose regular $\mathcal{D}$-classes form aperiodic semigroups. This variety received a lot of attention due to its many different characterizations; see e.g. [5, 27]. Most notably is its connection to two-variable first-order logic [28]. This logic is a natural restriction of first-order logic over finite words, which in turn is the logic characterization of **A**.

In this paper, we consider the word problem for ω-terms over the varieties in the *Trotter-Weil Hierarchy*. It was introduced by Trotter and Weil [29] with the idea of using the good understanding of the band varieties (cf. [7]) for studying the lattice of sub-varieties of **DA**; bands are semigroups satisfying $x^2 = x$. The levels of this hierarchy exhaust **DA**. As it turns out, the Trotter-Weil Hierarchy has tight connections to the quantifier alternation hierarchy inside two-variable first-order logic [14]. In addition, many characterizations of **DA** admit natural restrictions which allow climbing up this hiearchy (see [12]).

Please note that, in spite of this paper's title, we will refer to ω-terms as π-terms for most parts of the paper, the only exception being this introduction. We follow this notation introduced by Perrin and Pin [19] to avoid notational conflicts. Accordingly, we use π for the formal power in π-terms and speak of the word problem for π-terms.

Results. In this paper, we present the following results.
-   Our main tool for studying a variety **V** of the Trotter-Weil Hierarchy is a family of finite index congruences $\equiv_{\mathbf{V},n}$ for $n \in \mathbb{N}$. These congruences have the property that a monoid M is in **V** if and only if there exists n for which M divides a quotient by $\equiv_{\mathbf{V},n}$. The congruences are not new but they differ in some details from the ones usually found in the literature,

where they are introduced in terms of rankers [12, 14, 15]. Unfortunately, these differences necessitate new proofs.

- We lift the combinatorics from finite words to ω-terms using the "linear order approach" introduced by Huschenbett and the first author [10]. They showed that, over varieties of aperiodic monoids, one can use the order $\mathbb{N} + \mathbb{Z} \cdot \mathbb{Q} + (-\mathbb{N})$ for the formal ω-power. In this paper, we use the simpler order $\mathbb{N} + (-\mathbb{N})$. We show that two ω-terms α and β are equal in some variety $\mathbf{V}$ of the Trotter-Weil Hierarchy if and only if $[\![\alpha]\!]_{\mathbb{N}+(-\mathbb{N})} \equiv_{\mathbf{V},n} [\![\beta]\!]_{\mathbb{N}+(-\mathbb{N})}$ for all $n \in \mathbb{N}$. Here, $[\![\alpha]\!]_{\mathbb{N}+(-\mathbb{N})}$ denotes the generalized word (i. e. the labeled linear order) obtained from replacing every occurrence of the formal ω-power by the linear order $\mathbb{N} + (-\mathbb{N})$. Note that this order is tailor-made for the Trotter-Weil Hierarchy and does not result from simple arguments which work in any variety.
- We show that one can effectively check whether $[\![\alpha]\!]_{\mathbb{N}+(-\mathbb{N})} \equiv_{\mathbf{V},n} [\![\beta]\!]_{\mathbb{N}+(-\mathbb{N})}$ for all $n \in \mathbb{N}$.
- We further improve the algorithms and show that, for every variety $\mathbf{V}$ of the Trotter-Weil Hierarchy, the word problem for ω-terms over $\mathbf{V}$ is decidable in nondeterministic logarithmic space. The main difficulty is to avoid some blow-up which (naively) is caused by the nesting depth of the ω-power. For $\mathbf{R}$, which appears in the hierarchy, this result is incomparable to Almeida and Zeitoun's linear time algorithm [4].
- We also introduce polynomial time algorithms, which are more efficient than the direct translation of these NL algorithms.
- As an application, we prove that the separation problems for the so-called *corners* of the Trotter-Weil Hierarchy are decidable by showing ω-reducibility. For $\mathbf{J}$, we adapt the proof of van Rooijen and Zeitoun [30].
- With little additional effort, we also obtain all of the above results for $\mathbf{DA}$, the limit of the Trotter-Weil Hierarchy. The decidability of the separation problem re-proves a result of Place, van Rooijen and Zeitoun [22]. The algorithms for the word problem for ω-terms are more efficient than those of Moura [18].

Separability of the join levels and the intersection levels is still open. We conjecture that these problems can be solved with similar but more technical reductions.

2 Preliminaries

Natural Numbers and Finite Words. Let $\mathbb{N} = \{1, 2, \dots\}$, $\mathbb{N}_0 = \{0, 1, \dots\}$ and $-\mathbb{N} = \{-1, -2, \dots\}$. For the rest of this paper, we fix a finite alphabet Σ. By Σ^*, we denote the set of all finite words over the alphabet Σ, including the empty word ε; Σ^+ denotes $\Sigma^* \setminus \{\varepsilon\}$.

Order Types. A linearly ordered set $(P, \leq_P)$ consists of a (possibly infinite) set P and a linear ordering relation $\leq_P$ of P, i. e. a reflexive, anti-symmetric, transitive and total binary relation $\leq_P \subseteq P \times P$. To simplify notation we

define two special objects $-\infty$ and $+\infty$. The former is always smaller with regard to $\leq_P$ than any element in P while the latter is always larger. We call two linearly ordered sets $(P, \leq_P)$ and $(Q, \leq_Q)$ *isomorphic* if there is an order-preserving bijection $\varphi : P \to Q$. Isomorphism between linearly order sets is an equivalence relation; its classes are called (linear) *order types*.

The *sum* of two linearly ordered sets $(P, \leq_P)$ and $(Q, \leq_Q)$ is $(P \uplus Q, \leq_{P+Q})$ where $P \uplus Q$ is the disjoint union of P and Q and $\leq_{P+Q}$ orders all elements of P to be smaller than those of Q while it behaves as $\leq_P$ and $\leq_Q$ on elements from their respective sets. Similarly, the *product* of $(P, \leq_P)$ and $(Q, \leq_Q)$ is $(P \times Q, \leq_{P*Q})$ where $(p, q) \leq_{P*Q} (\tilde{p}, \tilde{q})$ holds if and only if either $q \leq_Q \tilde{q}$ and $q \neq \tilde{q}$ or $q = \tilde{q}$ and $p \leq_P \tilde{p}$ holds. Sum and product of linearly ordered sets are compatible with taking the order type. This allows for writing $\mu + \nu$ and $\mu * \nu$ for order types μ and ν.

We re-use $n \in \mathbb{N}_0$ to denote the order type of $(\{1, 2, \ldots, n\}, \leq)$. One should note that this use of natural numbers to denote order types does not result in contradictions with sums and products: the usual calculation rules apply. Besides finite linear order types, we need ω, the order type of $(\mathbb{N}, \leq)$, and its dual ω^* the order type of $(-\mathbb{N}, \leq)$. Another important order type in the scope of this paper is $\omega + \omega^*$, whose underlying set is $\mathbb{N} \uplus (-\mathbb{N})$. Note that, here, natural numbers and the (strictly) negative numbers are ordered as $1, 2, 3, \ldots, \ldots, -3, -2, -1$; therefore, in this order type, we have for example $-1 \geq_{\omega+\omega^*} 1$.

Generalized Words. Any finite word $w = a_1 a_2 \ldots a_n$ of length $n \in \mathbb{N}_0$ with $a_i \in \Sigma$ can be seen as a function which maps a *position* $i \in \{1, 2, \ldots, n\}$ to the corresponding letter a_i (or, possibly, the empty map). Therefore, it is natural to denote the positions in a word w by $\mathrm{dom}(w)$. By relaxing the requirement of $\mathrm{dom}(w)$ to be finite, one obtains the notion of *generalized words*: a (generalized) word w over the alphabet Σ of order type μ is a function $w : \mathrm{dom}(w) \to \Sigma$, where $\mathrm{dom}(w)$ is a linearly ordered set in μ. For $\mathrm{dom}(w)$, we usually choose $(\mathbb{N}, \leq)$, $(-\mathbb{N}, \leq)$ and $(\mathbb{N} \uplus (-\mathbb{N}), \leq_{\omega+\omega^*})$ as representative of ω, ω^* and $\omega + \omega^*$, respectively. The order type of a finite word of length n is n.

Like finite words, generalized words can be concatenated, i.e. we write u to the left of v and obtain uv. In that case, the order type of uv is the sum of the order types of u and v. Beside concatenation, we can also take powers of generalized words. Let w be a generalized word of order type μ which belongs to $(P_\mu, \leq_\mu)$ and let ν be an arbitrary order type belonging to $(P_\nu, \leq_\nu)$. Then, w^ν is a generalized word of order type $\mu * \nu$ which determines the ordering of its letters; w maps $(p_1, p_2) \in P_\mu \times P_\nu$ to $w(p_1)$. If $\nu = n$ for some $n \in \mathbb{N}$, then $w^\nu = w^n$ is equal to the n-fold concatenation of w.

In this paper, the term *word* refers to a generalized word. If it is important for a word to be finite, it is referred to explicitly as a *finite word*. As a counterpart to the positions $\mathrm{dom}(w)$ in w, we define the set of *letters appearing in a word* $\mathrm{alph}(w)$ as the image of w seen as a function. For example, for a

finite word $w = a_1 a_2 \ldots a_n$ of length $n \in \mathbb{N}_0$ (with $a_1, a_2, \ldots, a_n \in \Sigma$), we have $\mathrm{alph}(w) = \{a_1, a_2, \ldots, a_n\}$.

We also introduce notation for factors of words. For a pair $(l, r) \in (\{-\infty\} \uplus \mathrm{dom}(w)) \times (\mathrm{dom}(w) \uplus \{+\infty\})$, define $w_{(l,r)}$ as the restriction of the word w (seen as a mapping) to the set of positions (strictly) larger than l and (strictly) smaller than r. Note that $w = w_{(-\infty,+\infty)}$ and $w_{(l,r)} = \varepsilon$ for any pair (l, r) with no position between l and r.

Monoids, Divisors, Congruences and Recognition. In this paper, the term *monoid* refers to a finite monoid (except when stated otherwise). It is well known that, for any monoid M, there is a smallest number $n \in \mathbb{N}$ such that m^n is idempotent (i.e. $m^{2n} = m^n$) for every element $m \in M$; this number is called the *exponent* of M and shall be denoted by $M! = n$.[1] A monoid N is a *divisor* of (another) monoid M, written as $N \prec M$, if N is a homomorphic image of a submonoid of M.

A *congruence* (relation) in a (not necessarily finite) monoid M is an equivalence relation $\mathcal{C} \subseteq M \times M$ such that $x_1 \, \mathcal{C} \, x_2$ and $y_1 \, \mathcal{C} \, y_2$ implies $x_1 y_1 \, \mathcal{C} \, x_2 y_2$ for all $x_1, x_2, y_1, y_2 \in M$. If M is a (possibly infinite) monoid and $\mathcal{C} \subseteq M \times M$ is a congruence, then the set of equivalence classes of $\mathcal{C}$, denoted by $M/\mathcal{C}$, is a well-defined monoid (which might still be infinite), whose size is called the *index* of $\mathcal{C}$. For any two congruences $\mathcal{C}_1$ and $\mathcal{C}_2$, one can define their *join* $\mathcal{C}_1 \vee \mathcal{C}_2$ as the smallest congruence which includes $\mathcal{C}_1$ and $\mathcal{C}_2$; its index is at most as large as the index of $\mathcal{C}_1$ and the index of $\mathcal{C}_2$.

A (possibly infinite) monoid M *recognizes* a language (of finite words) $L \subseteq \Sigma^*$ if there is a homomorphism $\varphi : \Sigma^* \to M$ with $L = \varphi^{-1}(\varphi(L))$. A language is *regular* if and only if it is recognized by a finite monoid. It is well known that there is a unique smallest monoid which recognizes a given regular language: the *syntactic monoid*.

Green's Relations. Among the most important tools for studying monoids are Green's Relations. Let x and y be elements of a monoid M. Define

$$\begin{aligned}
x \, \mathcal{R} \, y &\Leftrightarrow xM = yM, \\
x \, \mathcal{L} \, y &\Leftrightarrow Mx = My \qquad \text{and} \\
x \, \mathcal{J} \, y &\Leftrightarrow MxM = MyM
\end{aligned}$$

where $xM = \{xm \mid m \in M\}$ is the *right-ideal* of x, $Mx = \{mx \mid m \in M\}$ its *left-ideal* and $MxM = \{m_1 x m_2 \mid m_1, m_2 \in M\}$ its (two-sided) *ideal*.

By simple calculation, one can see that $x \, \mathcal{R} \, y$ holds if and only if there are $z, z' \in M$ such that $xz = y$ and $yz' = x$ and, symmetrically, that $x \, \mathcal{L} \, y$ holds if and only if there are $z, z' \in M$ such that $zx = y$ and $z'y = x$.

---

[1] Note that all statements remain valid if one assumes that $M!$ is used to denote $|M|!$.

Varieties, π-Terms, Equations and Word Problem for π-terms. A *variety* (of finite monoids) – sometimes also referred to as a *pseudo-variety* – is a class of monoids which is closed under submonoids, homomorphic images and – possibly empty – finite direct products. For example, the class $\mathbf{R}$ of $\mathcal{R}$-trivial monoids and the class $\mathbf{L}$ of $\mathcal{L}$-trivial monoids both form a variety, see e.g. [20]. Clearly, if $\mathbf{V}$ and $\mathbf{W}$ are varieties, then so is $\mathbf{V} \cap \mathbf{W}$. For example, the class $\mathbf{J} = \mathbf{R} \cap \mathbf{L}$ is a variety; in fact, it is the variety of all $\mathcal{J}$-trivial monoids. For two varieties $\mathbf{V}$ and $\mathbf{W}$, the smallest variety containing $\mathbf{V} \cup \mathbf{W}$, the so called *join*, is denoted by $\mathbf{V} \vee \mathbf{W}$.

Many varieties can be defined in terms of *equations* (or *identities*). Because it will be useful later, we take a more formal approach towards equations by using *π-terms*[2]. A π-term is a finite word, built using letters, concatenation and an additional formal π-power (and appropriate parentheses), whose π-exponents act as a placeholder for a substitution value. Formally, every letter $a \in \Sigma$ is a π-term (over Σ). As a special case, also ε is a π-term. For two π-terms α and β, their concatenation $\alpha\beta$ is a π-term as well and, if γ is a π-term, then so is $(\gamma)^\pi$, where π is a formal exponent.

To state equations using π-terms, one needs to substitute the formal π-exponents by actual values resulting in a word. We define $[\![\gamma]\!]_\mu$ as the result of substituting the π-exponents in γ by an order type μ, i.e. we have $[\![\varepsilon]\!]_\mu = \varepsilon$, $[\![a]\!]_\mu = a$ for all $a \in \Sigma$, $[\![\alpha\beta]\!]_\mu = [\![\alpha]\!]_\mu [\![\beta]\!]_\mu$ and $[\![(\gamma)^\pi]\!]_\mu = ([\![\gamma]\!]_\mu)^\mu$. For example[3], we have

$$[\![a\left((b)^\pi c\right)^\pi]\!]_3 = a\left((bbb)c\right)\left((bbb)c\right)\left((bbb)c\right).$$

An equation $\alpha = \beta$ consists of two π-terms α and β over the same alphabet Σ, which, here, can be seen as a set of *variables*. A homomorphism $\sigma : \Sigma^* \to M$ is called an *assignment of variables* in this context. An equation $\alpha = \beta$ *holds in a monoid M* if for every assignment of variables $\sigma\left([\![\alpha]\!]_{M!}\right) = \sigma\left([\![\beta]\!]_{M!}\right)$ is satisfied. It holds in a variety $\mathbf{V}$ if it holds in all monoids in $\mathbf{V}$. The *word problem for π-terms* over a variety $\mathbf{V}$ is the problem to decide whether $\alpha = \beta$ holds in $\mathbf{V}$ for the input π-terms α and β.

Mal'cev Products. Besides intersection and join, we need one more constructions for varieties: the *Mal'cev product*, which is often defined using *relational morphisms*. In this paper, we use a different, yet equivalent, approach based on the congruences $\sim_K$ and $\sim_D$, see [11] or [8, Corollary 4.3]. For their definition, let x and y be elements of a monoid M and define

$$x \sim_K y \iff \forall e \in E(S) : ex \mathcal{R} e \text{ or } ey \mathcal{R} e \implies ex = ey$$
$$\text{and } x \sim_D y \iff \forall e \in E(S) : xe \mathcal{L} e \text{ or } ye \mathcal{L} e \implies xe = ye,$$

where $E(S)$ denotes the set of idempotents in S.

---

[2] As mentioned in the introduction, π-terms are usually referred to as ω-terms. In this paper, however, we use ω to denote the order type of the natural numbers. Therefore, we follow the approach of Perrin and Pin [19] and use π instead of ω.

[3] For a more elaborate example (involving $\omega + \omega^*$) see Example 1 on page 36.

Obviously, $\sim_K$ and $\sim_D$ are of finite index in any (finite) monoid M. Thus, we have that $M/\sim_K$ and $M/\sim_D$ are (finite) monoids and can define Mal'cev products of varieties. Let $\mathbf{V}$ be a variety. The varieties $\mathbf{K}\,\textcircled{m}\,\mathbf{V}$ and $\mathbf{D}\,\textcircled{m}\,\mathbf{V}$ are defined by

$$M \in \mathbf{K}\,\textcircled{m}\,\mathbf{V} \;\Leftrightarrow\; M/\sim_K \in \mathbf{V} \text{ and}$$
$$M \in \mathbf{D}\,\textcircled{m}\,\mathbf{V} \;\Leftrightarrow\; M/\sim_D \in \mathbf{V},$$

where M is a monoid. Note that, indeed, $\mathbf{K}\,\textcircled{m}\,\mathbf{V}$ and $\mathbf{D}\,\textcircled{m}\,\mathbf{V}$ are varieties for any variety $\mathbf{V}$ and that, furthermore, we have $\mathbf{V} \subseteq \mathbf{K}\,\textcircled{m}\,\mathbf{V}$ and $\mathbf{V} \subseteq \mathbf{D}\,\textcircled{m}\,\mathbf{V}$.

3 The Trotter-Weil Hierarchy

The main object of study in this paper, is the so-called *Trotter-Weil Hierarchy*. We will approach it primarily using certain combinatorial congruence. We will show that this approach is equivalent to the more common algebraic approach, which we will use as the definition of the hierarchy.

The Trotter-Weil Hierarchy. As the name implies, this hierarchy was first studied by Trotter and Weil [29]. We will define it using Mal'cev products. Though this approach is different to the original one used by Trotter and Weil, both are equivalent [15]. We define:

$$\mathbf{R_1} = \mathbf{L_1} = \mathbf{J},$$
$$\mathbf{R_{m+1}} = \mathbf{K}\,\textcircled{m}\,\mathbf{L_m} \text{ and}$$
$$\mathbf{L_{m+1}} = \mathbf{D}\,\textcircled{m}\,\mathbf{R_m}.$$

These varieties form the so-called *corners* of the hierarchy. Additionally, it contains the *join levels* $\mathbf{R_m} \vee \mathbf{L_m}$ and the *intersection levels* $\mathbf{R_m} \cap \mathbf{L_m}$. The term "hierarchy" is justified by the following inclusions: we have $\mathbf{R_m} \cap \mathbf{L_m} \subseteq \mathbf{R_m}, \mathbf{L_m} \subseteq \mathbf{R_m} \vee \mathbf{L_m}$ and $\mathbf{R_m} \vee \mathbf{L_m} \subseteq \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$; the latter can be seen by induction.

Among the corners of the Trotter-Weil Hierarchy are some well known varieties: we have $\mathbf{R_1} = \mathbf{L_1} = \mathbf{J}$, $\mathbf{R_2} = \mathbf{R}$ and $\mathbf{L_2} = \mathbf{L}$ (for the last two, see [20]; the others are straightforward).

By taking the union of all varieties in the hierarchy, one gets the variety $\mathbf{DA}$ [13], which is usually defined as the class of monoids whose regular $\mathcal{D}$-classes form aperiodic semigroups[4]:
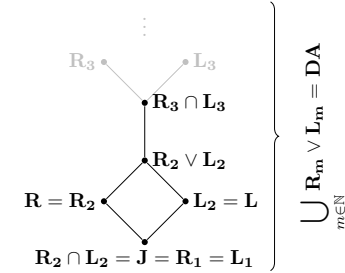

Fig. 1 Trotter-Weil Hierarchy

---

[4] In finite monoids, $\mathcal{D}$-classes coincide with $\mathcal{J}$-classes; a $\mathcal{D}$-class is called *regular* if it contains an idempotent. A semigroup is called *aperiodic* (or *group-free*) if it has no divisor which is a nontrivial group.

7

Fact 1
$$\mathbf{DA} = \bigcup_{m\in\mathbb{N}} \mathbf{R_m} \vee \mathbf{L_m} = \bigcup_{m\in\mathbb{N}} \mathbf{R_m} = \bigcup_{m\in\mathbb{N}} \mathbf{L_m}$$

These considerations yield the graphic representation given in Figure 1.

Connections to Two-Variable Logic. The variety $\mathbf{DA}$ is closely connected to two-variable first-order logic. By $\mathsf{FO}^2[<]$, denote the set of all first-order sentences over finite words which may only use the $<$ predicate (and equality) and no more than two variables. A language $L \subseteq \Sigma^*$ of finite words is definable by a sentence $\varphi \in \mathsf{FO}^2[<]$ if and only if its syntactic monoid is in $\mathbf{DA}$ [28], which it is if and only if it is in one of the Trotter-Weil Hierarchy's varieties.

The intersection levels corresponds to the quantifier alternation hierarchy within $\mathsf{FO}^2[<]$ [14]: a first-order sentence using at most two variables belongs to $\mathsf{FO}^2_m[<]$ if, on any path in its syntax tree, there is no quantifier after the first negation and there are at most m blocks of quantifiers. A language is definable by a sentence in $\mathsf{FO}^2_m[<]$ if and only if its syntactic monoid is in $\mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$.

Equational Characterization. Besides its definition using Mal'cev products and its connections to logic, the Trotter-Weil Hierarchy can also be characterized in terms of equations. For our proofs, we only need the direction of this characterization stated in the lemma below. The other direction does hold as well; we will see later on in Lemma 5 that it is an easy consequence of the hierarchy's combinatorial characterization which we state below.

Lemma 1 *Define the π-terms*

$$U_1 = (sx_1)^\pi s(y_1 t)^\pi \quad and \quad V_1 = (sx_1)^\pi t(y_1 t)^\pi$$

over the alphabet $\Sigma_1 = \{s, t, x_1\}$. For $m \in \mathbb{N}$, let x_{m+1} and y_{m+1} be new characters not in the alphabet Σ_m and define the π-terms

$$U_{m+1} = (U_m x_{m+1})^\pi U_m (y_{m+1} U_m)^\pi \quad and \quad V_{m+1} = (U_m x_{m+1})^\pi V_m (y_{m+1} U_m)^\pi$$

over the alphabet $\Sigma_{m+1} = \Sigma_m \uplus \{x_{m+1}, y_{m+1}\}$.
Then we have

$$M \in \mathbf{R_1} = \mathbf{L_1} = \mathbf{J} \Leftarrow U_1 = V_1 \text{ holds in } M,$$
$$M \in \mathbf{R_{m+1}} \Leftarrow (U_m x_{m+1})^\pi U_m = (U_m x_{m+1})^\pi V_m \text{ holds in } M,$$
$$M \in \mathbf{L_{m+1}} \Leftarrow U_m (y_{m+1} U_m)^\pi = V_m (y_{m+1} U_m)^\pi \text{ holds in } M \text{ and}$$
$$M \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}} \Leftarrow U_m = V_m \text{ holds in } M$$

for all $m \in \mathbb{N}$.

Proof The first implication is a well-known characterization of $\mathbf{J}$ [21]. We show the next two implications by induction over m (see also [12]).

First, assume $m = 1$ and consider a monoid M in which $(U_1 x_2)^\pi U_1 = (U_1 x_2)^\pi V_1$ holds. We need to show $M \in \mathbf{R_2}$, which is equivalent to showing

$M/\!\sim_K \in \mathbf{L_1} = \mathbf{J}$. For this, we show $u_1 = \sigma\left(\llbracket U_1 \rrbracket_{M!}\right) \sim_K \sigma\left(\llbracket V_1 \rrbracket_{M!}\right) = v_1$ for an arbitrary assignment of variables $\sigma : \Sigma_1^* \to M$. Afterwards, we are done by the first implication since the exponent of $M/\!\sim_K$ is a divisor of $M!$. Let e be an arbitrary idempotent of M and assume $eu_1 \mathrel{\mathcal{R}} e$ (the other case from the definition of $\sim_K$ is symmetrical). Thus, there is an element $x_2 \in M$ such that $eu_1 x_2 = e$ holds. We can extend σ by mapping the letter $x_2 \in \Sigma_2$ to the just defined monoid element x_2. We then have

$$eu_1 = e(u_1 x_2)u_1 = \ldots = e(u_1 x_2)^{M!} u_1$$
$$= e(u_1 x_2)^{M!} v_1 = \cdots = e(u_1 x_2) v_1 = ev_1.$$

The equality in the middle holds because $(U_1 x_2)^\pi U_1 = (U_1 x_2)^\pi V_1$ holds in M by assumption. This concludes the $m = 1$ case because the third implication is symmetrical.

Now, assume $m > 1$ and consider a monoid M in which $(U_m x_{m+1})^\pi U_m = (U_m x_{m+1})^\pi V_m$ holds. We need to show $M \in \mathbf{R_{m+1}}$ and we do this by showing $M/\!\sim_K \in \mathbf{L_m}$. By induction, we only have to show that $U_{m-1}(y_m U_{m-1})^\pi = V_{m-1}(y_m U_{m-1})^\pi$ holds in $M/\!\sim_K$. As before, let $\sigma : \left(\Sigma_{m-1} \uplus \{y_m\}\right)^* \to M$ be an arbitrary assignment of variables. For convenience, let $u_{m-1} = \sigma\left(\llbracket U_{m-1} \rrbracket_{M!}\right)$ and $v_{m-1} = \sigma\left(\llbracket V_{m-1} \rrbracket_{M!}\right)$ and identify y_m with $\sigma(y_m)$. Let e be an arbitrary idempotent in M such that $eu_{m-1}(y_m u_{m-1})^{M!} \mathrel{\mathcal{R}} e$ holds. Clearly, this implies $eu_{m-1} \mathrel{\mathcal{R}} e$ and there are elements $x_m, x_{m+1} \in M$ such that $e = eu_{m-1}x_m$ and $e = eu_{m-1}(y_m u_{m-1})^{M!} x_{m+1}$ holds. Extend σ to map the letters x_m and x_{m+1} to the respective monoid elements. Then, we have

$$eu_{m-1}(y_m u_{m-1})^{M!} = e(u_{m-1} x_m)u_{m-1}(y_m u_{m-1})^{M!}$$
$$= \cdots = e(u_{m-1} x_m)^{M!} u_{m-1}(y_m u_{m-1})^{M!} = eu_m$$

with $u_m = \sigma\left(\llbracket U_m \rrbracket_{M!}\right)$. This yields

$$e = \underbrace{eu_{m-1}(y_m u_{m-1})^{M!}}_{eu_m} x_{m+1} = e\left(u_m x_{m+1}\right) = \cdots = e\left(u_m x_{m+1}\right)^{M!}.$$

Using the fact that $(U_m x_{m+1})^\pi U_m = (U_m x_{m+1})^\pi V_m$ holds in M (by assumption), we get

$$eu_m = e\left(u_m x_{m+1}\right)^{M!} u_m = e\left(u_m x_{m+1}\right)^{M!} v_m = ev_m$$

where $v_m = \sigma\left(\llbracket V_m \rrbracket_{M!}\right)$. In combination, we have

$$eu_{m-1}(y_m u_{m-1})^{M!} = eu_m = ev_m = e(u_{m-1} x_m)^{M!} v_{m-1}(y_m u_{m-1})^{M!}$$

where the last equality holds due to the definition of V_m. Finally, we get

$$eu_{m-1}(y_m u_{m-1})^{M!} = e(u_{m-1} x_m)^{M!} v_{m-1}(y_m u_{m-1})^{M!}$$
$$= \cdots = e(u_{m-1} x_m) v_{m-1}(y_m u_{m-1})^{M!}$$
$$= ev_{m-1}(y_m u_{m-1})^{M!}$$

Thus, we have shown $u_{m-1}(y_m u_{m-1})^{M!} \sim_K v_{m-1}(y_m u_{m-1})^{M!}$ (the other case from the definition of $\sim_K$ is symmetrical) and are done. The implication for $\mathbf{L_{m+1}}$ in the case $m > 1$ follows by symmetry again.

Finally, for the intersection levels, suppose that $U_m = V_m$ holds in a monoid M. By the identities for the corners, we directly have $M \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$. $\square$

Besides the equational characterization of the individual varieties in the hierarchy, one can also characterize their union $\mathbf{DA}$ in terms of an equation:

Fact 2 *Let M be a monoid. Then, we have*

$$M \in \mathbf{DA} \;\Leftrightarrow\; (xyz)^\pi y(xyz)^\pi = (xyz)^\pi \;\text{ holds in } M.$$

A proof of this fact can be found in [27].

4 Relations for the Trotter-Weil Hierarchy

Although we define the Trotter-Weil Hierarchy algebraically using Mal'cev products, we will primarily use a different characterization which is based on certain combinatorial congruences. Before we can finally introduce these congruences, however, we need to give some definitions for factorizations of words at the first or last a-position (i.e. an a-labeled position).

Factorizations and accessible words. For a word w, a position $p \in \mathrm{dom}(w) \uplus \{-\infty\}$ and a letter $a \in \mathrm{alph}(w)$, let $X_a(w; p)$ denote the first a-position (strictly) larger than p (or the first a-position in w if $p = -\infty$). It is undefined if there is no such position (i.e. there is no a to the right of p in w) or if the position is not well-defined. Define $Y_a(w; p)$ symmetrically as the first a-position from the right which is (strictly) smaller than p. Notice that, with generalized words, the first a-position to the right of a position p is not necessarily well-defined even if there is an a-position larger than p. For example, $a^{\omega^*} = \ldots aa$ does not have a first a-position. We call words for which this situation does not occur *accessible*; i.e. a word w is accessible if, for every position p in w (including the special cases $p \in \{\pm\infty\}$), $X_a(w; p)$ is defined if and only if there is an a-position in $w_{(p, +\infty)}$ and $Y_a(w; p)$ is defined if and only if there is an a-position in $w_{(-\infty, p)}$. Note that all finite words are accessible and that so are all words of the form $[\![\gamma]\!]_{\omega + \omega^*}$ for a π-term γ.

Let w be an accessible word, define

$$w \cdot X_a^L = w_{(-\infty, X_a(w; -\infty))}, \qquad w \cdot X_a^R = w_{(X_a(w; -\infty), +\infty)},$$
$$w \cdot Y_a^L = w_{(-\infty, Y_a(w; +\infty))} \text{ and } \qquad w \cdot Y_a^R = w_{(Y_a(w; +\infty), +\infty)}$$

for all $a \in \mathrm{alph}(w)$. Additionally, define $C_{a,b}$ as a special form of applying X_a^L first and then Y_b^R which is only defined if $X_a(w; -\infty)$ is strictly larger than $Y_b(w; +\infty)$. For an example of X_a^L and X_a^R acting on a word see Figure 2. Note that we have $w = (w \cdot X_a^L)a(w \cdot X_a^R) = (w \cdot Y_a^L)a(w \cdot Y_a^R) = (w \cdot Y_b^L)b(w \cdot C_{a,b})a(w \cdot X_a^R)$ (whenever these factors are defined). For example,
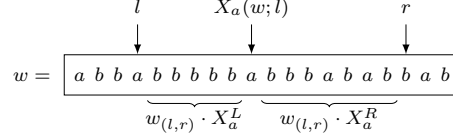
Fig. 2 Application of X_a^L and X_a^R to an example word.

we have $cbbcdcaca \cdot C_{a,b} = cdc$; see the upper part of Figure 4 for a graphical representation of this example. If we apply a sequence of factorizations, we omit the $\cdot$ between them, e. g. we write $w \cdot X_a^L Y_b^R = w \cdot X_a^L \cdot Y_b^R = w \cdot C_{a,b}$.

Relations for the Trotter-Weil Hierarchy. With these definitions in place, we define for $m, n \in \mathbb{N}$ the relations $\equiv_{m,n}^X$, $\equiv_{m,n}^Y$ and $\equiv_{m,n}^{\mathrm{WI}}$ of accessible words.[5] The idea is that these relations hold on two words u and v if both words allow for the same sequence of factorizations at the first or last occurrence of a letter. For this, we use the following recursive definition.

Definition 1 Let $m, n \in \mathbb{N}$ and let u and v be accessible words. Define recursively:

1. $u \equiv_{0,0}^Z v$, $u \equiv_{m,0}^Z v$ and $u \equiv_{0,n}^Z v$ for $Z \in \{X, Y, \mathrm{WI}\}$ always hold.
2. $u \equiv_{m,n}^X v \Leftrightarrow \mathrm{alph}(u) = \mathrm{alph}(v)$, $u \equiv_{m-1,n-1}^Y v$ and
$$\forall a \in \mathrm{alph}\,(u): u \cdot X_a^L \equiv_{m-1,n-1}^Y v \cdot X_a^L \text{ and}$$
$$u \cdot X_a^R \equiv_{m,n-1}^X v \cdot X_a^R$$
$u \equiv_{m,n}^Y v \Leftrightarrow \mathrm{alph}(u) = \mathrm{alph}(v)$, $u \equiv_{m-1,n-1}^X v$ and
$$\forall a \in \mathrm{alph}\,(u): u \cdot Y_a^L \equiv_{m,n-1}^Y v \cdot Y_a^L \text{ and}$$
$$u \cdot Y_a^R \equiv_{m-1,n-1}^X v \cdot Y_a^R$$
$u \equiv_{m,n}^{\mathrm{WI}} v \Leftrightarrow \mathrm{alph}(u) = \mathrm{alph}(v)$,
$$\forall a \in \mathrm{alph}\,(u): u \cdot X_a^L \equiv_{m-1,n-1}^{\mathrm{WI}} v \cdot X_a^L \text{ and}$$
$$u \cdot X_a^R \equiv_{m,n-1}^{\mathrm{WI}} v \cdot X_a^R,$$
$$\forall a \in \mathrm{alph}\,(u): u \cdot Y_a^L \equiv_{m,n-1}^{\mathrm{WI}} v \cdot Y_a^L \text{ and}$$
$$u \cdot Y_a^R \equiv_{m-1,n-1}^{\mathrm{WI}} v \cdot Y_a^R \text{ and}$$
$$\forall a, b \in \mathrm{alph}\,(u): u \cdot C_{a,b} \text{ and } v \cdot C_{a,b} \text{ are either both undefined}$$
$$\text{or both defined and } u \cdot C_{a,b} \equiv_{m-1,n-1}^{\mathrm{WI}} v \cdot C_{a,b}$$
$$\text{holds.}$$

Additionally, define $u \equiv_{m,n}^{XY} v \Leftrightarrow u \equiv_{m,n}^X v$ and $u \equiv_{m,n}^Y v$ for all $m, n \in \mathbb{N}_0$.

---

[5] The presented relations could also be defined by (condensed) *rankers* (as it is done in [14] and [15]). Rankers were introduced by Weis and Immerman [31] (thus, the WI exponent in $\equiv_{m,n}^{\mathrm{WI}}$) who reused the *turtle programs* by Schwentick, Thérien and Vollmer [24]. Another concept related to condensed rankers is the *unambiguous interval temporal logic* by Lodaya, Pandya and Shah [16].

11

As this definition is vital for the understanding of the rest of this paper, we try to give an intuitive understanding of how the relations work. In the parameter m, we remember the remaining number of direction changes (which are caused by an X_a^L or Y_a^R factorizations) in a factorization sequence and the parameter n is the number of remaining factorization moves (independent of their direction). Thus, if m or n is zero, then the relations shall be satisfied on all accessible word pairs. For m and n larger than zero, our first assertion is that both words have the same alphabet; otherwise, one of them would admit a factorization at

Fig. 3 $\equiv_{m,n}^X$ illustrated.

a letter while the other would not, as the letter is not in its alphabet. For the letters of the common alphabet, we want to be able to perform further factorization (until we reach $n = 0$). The X or Y exponent of the relation indicates whether we start at the beginnings or at the ends of the two words; the WI exponent is a special case, which we will discuss below. So, if we want $u \equiv_{m,n}^X v$ to hold and the alphabets of u and v coincide, then we can continue factorizing at the first a in u and in v. In the next step, we continue either in the two left parts or in the two right parts. If we continue in the right parts (i. e. we do an X_a^R factorization), then we require $u \cdot X_a^R \equiv_{m,n-1}^X v \cdot X_a^R$. The X in the exponent indicates that the last factorization position (the first a in u and v, respectively) was at the beginning of the words. We have made one additional factorization – thus, we only have $n-1$ remaining factorizations in this part – but we did not change the direction because we were at the beginnings of the words before and still are for the new word pair – thus, we still have m such changes of direction. If, instead of taking the right parts after the first a, we had taken the left parts, then we would still have made a single factorization; thus we have $n-1$ remaining factorizations. However, we would also have a direction change: the factorization position (i. e. the first a) is to the right of the new words $u \cdot X_a^L$ and $v \cdot X_a^L$. Therefore, we have $m-1$ remaining direction changes and we change the X exponent into a Y exponent. This is summed up by requiring $u \cdot X_a^L \equiv_{m-1,n-1}^Y v \cdot X_a^L$. Both situations are illustrated in Figure 3. Additionally, we also have the choice to switch from the beginning to the end at the cost of one direction change and one factorization. This is reflected in the fact that we require $u \equiv_{m-1,n-1}^Y v$ for $u \equiv_{m,n}^X v$.
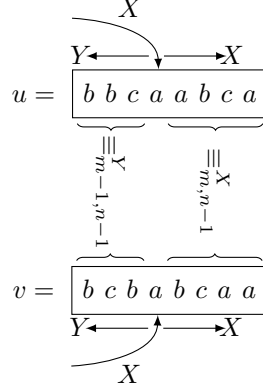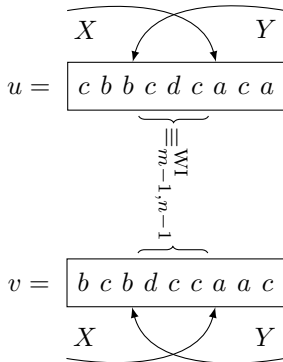

Fig. 4 $\equiv_{m,n}^{WI}$ illustrated.

The relation $\equiv_{m,n}^Y$ works symmetrically to $\equiv_{m,n}^X$. However, $\equiv_{m,n}^{WI}$ is a bit different. Here, we still lose one direction change after an X_a^L or Y_a^R factorization while the number keeps the same for X_a^R and Y_a^L factorizations but we also have the special $C_{a,b}$ factorizations, which start simultaneously at the beginning and the end. Consider the situation given in Figure 4. We go

12

to the first a and the last b; the former is to the right of the latter in both words (or the $\equiv_{m,n}^{\mathrm{WI}}$ relation does not hold). For the pair of parts in the middle, we require $u \cdot C_{a,b} \equiv_{m-1,n-1}^{\mathrm{WI}} v \cdot C_{a,b}$,

i. e. we count this as a single factorization move which loses a single direction change.

Finally, there is $\equiv_{m,n}^{XY}$ where we can start factorizing at the beginning or at the end.

By simple inductions, one can see that the relations are congruences of finite index over Σ^*. Also note that $u \equiv_{m,n}^{Z} v$ implies $u \equiv_{m,k}^{Z} v$ and, if $m > 0$, also $u \equiv_{m-1,k}^{Z} v$ for all $k \leq n$ and $Z \in \{X, Y, XY, \mathrm{WI}\}$.

The importance of the just defined relations for this paper yields from their connection to the Trotter-Weil Hierarchy (and, thus, to **DA**), which we state in the next theorem.

Theorem 1 *Let M be a monoid and $m \in \mathbb{N}$. Then*

- $M \prec \Sigma^*/\equiv_{m,n}^{X}$ *for some $n \in \mathbb{N}_0$* $\Leftrightarrow$ $M \in \mathbf{R_m}$,
- $M \prec \Sigma^*/\equiv_{m,n}^{Y}$ *for some $n \in \mathbb{N}_0$* $\Leftrightarrow$ $M \in \mathbf{L_m}$,
- $M \prec \Sigma^*/\equiv_{m,n}^{XY}$ *for some $n \in \mathbb{N}_0$* $\Leftrightarrow$ $M \in \mathbf{R_m} \vee \mathbf{L_m}$ *and*
- $M \prec \Sigma^*/\equiv_{m,n}^{\mathrm{WI}}$ *for some $n \in \mathbb{N}_0$* $\Leftrightarrow$ $M \in \mathbf{R_m} \cap \mathbf{L_m}$ *hold.*

The combinatorial nature of the relations will turn out to be useful in the remainder of this paper since it allow us, for example, to obtain efficient algorithms for the problems we consider. Unfortunately, on the other hand, it makes proving Theorem 1 quite technical. In fact, we will dedicate the next section to this proof.

5 A Proof for Theorem 1

We will prove both directions of Theorem 1 individually. Before we can do this, however, we need to introduce some more concepts.

$\mathcal{R}$- and $\mathcal{L}$-factorizations. Let $\varphi : \Sigma^* \to M$ be a (monoid) homomorphism into a monoid M. The *$\mathcal{R}$-factorization* of a finite word w is the (unique) factorization $w = w_0 a_1 w_1 a_2 w_2 \ldots a_k w_k$ with $w_0, w_1 \ldots, w_k \in \Sigma^*$ and $a_1, a_2, \ldots, a_k \in \Sigma$ such that, on the one hand,

$$\varphi(\varepsilon) \; \mathcal{R} \; \varphi(w_0) \text{ and}$$
$$\varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_i) \; \mathcal{R} \; \varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_i w_i)$$

hold for $i = 1, 2, \ldots, k$ and, on the other hand,

$$\varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_i w_i) \; \mathcal{R} \; \varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_i w_i a_{i+1})$$

holds for $i = 0, 1, \ldots, k - 1$. Symmetrically, the $\mathcal{L}$-*factorization* of w is the factorization $w = w_0 a_1 w_1 a_2 w_2 \ldots a_k w_k$ with $w_0, w_1 \ldots, w_k \in \Sigma^*$ and $a_1, a_2, \ldots, a_k \in \Sigma$ such that, on the one hand,

$$\varphi(w_k) \mathrel{\mathcal{L}} \varphi(\varepsilon) \text{ and}$$

$$\varphi(w_{i-1} a_i w_i a_{i+1} w_{i+1} \ldots a_k w_k) \mathrel{\mathcal{L}} \varphi(a_i w_i a_{i+1} w_{i+1} \ldots a_k w_k)$$

hold for $i = 1, 2, \ldots, k$ and, on the other hand,

$$\varphi(a_i w_i a_{i+1} w_{i+1} a_{i+2} w_{i+2} \ldots a_k w_k) \mathrel{\not{\mathcal{L}}} \varphi(w_i a_{i+1} w_{i+1} a_{i+2} w_{i+2} \ldots a_k w_k)$$

holds for $i = 1, 2, \ldots, k$.

DA *and* $\mathcal{R}$-*classes.* In **DA**, getting into a new $\mathcal{R}$-class is strictly coupled to an element's alphabet, as the following lemma shows[6], where a can be seen as one of the monoids generators (i. e. a letter in its alphabet).

Lemma 2 *Let* $M \in \mathbf{DA}$ *be a monoid and let* $s, t \in M$ *such that* $s \mathrel{\mathcal{R}} t$. *Then*

$$s \mathrel{\mathcal{R}} sa \;\Rightarrow\; t \mathrel{\mathcal{R}} ta$$

holds for all $a \in M$.

Proof Since we have $t \mathrel{\mathcal{R}} s \mathrel{\mathcal{R}} sa$, there are $x, y \in M$ with $s = tx$ and $t = say$. We then have

$$t = txay = t(xay)^2 = \cdots = t(xay)^{M!},$$

which yields

$$ta(xay)^{M!} = t(xay)^{M!}a(xay)^{M!} = t(xay)^{M!} = t.$$

using the equation from Fact 2. Thus, we have $ta \mathrel{\mathcal{R}} t$. □

One of the main applications of the previous lemma is the following. If we have a monoid $M \in \mathbf{DA}$, a homomorphism $\varphi : \Sigma^* \to M$ and the $\mathcal{R}$-factorization $w = w_0 a_1 w_1 a_2 w_2 \ldots a_k w_k$ of a finite word $w \in \Sigma^*$, then we know that $a_i \notin \mathrm{alph}(w_{i-1})$ for $i = 1, 2, \ldots, k$. If we had $a_i \in \mathrm{alph}(w_{i-1})$, we could factorize $w_{i-1} = u a_i v$ and would have

$$\varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_{i-1} u) \mathrel{\mathcal{R}} \varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_{i-1} u a_i)$$

and, by the previous lemma, also

$$\varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_{i-1} u a_i v) \mathrel{\mathcal{R}} \varphi(w_0 a_1 w_1 a_2 w_2 \ldots a_{i-1} u a_i v a_i),$$

which results in a contradiction to the definition of $\mathcal{R}$-factorizations. Of course, we can apply a left-right dual of the lemma to get an analogue statement for $\mathcal{L}$-factorizations.

Another application of Lemma 2 is stated in the following lemma.

---

[6] The curious reader might be interested in the fact that the lemma's assertion also holds for monoids in **DS**, the variety of monoids whose regular $\mathcal{D}$-classes form (arbitrary, but finite) semigroups. More on **DS** can, for example, be found in [2].

Lemma 3 *Let $M \in \mathbf{DA}$ be a monoid, $s \in M$ and let $\varphi : \Sigma^* \to M$ be a homomorphism. Then, for all finite words $u, v \in \Sigma^*$ with $\mathrm{alph}(u) = \mathrm{alph}(v)$, we have*

$$s \mathrel{\mathcal{R}} s\varphi(u) \;\Rightarrow\; s \mathrel{\mathcal{R}} s\varphi(v).$$

Proof The case $u = v = \varepsilon$ is trivial. Therefore, assume $s \mathrel{\mathcal{R}} s\varphi(u)$ but $s \mathrel{\not\mathcal{R}} s\varphi(v'a)$ for some prefix $v'a$ of v with $a \in \Sigma$. Without loss of generality, let $v'a$ be the shortest such prefix; thus, we have $s \mathrel{\mathcal{R}} s\varphi(v')$. Since we have $\mathrm{alph}(u) = \mathrm{alph}(v)$, the is a prefix $u'a$ of u. Because of $s \mathrel{\mathcal{R}} s\varphi(u)$, we have $s \mathrel{\mathcal{R}} s\varphi(u') \mathrel{\mathcal{R}} s\varphi(u'a)$. Thus, we have $s\varphi(v') \mathrel{\mathcal{R}} s \mathrel{\mathcal{R}} s\varphi(u') \mathrel{\mathcal{R}} s\varphi(u')\varphi(a)$. Now, Lemma 2 yields $s\varphi(v') \mathrel{\mathcal{R}} s\varphi(v')\varphi(a)$, which is a contradiction to our assumption. $\qquad\square$

Now, we are prepared to prove the characterization of the Trotter-Weil Hierarchy stated in Theorem 1. This is done in the following two theorems (see also [15] for the corners and [14] for the intersection levels). We use the notations $X_\Sigma^D = \{X_a^L, X_a^R \mid a \in \Sigma\}$, $Y_\Sigma^D = \{Y_a^L, Y_a^R \mid a \in \Sigma\}$ and some natural variations of it.

Theorem 2 *Let M be a finite monoid, $\varphi : \Sigma^* \to M$ a homomorphism and $m \in \mathbb{N}$. Then:*

- $M \in \mathbf{R_m} \;\Rightarrow\; \big(\exists n \in \mathbb{N} \, \forall u, v \in \Sigma^* : u \equiv_{m,n}^X v \;\Rightarrow\; \varphi(u) = \varphi(v)\big)$
- $M \in \mathbf{L_m} \;\Rightarrow\; \big(\exists n \in \mathbb{N} \, \forall u, v \in \Sigma^* : u \equiv_{m,n}^Y v \;\Rightarrow\; \varphi(u) = \varphi(v)\big)$
- $M \in \mathbf{R_m} \vee \mathbf{L_m} \;\Rightarrow\; \big(\exists n \in \mathbb{N} \, \forall u, v \in \Sigma^* : u \equiv_{m,n}^{XY} v \;\Rightarrow\; \varphi(u) = \varphi(v)\big)$
- $M \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}} \;\Rightarrow\; \big(\exists n \in \mathbb{N} \, \forall u, v \in \Sigma^* : u \equiv_{m,n}^{\mathrm{WI}} v \;\Rightarrow\; \varphi(u) = \varphi(v)\big)$

Proof We fix a homomorphism $\varphi : \Sigma^* \to M$ and proceed by induction over m. For $m = 1$, we have $\mathbf{R_1} = \mathbf{L_1} = \mathbf{R_1} \vee \mathbf{L_1} = \mathbf{R_2} \cap \mathbf{L_2} = \mathbf{J}$. Thus, the assertion follows from a result of Simon [25]. However, we also include a full proof for completeness. Let $M \in \mathbf{J}$ and $n = |M|$, which is the number of $\mathcal{J}$-classes in M (and equal to the number of $\mathcal{R}$-classes and the number of $\mathcal{L}$-classes). Assume that $u \equiv_{1,n}^X v$ for two finite words $u, v \in \Sigma^*$ and let $u = u_0 a_1 u_1 a_2 u_2 \ldots a_k u_k$ be the $\mathcal{R}$-factorization of u. We have $k + 1 \leq n$ and, because M is $\mathcal{R}$-trivial, $\varphi(u_0) = \varphi(u_1) = \cdots = \varphi(u_k) = 1$ or, more precisely, that no letter in $u_0 u_1 \ldots u_k$ can be mapped by φ to a value different from 1. Notice that this implies $\varphi(u) = \varphi(a_1 a_2 \ldots a_k)$. Furthermore, we have $\varphi(a_1), \varphi(a_2), \ldots, \varphi(a_k) \neq 1$ due to the definition of an $\mathcal{R}$-factorization.

By definition of $\equiv_{m,n}^X$, we have $a_1 \in \mathrm{alph}(v)$ and $u \cdot X_{a_1}^R = a_2 u_2 a_3 u_3 \ldots a_k u_k \equiv_{1,n-1}^X v \cdot X_{a_1}^R$. Therefore, we can find a_2 in $v \cdot X_{a_1}^R$ and have $u \cdot X_{a_1}^R \cdot X_{a_2}^R = a_3 u_3 a_4 u_4 \ldots a_k u_k \equiv_{m,n-2}^X v \cdot X_{a_1}^R \cdot X_{a_2}^R$. Iterating this approach yields that $a_1 a_2 \ldots a_k$ is a subword[7] of v. Now, let $v = v_0 b_1 v_1 b_2 v_2 \ldots b_l v_l$ be the $\mathcal{R}$-factorization of v. By symmetry, we get $\varphi(b_1), \varphi(b_2), \ldots, \varphi(b_l) \neq 1$ and that $b_1 b_2 \ldots b_l$ is a subword of u. However, for no $j \in \{1, 2, \ldots, l\}$, the letter b_j can occur in $u_0 u_1 \ldots u_k$ since all letters in that word must be mapped to 1.

---

[7] A finite word $c_1 c_2 \ldots c_s$ with $c_i \in \Sigma$ is a subword of a (not necessarily finite) word w if we can write $w = w_0 c_1 w_1 c_2 w_2 \ldots c_s w_s$ for some words $w_0, w_1, \ldots, w_s$.

Thus, $b_1 b_2 \ldots b_l$ must in fact be a subword of $a_1 a_2 \ldots a_k$. Again by symmetry, $a_1 a_2 \ldots a_k$ must be a subword of $b_1 b_2 \ldots b_l$ and, thus, the two words must be equal. This implies $\varphi(u) = \varphi(a_1 a_2 \ldots a_k) = \varphi(b_1 b_2 \ldots b_l) = \varphi(v)$ where the last equality follows from $\varphi(v_0) = \varphi(v_1) = \cdots = \varphi(v_l) = 1$, which holds due to the $\mathcal{R}$-triviality of M.

The argumentation for $u \equiv_{1,n}^Y v$ is symmetric using the $\mathcal{L}$-factorization, the case for $u \equiv_{1,n}^{XY} v$ follows trivially and the case for $u \equiv_{1,n}^{\mathrm{WI}} v$ uses the same argumentation.

Now, let $M \in \mathbf{R_m}$ for an $m > 1$. This implies $M/\sim_K \in \mathbf{L_{m-1}}$ and there is an $n' \in \mathbb{N}$ such that $u' \equiv_{m-1,n'}^Y v' \Rightarrow \varphi(u') \sim_K \varphi(v')$ holds for all $u', v' \in \Sigma^*$. Let r be the number of $\mathcal{R}$-classes in M and let $n = n' + r$. Consider the $\mathcal{R}$-factorization $u = u_0 a_1 u_1 a_2 u_2 \ldots a_k u_k$ of a finite word $u \in \Sigma^*$; note that $k + 1 \leq r$ must hold. We have

$$u_i = u \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_{i-1}}^R X_{a_i}^L \text{ for } i = 0, 1, \ldots, k-1 \text{ and}$$
$$u_k = u \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_k}^R.$$

For a second finite word $v \in \Sigma^*$ with $u \equiv_{m,n}^X v$, we know that $\mathrm{alph}(u) = \mathrm{alph}(v)$. Thus, we can apply $X_{a_1}^L$ and $X_{a_1}^R$ to v and obtain

$$v_0 = v \cdot X_{a_1}^L \quad \text{and} \quad v' = v \cdot X_{a_1}^R.$$

By definition of $\equiv_{m,n}^X$, we have $v_0 \equiv_{m-1,n-1}^Y u_0$ and $v' \equiv_{m,n-1}^X u_1 a_2 u_2 a_3 u_3 \ldots a_k u_k$. Because of $k \leq r < n$, we can apply the same argument on v' and, by iteration, get

$$v_i = v \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_{i-1}}^R X_{a_i}^L \text{ for } i = 0, 1, \ldots, k-1 \text{ and}$$
$$v_k = v \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_k}^R$$

with $u_i \equiv_{m-1,n-i-1}^Y v_i$ for $i = 0, 1, \ldots, k-1$ and $u_k \equiv_{m,n-k}^X v_k$. Because of $i \leq k \leq r-1$, we have $n - i - 1 = n' + r - i - 1 \geq n' + r - (r-1) - 1 = n'$ and $u_i \equiv_{m-1,n'}^Y v_i$ for $i = 0, 1, \ldots, k-1$. For u_k and v_k, we have $u_k \equiv_{m-1,n-k-1}^Y v_k$ by the definition of the congruences and, therefore, $u_k \equiv_{m-1,n'}^Y v_k$ because of $n - k - 1 \geq n - i - 1 \geq n'$. Summing this up, we have $u_i \equiv_{m-1,n'}^Y v_i$ and, thus, $\varphi(u_i) \sim_K \varphi(v_i)$ for all $i = 0, 1, \ldots, k$.

Since we have defined u_i by the $\mathcal{R}$-factorization of u, there is an $s_i \in M$ for any $i \in \{0, 1, \ldots, k\}$ such that $\varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_i u_i) s_i = \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_i)$ holds. For these, we have

$$(\varphi(u_i) s_i)^{M!} \varphi(u_i) \; \mathcal{R} \; (\varphi(u_i) s_i)^{M!}$$

because of $(\varphi(u_i) s_i)^{M!} \varphi(u_i) s_i (\varphi(u_i) s_i)^{M!-1} = (\varphi(u_i) s_i)^{M!}$, which yields

$$(\varphi(u_i) s_i)^{M!} \varphi(u_i) = (\varphi(u_i) s_i)^{M!} \varphi(v_i)$$

by $\varphi(u_i) \sim_K \varphi(v_i)$. Thus, we have

$$
\begin{aligned}
\varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_k u_k) &= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_k u_k) \left(s_k \varphi(u_k)\right)^{M!} \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_k) \left(\varphi(u_k) s_k\right)^{M!} \varphi(u_k) \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_k) \left(\varphi(u_k) s_k\right)^{M!} \varphi(v_k) \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_k) \varphi(v_k) \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_{k-1} u_{k-1}) \varphi(a_k v_k) \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_{k-1}) \left(\varphi(u_{k-1}) s_{k-1}\right)^{M!} \varphi(u_{k-1}) \varphi(a_k v_k) \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_{k-1}) \left(\varphi(u_{k-1}) s_{k-1}\right)^{M!} \varphi(v_{k-1}) \varphi(a_k v_k) \\
&= \varphi(u_0 a_1 u_1 a_2 u_2 \ldots a_{k-2} u_{k-2}) \varphi(a_{k-1} v_{k-1} a_k v_k) \\
&= \ldots \\
&= \varphi(v_0 a_1 v_1 a_2 v_2 \ldots a_k v_k),
\end{aligned}
$$

which concludes the proof for $\mathbf{R_m}$.

The proof for $\mathbf{L_m}$ is symmetrical. For $\mathbf{R_m} \vee \mathbf{L_m}$, we observe that a monoid is in the join $\mathbf{V} \vee \mathbf{W}$ of two varieties $\mathbf{V}$ and $\mathbf{W}$ if and only if it is a divisor (i.e. the homomorphic image of a submonoid) of a direct product $M_1 \times M_2$ such that $M_1 \in \mathbf{V}$ and $M_2 \in \mathbf{W}$ [6, Exercise 1.1]. Therefore, if we have a monoid $M \in \mathbf{R_m} \vee \mathbf{L_m}$, there are monoids $M_1 \in \mathbf{R_m}$ and $M_2 \in \mathbf{L_m}$ such that M is a divisor of $M_1 \times M_2$; i.e. there is a submonoid N of $M_1 \times M_2$ and a surjective monoid homomorphism $\psi : N \twoheadrightarrow M$. For every $a \in \Sigma$, we can find elements $m_{a,1} \in M_1$ and $m_{a,2} \in M_2$ with $(m_{a,1}, m_{a,2}) \in N$ such that $\varphi(a) = \psi(m_{a,1}, m_{a,2})$. Indeed, we can define the maps $\varphi_1 : \Sigma \to M_1$ and $\varphi_2 : \Sigma \to M_2$ by setting $\varphi_1(a) := m_{a,1}$ and $\varphi_2(a) := m_{a,2}$. These maps can be lifted into homomorphisms $\varphi_1 : \Sigma^* \to M_1$ and $\varphi_2 : \Sigma^* \to M_2$. By induction, there are n_1 and n_2 such that $u \equiv_{m,n_1}^{X} v$ implies $\varphi_1(u) = \varphi_2(v)$ and $u \equiv_{m,n_2}^{Y} v$ implies $\varphi_2(u) = \varphi_2(v)$ for any two finite words $u, v \in \Sigma^*$. By setting $n = \max\{n_1, n_2\}$, we have

$$
u \equiv_{m,n}^{XY} v \;\Rightarrow\; \varphi_1(u) = \varphi_1(v) \text{ and } \varphi_2(u) = \varphi_2(v)
$$

for all $u, v \in \Sigma^*$. For all $u, v \in \Sigma^*$ with $u \equiv_{m,n}^{XY} v$, this yields

$$
\begin{aligned}
\varphi(a_1 a_2 \ldots a_k) &= \varphi(a_1) \varphi(a_2) \ldots \varphi(a_k) \\
&= \psi(m_{a_1,1}, m_{a_1,2}) \psi(m_{a_2,1}, m_{a_2,2}) \ldots \psi(m_{a_k,1}, m_{a_k,2}) \\
&= \psi((m_{a_1,1}, m_{a_1,2})(m_{a_2,1}, m_{a_2,2}) \ldots (m_{a_k,1}, m_{a_k,2})) \\
&= \psi(m_{a_1,1} m_{a_2,1} \ldots m_{a_k,1}, m_{a_1,2} m_{a_2,2} \ldots m_{a_k,2}) \\
&= \psi\left(\varphi_1(u), \varphi_2(u)\right) \\
&= \psi\left(\varphi_1(v), \varphi_2(v)\right) \\
&= \varphi(b_1 b_2 \ldots b_l)
\end{aligned}
$$

where $u = a_1 a_2 \ldots a_k$, $v = b_1 b_2 \ldots b_l$ and $a_1, a_2, \ldots, a_k, b_1, b_2, \ldots, b_l \in \Sigma$.

Finally, let $M \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$ with $m > 1$. Denote by 2^Σ the monoid of subsets of Σ whose binary operation is the union of sets. It is easy to see that 2^Σ is $\mathcal{J}$-trivial. Therefore, we have $M \times 2^\Sigma \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$. Next, we lift $\varphi : \Sigma^* \to M$ into a homomorphism $\hat{\varphi} : \Sigma^* \to M \times 2^\Sigma$ by taking the word's alphabet as the entry in the second component. If we show $u \equiv^{\mathrm{WI}}_{m,n} v \Rightarrow \hat{\varphi}(u) = \hat{\varphi}(v)$ for a suitable $n \in \mathbb{N}$, we have, in particular, $u \equiv^{\mathrm{WI}}_{m,n} v \Rightarrow \varphi(u) = \varphi(v)$. The advantage of this approach is that we have $\hat{\varphi}(u) = \hat{\varphi}(v) \Rightarrow \mathrm{alph}(u) = \mathrm{alph}(v)$ for all $u, v \in \Sigma^*$ by the construction of $\hat{\varphi}$. Instead of continuing to write $\hat{\varphi}$, we simply substitute M by $M \times 2^\Sigma$ and φ by $\hat{\varphi}$.

We have $M/\!\!\sim_K \in \mathbf{L_m}$ and $M/\!\!\sim_D \in \mathbf{R_m}$. By $\approx$, denote the join of $\sim_K$ and $\sim_D$. Since it is a homomorphic image of both, $M/\!\!\sim_K$ and $M/\!\!\sim_D$, the monoid $M/\!\!\approx$ is in $\mathbf{R_m} \cap \mathbf{L_m}$ and we can apply induction, which yields an $n' \in \mathbb{N}$ such that $u \equiv^{\mathrm{WI}}_{m-1,n'} v$ implies $\varphi(u) \approx \varphi(v)$ for all finite words $u, v \in \Sigma^*$. Let c be the sum of the number of $\mathcal{R}$-classes and the number of $\mathcal{L}$-classes in M and set $n = n' + c$. Suppose we have $u \equiv^{\mathrm{WI}}_{m,n} v$ for two finite words $u, v \in \Sigma^*$. Consider the $\mathcal{R}$-factorization $u = u'_0 a_1 u'_1 a_2 u'_2 \ldots a_r u'_r$ of u and the $\mathcal{L}$-factorization $v = v'_0 b_1 v'_1 b_2 v'_2 \ldots b_l v'_l$ of v. Clearly, we have $r + 1 + l + 1 \le c$. Define the positions $p^w_0 = -\infty$, $p^w_{r+1} = +\infty$ and $p^w_i = X_{a_i}(w; p^w_{i-1})$ for $i = 1, 2, \ldots, r$ and $w = u, v$. By Lemma 2, we know that p^u_i denotes the position of a_i in the $\mathcal{R}$-factorization for $i = 1, 2, \ldots, r$. Symmetrically, we can define $q^w_{l+1} = +\infty$, $q^w_0 = -\infty$ and $q^w_j = Y_{a_j}(w; q^w_{j+1})$ for $j = l, l-1, \ldots, 1$ and $w = u, v$. Again, we know that q^v_j is the position of b_j in the $\mathcal{L}$-factorization of v for $j = 1, 2, \ldots, l$. Additionally, we have

$$p^w_0 < p^w_1 < \cdots < p^w_r < p^w_{r+1} \text{ and}$$
$$q^w_0 < q^w_1 < \cdots < q^w_l < q^w_{l+1}$$

for $w = u$ and $w = v$ by their definition. We are going to show that we have $p^u_i \triangledown q^u_j \Leftrightarrow p^v_i \triangledown q^v_j$ for $\triangledown \in \{<, =, >\}$ and all $i = 1, 2, \ldots, r$ and $j = 1, 2, \ldots, l$. Together, these results yield that the sequence which is obtained by ordering the p_i and q_j positions in u is equal to the corresponding sequence in v. To prove this assertion, assume that we have $q^u_j \le p^u_i$ but $q^v_j > p^v_i$ for an $i \in \{1, 2, \ldots, r\}$ and a $j \in \{1, 2, \ldots, l\}$ (all other cases are symmetric or analogous). Without loss of generality, we may assume that $p^u_{i-1} < q^u_j \le p^u_i$ holds since, otherwise, we can substitute i by a smaller i for which the former holds. Note that this substitution does not violate the condition $q^v_j > p^v_i$ as p^v_i gets strictly smaller if i decreases. Equally without loss of generality, we may assume $q^u_j \le p^u_i < q^u_{j+1}$ by a dual argumentation. The situation is presented in Figure 5. We have

$$u_{(p^u_{i-1}, q^u_{j+1})} = u \cdot X^R_{a_1} X^R_{a_2} \ldots X^R_{a_{i-1}} Y^L_{b_l} Y^L_{b_{l-1}} \ldots Y^L_{b_{j+1}} \text{ and}$$
$$v_{(p^v_{i-1}, q^v_{j+1})} = v \cdot X^R_{a_1} X^R_{a_2} \ldots X^R_{a_{i-1}} Y^L_{b_l} Y^L_{b_{l-1}} \ldots Y^L_{b_{j+1}}$$
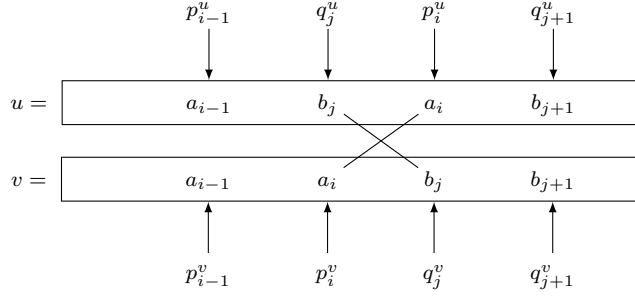
18

Fig. 5 Contradiction: p_i is to the right of q_j in u but to its left in v.

and $u_{(p_{i-1}^u, q_{j+1}^u)} \equiv_{m, n-(i-1)-(l-(j+1)+1)}^{\mathrm{WI}} v_{(p_{i-1}^v, q_{j+1}^v)}$, which yields $u_{(p_{i-1}^u, q_{j+1}^u)} \equiv_{m,2}^{\mathrm{WI}} v_{(p_{i-1}^v, q_{j+1}^v)}$ because of

$$
\begin{aligned}
n - (i-1) - (l - (j+1) + 1) &= n' + c - i + 1 - l + j + 1 - 1 \\
&= n' + c - i - l + j + 1 \\
&\geq n' + c - (r + l) + 1 \\
&\geq n' + c - (c-2) + 1 = n' + 3 \\
&> 2.
\end{aligned}
$$

If $q_j^u = p_i^u$, we have a contradiction since $u_{(p_{i-1}^u, q_{j+1}^u)} \cdot Y_{b_j}^L$ contains no a_i while $v_{(p_{i-1}^v, q_{j+1}^v)} \cdot Y_{b_j}^L$ does. For $q_j^u < p_i^u$, we can apply C_{a_i, b_j} to $u_{(p_{i-1}^u, q_{j+1}^u)}$ while we cannot apply it to $v_{(p_{i-1}^v, q_{j+1}^v)}$ by its definition. Both situations constitute a contradiction.

We have proved that if we order the set $\{p_i^u, q_j^u \mid i = 1, 2, \ldots, r, j = 1, 2, \ldots, l\} = \{P_1^u, P_2^u, \ldots, P_t^u\}$ (with $t \in \mathbb{N}_0$) such that

$$ P_1^u < P_2^u < \cdots < P_t^u $$

holds, then we can set

$$
P_s^v = \begin{cases} p_i^v & P_s^u = p_i^u \text{ for some } i \in \{1, 2, \ldots, r\} \\ q_j^v & P_s^u = q_j^u \text{ for some } j \in \{1, 2, \ldots, l\} \end{cases}
$$

for $s = 1, 2, \ldots, t$ and get

$$ P_1^v < P_2^v < \cdots < P_t^v. $$

These positions yield factorizations $u = u_0 c_1 u_1 c_2 u_2 \ldots c_t u_t$ and $v = v_0 c_1 v_1 c_2 v_2 \ldots c_t v_t$ such that $c_s \in \{a_i, b_j \mid i = 1, 2, \ldots, r, j = 1, 2, \ldots, l\}$ and P_s^w denotes the position of c_s in $w \in \{u, v\}$ for $s = 1, 2, \ldots, t$. To apply induction, we are going to show $u_s \equiv_{m-1, n'}^{\mathrm{WI}} v_s$ for all $s = 1, 2, \ldots, t$ next.

To simplify notation, we say "P_s is an $\mathcal{R}$-position" for any $s \in \{1, 2, \ldots, t\}$ if $P_s^u = p_i^u$ for some $i \in \{1, 2, \ldots, r\}$ (or, equivalently, if $P_s^v = p_i^v$ for some

19

i) and we say "P_s is an $\mathcal{L}$-position" if $P_s^u = q_j^u$ for some $j \in \{1, 2, \ldots, l\}$ (or, equivalently again, if $P_s^v = q_j^v$ for some j). Note that this definition is *not* exclusive, i.e. there can be a position which is both, an $\mathcal{R}$-position and an $\mathcal{L}$-position.

Next, we consider the corner cases of u_0/v_0 and u_t/v_t. If P_1 is an $\mathcal{R}$-position, we have $c_1 = a_1$ and

$$u_0 = u \cdot X_{a_1}^L \quad \text{as well as} \quad v_0 = v \cdot X_{a_1}^L,$$

which yields $u_0 \equiv_{m-1,n'}^{\mathrm{WI}} v_0$ by definition of $\equiv_{m,n}^{\mathrm{WI}}$ and because of $c > 0$. If P_1 is an $\mathcal{L}$-position, we have $c_1 = b_1$ and

$$u_0 = u \cdot Y_{b_l}^L Y_{b_{l-1}}^L \ldots Y_{b_1}^L \text{ as well as}$$
$$v_0 = v \cdot Y_{b_l}^L Y_{b_{l-1}}^L \ldots Y_{b_1}^L.$$

Because $l < c$, $u_0 \equiv_{m-1,n'}^{\mathrm{WI}} v_0$ holds also in this case. For u_t and v_t, we can apply a symmetric argumentation.

Finally, we distinguish four cases for a fixed $s \in \{1, 2, \ldots, t-1\}$. If P_s and P_{s+1} are both $\mathcal{R}$-positions, then we have $c_s = a_i$ and $c_{s+1} = a_{i+1}$ for some $i \in \{1, 2, \ldots, r\}$ and also

$$u_s = u \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_i}^R X_{a_{i+1}}^L \text{ as well as}$$
$$v_s = v \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_i}^R X_{a_{i+1}}^L.$$

By definition of $\equiv_{m,n}^{\mathrm{WI}}$, because of $i + 1 \leq c$, we thus have $u_s \equiv_{m-1,n'}^{\mathrm{WI}} v_s$. A symmetric argument applies if both, P_s and P_{s+1}, are $\mathcal{L}$-positions. If P_s is an $\mathcal{R}$-position but P_{s+1} is an $\mathcal{L}$-position, then $c_s = a_i$ for some $i \in \{1, 2, \ldots, r\}$ and $c_{s+1} = b_j$ for some $j \in \{1, 2, \ldots, l\}$, which yields

$$u_s = u \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_i}^R Y_{b_l}^L Y_{b_{l-1}}^L \ldots Y_{b_j}^L \text{ as well as}$$
$$v_s = v \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_i}^R Y_{b_l}^L Y_{b_{l-1}}^L \ldots Y_{b_j}^L.$$

Therefore, we have $u_s \equiv_{m-1,n'}^{\mathrm{WI}} v_s$ because of the definition of $\equiv_{m,n}^{\mathrm{WI}}$ and $n - i - (l - j + 1) = n' + c - (i + 1 + l) + j \geq n' + c - c + 0 = n'$. The fourth case is the most interesting: if P_s is an $\mathcal{L}$ but not an $\mathcal{R}$-position while P_{s+1} is an $\mathcal{R}$ but not an $\mathcal{L}$-position, then $c_s = b_j$ for some $j \in \{1, 2, \ldots, l\}$ and $c_{s+1} = a_i$ for some $i \in \{1, 2, \ldots, r\}$. Additionally, we have $p_{i-1}^w < P_s^w = q_j^w < P_{s+1}^w = p_i^w < q_{j+1}$ for $w = u$ and for $w = v$. We define

$$\tilde{u} = u \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_{i-1}}^R Y_{b_l}^L Y_{b_{l-1}}^L \ldots Y_{b_{j+1}}^L \text{ as well as}$$
$$\tilde{v} = v \cdot X_{a_1}^R X_{a_2}^R \ldots X_{a_{i-1}}^R Y_{b_l}^L Y_{b_{l-1}}^L \ldots Y_{b_{j+1}}^L$$

(we consider the X-blocks as empty – meaning that we do not factorize – if $i = 1$ and the Y-blocks as empty if $j = l$). We have $\tilde{u} \equiv_{m,n-(i-1)-(l-j)}^{\mathrm{WI}} \tilde{v}$. Because of $n - (i-1) - (l-j) = n' + c - (i+l) + j + 1 \geq n' + c - (r+l) + 1 \geq n' + 1$, $u_s = \tilde{u} \cdot C_{a_i,b_j}$, $v_s = \tilde{v} \cdot C_{a_i,b_j}$ and the definition of $\equiv_{m,n}^{\mathrm{WI}}$, we have $u_s \equiv_{m-1,n'}^{\mathrm{WI}} v_s$.

We have shown $u_s \equiv_{m-1,n'}^{\text{WI}} v_s$ for all $s = 1, 2, \ldots, t$ and, by induction, therefore, know that $\varphi(u_s) \approx \varphi(v_s)$, i.e. for a fixed $s \in \{1, 2, \ldots, t\}$, there are $w_1, w_2, \ldots, w_k \in \Sigma^*$ such that

$$\varphi(u_s) = \varphi(w_1) \sim_K \varphi(w_2) \sim_D \cdots \sim_K \varphi(w_{k-1}) \sim_D \varphi(w_k) = \varphi(v_s)$$

holds.

Remember that we substituted M by $M \times 2^\Sigma$ so that we can assume $\varphi(u) = \varphi(v) \Rightarrow \text{alph}(u) = \text{alph}(v)$ for all $u, v \in \Sigma^*$. We can extend this implication: if we have $\varphi(u) \sim_K \varphi(v)$ for two $u, v \in \Sigma^*$, then, by definition of $\sim_K$, we also have $\varphi(u)^{M!}\varphi(u) = \varphi(u)^{M!}\varphi(v)$ because of $\varphi(u)^{M!}\varphi(u) \; \mathcal{R} \; \varphi(u)^{M!}$. Therefore, we have $\text{alph}(u) = \text{alph}(u) \cup \text{alph}(v)$ by the implication stated above. By symmetry, we, thus, have $\text{alph}(u) = \text{alph}(v)$. Since we can apply a similar argumentation for $\sim_D$, we have $\varphi(u) \sim_K \varphi(v)$ or $\varphi(u) \sim_D \varphi(v) \Rightarrow \text{alph}(u) = \text{alph}(v)$ for all $u, v \in \Sigma^*$. This yields $\text{alph}(u_s) = \text{alph}(w_1) = \text{alph}(w_2) = \cdots = \text{alph}(w_k) = \text{alph}(v_s)$.

Since the factorizations $u = u_0 c_1 u_1 c_2 u_2 \ldots c_t u_t$ and $v = v_0 c_1 v_1 c_2 v_2 \ldots c_t v_t$ are subfactorizations from the $\mathcal{R}$-factorization of u and the $\mathcal{L}$-factorization of v, there are $x_s, y_s \in M$ with

$$\varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s) = \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s u_s) x_s \text{ and}$$
$$\varphi(c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t) = y_s \varphi(v_s c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t).$$

Because of $\text{alph}(u_s) = \text{alph}(w_i)$ for all $i \in \{1, 2, \ldots, k\}$ and by Lemma 3,

$$(\varphi(u_s)x_s)^{M!} \; \mathcal{R} \; (\varphi(u_s)x_s)^{M!} \varphi(u_s) \text{ implies } (\varphi(u_s)x_s)^{M!} \; \mathcal{R} \; (\varphi(u_s)x_s)^{M!} \varphi(w_i).$$

Similarly, we have

$$(y_s \varphi(v_s))^{M!} \; \mathcal{L} \; \varphi(w_i) (y_s \varphi(v_s))^{M!}$$

for all $i \in \{1, 2, \ldots, k\}$. For $\varphi(w_i) \sim_K \varphi(w_{i+1})$, this implies

$$(\varphi(u_s)x_s)^{M!} \varphi(w_i) = (\varphi(u_s)x_s)^{M!} \varphi(w_{i+1})$$

and

$$\varphi(w_i) (y_s \varphi(v_s))^{M!} = \varphi(w_{i+1}) (y_s \varphi(v_s))^{M!}$$

for $\varphi(w_i) \sim_D \varphi(w_{i+1})$. In either case, we have

$$(\varphi(u_s)x_s)^{M!} \varphi(w_i) (y_s \varphi(v_s))^{M!} = (\varphi(u_s)x_s)^{M!} \varphi(w_{i+1}) (y_s \varphi(v_s))^{M!},$$

which yields for any $i \in \{1, 2, \ldots, k-1\}$:

$$\begin{aligned}
&\varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s w_i c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t) \\
&= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s) (\varphi(u_s)x_s)^{M!} \varphi(w_i) (y_s \varphi(v_s))^{M!} \varphi(c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t) \\
&= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s) (\varphi(u_s)x_s)^{M!} \varphi(w_{i+1}) (y_s \varphi(v_s))^{M!} \varphi(c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t) \\
&= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s w_{i+1} c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t)
\end{aligned}$$

So, we can substitute w_i by w_{i+1} and, therefore, also u_s by v_s, i.e. we have

$$\varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s u_s c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t u_t)$$
$$= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_s v_s c_{s+1} v_{s+1} c_{s+2} v_{s+2} \ldots c_t v_t).$$

Consecutively applying the former equation for $s = t$, then for $s = t - 1$ and so on yields

$$\begin{aligned}
\varphi(u) &= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_{t-1} u_{t-1} c_t u_t) \\
&= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_{t-1} u_{t-1} c_t v_t) \\
&= \varphi(u_0 c_1 u_1 c_2 u_2 \ldots c_{t-1} v_{t-1} c_t v_t) \\
&\ \ \vdots \\
&= \varphi(v_0 c_1 v_1 c_2 v_2 \ldots c_{t-1} v_{t-1} c_t v_t) \\
&= \varphi(v),
\end{aligned}$$

which concludes the proof. □

It remains to show the other direction of Theorem 1 (i.e. the converse of Theorem 2). We will do this by using the equations from Lemma 1. Therefore, we begin with the following lemma.

Lemma 4 *Let Σ_m, U_m and V_m be as in Lemma 1 and let $k \in \mathbb{N}$ and $n \in \mathbb{N}_0$. Then, for every assignment of variables $\sigma : \Sigma_m^* \to \Sigma^*$ and all $Z \in \{X, Y, XY, \mathrm{WI}\}$, we have*

$$\sigma\left(\llbracket U_m \rrbracket_{nk}\right) \equiv_{m,n}^{Z} \sigma\left(\llbracket V_1 \rrbracket_{nk}\right)$$
$$\sigma\left(\llbracket (U_m x_{m+1})^\pi U_m \rrbracket_{nk}\right) \equiv_{m+1,n}^{X} \sigma\left(\llbracket (U_m x_{m+1})^\pi V_m \rrbracket_{nk}\right)$$
$$\sigma\left(\llbracket U_m (U_m x_{m+1})^\pi \rrbracket_{nk}\right) \equiv_{m+1,n}^{Y} \sigma\left(\llbracket V_m (U_m x_{m+1})^\pi \rrbracket_{nk}\right)$$

for all $m \in \mathbb{N}$.

Proof We only show the first assertion by induction over m. The other assertions follow by similar arguments. For $m = 1$, we may assume $n > 0$ since, otherwise, there is nothing to show. We have

$$U_1 = (sx_1)^\pi s(y_1 t)^\pi \quad \text{and} \quad V_1 = (sx_1)^\pi t(y_1 t)^\pi.$$

Let $u_1 = \sigma\left(\llbracket U_1 \rrbracket_{nk}\right)$ and $v_1 = \sigma\left(\llbracket V_1 \rrbracket_{nk}\right)$. First, assume $Z = X$. We are only interested in at most n consecutive simultaneous X_Σ^R factorizations of u_1 and v_1 because, as soon as we apply at least one X_Σ^L factorization, we know that $\equiv_{0,n}^{X}$ holds. As long as we apply only factorizations X_a^R with $a \in \mathrm{alph}(\sigma(sx_1))$, the factorization position stays in the $(sx_1)^\pi$ part of u_1 and v_1. Since the number of remaining factorizations decreases, the right parts will eventually be in relation under $\equiv_{m,0}^{X}$. If there is at least one X_a^R factorization in the sequence where a is in $\mathrm{alph}(\sigma(y_1 t)) \setminus \mathrm{alph}(\sigma(sx_1))$, the right-hand side of u_1 belongs to the $(y_1 t)^\pi$ part and the right-hand side of v_1 belongs to the $t(y_1 t)^\pi$ part; but in

22

$$u_{m+1} = \boxed{(u_m x_{m+1})\ (u_m x_{m+1})\ \ldots\ (u_m x_{m+1})\ u_m\ (y_{m+1} u_m)\ (y_{m+1} u_m)\ \ldots\ (y_{m+1} u_m)}$$

$$v_{m+1} = \boxed{\underbrace{(u_m x_{m+1})\ (u_m x_{m+1})\ \ldots\ (u_m x_{m+1})}_{\substack{\text{Repetition of the same word} \\ \text{for } nk \geq n \text{ times.}}}\ v_m\ \underbrace{(y_{m+1} u_m)\ (y_{m+1} u_m)\ \ldots\ (y_{m+1} u_m)}_{\substack{\text{Repetition of the same word} \\ \text{for } nk \geq n \text{ times.}}}}$$

Fig. 6 Schematic representation of u_{m+1} and v_{m+1}. Note that x_{m+1} and y_{m+1} are identified with $\sigma(x_{m+1})$ and $\sigma(y_{m+1})$, respectively.

both words, there are still at least $n-1$ instances of $\sigma(y_1 t)$, which implies that the right-hand sides are equal under $\equiv_{m,n-1}^{X}$. For $Z = Y$, the argumentation is symmetric, which also handles the $Z = XY$ case. The additional $C_{a,b}$ of $Z = \mathrm{WI}$ needs no special handling since it decreases the first index of $\equiv_{m,n}^{\mathrm{WI}}$ to $m - 1 = 0$ anyway.

To conclude the induction, we show

$$u_{m+1} = \sigma\left(\llbracket U_{m+1} \rrbracket_{nk}\right) \equiv_{m+1,n}^{Z} \sigma\left(\llbracket V_{m+1} \rrbracket_{nk}\right) = v_{m+1}$$

next. For a schematic representation of u_{m+1} and v_{m+1}, the reader is advised to refer to Figure 6. For convenience, we identify x_{m+1} with $\sigma(x_{m+1})$ and y_{m+1} with $\sigma(y_{m+1})$. We will only show the case $Z = \mathrm{WI}$ since this is the most difficult one and the other ones are similar. In fact, we will proof the following claim by an inner induction on l for $0 \leq l \leq n$:

$$(u_m x_{m+1})^l u_m (y_{m+1} u_m)^l \equiv_{m+1,l}^{\mathrm{WI}} (u_m x_{m+1})^l v_m (y_{m+1} u_m)^l$$

The actual assertion then follows for the case $l = n$. For $l = 0$, there is nothing to show. So, let $l > 0$. First, consider an X_a^L or X_a^R factorizations. Only two cases can emerge: the factorization happens in the $(u_m x_{m+1})^l$ part or the factorization happens in the $(y_{m+1} u_m)^l$ part (in both words simultaneously). The factorization cannot happen in the central u_m or v_m part because we have $\mathrm{alph}(v_m) = \mathrm{alph}(u_m) \subseteq \mathrm{alph}(u_m x_{m+1})$.

First, consider the case in which the factorization happens at the beginning. Clearly, if this is the case, then the factorization must occur in the first $(u_m x_{m+1})$ factor of both words at the same position:
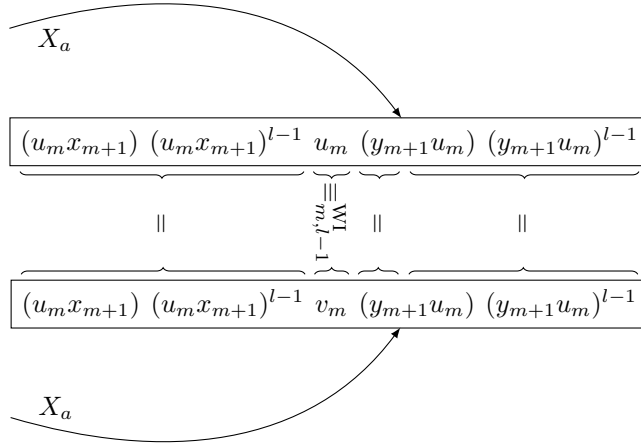

23

So, we have $(u_m x_{m+1})^l u_m (y_{m+1} u_m)^l \cdot X_a^L \equiv_{m,l-1}^{\mathrm{WI}} (u_m x_{m+1})^l v_m (y_{m+1} u_m)^l \cdot X_a^L$ because the two words are equal. For an X_a^R factorization, we have to apply induction on l and get

$$(u_m x_{m+1})^{l-1} u_m (y_{m+1} u_m)^{l-1} \equiv_{m+1,l-1}^{\mathrm{WI}} (u_m x_{m+1})^{l-1} v_m (y_{m+1} u_m)^{l-1}$$
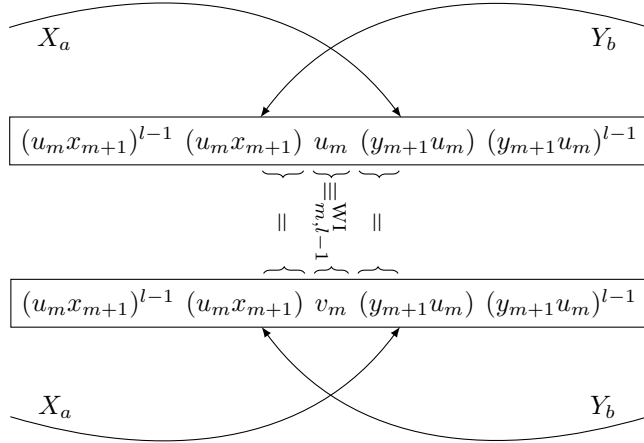
in the middle. Thus, we also get $(u_m x_{m+1})^l u_m (y_{m+1} u_m)^l \cdot X_a^R \equiv_{m+1,l-1}^{\mathrm{WI}} (u_m x_{m+1})^l v_m (y_{m+1} u_m)^l \cdot X_a^R$ (because the other word parts are equal and because $\equiv_{m+1,l-1}^{\mathrm{WI}}$ is a congruence).

If the factorization happens at the end, the situation is similar. Clearly, the factorization can only happen at the same position in the first $(y_{m+1} u_m)$ factor of each word, respectively (in fact, it can only happen within the first y_{m+1}):

We have $u_m \equiv_{m,n}^{\mathrm{WI}} v_m$ by induction on m and (because $l \leq n$) also $u_m \equiv_{m,l-1}^{\mathrm{WI}} v_m$. Because the other parts of the words are equal, respectively, and because $\equiv_{m,l-1}^{\mathrm{WI}}$ is a congruence, we get $(u_m x_{m+1})^l u_m (y_{m+1} u_m)^l \cdot X_a^L \equiv_{m,l-1}^{\mathrm{WI}} (u_m x_{m+1})^l v_m (y_{m+1} u_m)^l \cdot X_a^L$. We have $(u_m x_{m+1})^l u_m (y_{m+1} u_m)^l \cdot X_a^R \equiv_{m+1,l-1}^{\mathrm{WI}} (u_m x_{m+1})^l v_m (y_{m+1} u_m)^l \cdot X_a^R$ directly because the word parts on the right coincide.

Because Y_a^L and Y_a^R factorizations are symmetrical, the only remaining case is a $C_{a,b}$ factorization. It is not possible that $C_{a,b}$ is defined on one of the words but not on the other because in each respective part the same letters appear. If $C_{a,b}$ is defined on $(u_m x_{m+1})^l u_m (y_{m+1} u_m)^l$ and on $(u_m x_{m+1})^l v_m (y_{m+1} u_m)^l$, then the only possible situation is that the X_a^L factorization happens at the same position of the first $(y_{m+1} u_m)$ part in each word and the Y_b^R factorization happens at the same position of the last $(u_m x_{m+1})$ factor in each word:

X_a $\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ Y_b

$$\boxed{(u_m x_{m+1})^{l-1}\ (u_m x_{m+1})\ u_m\ (y_{m+1}u_m)\ (y_{m+1}u_m)^{l-1}}$$

$\qquad\qquad\qquad \| \qquad \equiv^{\mathrm{WI}}_{m,l-1} \qquad \|$

$$\boxed{(u_m x_{m+1})^{l-1}\ (u_m x_{m+1})\ v_m\ (y_{m+1}u_m)\ (y_{m+1}u_m)^{l-1}}$$

X_a $\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ Y_b

As in the previous case, we have $u_m \equiv^{\mathrm{WI}}_{m,l-1} v_m$ and, because the other parts of the words coincide respectively, also $(u_m x_{m+1})^l u_m (y_{m+1}u_m)^l \cdot C_{a,b} \equiv^{\mathrm{WI}}_{m,l-1} (u_m x_{m+1})^l v_m (y_{m+1}u_m)^l \cdot C_{a,b}$. $\qquad\qquad\square$

Theorem 3 *Let $m,n \in \mathbb{N}$. Then:*

- $\Sigma^* / \equiv^X_{m,n} \in \mathbf{R_m}$
- $\Sigma^* / \equiv^Y_{m,n} \in \mathbf{L_m}$

- $\Sigma^* / \equiv^{XY}_{m,n} \in \mathbf{R_m} \vee \mathbf{L_m}$
- $\Sigma^* / \equiv^{\mathrm{WI}}_{m,n} \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$

Proof To prove the theorem, one needs to show that the equations from Lemma 1 hold in the respective monoid. To do this, it is worthwhile to make an observation: choose $m,n \in \mathbb{N}$ and $Z \in \{X,Y,XY,\mathrm{WI}\}$ arbitrarily and let $M = \Sigma^* / \equiv^Z_{m,n}$. The observation is that an equation $\alpha = \beta$ holds in M if and only if $\sigma\left(\llbracket\alpha\rrbracket_{n\cdot M!}\right) \equiv^Z_{m,n} \sigma\left(\llbracket\beta\rrbracket_{n\cdot M!}\right)$ holds for all assignments $\sigma : \Gamma \to \Sigma^*$ where Γ is the alphabet of α and β (i.e. the set of variables appearing in α and β).

So, to show $\Sigma^* / \equiv^{\mathrm{WI}}_{m,n} \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$, it suffices to show that $\sigma\left(\llbracket U_m\rrbracket_{n\cdot M!}\right) \equiv^{\mathrm{WI}}_{m,n} \sigma\left(\llbracket V_m\rrbracket_{n\cdot M!}\right)$ holds for all assignments of variables $\sigma : \Sigma^*_m \to \Sigma^*$. However, this follows from the first assertion of Lemma 4. The same is true for $\mathbf{R_1}$ and $\mathbf{L_1}$. To show $\Sigma^* / \equiv^X_{m+1,n} \in \mathbf{R_{m+1}}$ for $m \in \mathbb{N}_0$, we use the second assertion of Lemma 4 and, for $\Sigma^* / \equiv^Y_{m+1,n} \in \mathbf{L_{m+1}}$, we use the third one.

To prove that $M/\equiv^{XY}_{m,n}$ is in $\mathbf{R_m} \vee \mathbf{L_m}$, one can recycle an observation from the proof of Theorem 2: a monoid is in the join $\mathbf{V} \vee \mathbf{W}$ of two varieties $\mathbf{V}$ and $\mathbf{W}$ if and only if it is a divisor of a direct product $M_1 \times M_2$ such that $M_1 \in \mathbf{V}$ and $M_2 \in \mathbf{W}$. Indeed, for any two congruences $\mathcal{C}_1$ and $\mathcal{C}_2$ over a monoid N, $N/(\mathcal{C}_1 \cap \mathcal{C}_2)$ is a divisor of $N/\mathcal{C}_1 \times N/\mathcal{C}_2$ (as can be shown easily). Therefore, $M/\equiv^{XY}_{m,n}$ is a divisor of the direct product of $M/\equiv^X_{m,n} \in \mathbf{R_m}$ and $M/\equiv^Y_{m,n} \in \mathbf{L_m}$. $\qquad\qquad\square$

25

To conclude this section, we finally state and prove the converse of Lemma 1. This gives us a full characterization of the Trotter-Weil Hierarchy in terms of equations (see also [12]). Note that we include this proof only for the sake of completeness. Strictly speaking, it is not necessary for the remainder of this paper.

Lemma 5 *Define the π-terms*

$$U_1 = (sx_1)^\pi s(y_1 t)^\pi \quad and \quad V_1 = (sx_1)^\pi t(y_1 t)^\pi$$

over the alphabet $\Sigma_1 = \{s, t, x_1\}$. For $m \in \mathbb{N}$, let x_{m+1} and y_{m+1} be new characters not in the alphabet Σ_m and define the π-terms

$$U_{m+1} = (U_m x_{m+1})^\pi U_m (y_{m+1} U_m)^\pi \quad and \quad V_{m+1} = (U_m x_{m+1})^\pi V_m (y_{m+1} U_m)^\pi$$

over the alphabet $\Sigma_{m+1} = \Sigma_m \uplus \{x_{m+1}, y_{m+1}\}$.

Then we have

$$M \in \mathbf{R_1} = \mathbf{L_1} = \mathbf{J} \; \Rightarrow \; U_1 = V_1 \; holds \; in \; M,$$
$$M \in \mathbf{R_{m+1}} \; \Rightarrow \; (U_m x_{m+1})^\pi U_m = (U_m x_{m+1})^\pi V_m \; holds \; in \; M,$$
$$M \in \mathbf{L_{m+1}} \; \Rightarrow \; U_m (y_{m+1} U_m)^\pi = V_m (y_{m+1} U_m)^\pi \; holds \; in \; M \; and$$
$$M \in \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}} \; \Rightarrow \; U_m = V_m \; holds \; in \; M$$

for all $m \in \mathbb{N}$.

Proof As stated in the proof of Lemma 1, the first implication is due to a well-known characterization in terms of equations of $\mathbf{J}$ [21]. The other implications can be proved by combining Lemma 4 and Theorem 2. For example, suppose $M \in \mathbf{R_{m+1}}$ and let $\sigma : (\Sigma_m \uplus \{x_{m+1}\})^* \to M$ be an arbitrary assignment of variables. We need to show $\sigma\left(\llbracket (U_m x_{m+1})^\pi U_m \rrbracket_{M!}\right) = \sigma\left(\llbracket (U_m x_{m+1})^\pi V_m \rrbracket_{M!}\right)$. By Theorem 2, there is an $n \in \mathbb{N}$ such that $u \equiv_{m+1,n}^X v$ implies $\sigma(u) = \sigma(v)$ for all $u, v \in (\Sigma_m \uplus \{x_{m+1}\})^*$. Thus, for this n, we have

$$\sigma\left(\llbracket (U_m x_{m+1})^\pi U_m \rrbracket_{M!}\right) = \sigma\left(\llbracket (U_m x_{m+1})^\pi U_m \rrbracket_{n \cdot M!}\right)$$
$$= \sigma\left(\llbracket (U_m x_{m+1})^\pi V_m \rrbracket_{n \cdot M!}\right) = \sigma\left(\llbracket (U_m x_{m+1})^\pi V_m \rrbracket_{M!}\right)$$

since we have $\llbracket (U_m x_{m+1})^\pi U_m \rrbracket_{n \cdot M!} \equiv_{m+1,n}^X \llbracket (U_m x_{m+1})^\pi V_m \rrbracket_{n \cdot M!}$ by Lemma 4. $\qquad\square$

6 Relations and Equations

Infinite Version of the Relations. So far, we have mainly considered finite words. In this section, which connects the relational approach outlined above with equations, we finally need to consider generalized words. Thus, we also define an infinite version of the above relations. Here, we allow an arbitrary number of factorizations: for $m \in \mathbb{N}_0$ and $Z \in \{X, Y, XY, \mathrm{WI}\}$, define

$$u \equiv_m^Z v \; \Leftrightarrow \; \forall n \in \mathbb{N} : u \equiv_{m,n}^Z v.$$

Notice that this definition is not useful for finite words as every one of them is in its own class.

Connecting the Relations and Equations. In order to solve the word problem for π-terms over the varieties in the Trotter-Weil Hierarchy, one can use the following connection between the relations defined above and equations in these varieties, which is straightforward if one makes the transition from finite to infinite words. Besides its use for the word problem for π-terms, this connection is also interesting in its own right as it can be used to prove or disprove equations in any of the varieties. As the class of monoids in which an equation $\alpha = \beta$ holds is a variety, one can see the assertion for the join levels as an implication of the ones for the corners.

Theorem 4 *Let α and β be two π-terms. For every $m \in \mathbb{N}$, we have:*

$$[\![\alpha]\!]_{\omega+\omega^*} \equiv_m^X [\![\beta]\!]_{\omega+\omega^*} \ \Leftrightarrow \ \alpha = \beta \ \text{holds in } \mathbf{R_m}$$

$$[\![\alpha]\!]_{\omega+\omega^*} \equiv_m^Y [\![\beta]\!]_{\omega+\omega^*} \ \Leftrightarrow \ \alpha = \beta \ \text{holds in } \mathbf{L_m}$$

$$[\![\alpha]\!]_{\omega+\omega^*} \equiv_m^{XY} [\![\beta]\!]_{\omega+\omega^*} \ \Leftrightarrow \ \alpha = \beta \ \text{holds in } \mathbf{R_m} \vee \mathbf{L_m}$$

$$[\![\alpha]\!]_{\omega+\omega^*} \equiv_m^{\text{WI}} [\![\beta]\!]_{\omega+\omega^*} \ \Leftrightarrow \ \alpha = \beta \ \text{holds in } \mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$$

Using Fact 1, which states that **DA** is equal to the union of all varieties in the Trotter-Weil Hierarchy, we immediately get the following corollary.

Corollary 1 $\left(\forall m \in \mathbb{N} : [\![\alpha]\!]_{\omega+\omega^*} \equiv_m^{XY} [\![\beta]\!]_{\omega+\omega^*} \right) \ \Leftrightarrow \ \alpha = \beta \ \text{holds in } \mathbf{DA}$

To prove Theorem 4, we need two technical lemmas. The first one basically says that a sufficiently large finite power is as good as an $\omega + \omega^*$ power.

Lemma 6 *Let $m \in \mathbb{N}_0$, $Z \in \{X, Y, XY, \text{WI}\}$ and let u and v be accessible words. Then:*

$$u \equiv_{m,n}^Z v \ \Rightarrow \ \forall 0 \le k \le n : u^{k+1} \equiv_{m,k}^Z v^{\omega+\omega^*}$$

Proof The case $m = 0$ is trivial. Therefore, let $m > 0$ and continue by induction over k. Again, the case $k = 0$ is trivial. To complete the induction, it remains to show that $uu^{k+1} \equiv_{m,k+1}^Z v^{\omega+\omega^*}$ holds for $k < n$. Obviously, $\text{alph}(u^{k+2}) = \text{alph}\left(v^{\omega+\omega^*}\right)$ is satisfied by assumption. Now assume $Z = X$. The assumption $u \equiv_{m,n}^X v$ implies $u \equiv_{m-1,n-1}^Y v$. By induction on m, this yields $u^{k+1} \equiv_{m-1,k}^Y v^{\omega+\omega^*}$ and $u \equiv_{m-1,k}^Y v$. Because $\equiv_{m-1,k}^Y$ is a congruence, this shows $uu^{k+1} \equiv_{m-1,k}^Y vv^{\omega+\omega^*} = v^{\omega+\omega^*}$. Let $a \in \text{alph}(u) = \text{alph}(v)$. It remains to show $uu^{k+1} \cdot X_a^L \equiv_{m-1,k}^Y v^{\omega+\omega^*} \cdot X_a^L$ and $uu^{k+1} \cdot X_a^R \equiv_{m,k}^X v^{\omega+\omega^*} \cdot X_a^R$. For the former, note that $u \equiv_{m,n}^X v$ implies $u \cdot X_a^L \equiv_{m-1,n-1}^Y v \cdot X_a^L$, which in turn implies $uu^{k+1} \cdot X_a^L = u \cdot X_a^L \equiv_{m-1,k}^Y v \cdot X_a^L = vv^{\omega+\omega^*} \cdot X_a^L = v^{\omega+\omega^*} \cdot X_a^L$. For the latter, note that $u \equiv_{m,n}^X v$ implies $u \cdot X_a^R \equiv_{m,n-1}^X v \cdot X_a^R$ and, thus, $u \cdot X_a^R \equiv_{m,k}^X v \cdot X_a^R$. By induction on k, we also have $u^{k+1} \equiv_{m,k}^X v^{\omega+\omega^*}$. Together these yield $uu^{k+1} \cdot X_a^R = (u \cdot X_a^R) u^{k+1} \equiv_{m,k}^X (v \cdot X_a^R) v^{\omega+\omega^*} = v^{\omega+\omega^*} \cdot X_a^R$.

The case for $Z = Y$ is symmetric and the case for $Z = XY$ follows directly. Finally, for $Z = \text{WI}$ the argumentation is analogous because, for $k > 0$, neither $uu^{k+1} \cdot C_{a,b}$ nor $v^{\omega+\omega^*} \cdot C_{a,b}$ is defined for any pair (a, b) of letters. $\square$

The second technical lemma states that, for finitely many factorization steps, only finitely many positions of $\omega + \omega^*$ are relevant.

Lemma 7 *Let $m, n \in \mathbb{N}_0$, $Z \in \{X, Y, XY, \mathrm{WI}\}$ and let γ be a π-term. Then*

$$[\![\gamma]\!]_k \equiv_{m,n}^Z [\![\gamma]\!]_{\omega+\omega^*}$$

holds for all $k \in \mathbb{N}_0$ with $k > n$.

Proof The cases for $m = 0$ or $n = 0$ are trivial. Thus, assume $m > 0$ and $n > 0$. If $\gamma = \varepsilon$ or $\gamma = a$ for an $a \in \Sigma$, then $[\![\gamma]\!]_k = \gamma = [\![\gamma]\!]_{\omega+\omega^*}$. If $\gamma = \alpha\beta$ for two π-terms α and β, then by induction $[\![\alpha]\!]_k \equiv_{m,n}^Z [\![\alpha]\!]_{\omega+\omega^*}$ and $[\![\beta]\!]_k \equiv_{m,n}^Z [\![\beta]\!]_{\omega+\omega^*}$ hold. As $\equiv_{m,n}^Z$ is a congruence, this implies $[\![\gamma]\!]_k \equiv_{m,n}^Z [\![\gamma]\!]_{\omega+\omega^*}$.

Finally, let $\gamma = (\alpha)^\pi$ for a π-term α. It remains to show that $[\![\alpha]\!]_k^k \equiv_{m,n}^Z [\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*}$. Clearly, the alphabetic condition is satisfied and, by induction, $[\![\alpha]\!]_k \equiv_{m,n}^Z [\![\alpha]\!]_{\omega+\omega^*}$ holds. For $Z = X$, let $a \in \mathrm{alph}(\alpha)$. By induction on m and n, we have $[\![\alpha]\!]_k^k = [\![\gamma]\!]_k \equiv_{m-1,n-1}^Y [\![\gamma]\!]_{\omega+\omega^*} = [\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*}$. We also have $[\![\alpha]\!]_k^k \cdot X_a^L = [\![\alpha]\!]_k \cdot X_a^L \equiv_{m-1,n-1}^Y [\![\alpha]\!]_{\omega+\omega^*} \cdot X_a^L = [\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*} \cdot X_a^L$ by definition of $[\![\alpha]\!]_k \equiv_{m,n}^X [\![\alpha]\!]_{\omega+\omega^*}$. To show $[\![\alpha]\!]_k^k \cdot X_a^R \equiv_{m,n-1}^X [\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*} \cdot X_a^R$, we use $[\![\alpha]\!]_k \cdot X_a^R \equiv_{m,n-1}^X [\![\alpha]\!]_{\omega+\omega^*} \cdot X_a^R$, which holds because of $[\![\alpha]\!]_k \equiv_{m,n}^X [\![\alpha]\!]_{\omega+\omega^*}$, and show $[\![\alpha]\!]_k^{k-1} \equiv_{m,n-1}^X [\![\alpha]\!]_{\omega+\omega}^{\omega+\omega^*}$. This is sufficient because $\equiv_{m,n-1}^X$ is a congruence and we have $uu^{\omega+\omega^*} = u^{\omega+\omega^*}$ for all words u. For showing $[\![\alpha]\!]_k^{k-1} \equiv_{m,n-1}^X [\![\alpha]\!]_{\omega+\omega}^{\omega+\omega^*}$, we write $k - 1 = k' + n$ for a $k' \in \mathbb{N}_0$ and

$$[\![\alpha]\!]_k^{k-1} = [\![\alpha]\!]_k^{k'} [\![\alpha]\!]_k^n \text{ and}$$
$$[\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*} = [\![\alpha]\!]_{\omega+\omega^*}^{k'} [\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*}.$$

Setting $k = n - 1$ in Lemma 6 yields $[\![\alpha]\!]_k^n \equiv_{m,n-1}^X [\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*}$, which concludes the proof for $Z = X$ since $\equiv_{m,n-1}^X$ is a congruence.

The case for $Z = Y$ is symmetric, which also shows the case for $Z = XY$. For $Z = \mathrm{WI}$, we note that $[\![\alpha]\!]_{\omega+\omega^*}^{\omega+\omega^*} \cdot C_{a,b}$ is defined for no pair a, b of letters. On the other hand, $[\![\alpha]\!]_k^k \cdot C_{a,b}$ can only be defined for $k = 1$, in which case we are done because we have $n = 0$.

Now, we are finally prepared to prove Theorem 4.

Proof (for Theorem 4) The proof is structurally identical for all stated varieties. Therefore, we limit our discussion to $\mathbf{R_m}$.

First, let $[\![\alpha]\!]_{\omega+\omega^*} \equiv_m^X [\![\beta]\!]_{\omega+\omega^*}$. Choose a monoid $M \in \mathbf{R_m}$ and an assignment of variables $\sigma : \Sigma^* \to M$. Because M is in $\mathbf{R_m}$, there is an $n \in \mathbb{N}$ such that $u \equiv_{m,n}^X v$ implies $\sigma(u) = \sigma(v)$ for any two words $u, v \in \Sigma^*$ (see also Theorem 2). Now, choose $c \in \mathbb{N}$ with $M! \cdot c > n$. Then by assumption and Lemma 7, we have

$$\Sigma^* \ni [\![\alpha]\!]_{M! \cdot c} \equiv_{m,n}^X [\![\alpha]\!]_{\omega+\omega^*} \equiv_{m,n}^X [\![\beta]\!]_{\omega+\omega^*} \equiv_{m,n}^X [\![\beta]\!]_{M! \cdot c} \in \Sigma^*$$

and, therefore, $\sigma([\![\alpha]\!]_{M!}) = \sigma([\![\alpha]\!]_{M! \cdot c}) = \sigma([\![\beta]\!]_{M! \cdot c}) = \sigma([\![\beta]\!]_{M!})$, which is equivalent to $\alpha = \beta$ holding in M.

Now, let $[\![\alpha]\!]_{\omega+\omega^*} \not\equiv^X_m [\![\beta]\!]_{\omega+\omega^*}$, which implies that there is an $n \in \mathbb{N}$ such that $[\![\alpha]\!]_{\omega+\omega^*} \not\equiv^X_{m,n} [\![\beta]\!]_{\omega+\omega^*}$. Define $M := \Sigma^*/{\equiv^X_{m,n}}$, which is in $\mathbf{R_m}$ (by Theorem 3), and choose $c \in \mathbb{N}$ such that $M! \cdot c > n$. Then, by assumption and Lemma 7, we have

$$\Sigma^* \ni [\![\alpha]\!]_{M! \cdot c} \equiv^X_{m,n} [\![\alpha]\!]_{\omega+\omega^*} \not\equiv^X_{m,n} [\![\beta]\!]_{\omega+\omega^*} \equiv^X_{m,n} [\![\beta]\!]_{M! \cdot c} \in \Sigma^*.$$

As assignment of variables $\sigma : \Sigma^* \to M$ choose the canonical projection. This yields $\sigma([\![\alpha]\!]_{M!}) = \sigma([\![\alpha]\!]_{M! \cdot c}) \neq \sigma([\![\beta]\!]_{M! \cdot c}) = \sigma([\![\beta]\!]_{M!})$, which means that $\alpha = \beta$ does *not* hold in M. □

7 Decidability

In the previous section, we saw that checking whether $\alpha = \beta$ holds in a variety of the Trotter-Weil Hierarchy boils down to checking $[\![\alpha]\!]_{\omega+\omega^*} \equiv^Z_m [\![\beta]\!]_{\omega+\omega^*}$ (where $\equiv^Z_m$ depends on the variety in question). In this section, we give an introduction on how to do this. The presented approach works uniformly for all varieties in the Trotter-Weil Hierarchy (in particular, it also works for the intersection levels, which tend to be more complicated) and is designed to yield efficient algorithms.

The definition of the relations which need to be tested is inherently recursive. One would factorize $[\![\alpha]\!]_{\omega+\omega^*}$ and $[\![\beta]\!]_{\omega+\omega^*}$ on the first a and/or last b (for $a, b \in \Sigma$) and test the factors recursively. Therefore, the computation is based on working with factors of words of the form $[\![\gamma]\!]_{\omega+\omega^*}$ where γ is a π-term.

What happens if we consecutively factorize at a first/last a is best understood if one considers the structure of $[\![(\alpha)^\pi]\!]_{\omega+\omega^*} = [\![\alpha]\!]^{\omega+\omega^*}_{\omega+\omega^*} = u^{\omega+\omega^*} = w$, which is schematically represented in Figure 7.
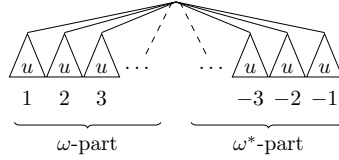
Fig. 7 Representation of $u^{\omega+\omega^*}$

Suppose u only contains a single a and we start with the whole word $w_{(-\infty,+\infty)}$. If we factorize on the first a taking the part to the right, then we end up with the factor $w_{(X_a(w;-\infty),+\infty)}$ with $X_a(w; -\infty) = (p, 1)$ where p is the single a-position in u. If we do this again, we obtain $w_{((p,2),+\infty)}$. If we now factorize on the next a but take the part to the left, then we get $w_{((p,2),(p,3))}$. Notice that the difference between 2 and 3 is 1 and that there is no way of getting a (finite) difference larger than one by factorizing on the respective first a. On the other hand, we can reach any number in $\mathbb{N}$ as long as the right position is not in the ω-part.

Notice that there is also no way of reaching $(p, -2)$ as left border without having $(q, -1)$ or $(q, -2)$ as right border for a position $q \in \mathrm{dom}(u)$. These observations (and their symmetrical duals) lead to the notion of *normalizable* pairs of positions.

Definition 2 Let γ be a π-term and let $w = [\![\gamma]\!]_{\omega+\omega^*}$. A pair (l, r) of positions in w such that l is strictly smaller than r is called *normalizable* (with respect to γ) based on the following rules:

- Any pair is normalizable with respect to $\gamma = \varepsilon$ or $\gamma = a$ for an $a \in \Sigma$.
- $(-\infty, +\infty)$ is normalizable with respect to any π-term.
- If $\gamma = \alpha\beta$ for π-terms α and β, $l \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*}) \uplus \{-\infty\}$ and $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$, then (l, r) is normalizable with respect to γ if $(l, +\infty)$ is with respect to α and $(-\infty, r)$ is with respect to β.
- If $\gamma = \alpha\beta$ for π-terms α and β and $l \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*}) \uplus \{-\infty\}$ as well as $r \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ (or $l \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*})$ as well as $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$), then (l, r) is normalizable with respect to γ if it is with respect to α (or β, respectively).
- If $\gamma = (\alpha)^\pi$ for a π-term α, $l = (l', n)$ for $l' \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and $n \in \mathbb{N} \uplus -\mathbb{N}$ and $r = +\infty$, then (l, r) is normalizable with respect to γ if $(l', +\infty)$ is with respect to α and n is in $\mathbb{N} \uplus \{-1\}$.
- If $\gamma = (\alpha)^\pi$ for a π-term α, $l = -\infty$, and $r = (r', m)$ for $r' \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and $m \in \mathbb{N} \uplus -\mathbb{N}$, then (l, r) is normalizable with respect to γ if $(-\infty, r')$ is with respect to α and m is in $\{1\} \uplus -\mathbb{N}$.
- If $\gamma = (\alpha)^\pi$ for a π-term α, $l = (l', n)$ for $l' \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and $n \in \mathbb{N} \uplus -\mathbb{N}$ and $r = (r', m)$ for $r' \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and $m \in \mathbb{N} \uplus -\mathbb{N}$, then (l, r) is normalizable with respect to γ if
  - $n \in \mathbb{N}$, $m \in -\mathbb{N}$ and $(l', +\infty)$ and $(-\infty, r')$ are normalizable with respect to α,
  - $n, m \in \mathbb{N}$ or $n, m \in -\mathbb{N}$ and in both cases $m = n$ and (l', r') is normalizable with respect to α, or
  - $n, m \in \mathbb{N}$ or $n, m \in -\mathbb{N}$ and in both cases $m = n + 1$ and $(l', +\infty)$ and $(-\infty, r')$ are normalizable with respect to α.

This definition looks cumbersome at first. All it does, however, is formalizing our previous observations. This allows us to give a formal inductive proof that these observations hold for all positions reachable by iterated first/last a factorization. First, we need to extend our notation. For this, we consider (abstract) positions $p \in \{-\infty, +\infty\} \uplus \mathrm{dom}(w)$ in a word w to be implicitly linked to w. For a pair of positions $(l, r) \in (\{-\infty\} \uplus \mathrm{dom}(w)) \times (\mathrm{dom}(w) \uplus \{+\infty\})$ in an accessible word w, we write

$$(l, r) \cdot X_a^L = (l, X_a(w; l)), \qquad (l, r) \cdot X_a^R = (X_a(w; l), r),$$
$$(l, r) \cdot Y_a^L = (l, Y_a(w; r)) \text{ and} \qquad (l, r) \cdot Y_a^R = (Y_a(w; r), r)$$

for all $a \in \mathrm{alph}\, w_{(l,r)}$. Note that we now have $w_{(l,r)} \cdot Z_a^D = w_{(l,r) \cdot Z_a^D}$ for all $Z \in \{X, Y\}$ and all $D \in \{L, R\}$. In fact, we could have used this as the definition previously. We also write $(l, r) \cdot C_{a,b}$ for $(l, r) \cdot X_a^L \cdot Y_b^R$ if $X_a(w; l)$ is strictly larger than $Y_b(w; r)$. Therefore, we also have $w_{(l,r)} \cdot C_{a,b} = w_{(l,r) \cdot C_{a,b}}$. As with words, we omit the $\cdot$ if we apply a sequence of steps longer than one, e. g. we simply write $(l, r) \cdot X_a^L Y_b^R = (l, r) \cdot X_a^L \cdot Y_b^R = (l, r) \cdot C_{a,b}$.

Lemma 8 *Let γ be a π-term and let $w = [\![\gamma]\!]_{\omega+\omega^*}$. Additionally, let (l, r) be a normalizable (with respect to γ) pair of positions in w. Then the pairs*

$$(l, r) \cdot X_a^L, (l, r) \cdot X_a^R, (l, r) \cdot Y_a^L \text{ and } (l, r) \cdot Y_a^R$$

are normalizable with respect to γ for any $a \in \mathrm{alph}(w_{(l,r)})$.

Therefore, $(-\infty, +\infty) \cdot F_1 F_2 \ldots F_n$ is normalizable with respect to γ for any $F_1, F_2, \ldots, F_n \in \{X_a^L, X_a^R, Y_a^L, Y_a^R, C_{a,b} \mid a, b \in \Sigma\}$ (if it is defined).

Proof As the cases for Y_a^L and Y_a^R are symmetrical, we only show those for X_a^L and X_a^R. Let $p = X_a(w; l)$ for an $a \in \mathrm{alph}(w_{(l,r)})$. Clearly, we have $l <_\mu p <_\mu r$, where μ is the order type of w, and we need to show that (l, p) and (p, r) are normalizable. For this, we proceed by induction on the structure of γ. The base case $\gamma = \varepsilon$ or $\gamma \in \Sigma$ is trivial.

Case 1 ($\gamma = \alpha\beta$) Define $u = [\![\alpha]\!]_{\omega+\omega^*}$ and $v = [\![\beta]\!]_{\omega+\omega^*}$. For $l \in \mathrm{dom}(u) \uplus \{-\infty\}$ and $r \in \mathrm{dom}(u)$ we have $p \in \mathrm{dom}(u)$ as well. Additionally, (l, r) needs to be normalizable with respect to α by the definition of normalizability and we can apply induction. The same argument, but on β, works for $l \in \mathrm{dom}(v)$ and $r \in \mathrm{dom}(v) \uplus \{+\infty\}$. For $l \in \mathrm{dom}(u) \uplus \{-\infty\}$ and $r \in \mathrm{dom}(v) \uplus \{+\infty\}$ we know that $(l, +\infty)$ is normalizable with respect to α and $(-\infty, r)$ is with respect to β by the definition of normalizablity. If $p \in \mathrm{dom}(u)$, then $(p, +\infty) = (l, +\infty) \cdot X_a^R$ and $(l, p) = (l, +\infty) \cdot X_a^L$. Induction yields normalizability with respect to α for both and, by the definition of normalizability, we have that (p, r) and (l, p) are normalizable with respect to γ. For $p \in \mathrm{dom}(v)$, we can apply a similar argument, as then $(-\infty, p) = (-\infty, +\infty) \cdot X_a^L$ and $(p, r) = (-\infty, r) \cdot X_a^R$ are normalizable with respect to β.

Case 2 ($\gamma = (\alpha)^\pi$) Define $u = [\![\alpha]\!]_{\omega+\omega^*}$ and let $p = (p', k)$. If $l = (l', n)$ for an $n \in \mathbb{N} \uplus -\mathbb{N}$ and $r = +\infty$, then, by the definition of normalizability, we have that $(l', +\infty)$ is normalizable with respect to α and $n \in \mathbb{N} \uplus \{-1\}$. There are two cases: for $k = n \in \mathbb{N} \uplus \{-1\}$ we know that $p' = X_a(u; l')$ and, by induction, that $(l', p'), (p', +\infty)$ are normalizable with respect to α. This yields the normalizability with respect to γ of (l, p) and $(p, +\infty)$. For $k = n + 1$ we know that $n \neq -1$ and, therefore, that $n, k \in \mathbb{N}$. We also have $p' = X_a(u; -\infty)$ and, thus, that $(-\infty, p')$ and $(p', +\infty)$ are normalizable with respect to α by induction. By definition, $(p, +\infty)$ and (l, p) are normalizable with respect to γ then. Note that k cannot have any other value than n or $n + 1$ since otherwise it could not be the smallest a-position to the right of l.

If $l = -\infty$ and $r = (r', m)$, then $k = 1$, and $p' = X_a(u; -\infty)$, which yields $(-\infty, p') = (-\infty, +\infty) \cdot X_a^L$ and $(p', +\infty) = (-\infty, +\infty) \cdot X_a^R$. By induction, both of these pairs are normalizable with respect to α and, by definition of the normalizability, $(-\infty, p)$ is normalizable with respect to γ. Furthermore in this case, we know that $(-\infty, r')$ is normalizable with respect to α and that m is in $\{1\} \uplus -\mathbb{N}$. For $m \in -\mathbb{N}$, this shows the normalizability with respect to γ of (p, r). For $m = 1$, we have $(p', r') = (-\infty, r') \cdot X_a^R$ and, by induction, its normalizability with respect to α. This yields that (p, r) is normalizable with respect to γ.

If $l = (l', n)$ and $r = (r', m)$ for $n \in \mathbb{N}$ and $m \in -\mathbb{N}$, we know that $(l', +\infty)$ and $(-\infty, r')$ are normalizable with respect to α. For $k = n \in \mathbb{N}$, we also know that $p' = X_a(u; l')$ and, therefore, that $(l', p') = (l', +\infty) \cdot X_a^L$ and $(p', +\infty) = (l', +\infty) \cdot X_a^R$ are normalizable with respect to α by induction. Then, by definition, (l, p) and (p, r) are normalizable with respect to γ. For $k = n+1 \in \mathbb{N}$ we have that $p' = X_a(u; -\infty)$ and, therefore, the normalizability with respect to α of $(-\infty, p') = (-\infty, +\infty) \cdot X_a^L$ and $(p', +\infty) = (-\infty, +\infty) \cdot X_a^R$ by induction. This yields the normalizability with respect to γ of (l, p) and (p, r).

Finally, if $l = (l', n)$ and $r = (r', m)$ for $n, m \in \mathbb{N}$ or $n, m \in -\mathbb{N}$, we know that $0 \leq m - n \leq 1$. Because p must be in between l and r, $n = m$ also implies $n = m = k$ and that p' is in between l' and r' as well as $p' = X_a(u; l')$. In that case, we have that (l', r') and, by induction, also $(l', p') = (l', r') \cdot X_a^L$ and $(p', r') = (l', r') \cdot X_a^R$ are normalizable with respect to α. This yields the normalizability with respect to γ of (l, p) and (p, r). For $m = n + 1$, we know that $(l', +\infty)$ and $(-\infty, r')$ are normalizable with respect to α. Moreover, there are only two cases: $k = n$ and $k = m$. In the former case, we have $p' = X_a(u; l')$ and the normalizability with respect to α of $(l', p') = (l', +\infty) \cdot X_a^L$ and $(p', +\infty) = (l', +\infty) \cdot X_a^R$ by induction, which yields the normalizability of (l, p) and (p, r) with respect to γ. In the latter case, we have $p' = X_a(u; -\infty)$ and the normalizability with respect to α of $(-\infty, p') = (-\infty, +\infty) \cdot X_a^L$ and $(p', r') = (-\infty, r') \cdot X_a^R$, which yields the normalizability with respect to γ of (l, p) and (p, r). $\square$

The choice of words indicates that normalizability of a pair (l, r) can be used to define a normalization. Before we give a formal – unfortunately, quite technical – definition of this, we describe its idea informally. Let us refer back to the schematic representation of $[\![(\alpha)^\pi]\!]_{\omega + \omega^*} = w$ as given in Figure 7. Basically, there are three different cases for relative positions of the left border l and the right border r which describe the factor $w_{(l,r)}$:

1. l is in the ω-part and r is in the ω^*-part,
2. l and r are either both in the ω-part or both in the ω^*-part and have the same value there, or
3. l and r are either both in the ω-part or both in the ω^*-part but r has a value exactly larger by one than l.

This is ensured by the normalizability of (l, r). Now, in the first case, we can safely move l to value 1 (the first position) and r to value -1 (the last position) without changing the described factor. In the second and third case, we can move l and r to any value – as long as we retain the difference between the values – without changing the described factor. Here, we move them to the left-most values (which are $1, 1$ or $1, 2$). Afterwards, we go on recursively.

Unfortunately, things get a bit more complicated because l might be $-\infty$ and r might be $+\infty$. In these cases, we normalize to the left-most or right-most value without changing the factor.

For concatenations of π-terms, we have a similar situation: either l and r belong both to the left or to the right factor, in which case we can continue

by normalization with respect to that, or l belongs to the left factor and r belongs to the right one. In this case, we have to continue the normalization with $(l, +\infty)$ and $(-\infty, r)$ in the respective concatenation parts, as this ensures that the described factor remains unchanged.

Formalizing these ideas results in the following inductive definition.

Definition 3 Let γ be a π-term, $w = [\![\gamma]\!]_{\omega+\omega^*}$ and (l, r) a normalizable pair of positions in w. The normalized pair $\overline{(l, r)}^\gamma = (\bar{l}, \bar{r})$ with respect to γ is defined recursively:

- For $\gamma = \varepsilon$ or $\gamma = a \in \Sigma$ define $\bar{l} = l$ and $\bar{r} = r$.
- If $\gamma = \alpha\beta$ for π-terms α and β, $l \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*}) \uplus \{-\infty\}$ and $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$, then define $\bar{l}$ as the first component of $\overline{(l, +\infty)}^\alpha$ and $\bar{r}$ as the second component of $\overline{(-\infty, r)}^\beta$.
- If $\gamma = \alpha\beta$ for π-terms α and β and $l \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*}) \uplus \{-\infty\}$ as well as $r \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ (or $l \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*})$ as well as $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$), then define $(\bar{l}, \bar{r}) = \overline{(l, r)}^\alpha$ (or $(\bar{l}, \bar{r}) = \overline{(l, r)}^\beta$, respectively).
- If $\gamma = (\alpha)^\pi$ for a π-term α, then:
  - if $l = -\infty$, define $\bar{l} = -\infty$,
  - if $r = +\infty$, define $\bar{r} = +\infty$,
  - if $l = (l', n)$ and $r = +\infty$, define $\bar{l} = (\bar{l}', \bar{n})$ with $\bar{l}'$ given by the first component of $\overline{(l', +\infty)}^\alpha$ and $\bar{n}$ given by

$$\bar{n} = \begin{cases} 1 & \text{if } n \in \mathbb{N} \\ -1 & \text{if } n = -1, \end{cases}$$

  - if $l = -\infty$ and $r = (r', m)$, define $\bar{r} = (\bar{r}', \bar{m})$ with $\bar{r}'$ given by the second component of $\overline{(-\infty, r')}^\alpha$ and $\bar{m}$ given by

$$\bar{m} = \begin{cases} 1 & \text{if } n = 1 \\ -1 & \text{if } n \in -\mathbb{N}, \end{cases}$$

  - if $l = (l', n)$ and $r = (r', m)$ with $n \in \mathbb{N}$ and $m \in -\mathbb{N}$, define $\bar{l} = (\bar{l}', 1)$ with $\bar{l}'$ being by the first component of $\overline{(l', +\infty)}^\alpha$ and define $\bar{r} = (\bar{r}', -1)$ with $\bar{r}'$ given by the second component of $\overline{(-\infty, r')}^\alpha$,
  - if $l = (l', n)$ and $r = (r', m)$ with $n = m$, define $\bar{l} = (\bar{l}', \bar{n})$ and $\bar{r} = (\bar{r}', \bar{m})$ with $(\bar{l}', \bar{r}') = \overline{(l', r')}^\alpha$ and $\bar{n} = \bar{m} = 1$, and
  - if $l = (l', n)$ and $r = (r', m)$ with $m = n + 1$, define $\bar{l} = (\bar{l}', \bar{n})$ and $\bar{r} = (\bar{r}', \bar{m})$ with $\bar{l}'$ given by the first component of $\overline{(l', +\infty)}^\alpha$, $\bar{r}'$ given by the second component of $\overline{(-\infty, r')}^\alpha$, $\bar{n} = 1$ and $\bar{m} = \bar{n} + 1 = 2$.

One should note that if we normalize a normalizable pair (l, r), then the resulting pair is normalizable itself. Indeed, if we normalize an already normalized pair again, we do not change any values. Formally, this can be proved by an induction on the structure of the π-term. As an example for such an induction, we prove the following lemma which states that normalizing a pair of positions does not change the described factor.

Lemma 9 *Let γ be a π-term and let (l, r) be a normalizable pair of positions in $w = [\![\gamma]\!]_{\omega + \omega^*}$. Then*

$$w_{(l,r)} = w_{\overline{(l,r)}^\gamma}$$

holds.

Proof Define $\overline{(l, r)}^\gamma = (\bar{l}, \bar{r})$ and proceed by induction on the structure of γ. The base cases for $\gamma = \varepsilon$ and $\gamma \in \Sigma$ are trivial.

If $\gamma = \alpha\beta$ for π-terms α and β, then define $u = [\![\alpha]\!]_{\omega + \omega^*}$ and $v = [\![\beta]\!]_{\omega + \omega^*}$. If $l \in \mathrm{dom}(u) \uplus \{-\infty\}$ and $r \in \mathrm{dom}(v) \uplus \{+\infty\}$, then

$$w_{(l,r)} = u_{(l,+\infty)} v_{(-\infty,r)} = u_{\overline{(l,+\infty)}^\alpha} v_{\overline{(-\infty,r)}^\beta} = w_{\overline{(l,r)}^\gamma}.$$

If $l \in \mathrm{dom}(u) \uplus \{-\infty\}$ and $r \in \mathrm{dom}(u)$, then

$$w_{(l,r)} = u_{(l,r)} = u_{\overline{(l,r)}^\alpha} = w_{\overline{(l,r)}^\gamma}.$$

The case $l \in \mathrm{dom}(v)$ and $r \in \mathrm{dom}(v) \uplus \{+\infty\}$ is symmetrical.

If $\gamma = (\alpha)^\pi$ for a π-term α, then define $u = [\![\alpha]\!]_{\omega + \omega^*}$. The case $l = -\infty$ and $r = +\infty$ is trivial. If $l = (l', n)$ for an $n \in \mathbb{N} \uplus -\mathbb{N}$ and $r = +\infty$, define $\bar{l}'$ by $\overline{(l', +\infty)}^\alpha = (\bar{l}', +\infty)$. For $n \in \mathbb{N}$ we then have

$$w_{(l,r)} = w_{((l',n),+\infty)} = \left(u^{\omega + \omega^*}\right)_{((l',n),+\infty)} = \left(u^{\omega + \omega^*}\right)_{((l',1),+\infty)}$$

because of $u^{\omega + \omega^*} = u u^{\omega + \omega^*}$ and further

$$w_{(l,r)} = u_{(l',+\infty)} u^{\omega + \omega^*} = u_{\overline{(l',+\infty)}^\alpha} u^{\omega + \omega^*} = u_{(\bar{l}',+\infty)} u^{\omega + \omega^*}$$
$$= \left(u^{\omega + \omega^*}\right)_{((\bar{l}',1),+\infty)} = w_{\overline{((l',n),+\infty)}^\gamma} = w_{\overline{(l,r)}^\gamma}$$

and for $n = -1$ – the only remaining case – we have

$$w_{(l,r)} = w_{((l',-1),+\infty)} = u_{(l',+\infty)} = u_{\overline{(l',+\infty)}^\alpha} = u_{(\bar{l}',+\infty)} = w_{((\bar{l}',-1),+\infty)} = w_{\overline{(l,r)}^\gamma}.$$

The case for $l = -\infty$ and $r = (r', m)$ is symmetrical.

Therefore, we can assume $l = (l', n)$ and $r = (r', m)$. The case $n \in \mathbb{N}$ and $m \in -\mathbb{N}$ is proved by a calculation similar to the one given above. For $n = m$ we have

$$w_{(l,r)} = w_{((l',n),(r',n))} = u_{(l',r')} = u_{\overline{(l',r')}^\alpha} = w_{\overline{(l,r)}^\gamma}$$

and for $m = n + 1$ we have

$$w_{(l,r)} = u_{(l',+\infty)} u_{(-\infty,r')} = u_{\overline{(l',+\infty)}^\alpha} u_{\overline{(-\infty,r')}^\alpha} = w_{\overline{(l,r)}^\gamma}.$$

$\square$

Another observation is crucial for the proof of the decidability: after normalizing a pair (l, r) the values belonging to the $\omega + \omega^*$ parts for the two positions are all in $\{1, 2, -2, -1\}$. But: there are only finitely many such positions in any word $w = \llbracket \gamma \rrbracket_{\omega + \omega^*}$ for a π-term γ. Because the normalization preserves the described factor, this means that there are only finitely many factors which can result from a sequence of first/last a factorizations.

Plugging all these ideas and observations together yields a proof for the next theorem.

Theorem 5 *For two π-terms α and β over the same alphabet Σ, it is decidable for all $m \in \mathbb{N}_0$ and all $Z \in \{X, Y, XY, \mathrm{WI}\}$ whether*

$$\llbracket \alpha \rrbracket_{\omega + \omega^*} \equiv_m^Z \llbracket \beta \rrbracket_{\omega + \omega^*}$$

holds. Furthermore, it is decidable whether

$$\forall m \in \mathbb{N} : \llbracket \alpha \rrbracket_{\omega + \omega^*} \equiv_m^{XY} \llbracket \beta \rrbracket_{\omega + \omega^*}$$

holds.

Proof We decide whether $u = \llbracket \alpha \rrbracket_{\omega + \omega^*} \equiv_m^Z \llbracket \beta \rrbracket_{\omega + \omega^*} = v$ holds by trying to find a factorization sequence $F_1 F_2 \ldots F_n$ with $F_1, F_2, \ldots, F_n \in \{X_a^L, X_a^R, Y_a^L, Y_a^R, C_{a,b} \mid a, b \in \Sigma\}$ that can be applied to u but not to v (or vice versa), i.e. we rather try to decide $u \not\equiv_m^Z v$ instead.

Which sequences need to be tested depends on the actual relation. For example, for $\equiv_m^X$, we can start with arbitrary many X_a^R factorizations but, as soon as we apply an X_a^L, Y_a^L or Y_a^R factorization, we have changed the direction and we have to decrease m by one. If we did this because of a Y_a^D factorization (with $D \in \{R, L\}$), we also need to switch to a $\equiv_{m-1}^Y$ mode, in which case we can continue with arbitrary many Y_a^L factorizations while any other factorization decreases the remaining number of direction changes and might also switch back to a $\equiv_{m-2}^X$ mode. If we want to test $\equiv_m^{\mathrm{WI}}$, we also need to allow $C_{a,b}$ factorizations and we need to count the number of direction changes appropriately. For testing $\equiv_m^{XY}$ for all $m \in \mathbb{N}$, the situation is simpler: here, we do not need to keep track of the remaining direction changes as there always is an arbitrary number of them left. Clearly, we can construct a deterministic finite automaton for any of the relations which accepts exactly those sequences $F_1 F_2 \ldots F_n$ which need to be tested.

To test whether a factorization sequence can be applied on u, we construct an additional deterministic finite automaton. The states of this automaton are the normalized pairs of positions in u (of which there only finitely many, as discussed above). The initial state is $(-\infty, +\infty)$ and all states are final. This does not result in a trivial automaton because it will not be complete in general. We have an F-labeled transition with $F \in \{X_a^D, Y_a^D, C_{a,b} \mid a, b \in \Sigma, D \in \{L, R\}\}$ from (l, r) to $\overline{(l, r) \cdot F}^\alpha$ if $(l, r) \cdot F$ is defined. Note that normalization does not change the implicitly stored factor of u by Lemma 9. This automaton, by construction, accepts exactly those factorization sequences which can be applied to u.

Using the same construction, we can also get such an automaton for v. We intersect both automata with the one which accepts the relevant factorization sequences. For the resulting pair of automata, we check the symmetric difference of the accepted languages for emptiness. It is empty if any only if u and v are in relation. □

Together with Theorem 4, this gives the following decidability result. Note that previous partial results exist: Almeida proved decidability for $\mathbf{J}$ [1,3], Almeida and Zeitoun proved it for $\mathbf{R}$ [4] and Moura for $\mathbf{DA}$ [18].

Corollary 2 *The word problems for π-terms over $\mathbf{R_m}$, $\mathbf{L_m}$, $\mathbf{R_m} \vee \mathbf{L_m}$ and $\mathbf{R_m} \cap \mathbf{L_m}$ are decidable for any $m \in \mathbb{N}$. Moreover, the word problem for π-terms over $\mathbf{DA}$ is decidable.*

8 Nondeterministic Logarithmic Space

In the presented algorithm, we construct for a π-term γ over the alphabet Σ a deterministic finite automaton which accepts exactly those factorization sequences which can be applied to $[\![\gamma]\!]_{\omega+\omega^*} = w$. For this, we had to store normalized pairs (l, r) of positions in w and we had to compute $\overline{(l, r) \cdot F}^{\gamma}$ for a factorization $F \in \{X_a^D, Y_a^D, C_{a,b} \mid a, b \in \Sigma, D \in \{L, R\}\}$. In this section, we will show that both can be done by a deterministic Turing machine in logarithmic space. Afterwards, we will show that this yields membership of the word problems for π-terms to the class of problems which can be solved by nondeterministic Turing machines within logarithmic space.

Observations and Ideas. We will start by having a close look on how to store a position p (and, therefore, a pair of positions) in the word $w = [\![\gamma]\!]_{\omega+\omega^*}$ on a Turing machine. To do this, we first store the position in γ which corresponds to p in w; this, basically, is a simple pointer. Additionally, we need to store to which value in $\mathbb{N} \uplus -\mathbb{N}$ the position p belongs for all relevant π-exponents; this, we can do by storing a pointer to the π-position in γ together with the corresponding value. If p is part of a normalized pair of positions, the values can only be in $\{1, 2, -2, -1\}$, which is a finite set and, thus, needs only finite information.

Example 1 Have a look at the π-term $\gamma = a \left(b(c)^{\pi} \right)^{\pi} a(b)^{\pi} c$. The word $[\![\gamma]\!]_{\omega+\omega^*}$ can be represented by the following tree:
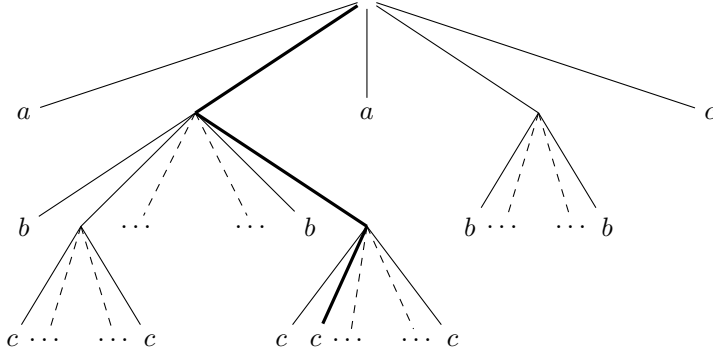
The highlighted position can also be represented this way:

$$\boxed{a \ (\ b \ (\ c \ )^{\ \pi} \ )^{\ \pi} \ a \ (\ b \ )^{\ \pi} \ c}$$

$$\uparrow \quad 2 \quad -1$$

Note that the last π-position does not have a value since it is not relevant for the position. Also note that storing the values on a second tape directly under the π-position would require linear space (for a position belonging to a normalized pair). We store them in a position/value list which will turn out to be more efficient after some modifications.

Unfortunately, this approach still requires at least linear space. But for normalized pairs, we can optimize it further if we look at the definition of the normalization. At some point the two positions l and r in the normalized pair (l, r) branch since l is a smaller position than r. This can either happen because $l = -\infty$ or $r = +\infty$, because there are sub-π-terms α and β and l belongs to $[\![\alpha]\!]_{\omega+\omega^*}$ while r belongs to $[\![\beta]\!]_{\omega+\omega^*}$, or because of a sub-$\pi$-term of the form $(\alpha)^\pi$ where l has a different value compared to r. In the former two cases, we call the pair *c-branching*[8] and in the latter case *π-branching*. Whichever is the case, we know that the values of l and r belonging to hierarchically higher π-positions are always equal and, by definition of the normalization, are equal to 1. Thus, we do not need to store these values explicitly; instead, we are going to store the branching position.

If the branching position is a π-position (i.e. the pair is π-branching), then, for a normalized pair, the values of l and r at this position can, by definition, only be 1 for l and 2 for r or 1 for l and -1 for r. This information can be stored alongside the branching position in constant space.

To store the values hierarchically below the branching position, we need to have an even closer look at normalized position pairs. Before we do this, however, it is convenient to define four position *forms*: a position is in +-form if all its values for relevant π-positions are from $\mathbb{N}$ and it is in −-form if they are from $-\mathbb{N}$. Positions in $\mp$-form may only have values from $\mathbb{N}$ for π-positions

---

[8] The "c" is for *concatenation*.

which are hierarchically lower than the first π-position with a value from $\mathbb{N}$ and positions in $\pm$-form are defined symmetrically. More formally, we define:

Definition 4 A position $p \in \mathrm{dom}(\llbracket \gamma \rrbracket_{\omega + \omega^*})$ for a π-term γ is in

- $+$-form (with respect to γ) if
    - $\gamma = \varepsilon$ or $\gamma \in \Sigma$,
    - $\gamma = \alpha\beta$ for π-terms α and β and p is in $+$-form with respect to its respective sub-π-term, or
    - $\gamma = (\alpha)^\pi$ for a π-term α, $p = (p', n)$ with $n \in \mathbb{N}$ and p' is in $+$-form with respect to α,
- $-$-form (with respect to γ) if
    - $\gamma = \varepsilon$ or $\gamma \in \Sigma$,
    - $\gamma = \alpha\beta$ for π-terms α and β and p is in $-$-form with respect to its respective sub-π-term, or
    - $\gamma = (\alpha)^\pi$ for a π-term α, $p = (p', n)$ with $n \in -\mathbb{N}$ and p' is in $-$-form with respect to α,
- $\mp$-form (with respect to γ) if
    - $\gamma = \varepsilon$ or $\gamma \in \Sigma$,
    - $\gamma = \alpha\beta$ for π-terms α and β and p is in $\mp$-form with respect to its respective sub-π-term,
    - $\gamma = (\alpha)^\pi$ for a π-term α, $p = (p', n)$ with $n \in \mathbb{N}$ and p' has $+$-form, or
    - $\gamma = (\alpha)^\pi$ for a π-term α, $p = (p', n)$ with $n \in -\mathbb{N}$ and p' has $\mp$-form, or
- $\pm$-form (with respect to γ) if
    - $\gamma = \varepsilon$ or $\gamma \in \Sigma$,
    - $\gamma = \alpha\beta$ for π-terms α and β and p is in $\pm$-form with respect to its respective sub-π-term,
    - $\gamma = (\alpha)^\pi$ for a π-term α, $p = (p', n)$ with $n \in -\mathbb{N}$ and p' has $-$-form, or
    - $\gamma = (\alpha)^\pi$ for a π-term α, $p = (p', n)$ with $n \in \mathbb{N}$ and p' has $\pm$-form.

Example 2 The position

$$\boxed{a\ (\ b\ (\ c\ )^{\ \pi}\ )^{\ \pi}\ a\ (\ b\ )^{\ \pi}\ c}$$
$$\uparrow\quad 2\quad -1$$

from the previous example is in $\mp$-form but not in any of the other three forms.

The similar position represented by

$$\boxed{a\ (\ b\ (\ c\ )^{\ \pi}\ )^{\ \pi}\ a\ (\ b\ )^{\ \pi}\ c}$$
$$\uparrow\quad 2\quad 3$$

is in $+$-form, in $\mp$-form and in $\pm$-form but it is not in $-$-form.

What use are these definitions for our goal of storing positions efficiently? If we know that a position is in $+$-form or $-$-form, then we do not need to store

whether a value is from $\mathbb{N}$ or from $-\mathbb{N}$. Similarly, if a position is in $\mp$-form or in $\pm$-form, then we only need to store one potential π-position at which the values switch form $-\mathbb{N}$ to $\mathbb{N}$ or vice versa. While this does not seem to be a huge gain since we still need to store the actual value, it will turn out to be crucial later on.

Next, we need to make some further observations. Consider a π-term γ and define $w = \llbracket \gamma \rrbracket_{\omega + \omega^*}$. If we start with a position $p \in \mathrm{dom}(w)$ in $\mp$-form and we go to the next a on the right (i. e. we compute $X_a(w; p)$), then the follow-up position p' is in $\mp$-form as well because strictly after the branching of p and p' all values for relevant π-position have to be 1 (since otherwise there would already have been an a before). By symmetry, if we start in $\pm$-form and go to the previous a on the left, the resulting position will also be in $\pm$-form.

Example 3 Look again at the π-term $\gamma = a\,(b(c)^\pi)^\pi\,a(b)^\pi c$. Suppose we are in the position p in $w = \llbracket \gamma \rrbracket_{\omega + \omega^*}$ and advance to $p' = X_b(w; p)$:
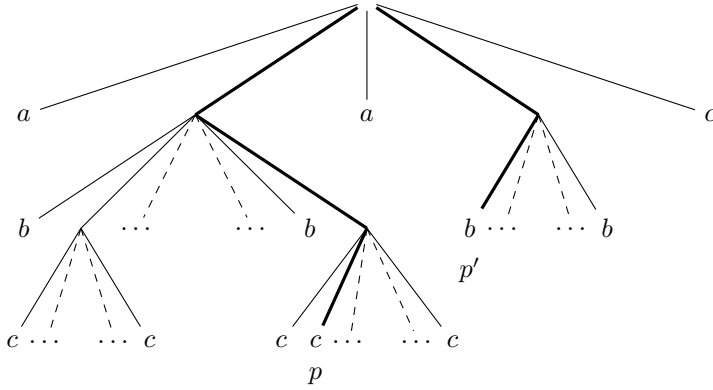
Clearly, p' is in $\mp$-form.

If we advanced to $p'' = X_c(w; p)$ instead, the situation would be as follows:
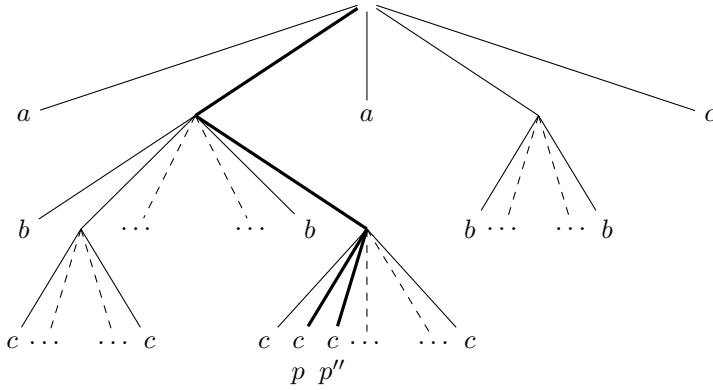
Again, p'' clearly has $\mp$-form.

More formally, we can prove the following lemma:

Lemma 10 *Let γ be a π-term and $w = [\![\gamma]\!]_{\omega+\omega^*}$. Then*

- $X_a(w; -\infty)$ *is in +-form for any $a \in \mathrm{alph}(w)$,*
- *if $l \in \mathrm{dom}(w)$ is in +-form, then so is $X_a(w; l)$ for all $a \in \mathrm{alph}(w_{(l,+\infty)})$,*
- *if $l \in \mathrm{dom}(w)$ is in $\mp$-form, then so is $X_a(w; l)$ for all $a \in \mathrm{alph}(w_{(l,+\infty)})$,*
- $Y_a(w; +\infty)$ *is in $-$-form for any $a \in \mathrm{alph}(w)$,*
- *if $r \in \mathrm{dom}(w)$ is in $-$-form, then so is $Y_a(w; r)$ for all $a \in \mathrm{alph}(w_{(-\infty,r)})$, and*
- *if $r \in \mathrm{dom}(w)$ is in $\mp$-form, then so is $Y_a(w; r)$ for all $a \in \mathrm{alph}(w_{(-\infty,r)})$.*

Proof By symmetry, we only have to prove the first three assertions.

We start by proving the first one. The cases $\gamma = \varepsilon$ and $\gamma \in \Sigma$ are trivial. For the other cases, let $a \in \mathrm{alph}(w)$ and define $p = X_a(w; -\infty)$. If $\gamma = \alpha\beta$ for two π-terms α and β, we define $u = [\![\alpha]\!]_{\omega+\omega^*}$ and $v = [\![\beta]\!]_{\omega+\omega^*}$. Now, we have $p = X_a(u; -\infty)$ or $p = X_a(v; -\infty)$. In either case, we can apply induction, which yields that p is in +-form with respect to its sub-π-term, and we are done. If $\gamma = (\alpha)^\pi$ for a π-term α, define $u = [\![\alpha]\!]_{\omega+\omega^*}$. Clearly, p has to be equal to $(p', 1)$ for $p' = X_a(u; -\infty)$. By induction, we have that p' is in +-form and, since $1 \in \mathbb{N}$, we are done.

The second assertion can be proved similarly. Let $l \in \mathrm{dom}(w)$ be in +-form. If $\gamma = \varepsilon$ or $\gamma \in \Sigma$, we do not have any remaining positions in $\mathrm{dom}(w)$ for $p = X_a(w; l)$. In the other cases, let $a \in \mathrm{alph}(w_{(l,+\infty)})$. If $\gamma = \alpha\beta$ for two π-terms α and β, we define $u = [\![\alpha]\!]_{\omega+\omega^*}$ and $v = [\![\beta]\!]_{\omega+\omega^*}$. We have to distinguish: if l and p both are in $\mathrm{dom}(u)$ (or, symmetrically, in $\mathrm{dom}(v)$), then we have $p = X_a(u; l)$ and we can apply induction. Therefore, p is in +-form with respect to α, which yields that p is also in +-form with respect to γ. If $l \in \mathrm{dom}(u)$ and $p \in \mathrm{dom}(v)$, then we know that $p = X_a(v; -\infty)$, which is in +-form with respect to β by the first assertion and, therefore, also in +-form with respect to γ.

If $\gamma = (\alpha)^\pi$ for a π-term α, define $u = [\![\alpha]\!]_{\omega+\omega^*}$. We can write $l = (l', m)$ and $p = (p', k)$ for some $m, k \in \mathbb{N} \uplus -\mathbb{N}$ and $l', p' \in \mathrm{dom}(u)$. By definition, l' is in +-form because l is so. For k, there are only two possible cases: $k = m$ and $k = m + 1$. In the former case, we know that $p' = X_a(u; l')$ and can apply induction to get that p' is in +-form with respect to α. Since l is in +-form, we also know that $k = m \in \mathbb{N}$. Together, this yields that p is in +-form with respect to γ. In the latter case $k = m + 1$, we know that $p' = X_a(u; -\infty)$ is in +-form by the first assertion. Since l is in +-form, we have $m \in \mathbb{N}$ and, therefore, also $k = m + 1 \in \mathbb{N}$. Thus, p is in +-form with respect to γ.

Now, we prove the third assertion. Let l be in $\mp$-form with respect to γ. Again, for $\gamma = \varepsilon$ or $\gamma \in \Sigma$, there is nothing to show. The case for $\gamma = \alpha\beta$ for two π-terms α and β can be proved analogously to the corresponding case in the proof for the second assertion.

If $\gamma = (\alpha)^\pi$ for a π-term α, define $u = [\![\alpha]\!]_{\omega+\omega^*}$ and let $p = X_a(w; l)$ for an $a \in \mathrm{alph}(w_{(l,+\infty)})$. We can write $l = (l', m)$ and $p = (p', k)$ for some $m, k \in \mathbb{N} \uplus -\mathbb{N}$ and $l', p' \in \mathrm{dom}(u)$. Again, there are only two possible cases:

$k = m$ and $k = m+1$. In the former case we have $p' = X_a(u; l')$. If $k = m \in \mathbb{N}$, we know that l' is in $+$-form since l has to be in $\mp$-form. By the second assertion, this yields that p' is in $+$-form as well. Therefore, we have that p is in $\mp$-form. If $k = m \in -\mathbb{N}$, we can simply apply induction and get that p' is in $\mp$-form with respect to α. This yields that p is in $\mp$-form with respect to γ. In the latter case $k = m + 1$, we know that $p' = X_a(u; -\infty)$ is in $+$-form by the first assertion and that $m \neq -1$. This yields $\mp$-form for p if $k = m+1 \in \mathbb{N}$ or $k = m + 1 \in -\mathbb{N}$. $\qquad\square$

This observation is also important for pairs which arise by consecutive factorization at the first/last a. If we start in $(-\infty, +\infty)$ and apply a sequence X of elements from $X_\Sigma^R = \left\{ X_a^R \mid a \in \Sigma \right\}$, then l' in the resulting pair $(l', +\infty) = (-\infty, +\infty) \cdot X$ will be in $\mp$-form. Equally, for a sequence over $Y_\Sigma^L = \left\{ Y_a^L \mid a \in \Sigma \right\}$, the right position will be in $\pm$-form. Can we assume that the left position is always in $\mp$-form and the right one is always in $\pm$-form? Unfortunately, the answer to this question is "no". Suppose we start in the pair (l, r) where l is in $\mp$-form and r is in $\pm$-form. If we apply X_a^L for some $a \in \Sigma$, then, obviously, we end up in a pair (l, r') whose right position is in $\mp$-form. But: the right position r' branches form l at some point and strictly below that point values for relevant π-positions are equal to 1. So, we could say that for this lower part r', indeed, is in $\pm$-form. In fact, we will prove that, strictly below the branching, l is always in $\mp$-form and r is always in $\pm$-form for any pair $(l, r) = (-\infty, +\infty) \cdot F$ where F is a sequence of elements from $F_\Sigma = \{X_a^D, Y_a^D, C_{a,b} \mid a, b \in \Sigma, D \in \{L, R\}\}$. The situation at the branching point itself depends on whether we have a c-branching or a π-branching pair. For a π-branching pair, we cannot make an assumption on the value of l and r at the actual branching π-position. We accommodate for this by having two definitions.

Definition 5 Let (l, r) be a pair of positions in $[\![\gamma]\!]_{\omega + \omega^*}$ for a π-term γ such that l is strictly smaller than r.

The pair is called *well-c-shaped* (with respect to γ) if

- $\gamma = \varepsilon$ or $\gamma \in \Sigma$,
- $\gamma = \alpha\beta$ for π-terms α and β, $l \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*}) \uplus \{-\infty\}$, $r \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and (l, r) is well-c-shaped with respect to α,
- $\gamma = \alpha\beta$ for π-terms α and β, $l \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*})$, $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$ and (l, r) is well-c-shaped with respect to β,
- $\gamma = \alpha\beta$ for π-terms α and β, $l \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*}) \uplus \{-\infty\}$, $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$, and
  - $l = -\infty$ or l is in $\mp$-form with respect to α, and
  - $r = +\infty$ or r is in $\pm$-form with respect to β,
  or
- $\gamma = (\alpha)^\pi$ for a π-term α and
  - $l = -\infty$ and $r = +\infty$,
  - $l = -\infty$ and r is in $\pm$-form,
  - l is in $\mp$-form and $r = +\infty$, or

- $l = (l', n)$ and $r = (r', n)$ for an $n \in \mathbb{N} \uplus -\mathbb{N}$ and (l', r') is well-c-shaped with respect to α,

It is called *well-π-shaped* (with respect to γ) if $\gamma \neq \varepsilon$ and $\gamma \notin \Sigma$ as well as

- $\gamma = \alpha\beta$ for π-terms α and β, $l, r \in \mathrm{dom}(\llbracket \alpha \rrbracket_{\omega+\omega^*})$ and (l, r) is well-π-shaped with respect to α,
- $\gamma = \alpha\beta$ for π-terms α and β, $l, r \in \mathrm{dom}(\llbracket \beta \rrbracket_{\omega+\omega^*})$ and (l, r) is well-π-shaped with respect to β, or
- $\gamma = (\alpha)^\pi$ for a π-term α, $l = (l', n)$, $r = (r', m)$ for $n, m \in \mathbb{N} \uplus -\mathbb{N}$ and $l', r' \in \mathrm{dom}(\llbracket \alpha \rrbracket_{\omega+\omega^*})$ and
  - $n = m$ and (l', r') is well-π-shaped with respect to α, or
  - $n \neq m$, l' is in $\mp$-form and r' is in $\pm$-form.

Finally, it is called *well-shaped* (with respect to γ) if it is well-c-shaped or well-π-shaped (with respect to γ).

The definition of well-c-shapedness is related to c-branching pairs and the definition of well-π-shapedness is related to π-branching pairs. Note that any pair (l, r) of positions in $\llbracket \gamma \rrbracket_{\omega+\omega^*}$ for a π-term is well-c-shaped if $l = -\infty$ or $r = +\infty$. This results in some asymmetry in the definition.

We proceed by showing that any pair (l, r) which arises by consecutive factorization at the first/last a is well-c-shaped or well-π-shaped.

Lemma 11 *Let (l, r) be a pair of positions in $\llbracket \gamma \rrbracket_{\omega+\omega^*}$ for a π-term γ. If (l, r) is well-shaped, then so is $(l, r) \cdot F$ for any $F \in F_\Sigma$ (if it is defined).*

Therefore, $(-\infty, +\infty) \cdot F_1 F_2 \ldots F_n$ is well-shaped with respect to γ for any $F_1, F_2, \ldots, F_n \in F_\Sigma$ (if it is defined).

Proof The second part follows from the first since $(-\infty, +\infty)$ is well-c-shaped by definition.

Let (l, r) be well-shaped and let $p = X_a(w; l)$ be defined for an $a \in \Sigma$. Due to symmetry, it remains to show that (l, p) and (p, r) are well-shaped. Note that this also includes the $C_{a,b}$ factorization.

First, consider (l, p). If $l = -\infty$, then p is in $+$-form by Lemma 10 and, therefore, also in $\pm$-form. This yields that $(-\infty, p)$ is well-c-shaped. Thus, we may safely assume that $l \neq -\infty$ and proceed by induction on the structure of γ. With this assumption, the cases $\gamma = \varepsilon$ and $\gamma \in \Sigma$ cannot occur.

If $\gamma = \alpha\beta$ for two π-terms α and β, define $u = \llbracket \alpha \rrbracket_{\omega+\omega^*}$ and $v = \llbracket \beta \rrbracket_{\omega+\omega^*}$. If l and r are both in $\mathrm{dom}(u)$, then we know, by definition of well-shapedness, that (l, r) is well-shaped with respect to α. We also know that $p = X_a(u; l)$ since p must be between l and r. By induction, we have that (l, p) is well-shaped with respect to α, which yields that (l, p) is also well-shaped with respect to γ. If $l \in \mathrm{dom}(v)$ and $r \in \mathrm{dom}(v) \uplus \{+\infty\}$, we can apply a similar argument. For $l \in \mathrm{dom}(u)$ and $r \in \mathrm{dom}(v) \uplus \{+\infty\}$, we know that (l, r) is well-c-shaped and that l is in $\mp$-form with respect to α. This yields that $(l, +\infty)$ is well-c-shaped with respect to α. By induction, we then have that $(l, p) = (l, X_a(u; l))$ is well-shaped with respect to α if $p \in \mathrm{dom}(u)$. Well-shapedness with respect to γ follows in both cases, i.e. (l, p) is well-c-shaped or (l, p) is well-π-shaped,

by definition of well-shapedness. If $p \in \mathrm{dom}(v)$, then we have $p = X_a(v; -\infty)$, which is in $+$-form (and, thus, in $\pm$-form) by Lemma 10. By definition, we have the well-c-shapedness of (l, p) with respect to γ.

If $\gamma = (\alpha)^\pi$ for a π-term α, then define $u = [\![\alpha]\!]_{\omega+\omega^*}$. We can write $l = (l', m)$ and $p = (p', k)$ for some $m, k \in \mathbb{N} \uplus -\mathbb{N}$ and $l', p' \in \mathrm{dom}(u)$. For $k = m$, we have $p' = X_a(u; l')$. This yields well-shapedness with respect to α of (l', p') by induction. By definition, we also have well-shapedness with respect to γ. For $k = m + 1$ (the only other possible case), we know $p' = X_a(u; -\infty)$ which is in $+$-form with respect to α by Lemma 10 and, therefore, also in $\pm$-form. If $r = +\infty$, then (l, r) had to be well-c-shaped and l has to be in $\mp$-form. Then, (l, p) is well-π-shaped. If $r = (r', n)$ for a $n \in \mathbb{N} \uplus -\mathbb{N}$ and $r' \in \mathrm{dom}(u)$, then n must be greater than m (with respect to $\omega + \omega^*$), because it must be greater than or equal to k. Therefore, (l, r) has to be well-π-shaped and l has to be in $\mp$-form. Again, (l, p) is well-π-shaped then, which concludes the proof that (l, p) is always well-shaped.

Next, consider (p, r). If $r = +\infty$, then (l, r) has to be well-c-shaped. This implies that $l = -\infty$ or l is in $\mp$-form. In either case, we have that $p = X_a(w; l)$ is in $\mp$-form by Lemma 10. Thus, $(p, +\infty)$ is well-c-shaped. Again, we can safely assume that $r \neq +\infty$ and proceed by induction on the structure of γ where the cases $\gamma = \varepsilon$ and $\gamma \in \Sigma$ do not occur.

If $\gamma = \alpha\beta$ for two π-terms α and β, then define $u = [\![\alpha]\!]_{\omega+\omega^*}$ and $v = [\![\beta]\!]_{\omega+\omega^*}$. The case $l \in \mathrm{dom}(u) \uplus \{-\infty\}$ and $r \in \mathrm{dom}(u)$ and the case $l, r \in \mathrm{dom}(v)$ are similar to the argumentation for (l, p). For $l \in \mathrm{dom}(u) \uplus \{-\infty\}$ and $r \in \mathrm{dom}(v)$, (l, r) has to be well-c-shaped. Therefore, r has to be in $\pm$-form with respect to β and $(-\infty, r)$ is well-c-shaped with respect to β. Induction yields the well-shapedness with respect to β, and, thus, also with respect to γ, of (p, r) if $p \in \mathrm{dom}(v)$. If $p \in \mathrm{dom}(u)$, we observe that well-c-shapedness of (l, r) yields $l = -\infty$ or l in $\mp$-form. In either case, p is in $\mp$-form by Lemma 10. Thus, (p, r) is well-c-shaped with respect to γ.

If $\gamma = (\alpha)^\pi$ for a π-term α, define $u = [\![\alpha]\!]_{\omega+\omega^*}$. We can write $p = (p', k)$ and $r = (r', n)$ for $k, n \in \mathbb{N} \uplus -\mathbb{N}$. For $k = n$, distinguish: if $l = -\infty$, then (l, r) is well-c-shaped and r has to be in $\pm$-form with respect to γ. Therefore, r' has to be in $\pm$-form with respect to α. If $l = (l', m)$ for $l' \in \mathrm{dom}(u)$ and $m \in \mathbb{N} \uplus -\mathbb{N}$ with $m <_{\omega+\omega^*} k = n$, then (l, r) is well-π-shaped and r', again, is in $\pm$-form. In both cases, this yields that $(-\infty, r')$ is well-c-shaped with respect to α. Induction yields that $(p', r') = (X_a(u; -\infty), r')$ is well-shaped with respect to α. Since $k = n$, this implies that (p, r) is well-shaped. If $l = (l', m)$ but $m = k = n$, then (l', r') has to be well-shaped with respect to α and so has to be (p', r') by induction, which again means that (p, r) is well-shaped with respect to γ. If $k <_{\omega+\omega^*} n$ and $l = -\infty$, then (l, r) is well-c-shaped, r is in $\pm$-form and p' is in $+$-form by Lemma 10. Therefore, (p, r) is well-π-shaped. If $l = (l', m)$ for $l' \in \mathrm{dom}(u)$ and $m \in \mathbb{N} \uplus -\mathbb{N}$ and $m \leq_{\omega+\omega^*} k <_{\omega+\omega^*} n$, then l' and, thus, also p' by Lemma 10 has to be in $\mp$-form while r' has to be in $\pm$-form. This concludes the proof because, then, (p, r) is well-π-shaped. $\quad\square$

By the previous lemma, we know that there is maximally one switching between values form $-\mathbb{N}$ to values from $\mathbb{N}$ in l below the branching position and that the same – but in reverse – is true for r when we consider a pair (l, r) which arises from $(-\infty, +\infty)$ by applying a sequence of factorizations from F_Σ. However, the algorithm described in the proof of Theorem 5 performs a normalization after each factorization step. So far, we have ignored this normalization but the next lemma states that normalization preserves well-shapedness.

Lemma 12 *Let (l, r) be a well-shaped, normalizable pair of positions in $[\![\gamma]\!]_{\omega+\omega^*}$ for a π-term γ over Σ. Then, $\overline{(l, r)}^\gamma$ is well-shaped.*

Proof Let $\overline{(l, r)}^\gamma = (\bar{l}, \bar{r})$. In the special case $l = -\infty$ and $r = +\infty$, we have $\overline{(l, r)}^\gamma = (l, r)$ and we are done. For the other cases, we proceed by induction over γ.

For $\gamma = \varepsilon$ and $\gamma \in \Sigma$, we have that $\overline{(l, r)}^\gamma$ is well-c-shaped. In the case $\gamma = \alpha\beta$ for two π-terms α and β, we distinguish: if $l \in \{-\infty\} \uplus \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$, we have that (l, r) must be well-c-shaped, that $\bar{l}$ is given by the first component of $\overline{(l, +\infty)}^\alpha$ and that $\bar{r}$ is given by the second component of $\overline{(-\infty, r)}^\beta$. For $l = -\infty$, we also have $\bar{l} = -\infty$ and, for $r = +\infty$, we also have $\bar{r} = +\infty$. If $l \neq -\infty$, then l must be in $\mp$-form. Note that normalization preserves the form of l when normalizing $(l, +\infty)$, so $\bar{l}$ is in $\mp$-from as well. Symmetrically, $\bar{r}$ must be in $\pm$-form if $r \neq +\infty$. Thus, $\overline{(l, r)}^\gamma$ is well-c-shaped. For $l \in \{-\infty\} \uplus \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$ and also $r \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$, we have $\overline{(l, r)}^\gamma = \overline{(l, r)}^\alpha$, which is well-shaped by induction. The same argument proves the symmetric case $l \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*})$ and $r \in \mathrm{dom}([\![\beta]\!]_{\omega+\omega^*}) \uplus \{+\infty\}$.

The remaining case is $\gamma = (\alpha)^\pi$ for a π-term α. If we have $l = -\infty$ or $r = +\infty$, then we know that l is in $\mp$-form or that r is in $\pm$-form, respectively, because (l, r) has to be well-c-shaped. As before, normalizing $(-\infty, r)$ or $(l, +\infty)$ preserves this form, which makes the result well-c-shaped as well. Therefore, we have $l = (l', n)$ and $r = (r', m)$ for $n, m \in \mathbb{N} \uplus -\mathbb{N}$ and positions $l', r' \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$. For $n = m$, we have $\bar{l} = (\bar{l}', 1)$ and $\bar{r} = (\bar{r}', 1)$ with $(\bar{l}', \bar{r}') = \overline{(l', r')}^\alpha$. By induction, we have that $\overline{(l', r')}^\alpha$ is well-c-shaped or well-π-shaped with respect to α. In either case, $(\bar{l}, \bar{r})$ is also well-c-shaped or well-π-shaped with respect to γ. For $n \neq m$, the pair (l, r) has to be well-π-shaped, l' must be in $\mp$-form and r' must be in $\pm$-form. Additionally, we have $\bar{l} = (\bar{l}', \bar{n})$ and $\bar{r} = (\bar{r}', \bar{m})$ for $\bar{n}, \bar{m} \in \mathbb{N} \uplus -\mathbb{N}$ and positions $\bar{l}', \bar{r}' \in \mathrm{dom}([\![\alpha]\!]_{\omega+\omega^*})$. Note that we also have $\bar{n} \neq \bar{m}$ (in both of the cases which can occur when normalizing). Furthermore, normalization of $(l', +\infty)$ (with respect to α) preserves the $\mp$-form of l' and normalization of $(-\infty, r')$ preserves the $\pm$-form of r', i.e. we have $\bar{l}'$ in $\mp$-form and $\bar{r}'$ in $\pm$-form, which makes $\overline{(l, r)}^\gamma$ well-π-shaped. $\square$

Combining the previous two lemmas shows the following. Suppose we start with the position pair $(-\infty, +\infty)$ and apply a single factorization $F_1 \in F_\Sigma$, then we get $(l_1', r_1') = (-\infty, +\infty) \cdot F_1$, which is well-shaped. If we then normalize (l_1', r_1'), the resulting pair $(l_1, r_1) = \overline{(l_1', r_1')}^\gamma$ is also well-shaped. We can

Name	Values	Size
ShapeType	{"well-c-shaped", "well-π-shaped"}	$\mathcal{O}(1)$
BranchPosition	a π-position or $+\infty$	$\mathcal{O}(\log n)$
BranchValues	$\{(1,2),(1,-1)\}$	$\mathcal{O}(1)$
lSwitchPosition	a π-position or $\bot$	$\mathcal{O}(\log n)$
rSwitchPosition	a π-position or $\bot$	$\mathcal{O}(\log n)$
lEndPosition	a Σ-position or $-\infty$	$\mathcal{O}(\log n)$
rEndPosition	a Σ-position or $+\infty$	$\mathcal{O}(\log n)$

Table 1 The variables required to compute the values at π-positions and their sizes.

continue with another factorization and normalize again; the result (l_1, r_2) will still be well-shaped. Therefore, all the pairs of positions (l, r) which are states in the deterministic finite automaton constructed in the proof of Theorem 5 are well-shaped and normalized.

One last small observation is necessary before we discuss the details of storing those pairs: by definition of the normalization, l and r may only have values from $\{1, -1\}$ for π-positions which are hierarchically lower than the branching position. Therefore, it is sufficient to store the potential π-position at which the values switch (from -1 to 1 for l or form 1 to -1 for r).

Storing a Well-Shaped, Normalized Pair. Summing up all of our observations results in the situation which is schematically represented in Figure 8. There-
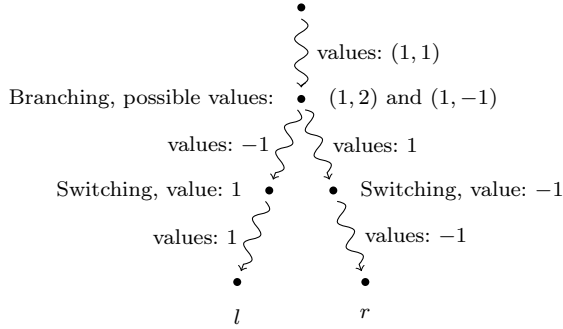

Fig. 8 Schematic representation of the possible values of a normalized pair (l, r)

fore, we can construct the values of l and r for a normalized, well-shaped pair (l, r) of positions in $[\![\gamma]\!]_{\omega+\omega^*}$ at all relevant π-positions in the π-term γ from the variables specified in Table 1.

We are going to explain the variables in more detail. Obviously, ShapeType gets the value "well-c-shaped" if (l, r) is well-c-shaped and the value "well-π-shaped" if (l, r) is well-π-shaped. For a well-c-shaped pair, we store the hierarchically lowest π-position for which l and r share the same value in BranchPosition. If l and r differ already in the hierarchically highest π-position, we store $+\infty$ in BranchPosition. For well-π-shaped pairs, we store

the π-position at which l and r branch in `BranchPosition` and their respective values there in `BranchValues`. The value of `BranchValues` is not relevant in other cases. If there is a π-position at which the values of l switch from -1 to 1 (that position is bound to be hierarchically lower than the branching position), then we store the position of the first occurrence of 1 in `lSwitchPosition`. `rSwitchPosition` stores the (potential) corresponding position for r. Finally, in `lEndPosition` and `rEndPosition` we store the position in the π-term γ which corresponds to l or r, respectively.

One may verify that all these values can be stored in the size specified in the table, where n is the length of the π-term γ seen as a finite word in $\Sigma \uplus \{\, (,), ^\pi \,\}$.

The variables are sufficient to compute the values of l and r at all relevant π-positions deterministically within logarithmic space bounds. We only discuss how to do this for r, as this version can easily be adapted for l. Assume we want to compute the value of r at a π-position given in `piPos`. Notice that `piPos` is relevant for r if and only if `rEndPosition` lies in between the opening and closing parenthesis belonging to `piPos`. Because finding the matching left or right parenthesis is a simple matter of counting the opening and closing parentheses this can be done deterministically in logarithmic space. Thus, we may safely assume that our routine is only ever called for π-positions which are relevant for r. To compute the value of r at `piPos`, we only have to know to which part of r it belongs with respect to the schematic representation in Figure 8. The first part where all values of r are equal to 1 consists of all π-positions which are hierarchically higher than `BranchPosition` (assuming that it is not equal to $\bot$), including `BranchPosition` if `ShapeType` is "well-c-shape" or excluding if `BranchPosition` is "well-π-shaped". Because we already know that both, `BranchPosition` and `piPos` are relevant for r, checking whether one is hierarchically higher than the other can be done by comparing their positions in γ, which is possible in logarithmic space. The hierarchically higher one is to the right of the lower one. If `piPos` belongs to the first part, we can return 1 immediately. Next, we check whether `BranchPosition` is equal to `piPos` and `ShapeType` is "well-π-shaped". If this is the case, we can return the value 2 or -1 depending on the value of `BranchValues`. The next part is from `BranchPosition` (excluding) to `rSwitchPosition` (excluding). Again, we can check whether `piPos` belongs to this part by comparing the position in the π-term and return 1 immediately. If `piPos` was not in any part so far, we know that it belongs to the last part from `rSwitchPosition` (including) onwards and can return -1. If `BranchPosition` is $+\infty$ or `rSwitchPosition` is $\bot$, then the corresponding parts in Figure 8 simply do not exist and we can omit the respective checks.

Compute a Follow-Up Pair. Now that we know how to store the well-shaped, normalized pair (l, r) efficiently, we need to find a way to compute the normalized follow-up pair if we apply an element from $Z_\Sigma^D = \{X_a^D, Y_a^D \mid a \in \Sigma, D \in \{L, R\}\}$ ($C_{a,b}$ needs to be handled a bit differently; see below). By symmetry, we can restrict our considerations to elements from X_Σ^D, which means that

we have to find the first a-position for an $a \in \Sigma$ on the right of the position l in $\llbracket \gamma \rrbracket_{\omega + \omega^*}$ for the π-term γ. For this, we use the additional variables from Table 2.

Name	Values	Size
CurrentPPosition	a position in γ	$\mathcal{O}(\log n)$
pBranchPosition	a π-position or $+\infty$	$\mathcal{O}(\log n)$
OpenParentheses	a value to count open parentheses	$\mathcal{O}(\log n)$

Table 2 The variables required to compute the follow-up pair and their sizes.

We start by setting pBranchPosition to $\bot$ and OpenParentheses to 0. CurrentPPosition gets assigned the next position to the right of the value of lEndPosition. The algorithm now iteratively moves CurrentPPosition to the right one position at a time. If CurrentPPosition reaches an a-position, we are done. If CurrentPPosition reaches the end of γ, then we know that there is no further a and we can stop. If CurrentPPosition reaches an opening parenthesis, then we increment OpenParentheses. If CurrentPPosition reaches a closing parenthesis and pBranchPosition is $\bot$, we distinguish: if OpenParentheses is > 0, then we decrement it and move on; if it is $= 0$, then we know that the corresponding π-position is relevant for l. We check if the value of l at that position is equal to -1. If that is the case, we can simply move on. If the value is from $\mathbb{N}$ or < -1, then we set pBranchPosition to the π-position which corresponds to the closing parenthesis and move CurrentPPosition to the matching opening parentheses using OpenParentheses to count the number of opening and closing parentheses. From that position we continue with the normal algorithm. If we then reach a closing parenthesis and pBranchPosition is equal to CurrentPPosition, then we set pBranchPosition to $\bot$ and OpenParenthesis to 0 and continue normally. Any other closing parenthesis which is reached while pBranchPosition is not equal to $\bot$ simply gets ignored.

Example 4 The only interesting part of the algorithm is that when a closing parenthesis is reached. Therefore, we explain this part in more detail. Suppose we are in the following situation[9] and want to move to the next a:

$$a \; (_1 \; (_2 \; a \;)_2 \; {}^{\pi_2} \; b \; (_3 \; c \; b \; (_4 \; b \;)_4 \; {}^{\pi_4} \;)_3 \; {}^{\pi_3} \;)_1 \; {}^{\pi_1} \; a \; (_5 \; b \;)_5 \; {}^{\pi_5} \; c$$

l: $\qquad\qquad\qquad\qquad\uparrow\qquad\qquad\qquad 2\qquad -1$

The algorithm would move CurrentPPosition to the right until it reaches $(_4$ where OpenParentheses gets increased to 1. When $)_4$ gets reached, OpenParentheses is set back to 0, which triggers a different behavior when reaching $)_3$ in the next step: pBranchPosition gets set to π_3 and CurrentPPosition gets moved to $(_3$ since l has the value 2 (and *not* -1) at π_3. From there,

---

[9] The numbers here do not have a special meaning other than to tell the parenthesis pairs and their exponents apart from one another.

`CurrentPPosition` moves back to the right until it reaches $)_3$ because $)_4$ simply gets ignored and because there is no a between $(_3$ and $)_3$. At $)_3$, `OpenPrantheses` and `pBranchPosition` are reset to 0 and $\perp$. At the next closing parenthesis $)_1$, the value of l gets checked again. But this time it is equal to -1 and we continue to the right where we find the sought-after next a.

If l had had the value -2 at π_1, then the algorithm would have moved to $(_1$ and the next a would have been the one between $(_2$ and $)_2$. In that case, `pBranchPosition` would have been set to π_1 when the final a is reached.

When the algorithm stops (and `CurrentPPosition` is not moved beyond the end of γ) then `CurrentPPosition` points to the position in γ which corresponds to the next a-position p in $[\![\gamma]\!]_{\omega+\omega^*}$. The values at the π-positions can be reconstructed from the stored variables. Again, this can be done by dividing the π-positions into parts for which the value is well-known. If a π-position is relevant for p but not for l, then the value of p at that position has to be 1.

Example 5 In the last example π_2, π_4 and π_5 are not relevant for l. If `CurrentPPosition` would end up within the pair of parentheses belonging to any of these positions, its value there would be 1.

If `pBranchPosition` has a value other than $\perp$, then p has the value of l plus 1 there. Note that the value of l cannot be -1 in that case by the definition of the algorithm. For π-positions which are relevant for l and for p and which are hierarchically higher than the position in `pBranchPosition`, we know that the values of l and p are equal. At those positions which are hierarchically lower than `pBranchPosition` and which are relevant for l and for p, the value of p has to be 1 just like in the first case. If `pBranchPosition` is $\perp$, then the values of p at positions relevant for l and for p are equal to the values of l.

Because – as said – it can be checked in logarithmic space whether a π-position is relevant for l, p or r, this yields a deterministic algorithm with logarithmic space bounds for calculating the values for p at all its relevant π-positions. It remains to test whether p is still strictly smaller than r and to compute the new values of the variables in Table 1 after an X_a^D-step and normalization. This test can be done by walking through all π-positions in γ from the right to the left and checking whether they are relevant for p and for r. If that is the case, we check their values. If the value of p is strictly smaller than that of r, we are done. If they are equal everywhere, then `CurrentPPosition` needs to be to the left of `rEndPosition` in γ.

For calculating the new values for the variables, a similar approach can be applied. Consider the (slightly more difficult) case of X_a^R where p gets the new l. We walk through the π-positions in γ from right to left and check their relevance for p and for r. If they are relevant for one or the other, we can compute the values. As long as the values are equal (which means that the position is relevant for both, p and r) we know that the π-positions belong to the part in Figure 8 where l and r share the value 1. If there is a π-position where the values are different, we can store that position in `BranchPosition`

and update `BranchValues` according to the value pair which normalization would yield. In that case, we can set `ShapeType` to "well-π-shaped". After the `BranchPosition`, we have to check p and r individually for a change from values in $\mathbb{N}$ to values in $-\mathbb{N}$ or vice versa and store the corresponding position. If the values at all positions have been equal so far and we reach a position which is relevant for r but not for p (the other way round is not possible because p has to be smaller than r), we can set `ShapeType` to "well-c-shaped" and `BranchPosition` to the last π-position where the values were equal. `ShapeType` gets also set to "well-c-shaped" if we reach `rEndPos` before there is a difference in the values.

Special Handling of $C_{a,b}$. By Theorem 3, we have to test whether $[\![\alpha]\!]_{\omega+\omega^*}$ $\equiv_m^{\text{WI}} [\![\beta]\!]_{\omega+\omega^*}$ holds in order to test whether $\alpha = \beta$ holds in $\mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$. The definition of $\equiv_m^{\text{WI}}$, however, does not only rely on factorizations at the first or last a. It uses an additional special factorization which behaves like first factorizing on the first a and then factorizing on the last b but which is only possible if the first a is to the right of the last b. As defined above, this kind of factorization is represented by $C_{a,b}$.

Clearly, $(l, r) \cdot C_{a,b}$ can only be defined if $(l, r) \cdot X_a Y_b$ is defined. But the other direction does not hold: it is possible that $(l, r) \cdot X_a Y_b$ is defined while $(l, r) \cdot C_{a,b}$ is undefined.

We have described how we can compute the follow-up values of the variables in Table 1 for a factorization at the first a and, by symmetry, also for factorizations at the last b when we start with a factor which is given by a normalized pair (l, r). We can combine these two calculations into a single one for $C_{a,b}$. For this, we assume that we have two instances of the variables in Table 2, one instance for the factorization at the first a and one instance for the factorization at the last b. For both instances, we apply the normal algorithm for the corresponding factorization. Afterwards, we check whether the position of the first a is to the left of r and whether the position of the last b is to the right of l just like we did before. For $C_{a,b}$, we simply add a third check which can be done in a similar manner to the other checks: we check whether the position of the first a is to the right of the position of the last b. If this check fails, then we know that $(l, r) \cdot C_{a,b}$ is undefined, otherwise it indeed *is* defined.

Decidability in Nondeterministic Logarithmic Space. Let us recapitulate the proof for decidability for the word problems for π-terms. First, we saw that the actual problem we needed to decide was whether $[\![\alpha]\!]_{\omega+\omega^*}$ and $[\![\beta]\!]_{\omega+\omega}$ are equivalent with respect to the relation belonging to the variety in question. To decide this, we constructed a deterministic finite automata $\mathcal{A}_\alpha$ and $\mathcal{A}_\beta$ for each of the input π-terms α and β. The automaton $\mathcal{A}_\alpha$ accepted exactly those factorization sequences $F \in F_\Sigma^*$ for which $[\![\alpha]\!]_{\omega+\omega^*} \cdot F$ (or $[\![\beta]\!]_{\omega+\omega^*} \cdot F$ in the case of $\mathcal{A}_\beta$) is defined. Additionally, we also constructed a deterministic finite automaton $\mathcal{B}$ which accepted those factorization sequences that need to be tested for the respective relation. Finally, we created for $\mathcal{A}_\alpha$ and for $\mathcal{A}_\beta$ an

automaton accepting the intersection with the sequences accepted by $\mathcal{B}$. For these automata, we checked the symmetric difference for emptiness.
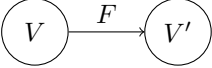
Checking the symmetric difference can be done in nondeterministic logarithmic space by a naïve (iterative) guess and check algorithm. The automaton $\mathcal{B}$ is fixed for every variety in the Trotter-Weil Hierarchy and for **DA** (or can be constructed deterministically in space logarithmic in m if one considers this as an input). Creating an automaton for the intersection can be done in deterministic logarithmic space. The only interesting operation is the construction of $\mathcal{A}_\alpha$ and $\mathcal{A}_\beta$. With the ideas from this section, we can describe how to do this in deterministic logarithmic space.

Input: a π-term α
Output: $\mathcal{A}_\alpha$ (which accepts a subset of F_Σ^*)
Algorithm:

> **for all** possible values V of the variables in Table 1 **do**
>> **for all** $F \in F_\Sigma$ **do**
>>> Compute the values V' of the variables for the follow-up pair after application of F and normalization.
>>> If there is such a follow-up pair, then output the transition

>> **end for**
> **end for**
> Mark the state which represents $(-\infty, +\infty)$ as initial state.
> Mark any state as final.

In difference to our previous construction, the states in this automaton are not the normalized pairs of positions anymore. Instead, we are representing them by their corresponding values for the variables in Table 1. Of course, not all possible values of the variables represent a valid normalized pair, but, since the corresponding states cannot be reached from the initial state, these errors do not affect the result.

These considerations allow us to state the following theorem.

Theorem 6 *Each of the word problems for π-terms over $\mathbf{R_m}$, $\mathbf{L_m}$, $\mathbf{R_m} \vee \mathbf{L_m}$, $\mathbf{R_m} \cap \mathbf{L_m}$ and $\mathbf{DA}$ can be solved by a nondeterministic Turing machine in logarithmic space (for every $m \in \mathbb{N}$).*

9 Deterministic Polynomial Time

While NL is quite efficient from a complexity class perspective, directly translating the algorithm to polynomial time does not result in a better running time than the algorithm for **DA** given by Moura [18]. However, with some additional tweaks, the algorithm's efficiency can be improved.

Encoding and Calculating Positions. Because we are not limited in space, we can encode a position $p \in \mathrm{dom}(w)$ with $w = \llbracket \gamma \rrbracket_{\omega + \omega^*}$ for a π-term γ simply by storing the values at the relevant π-positions and the corresponding position in γ. Clearly, we can obtain the values of p at a given π-position in constant time. Additionally, we can store a (possibly normalized) pair (l, r) by storing l and r. If we want to normalized such a pair, we walk through all relevant π-positions and update the values of l and r there, which requires at most linear time. Similarly, we can test whether a position is strictly smaller (or larger) than an other position in linear time.

Suppose we have stored a position $p \in \mathrm{dom}(w)$ and want to compute $X_a(w; p)$ for an $a \in \Sigma$. We can re-use the algorithm which we used previously to solve the problem in logarithmic space (see page 47). This algorithm moves a pointer, which belongs to a position in γ, to the right in every step. The only time it is moved to the left is when it hits a closing parenthesis and p has a value at the corresponding π-position which is *not* -1. In that case, we move to the matching opening parenthesis and continue to move to the right from there until we hit the closing parenthesis again. Note that there is no "back-setting to the left" in that process. Therefore, we can compute the position of the next a in at most quadratic time.

Computing the Automata and Equivalence Test. As we did before, we construct a (not necessarily complete) deterministic finite automaton for each input π-term γ. The automaton accepts exactly those sequences of factorizations which can be applied to $\llbracket \gamma \rrbracket_{\omega + \omega^*}$ and which need to be tested for the variety in question. We will only demonstrate the details of the construction for the variety $\mathbf{R_{m+1}} \cap \mathbf{L_{m+1}}$ and the input term α; the construction for the other varieties is similar. We need the variables `core` and `fringe`, which contain subsets of $(\mathrm{dom}(u) \times \mathrm{dom}(u)) \times \{1, 2, \ldots, m\}$ where $u = \llbracket \gamma \rrbracket_{\omega + \omega^*}$ such that all pairs $(l, r) \in \mathrm{dom}(u) \times \mathrm{dom}(u)$ are normalized. The algorithm works as follows:
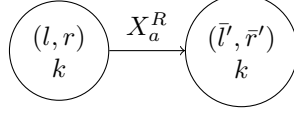
> `core` $\leftarrow \emptyset$
> `fringe` $\leftarrow \{((-\infty, +\infty), m)\}$
> **while** `fringe` $\neq \emptyset$ **do**
> > Remove $((l, r), k)$ from `fringe`
> > **if** $k > 0$ **then**
> > > **for all** $a \in \Sigma$ **do**
> > > > Compute $(l', r') = (l, r) \cdot X_a^L$ if defined $\triangleright$ time limited by $\mathcal{O}(n^2)$
> > > > **if** (l', r') is defined **then**
> > > > > $(\bar{l}', \bar{r}') \leftarrow \overline{(l', r')}^\alpha$ $\triangleright$ time limited by $\mathcal{O}(n)$
> > > > > Save transition

$$\overbrace{\substack{(l, r) \\ k}} \xrightarrow{X_a^L} \overbrace{\substack{(\bar{l}', \bar{r}') \\ k - 1}}$$
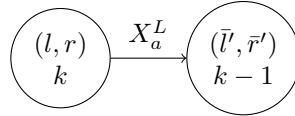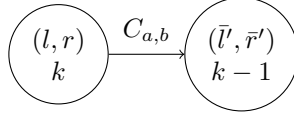
> > > > > Add $((\bar{l}', \bar{r}'), k - 1)$ to `fringe` unless it is in `core`

end if
Compute $(l', r') = (l, r) \cdot X_a^R$ if it is defined
if (l', r') is defined **then**
$(\bar{l}', \bar{r}') \leftarrow \overline{(l', r')}^{\alpha}$
Save transition

$$\left(\begin{matrix}(l,r)\\k\end{matrix}\right) \xrightarrow{X_a^R} \left(\begin{matrix}(\bar{l}',\bar{r}')\\k\end{matrix}\right)$$

Add $((\bar{l}', \bar{r}'), k)$ to `fringe` unless it is in `core`
end if
Handle Y_a^L and Y_a^R analogously
for all $b \in \Sigma$ **do** ▷ Special case which is only required for $\equiv_m^{\mathrm{WI}}$
Compute $(l', r') = (l, r) \cdot C_{a,b}$ if it is defined
if (l', r') is defined **then**
$(\bar{l}', \bar{r}') \leftarrow \overline{(l', r')}^{\alpha}$
Save transition

$$\left(\begin{matrix}(l,r)\\k\end{matrix}\right) \xrightarrow{C_{a,b}} \left(\begin{matrix}(\bar{l}',\bar{r}')\\k-1\end{matrix}\right)$$

Add $((\bar{l}', \bar{r}'), k-1)$ to `fringe` unless it is in `core`
end if
end for
end for
Add $((l, r), k)$ to `core`
end if
end while
Set $((-\infty, +\infty), m)$ as initial state
Mark all states as final states

Clearly, the resulting automaton accepts the desired factorization sequences. We know that any normalized pair can be encoded by the variables in Table 1. Therefore, the number of such pairs is limited by $\mathcal{O}(n^5)$ when n is the length of γ seen as a finite word. This directly yields that the constructed automaton has at most $\mathcal{O}(n^5 m)$ states. Because the outer loop handles any $((l, r), k)$ at most once the algorithm's running time is limited by $\mathcal{O}(n^7 m)$.

After the construction of the two automata for α and β, we need to test them for equivalence. They are equivalent if and only if $[\![\alpha]\!]_{\omega + \omega^*} \equiv_m^{\mathrm{WI}} [\![\beta]\!]_{\omega + \omega^*}$ holds. The test for equivalence can be done by the algorithm of Hopcroft and Karp [9] in almost linear time in the size of the automata. Since the number of states is bounded by $\mathcal{O}(n^5 m)$, the total running time of our complete algorithm is dominated by $\mathcal{O}(n^7 m^2)$.

The automata for the varieties $\mathbf{R_m}$, $\mathbf{L_m}$ and $\mathbf{R_m} \vee \mathbf{L_m}$ need to store whether the last factor was obtained by an element from X_Σ^D or by an element from Y_Σ^D. This information, however, is only of constant size and, therefore, does

not change the asymptotic running time of the overall algorithm. For **DA** we can omit the counting for m in k by Fact 1, which even reduces the number of states.

This finally shows the following theorem.

Theorem 7 *The word problems for π-terms over $\mathbf{R_m}$, $\mathbf{L_m}$, $\mathbf{R_m} \vee \mathbf{L_m}$ and $\mathbf{R_m} \cap \mathbf{L_m}$ can be solved by a deterministic algorithm with running time in $\mathcal{O}(n^7 m^2)$ where n is the length of the input π-terms. Moreover, the word problem for π-terms over $\mathbf{DA}$ can be solved by a deterministic algorithms in time $\mathcal{O}(n^7)$.*

10 Separability

Two languages $L_1, L_2 \subseteq \Sigma^*$ are *separable* by a variety $\mathbf{V}$ if there is a language $S \subseteq \Sigma^*$ with $L_1 \subseteq S$ and $L_2 \cap S = \emptyset$ such that S can be recognized by a monoid $M \in \mathbf{V}$. The *separation problem* of a variety $\mathbf{V}$ is the problem to decide whether two regular input languages of finite words are separable by $\mathbf{V}$.

We are going to show the decidability of the separations problems of $\mathbf{R_m}$ for all $m \in \mathbb{N}$ as well as for $\mathbf{DA}$ using the techniques presented in this paper[10]. Note that, by symmetry, this also shows decidability for $\mathbf{L_m}$.

The general idea is as follows. If the input languages are separable, then we can find a separating language S which is recognized by a monoid in the variety in question. We can do this by recursively enumerating all monoids and all languages in a suitable representation. For the other direction, we show that, if the input languages are inseparable, then there are π-terms α and β which witness their inseparability. Since we can also recursively enumerate these π-terms, we have decidability.

To construct suitable π-terms we need an additional combinatorial property of the $\equiv_{m,n}^X$ relations (which, in a slightly different form, can also be found in [15]).

Lemma 13 *Let $n, m \in \mathbb{N}$ with $m \geq 2$ and let $u \equiv_{m,n}^X v$ for two accessible words u and v. Then, $u \cdot X_a^L \equiv_{m,n-1}^X v \cdot X_a^L$ holds for all $a \in \mathrm{alph}(u) = \mathrm{alph}(v)$.*

Proof We prove the lemma by induction over n. For $n = 1$, the assertion is satisfied by definition. Therefore, assume we have $u \equiv_{m,n+1}^X v$ and we want to show $u_0 := u \cdot X_a^L \equiv_{m,n}^X v \cdot X_a^L =: v_0$. We already have $u_0 \equiv_{m-1,n}^Y v_0$ by definition of $\equiv_{m,n+1}^X$. This especially implies $\mathrm{alph}(u_0) = \mathrm{alph}(v_0)$ since we have $m \geq 2$ and $n \geq 1$, as well as $u_0 \equiv_{m-1,n-1}^Y v_0$. Additionally, we have

$$u_0 \cdot X_b^L = u \cdot X_b^L \equiv_{m-1,n}^Y v \cdot X_b^L = v_0 \cdot X_b^L$$

for all $b \in \mathrm{alph}(u_0) = \mathrm{alph}(v_0)$, which implies $u_0 \cdot X_b^L \equiv_{m-1,n-1}^Y v_0 \cdot X_b^L$. All that remains to be shown is that $u_0 \cdot X_b^R \equiv_{m,n-1}^X v_0 \cdot X_b^R$ holds for all

---

[10] Decidability of the separation problem of **DA** is already known [22]. The proof, however, uses a fixed point algorithm, which is different from our approach.

$b \in \mathrm{alph}(u_0) = \mathrm{alph}(v_0)$. Applying induction on $u \cdot X_b^R \equiv_{m,n}^X v \cdot X_b^R$ (for the same a) yields $u \cdot X_b^R X_a^L \equiv_{m,n-1}^X v \cdot X_b^R X_a^L$. Since we have $u_0 \cdot X_b^R = u \cdot X_a^L X_b^R = u \cdot X_b^R X_a^L$ and $v_0 \cdot X_b^R = v \cdot X_a^L X_b^R = v \cdot X_b^R X_a^L$, we are done. $\square$

Using this property, one can prove the following lemma about the π-term construction.

Lemma 14 *Let M be a monoid, $\varphi : \Sigma^* \to M$ a homomorphism and $m \in \mathbb{N}_0$. Let $(u_n, v_n)_{n \in \mathbb{N}_0}$ be an infinite sequence of word pairs $(u_n, v_n)_{n \in \mathbb{N}_0}$ with*

- $u_n, v_n \in \Sigma^*$,   – $\varphi(u_n) = m_u$ *and*
- $u_n \equiv_{m,n}^X v_n$,   – $\varphi(v_n) = m_v$

for fixed monoid elements $m_u, m_v \in M$ and all $n \in \mathbb{N}_0$. Then, the sequence yields π-terms α and β (over Σ) such that $\varphi(\llbracket \alpha \rrbracket_{M!}) = m_u$, $\varphi(\llbracket \beta \rrbracket_{M!}) = m_v$ and $\llbracket \alpha \rrbracket_{\omega + \omega^*} \equiv_m^X \llbracket \beta \rrbracket_{\omega + \omega^*}$ hold.

Before we give a proof of the general case, we give a separate one for the case $m = 1$. It is basically an adaption of the ideas from the proof showing decidability of the separation problem for the variety **J** of $\mathcal{J}$-trivial monoids given by van Rooijen and Zeitoun [30] to our setting.

Proof ($m = 1$) This proof is based on Simon's Factorization Forest Theorem [26]. For a finite word $w \in \Sigma^+$, a factorization tree is a rooted, finite, unranked, labeled ordered tree such that

- the tree's root is labeled with w,
- the leaves are labeled with letters (from Σ) and
- any internal node has at least two children and, if its children are labeled with $w_1, w_2, \ldots, w_k \in \Sigma^+$, then the node is labeled with $w_1 w_2 \ldots w_k$.

For every homomorphism $\psi : \Sigma^* \to N$ into a monoid N, Simon's Factorization Forest Theorem yields a factorization tree for every finite word $w \in \Sigma^+$ such that ψ maps the labels of a node's children to the same idempotent in N if the node has at least three children. Furthermore, the tree's height[11] is finite and limited by some constant that solely depends on $|N|$ (and, in particular, not on w).

Before we begin with the actual proof, we note that, if we remove pairs from the sequence $(u_n, v_n)_{n \in \mathbb{N}_0}$ and still have an infinite sequence, then the resulting sequence still satisfies all conditions stated above, in particular $u_n \equiv_{1,n}^X v_n$.

We extend φ into a homomorphism $\Sigma^* \to M \times 2^\Sigma$ which maps a word w to its alphabet $\mathrm{alph}(w)$ for the second component.[12] Then, we observe that there has to be an infinite subsequence such that all first components as well as all second components have the same alphabet. Indeed, these two alphabets have to coincide by the definition of $\equiv_{1,n}^X$! We remove all other words from the sequence. If the remaining words u_n and v_n are all empty (i.e. they have

---

[11] A single node has height 0.

[12] 2^Σ is the monoid of all subsets of Σ with taking union as the monoid's operation.

alphabet $\emptyset$), we can choose $\alpha = \beta = \varepsilon$ as well. Otherwise, we apply Simon's Factorization Forest Theorem to the remaining words u_n and v_n, which yields a sequence of factorization tree pairs $(T_{u,n}, T_{v,n})$. We first construct α from $(T_{u,n})_{n \in \mathbb{N}_0}$ such that we have $\varphi(\llbracket \alpha \rrbracket_{M!}) = m_u$ and the following conditions:

- If $w \in \Sigma^*$ is a subword[13] of u_n for an $n \in \mathbb{N}_0$, then w is a subword of $\llbracket \alpha \rrbracket_{\omega + \omega^*}$.
- If $w \in \Sigma^*$ is a subword of $\llbracket \alpha \rrbracket_{\omega + \omega^*}$, then it is a subword of all u_n with $n \geq n_0$ for an $n_0 \in \mathbb{N}_0$.

Afterwards, we proceed with $(T_{v,n})_{n \in \mathbb{N}_0}$ to construct β in the same manner.

We may assume that all trees $T_{u,n}$ have the same height H as the height is bounded by a constant and we can remove all words u_n from the underlying sequence which yield a tree not of height H. If H is zero, all trees consist of a single leaf and all words u_n consist of a single letter. Among these, one letter $a \in \Sigma$ has to appear infinitely often; we remove all other words from the sequence and choose $\alpha = a$. Clearly, all conditions for α are satisfied.

For $H > 0$, we consider the situation at the root of each $T_{u,n}$. Let $u_{n,1}, u_{n,2}, \ldots, u_{n,K_n}$ be the labels of the root's children in $T_{u,n}$. If the sequence $(K_n)_{n \in \mathbb{N}_0}$ is bounded, there is an infinite subsequence such that K_n is equal to a specific $K \geq 2$ for all indexes n of the subsequence; we remove all words not belonging to this subsequence. In the result, there is an infinite subsequence such that, for each sequence $(u_{n,k})_{n \in \mathbb{N}_0}$ with $1 \leq k \leq K$, all $u_{n,k}$ get mapped to the same monoid element by φ; we remove all other words. As each child of the root yields a subtree, taking these subtrees gives K infinite sequences of factorization trees of height $H - 1$. Applying induction on H yields $\alpha_1, \alpha_2, \ldots, \alpha_K$. We define $\alpha := \alpha_1 \alpha_2 \ldots \alpha_K$. Because $\alpha_1, \alpha_2, \ldots, \alpha_K$ satisfy the conditions stated above for their respective subtree sequence, so does α for $(T_{u,n})_{n \in \mathbb{N}_0}$.

If the sequence $(K_n)_{n \in \mathbb{N}_0}$ is unbounded, we can, without loss of generality, assume $K_n \geq 3$ for all $n \in \mathbb{N}_0$ and that it is strictly increasing – again taking the appropriate infinite subsequence. Also, we can assume that all $u_{n,1}, u_{n,2}, \ldots, u_{n,K_n}$ get mapped to the same idempotent $e \in M \times 2^\Sigma$. Choose $w \in \varphi^{-1}(e)$ arbitrarily and define $\alpha := (w)^\pi$. Note that we now have $\mathrm{alph}(u_{n,1}) = \mathrm{alph}(u_{n,2}) = \cdots = \mathrm{alph}(u_{n,K_n}) = \mathrm{alph}(u_n) = \mathrm{alph}(w)$ for all $n \in \mathbb{N}_0$. Therefore, α satisfies the conditions above.

All which remains to be shown is that we now have $\llbracket \alpha \rrbracket_{\omega + \omega^*} \equiv_1^X \llbracket \beta \rrbracket_{\omega + \omega^*}$. The important observation here is that $w_1 \equiv_{1,n}^X w_2$ with $n \in \mathbb{N}_0$ holds if and only if w_1 and w_2 have the same subwords of length $\leq n$. This means we have to show that $\llbracket \alpha \rrbracket_{\omega + \omega^*}$ and $\llbracket \beta \rrbracket_{\omega + \omega^*}$ have the same subwords (of arbitrary length). To show the subword equality, assume w is a subword of $\llbracket \alpha \rrbracket_{\omega + \omega^*}$ (without loss of generality). By the conditions above, w is a subword of all u_n with $n \geq n_0$ for an $n_0 \in \mathbb{N}_0$. Let $\tilde{n} = \max\{n_0, |w|\}$. Since we have $u_{\tilde{n}} \equiv_{1,\tilde{n}}^X v_{\tilde{n}}$ and by applying our observation regarding subwords and $\equiv_{1,\tilde{n}}^X$, w is a subword of $v_{\tilde{n}}$ and, thus, a subword of $\llbracket \beta \rrbracket_{\omega + \omega^*}$. $\qquad\square$

---

[13] Recall that a finite word $u = a_1 a_2 \ldots a_n$ with $a_i \in \Sigma$ is a subword of a (not necessarily finite) word v if we can write $v = v_0 a_1 v_1 a_2 v_2 \ldots a_n v_n$ for some words $v_0, v_1, \ldots, v_n$.

Next, we give a proof for the general case $m \geq 1$.

Proof (of Lemma 14) The assertion is trivial for $m = 0$. The case $m = 1$ has already been covered. For $m > 1$, we proceed by induction over $|\Sigma|$. For $\Sigma = \emptyset$, we set $\alpha = \beta = \varepsilon = u_n = v_n$. For $|\Sigma| > 0$, remember the observation from the previous proof: if we take an infinite subsequence $(u_n', v_n')_{n \in \mathbb{N}_0}$ of $(u_n, v_n)_{n \in \mathbb{N}_0}$, this sequence will still satisfy all conditions of the lemma. In particular, we will still have $u_n' \equiv_{m,n}^X v_n'$ for all $n \in \mathbb{N}_0$.

Now, we factorize $u_n = w_{n,0} a_{n,0} w_{n,1} a_{n,1} \ldots w_{n,K_n} a_{n,K_n} w_{n,K_n+1}$ for all $n \in \mathbb{N}_0$ such that $\mathrm{alph}(w_{n,k}) = \mathrm{alph}(u_n) \setminus \{a_{n,k}\}$ for all $k \in \{0, 1, \ldots, K_n\}$ and $\mathrm{alph}(w_{n,K_n+1}) \subsetneq \mathrm{alph}(u_n)$. If the sequence $(K_n)_{n \in \mathbb{N}_0}$ is bounded, let K be one of the numbers which appear infinitely often in it and restrict all further considerations to the corresponding subsequence of words. If $(K_n)_{n \in \mathbb{N}_0}$ is unbounded, let $K = |M|^2 + 1$ and remove all word pairs (u_n, v_n) for which K_n is smaller than K from the sequence. For all $k \in \{0, 1, \ldots, K\}$, a single letter $a_k \in \Sigma$ has to appear infinitely often in the sequence $(a_{n,k})_{n \in \mathbb{N}_0}$ because Σ is of finite size. We restrict our consideration to the corresponding subsequence. Then, we define $x_{n,k} = u_n \cdot X_{a_0}^R X_{a_1}^R \ldots X_{a_{k-1}}^R X_{a_k}^L$ and $y_{n,k} = v_n \cdot X_{a_0}^R X_{a_1}^R \ldots X_{a_{k-1}}^R X_{a_k}^L$ for $k \in \{0, 1, \ldots, K\}$ as well as $x_{n,K+1} = u_n \cdot X_{a_0}^R X_{a_1}^R \ldots X_{a_K}^R$ and $y_{n,K+1} = v_n \cdot X_{a_0}^R X_{a_1}^R \ldots X_{a_K}^R$. We, thus, have $u_n = x_{n,0} a_0 x_{n,1} a_1 \ldots x_{n,K} a_K x_{n,K+1}$ and $v_n = y_{n,0} a_0 y_{n,1} a_1 \ldots y_{n,K} a_K y_{n,K+1}$ for all $n \in \mathbb{N}_0$. Because K is constant, we can safely assume that φ maps all elements of the sequence $(x_{n,k})_{n \in \mathbb{N}_0}$ (for every $k \in \{0, 1, \ldots, K+1\}$) to the same element $s_k \in M$: one element has to appear infinitely often and we take the corresponding subsequence. In the same way, we can ensure that φ maps all element of $(y_{n,k})_{n \in \mathbb{N}_0}$ to the same element $t_k \in M$ (again, for all $k \in \{0, 1, \ldots, K+1\}$). By removing the first $K + 2$ pairs of words, we can also ensure $u_n \equiv_{m,n+K+2}^X v_n$ for all $n \in \mathbb{N}_0$. This implies $x_{n,k} \equiv_{m,n+K+2-k-1}^X y_{n,k}$ for all $n \in \mathbb{N}_0$ and all $k \in \{0, 1, \ldots, K\}$ by Lemma 13. Directly by the definition of $\equiv_{m,n}^X$, we already have $x_{n,K+1} \equiv_{m,n+K+2-K-1}^X y_{n,K+1}$ and, therefore, $x_{n,k} \equiv_{m,n}^X y_{n,k}$ for all $k \in \{0, 1, \ldots, K+1\}$. We can apply induction to $(x_{n,k}, y_{n,k})_{n \in \mathbb{N}_0}$ for $k \in \{0, 1, \ldots, K\}$ since we have $a_k \notin \mathrm{alph}(x_{n,k})$ by construction. This yields π-terms $\alpha_0, \alpha_1, \ldots, \alpha_K, \beta_0, \beta_1, \ldots, \beta_K$. If $(K_n)_{n \in \mathbb{N}_0}$ was bounded, then $\mathrm{alph}(x_{n,K+1}) = \mathrm{alph}(y_{n,K+1}) \subsetneq \mathrm{alph}(u_n) = \mathrm{alph}(v_n)$ holds and we can apply induction as well, which yields π-terms α_{K+1} and β_{K+1}. Setting $\alpha = \alpha_0 a_0 \alpha_1 a_1 \ldots \alpha_K a_K \alpha_{K+1}$ and $\beta = \beta_0 a_0 \beta_1 a_1 \ldots \beta_K a_K \beta_{K+1}$ satisfies $[\![\alpha]\!]_{\omega+\omega^*} \equiv_m^X [\![\beta]\!]_{\omega+\omega^*}$ since $\equiv_m^X$ is a congruence. If $(K_n)_{n \in \mathbb{N}_0}$ was unbounded, we set $K = |M|^2 + 1$ and, by the pigeon hole principle, there are $i, j \in \{0, 1, \ldots, K\}$ with $i < j$ and

$$s_0 \varphi(a_0) s_1 \varphi(a_1) \ldots s_i \varphi(a_i) = s_0 \varphi(a_0) s_1 \varphi(a_1) \ldots s_j \varphi(a_j) \text{ and}$$
$$t_0 \varphi(a_0) t_1 \varphi(a_1) \ldots t_i \varphi(a_i) = t_0 \varphi(a_0) t_1 \varphi(a_1) \ldots t_j \varphi(a_j).$$

We define

$$\alpha = \alpha_0 a_0 \alpha_1 a_1 \ldots \alpha_i a_i \left(\alpha_{i+1} a_{i+1} \alpha_{i+2} a_{i+2} \ldots \alpha_j a_j \right)^\pi \alpha_{K+1} \text{ and}$$
$$\beta = \beta_0 a_0 \beta_1 a_1 \ldots \beta_i a_i \left(\beta_{i+1} a_{i+1} \beta_{i+2} a_{i+2} \ldots \beta_j a_j \right)^\pi \beta_{K+1}$$

where α_{K+1} and β_{K+1} are obtained by using induction on m (and symmetry) for the sequences $(x_{n,K+1})_{n\in\mathbb{N}_0}$ and $(y_{n,K+1})_{n\in\mathbb{N}_0}$, i.e. we have $\varphi(\llbracket\alpha_{K+1}\rrbracket_{M!}) = s_{K+1}$, $\varphi(\llbracket\beta_{K+1}\rrbracket_{M!}) = t_{K+1}$ and $\llbracket\alpha_{K+1}\rrbracket_{\omega+\omega^*} \equiv^Y_{m-1} \llbracket\beta_{K+1}\rrbracket_{\omega+\omega^*}$. Therefore, we have $\varphi(\llbracket\alpha\rrbracket_{M!}) = m_u$ and $\varphi(\llbracket\beta\rrbracket_{M!}) = m_v$ by construction. We also have $\llbracket\alpha\rrbracket_{\omega+\omega^*} \equiv^X_m \llbracket\beta\rrbracket_{\omega+\omega^*}$: for the part left up to and including the $(\cdot)^\pi$, we have equivalence by induction and because $\equiv^X_m$ is a congruence; the right part, we cannot reach by arbitrarily many X_a^D factorizations since all letters appear infinitely often in the $(\cdot)^\pi$ part and, if we reach it by using at least one Y_a^D factorization, we are done since we have $\llbracket\alpha\rrbracket_{\omega+\omega^*} \equiv^Y_{m-1} \llbracket\beta\rrbracket_{\omega+\omega^*}$. $\qquad\square$

We can now plug everything together and prove the following theorem.

Theorem 8 *For each $m \in \mathbb{N}$, the separation problem for $\mathbf{R_m}$ is decidable and so is the one for $\mathbf{L_m}$.*

Proof We only consider $\mathbf{R_m}$ as the case for $\mathbf{L_m}$ is symmetric. If the input languages are separable, we can find a separating language by enumerating all candidates. If the languages are inseparable, we have to apply the previous lemma. As regular languages, the input languages $L_1 \subseteq \Sigma^*$ and $L_2 \subseteq \Sigma^*$ can be recognized by monoids M_1 and M_2 via the homomorphisms φ_1 and φ_2 and the homomorphism can be computed. Therefore, they are also recognized by $M := M_1 \times M_2$ via the homomorphism φ which maps a finite word to a pair whose first component is determined by φ_1 and whose second component is determined by φ_2. Let $n \in \mathbb{N}_0$ be arbitrary. Since we have $\Sigma^*/\equiv^X_{m,n} \in \mathbf{R_m}$ and since L_1 and L_2 cannot be separated by $\mathbf{R_m}$, there have to be finite words $u_n, v_n \in \Sigma^*$ with $u_n \in L_1$, $v_n \in L_2$ and $u_n \equiv^X_{m,n} v_n$; otherwise, we could construct a separating language. The homomorphism φ has to map infinitely many elements of the sequence $(u_n, v_n)_{n\in\mathbb{N}_0}$ to the same element in M since M is finite. If we remove all other elements, we still have an infinite sequence $(u_n, v_n)_{n\in\mathbb{N}_0}$ with $u_n \equiv^X_{m,n} v_n$ for all $n \in \mathbb{N}_0$ which also satisfies all conditions of Lemma 14. Therefore, there are π-terms α and β with $\llbracket\alpha\rrbracket_{\omega+\omega^*} \equiv^X_m \llbracket\beta\rrbracket_{\omega+\omega^*}$, $\varphi(\llbracket\alpha\rrbracket_{M!}) \in \varphi(L_1)$ and $\varphi(\llbracket\beta\rrbracket_{M!}) \in \varphi(L_2)$. Since we can test whether $\llbracket\alpha\rrbracket_{\omega+\omega^*} \equiv^X_m \llbracket\beta\rrbracket_{\omega+\omega^*}$ holds for any two π-terms α and β by Theorem 5, we can also recursively enumerate all possible π-term pairs and check whether the conditions above are met. We know that we can find such a pair if L_1 and L_2 are inseparable. On the other hand, suppose L_1 and L_2 can be separated by $S \subseteq \Sigma^*$ which is recognized by the monoid $N \in \mathbf{R_m}$ via a homomorphism $\psi : \Sigma^* \to N$ and we have found a pair α and β with $\llbracket\alpha\rrbracket_{\omega+\omega^*} \equiv^X_m \llbracket\beta\rrbracket_{\omega+\omega^*}$, $\varphi(\llbracket\alpha\rrbracket_{M!}) \in \varphi(L_1)$ and $\varphi(\llbracket\beta\rrbracket_{M!}) \in \varphi(L_2)$. Then, we have $\varphi(\llbracket\alpha\rrbracket_{N!\cdot M!}) = \varphi(\llbracket\alpha\rrbracket_{M!}) \in \varphi(L_1)$ and, thus, $\llbracket\alpha\rrbracket_{N!\cdot M!} \in L_1$ as well as $\llbracket\beta\rrbracket_{N!\cdot M!} \in L_2$ (by a similar argument). Also, $\alpha = \beta$ holds in $\mathbf{R_m}$ by Theorem 4, which implies $s := \psi(\llbracket\alpha\rrbracket_{N!\cdot M!}) = \psi(\llbracket\alpha\rrbracket_{N!}) = \psi(\llbracket\beta\rrbracket_{N!}) = \psi(\llbracket\beta\rrbracket_{N!\cdot M!})$. If we have $s \in \psi(S)$, then we have $\llbracket\beta\rrbracket_{N!\cdot M!} \in S \cap L_2$; otherwise, we have $\llbracket\alpha\rrbracket_{N!\cdot M!} \in L_1$ but $\llbracket\alpha\rrbracket_{N!\cdot M!} \notin S$ and, thus, a contradiction in either case. $\qquad\square$

Since two languages are separable by $\mathbf{R_m}$ for $m = |\Sigma| + 1$ if they are separable by $\mathbf{DA}$ [31], we also get decidability of the separation problem of $\mathbf{DA}$, which has already been shown by Place, van Rooijen and Zeitoun [22].

Corollary 3 *The separation problem for* **DA** *is decidable.*

References

1. Almeida, J.: Implicit operations on finite $\mathcal{J}$-trivial semigroups and a conjecture of I. Simon. Journal of Pure and Applied Algebra **69**(3), 205 – 218 (1991)
2. Almeida, J.: Finite Semigroups and Universal Algebra. World Scientific (1994)
3. Almeida, J.: Finite semigroups: an introduction to a unified theory of pseudovarieties. In: G.M. dos Gomes Moreira da Cunha, P.V.A. da Silva, J.É. Pin (eds.) Semigroups, Algorithms, Automata and Languages, pp. 3–64. World Scientific (2002)
4. Almeida, J., Zeitoun, M.: An automata-theoretic approach to the word problem for ω-terms over **R**. Theoretical Computer Science **370**(1), 131–169 (2007)
5. Diekert, V., Gastin, P., Kufleitner, M.: A survey on small fragments of first-order logic over finite words. Int. J. Found. Comput. Sci. **19**, 513–548 (2008)
6. Eilenberg, S.: Automata, Languages, and Machines, vol. B. Academic press (1976)
7. Gerhard, J., Petrich, M.: Varieties of bands revisited. Proceedings of the London Mathematical Society **58**(3), 323–350 (1989)
8. Hall, T., Weil, P.: On radical congruence systems. Semigroup Forum **59**(1), 56–73 (1999)
9. Hopcroft, J., Karp, R.: A linear algorithm for testing equivalence of finite automata. Tech. rep., Cornell University (1971)
10. Huschenbett, M., Kufleitner, M.: Ehrenfeucht-Fraïssé games on omega-terms. In: E.W. Mayr, N. Portier (eds.) STACS 2014, Proceedings, *LIPIcs*, vol. 25, pp. 374–385. Dagstuhl Publishing, Dagstuhl, Germany (2014)
11. Krohn, K., Rhodes, J.L., Tilson, B.: Homomorphisms and semilocal theory. In: M.A. Arbib (ed.) Algebraic Theory of Machines, Languages, and Semigroups, chap. 8, pp. 191–231. Academic Press, New York and London (1968)
12. Kufleitner, M., Lauser, A.: The join levels of the Trotter-Weil Hierarchy are decidable. In: B. Rovan, V. Sassone, P. Widmayer (eds.) MFCS 2012, Proceedings, *LNCS*, vol. 7464, pp. 603–614. Springer (2012)
13. Kufleitner, M., Weil, P.: On the lattice of sub-pseudovarieties of **DA**. Semigroup Forum **81**, 243–254 (2010)
14. Kufleitner, M., Weil, P.: The FO^2 alternation hierarchy is decidable. In: P. Cégielski, A. Durand (eds.) CSL 2012, Proceedings, *LIPIcs*, vol. 16, pp. 426–439. Dagstuhl Publishing, Dagstuhl, Germany (2012)
15. Kufleitner, M., Weil, P.: On logical hierarchies within FO^2-definable languages. Logical Methods in Computer Science **8**(3), 1–30 (2012)
16. Lodaya, K., Pandya, P., Shah, S.: Marking the chops: an unambiguous temporal logic. In: IFIP TCS 2008, Proceedings, IFIP, pp. 461–476. Springer (2008)
17. McCammond, J.P.: Normal forms for free aperiodic semigroups. Int. J. Algebra Comput. **11**(5), 581–625 (2001)
18. Moura, A.: The word problem for ω-terms over **DA**. Theoretical Computer Science **412**(46), 6556–6569 (2011)
19. Perrin, D., Pin, J.É.: Infinite words, *Pure and Applied Mathematics*, vol. 141. Elsevier, Amsterdam (2004)
20. Pin, J.É.: Varieties of Formal Languages. North Oxford Academic Publishers Ltd (1986)
21. Pin, J.É.: Varieties of Formal Languages. North Oxford Academic (1986)
22. Place, Th., van Rooijen, L., Zeitoun, M.: Separating regular languages by piecewise testable and unambiguous languages. In: MFCS 2013, Proceedings, pp. 729–740. Springer (2013)
23. Schützenberger, M.: On finite monoids having only trivial subgroups. Information and Control **8**, 190–194 (1965)
24. Schwentick, Th., Thérien, D., Vollmer, H.: Partially-ordered two-way automata: A new characterization of **DA**. In: W. Kuich, G. Rozenberg, A. Salomaa (eds.) DLT 2001, Proceedings, *LNCS*, vol. 2295, pp. 239–250. Springer (2002)
25. Simon, I.: Piecewise testable events. In: Autom. Theor. Form. Lang., 2nd GI Conf., *LNCS*, vol. 33, pp. 214–222. Springer (1975)

26. Simon, I.: Factorization forests of finite height. Theoretical Computer Science **72**(1), 65–94 (1990)
27. Tesson, P., Thérien, D.: Diamonds are forever: The variety **DA**. In: G.M. dos Gomes Moreira da Cunha, P.V.A. da Silva, J.É. Pin (eds.) Semigroups, Algorithms, Automata and Languages, pp. 475–500. World Scientific (2002)
28. Thérien, D., Wilke, Th.: Over words, two variables are as powerful as one quantifier alternation. In: Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, pp. 234–240. ACM (1998)
29. Trotter, P., Weil, P.: The lattice of pseudovarieties of idempotent semigroups and a non-regular analogue. Algebra Universalis **37**(4), 491–526 (1997)
30. van Rooijen, L., Zeitoun, M.: The separation problem for regular languages by piecewise testable languages. CoRR **abs/1303.2143** (2013)
31. Weis, Ph., Immerman, N.: Structure Theorem and Strict Alternation Hierarchy for FO^2 on Words. Logical Methods in Computer Science **5**(3), 1–23 (2009)