ORDERINGS ON WORDS AND PERMUTATIONS

Matthew McDevitt

A Thesis Submitted for the Degree of PhD
at the
University of St Andrews

2019

Orderings on Words and Permutations

Matthew McDevitt

University of
St Andrews

This thesis is submitted in partial fulfilment for the degree of

Doctor of Philosophy (PhD)

at the University of St Andrews

May 2019

Declarations

Candidate's declaration

I, Matthew McDevitt, do hereby certify that this thesis, submitted for the degree of PhD, which is approximately 40 000 words in length, has been written by me, and that it is the record of work carried out by me, or principally by myself in collaboration with others as acknowledged, and that it has not been submitted in any previous application for any degree.

I was admitted as a research student at the University of St Andrews in September 2015.

I received funding from an organisation or institution and have acknowledged the funder(s) in the full text of my thesis.

Date   Signature of candidate

Supervisor's declaration

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of PhD in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date   Signature of supervisor

Permission for publication

In submitting this thesis to the University of St Andrews we understand that we are giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. We also understand, unless exempt by an award of embargo as requested below, that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker, that this thesis will be electronically accessible for personal or research use and that the library has the right to migrate this thesis into new electronic forms as required to ensure continued access to the thesis.

I, Matthew McDevitt, confirm that my thesis does not contain any third-party material that requires copyright clearance.

The following is an agreed request by candidate and supervisor regarding the publication of this thesis:

Printed copy

No embargo on print copy.

Electronic copy

No embargo on electronic copy.

Date Signature of Candidate

Date Signature of Supervisor

Underpinning Research Data or Digital Outputs

Candidate's declaration

I, Matthew McDevitt, hereby certify that no requirements to deposit original research data or digital outputs apply to this thesis and that, where appropriate, secondary data used have been referenced in the full text of my thesis.

Date Signature of candidate

Abstract

Substructure orderings are ubiquitous throughout combinatorics and all of mathematics. In this thesis we consider various orderings on words, as well as the consecutive involvement ordering on permutations. Throughout there will be a focus on deciding certain order-theoretic properties, primarily the properties of being well-quasi-ordered (WQO) and of being atomic.

In Chapter 1, we establish the background material required for the remainder of the thesis. This will include concepts from order theory, formal language theory, automata theory, and the theory of permutations. We also introduce various orderings on words, and the consecutive involvement ordering on permutations.

In Chapter 2, we consider the prefix, suffix and factor orderings on words. For the prefix and suffix orderings, we give a characterisation of the regular languages which are WQO, and of those which are atomic. We then consider the factor ordering and show that the atomicity is decidable for finitely-based sets. We also give a new proof that WQO is decidable for finitely-based sets, which is a special case of a result of Atminas et al. [3].

In Chapters 3 and 4, we consider some general families of orderings on words. In Chapter 3 we consider orderings on words which are *rational*, meaning that they can be generated by transducers. We discuss the class of *insertion relations* introduced in a paper by the author [26], and introduce a generalisation. In Chapter 4, we consider three other variations of orderings on words. Throughout these chapters we prove various decidability results.

In Chapter 5, we consider the consecutive involvement on permutations. We generalise our results for the factor ordering on words to show that WQO and atomicity are decidable. Through this investigation we answer some questions which have been asked (and remain open) for the involvement on permutations [27].

Acknowledgements

I would like to thank my supervisor Nik Ruškuc for his patience and encouragement, and for suggesting interesting research questions for me to pursue. His advice on tackling complex problems and maintaining a hopeful attitude in times of difficulty will stay with me for years to come. I should also thank the EPSRC, without whose generous support this thesis would not have been possible.

My time at St Andrews would not have been what it was without the company of the friends and colleagues I spent it with. I would like to thank Raad Kohli, with whom I shared many long mathematical discussions and tremendously enjoyable conversations about all aspects of life. I would also like to thank Ashley Clayton, for the countless hours spent hanging out in his office and for introducing me to levels of pun-based humour I had previously not thought to be possible.

I would like to thank my parents and my brother Adam for their love and support, and for helping me to see the bigger picture. Finally I would like to thank my friends Chris Gerrard and Lewis Bage for sticking by me through the high and lows of my life over the last few years, and for all the immensely fun evenings spent together.

Contents

Chapter 1

Introduction and preliminaries

Substructure orderings are ubiquitous throughout combinatorics and all of mathematics. Many important classes of combinatorial objects are downward-closed under some substructure ordering, and so are defined by a minimal set of objects (an *anti-chain*) which their elements avoid under that ordering. The best known example of this is the set of planar graphs, which are precisely those avoiding K_5 and $K_{3,3}$ under the graph minor ordering [33]. Moreover, the Robertson-Seymour Theorem [31] asserts that any anti-chain under the graph minor ordering is finite. A consequence of this is that, as with planar graphs, the set of graphs embeddable on any fixed 2-dimensional manifold is defined by a finite anti-chain which it avoids under the graph minor ordering. By contrast, there exist infinite anti-chains under both the subgraph and induced subgraph orderings: an example for both is the set of cycle graphs C_n.

If a set does not contain any infinite anti-chains or infinite descending chains then it is said to be *well-quasi-ordered* (WQO). The WQO Problem for a particular ordering asks whether the downward-closed class avoiding a given finite set is WQO. This problem has received significant research attention, and the nature of the problem can change drastically depending on the chosen substructure ordering, even if one restricts their attention to a single combinatorial class. Returning to graphs, a result of Ding [10] asserts that a set of graphs which is downward-closed under the subgraph ordering is WQO if and only if it contains finitely many cycles and double-ended forks, whereas for the induced subgraph ordering, no such characterisation is known [23]. Elsewhere, the problem has been studied for the involvement ordering on permutations [6], the minor ordering on matroids [16] and the embedding ordering on tournaments [8]. In each of these cases, and in the case of the induced subgraph ordering, the problem has only been solved for bases containing one or two elements, and remains open in general. All of this points towards the need for an understanding of how different combinatorial classes, and different orderings on them, can produce different order-theoretic properties.

Words are perhaps the simplest combinatorial objects, and yet they present many interesting insights into WQO for combinatorics as a whole. The two best known orderings on words are the subword (subsequence) ordering and the factor (contiguous subsequence) ordering. Higman's Lemma [15] asserts that the subword ordering is WQO. By way of contrast, the factor (contiguous subword) ordering is not, since the set $\{ab^ia \mid i = 1, 2, \dots\}$ is an infinite anti-chain. Moreover, Higman's Lemma and its generalisation, Kruskal's Tree Theorem [24], underpin virtually every significant WQO result in combinatorics (see the survey article [18] for more details on this). It is therefore useful to understand orderings on words, as they can have implications for WQO and other order-theoretic properties in broader contexts.

In the literature, variations of orderings on words have also been considered, especially in relation to the WQO property. Ehrenfeucht et al. [11] introduced a class of orderings $\leq_I$, where I is a fixed finite set of words and $u \leq_I v$ if v can be obtained from u by repeated insertions of words from I. These orderings were introduced in connection with proving when certain context-free languages are regular. In their paper, Ehrenfeucht et al. prove the surprising result that the ordering $\leq_I$ is WQO if and only if the insertion closure of I is a regular language. Moreover, they generalise the famous Myhill-Nerode characterisation of regular languages [29] to say that a language is regular if and only if it is upward-closed under some WQO which respects concatenation. To give another example, Aichinger et al. [1] introduced what they call the *embedding* ordering $\leq_E$, where $u \leq_E v$ if v can be obtained from u by inserting letters after their first appearance. In their paper, it is proved that this ordering is WQO, and this is used to prove results on universal algebra. These results indicate the potential for further work in the intersection of WQO theory with formal language theory, and with wider areas of mathematics.

Permutations can be seen as a generalisation of words, and as relational structures are also akin to graphs. For permutations, the ordering which has received the most attention is the involvement ordering, which stipulates that $\sigma \leq \tau$ if τ has a subsequence which is order isomorphic to σ. This is an analogue of the subword ordering on words, but is not WQO. This ordering has been the subject of much investigation, especially in regards to the WQO property (see [2, 27]), and an understanding of permutation orderings could have implications for combinatorics in general. For instance, corresponding to each permutation σ is a graph G_σ such that G_σ is an induced subgraph of G_τ whenever $\sigma \leq \tau$ (see [3]). In the setting of permutations, the analogue of the factor ordering on words is the consecutive involvement ordering, where $\sigma \leq \tau$ if τ has a contiguous subsequence which is order isomorphic to σ. It is natural to investigate this simpler ordering, as it is interesting in its own right, and could lead to insights about the involvement ordering.

The purpose of this thesis is to examine various orderings on words, as well as the consecutive involvement ordering on permutations. Throughout there will be a focus on order-theoretic properties, primarily WQO and atomicity, and on decidability.

In Chapter 1, we establish the background material required for the remainder of the thesis. This will include concepts from order theory, formal language theory, automata theory and the theory of permutations. We also introduce the subword, factor, prefix and suffix orderings on words, and the consecutive involvement ordering on permutations. We also formally describe the two problems which are to permeate the thesis, namely the WQO Problem and the Atomicity Problem, which asks whether the downward-closed set avoiding a given finite set is atomic, i.e. if it is not the union of two downward-closed proper subsets.

In Chapter 2, we consider orderings on words which are known from the literature, namely the prefix, suffix and factor orderings. We begin with the prefix ordering and give a characterisation of the regular languages which are WQO, and of those which are atomic. The avoidance set of a finite (indeed, regular) set under the prefix ordering is always regular, so these results constitute solutions to both the WQO Problem and the Atomicity Problem for the prefix ordering. By appealing to the natural symmetry between the prefix and suffix orderings, we give analogous results for the latter. We then turn our attention to the factor ordering and show that the Atomicity Problem is decidable. To do this we introduce a graph $G(C)$ called the *factor graph* corresponding to a finitely-based set C, whereby every sufficiently long word in C corresponds uniquely to a path in $G(C)$. Deciding atomicity will then essentially boil down to testing atomicity in the contiguous subpath ordering on the set of paths in $G(C)$, and performing an analysis on the set of words in C which are not sufficiently long. Using these graphs, we also give a new proof that the WQO Problem is decidable, which is a special case of a result of Atminas et al. [3].

One feature that all of the aforementioned orderings on words have in common is that, given words u, v satisfying $u \leq v$, the word v can be obtained from u by inserting words according to some rule. Indeed, for the subword ordering, one can insert *any* word in *any position*, whereas for the factor ordering, one can insert words *only* before the first letter and after the last letter, and *no* insertions are allowed elsewhere. Similar descriptions can be formulated for the prefix and suffix orderings, and also for the orderings of Ehfenfeucht et al. and of Aichinger et al. In Chapters 3 and 4, we consider some general families of orderings on words, each of which arises from some set of insertion rules which further develop the ideas described above.

First, in Chapter 3, we consider certain orderings on words which are *rational*, meaning that they can be generated by transducers. In Section 3.1 we consider the class of *insertion relations*, which arise from certain transducers which can insert letters from a fixed alphabet at each state. This section will largely be based on the paper [26] written by the author, in which this class of relations was first introduced. We provide a number of examples, and show that it is decidable whether a given insertion relation is an ordering. We also show that being WQO is decidable for a specific subclass of insertion relations, called *left-most insertion relation*. In Section 3.2 we consider the class of *word-insertion relations*, which are a generalisation of insertion relations, and arise from certain transducers whose states each insert from a fixed set of words rather than from an alphabet. We draw some interesting comparisons between this class and the class of insertion relations, and give a sufficient condition for the new class to be orderings.

Next, in Chapter 4, we consider three other variations of orderings on words. First we consider the classes of *L-subword relations*, which arise from fixing a language L and inserting words from L. We show that it is decidable whether a given L-subword relation is an ordering when L is a regular language. We then consider the class of *(k,l)-factor orderings*, which arise from fixing two natural numbers k, l and inserting words amongst the first k and last l letters of a word. For this class of orderings, we show that the WQO Problem is decidable. We then consider the class of *I-factor orderings*, which arise from fixing a finite set of words I and applying repeated insertions from I in the middle of a word and arbitrary insertions at the start and end of a word. In this sense, these orderings mimic both the orderings of Ehrenfeucht et al. and the factor ordering. We conclude the chapter by drawing some interesting comparisons between these orderings, the orderings of Ehrenfeucht et al., and the factor ordering.

In Chapter 5 we turn our attention to the consecutive involvement ordering on permutations. Our main results are that both the WQO Problem and Atomicity Problem are decidable. To prove these we generalise our methods from the factor ordering on words, and again employ a graph $G(C)$ corresponding to a finitely-based set C. A distinction between this setting and the previous one is that a given path in $G(C)$ need not correspond to a unique element in C, in which case we say that the path is *ambiguous*. We will show that unless the graph $G(C)$ is strongly connected, the presence of an ambiguous path will prevent the class C from being atomic. We will therefore explore the ambiguity property in detail and show that it is decidable whether a given $G(C)$ exhibits it. Deciding atomicity will then essentially boil down to testing for this condition and performing the same analysis on $G(C)$ as we did in the word case. Deciding WQO will prove more complex, and will rely on a detailed understanding of how this non-uniqueness condition manifests within a particular set C.

1.1 Orderings

Let $\leq$ be a relation on a set S. We say $\leq$ is:

- *reflexive* if $x \leq x$ for all $x \in S$;
- *anti-symmetric* if $x \leq y$ and $y \leq x$ imply $x = y$;
- *transitive* if $x \leq y$ and $y \leq z$ imply $x \leq z$.

The relation $\leq$ is an *ordering* on S if it is reflexive, anti-symmetric and transitive. Some examples of orderings are the usual ordering on $\mathbb{Z}$ and the divisibility relation on $\mathbb{N}$. The divisibility relation on $\mathbb{Z}$ is not an ordering, since 1 and -1 are distinct elements of $\mathbb{Z}$ which divide each other, so the relation is not anti-symmetric.

The orderings we will be concerned with in this thesis will be *substructure relations* on words and permutations. Substructure relations are common throughout combinatorics and algebra, with some well-known examples being the subgraph relation, the induced subgraph relation and the subgroup relation. An example of a substructure relation which is *not* an ordering is the normal subgroup relation, since it is not transitive: if $H = \langle(12)(34)\rangle$ and

$$K = \langle(12)(34), (13)(24), (14)(23)\rangle$$

then $H \trianglelefteq K \trianglelefteq A_4$ but $H \ntrianglelefteq A_4$.

We say two elements $x, y \in S$ are *comparable* under $\leq$ if either $x \leq y$ or $y \leq x$. A subset of S is a *chain* under $\leq$ if any two of its elements are comparable, and an *anti-chain* if no two of its elements are comparable. A sequence of the form $x_1 < x_2 < \dots$ is an *ascending chain* and a sequence of the form $x_1 > x_2 > \dots$ is a *descending chain*.

Well-quasi-orderings

A major theme in our study of orderings will be the notion of a *well-quasi-ordering* (WQO).

Definition 1.1.1. Let $\leq$ be an ordering on a set S. We say that $\leq$ is a *well-quasi-ordering* (WQO) on S if:

- S contains no infinite anti-chains under $\leq$;
- S contains no infinite descending chains under $\leq$.

We may also say that S is *well-quasi-ordered* (WQO) under $\leq$. The two uses of the initialism 'WQO' will not cause any confusion as the meaning will always be clear from context.

Example 1.1.2. We present some examples and counter-examples of WQOs.

- The usual ordering on $\mathbb{N}$ is a WQO, since any two elements are comparable and any descending chain is clearly finite.
- The usual ordering on $\mathbb{Z}$ is not a WQO, because of the infinite descending chain $-1 > -2 > \ldots$.
- The divisibility ordering on $\mathbb{N}$ is not a WQO, since the set of prime numbers is an infinite anti-chain.
- The subgraph ordering on the set of all finite graphs is not a WQO, since the set of cycle graphs C_n is an infinite anti-chain.
- Hence also the induced subgraph ordering on the set of all finite graphs is not a WQO.
- The minor ordering[1] on the set of all finite graphs is a well-quasi-ordering, due to the celebrated Robertson-Seymour Theorem [31].

The following result of Higman [15] gives some alternative characterisations of the WQO property:

Proposition 1.1.3. *Let $\leq$ be an ordering on a set S. The following are equivalent:*

[1]We say G is a *minor* of H if G can be obtained from H by some sequence of vertex deletions, edge deletions and edge contractions.

(i) *The ordering* $\leq$ *is a WQO on* S.

(ii) *For any infinite sequence* $a_1, a_2, \ldots$ *of elements in* S*, there are indices* $i < j$ *such that* $a_i \leq a_j$.

(iii) *Any infinite sequence of elements in* S *has an infinite subsequence which is ascending under* $\leq$.

An infinite sequence $a_1, a_2, \ldots$, which does not satisfy condition (ii) of Proposition 1.1.3, so that $a_i \nleq a_j$ whenever $i < j$, is said to be a *bad* sequence. Hence being WQO is equivalent to the non-existence of bad sequences. We can use this to note the following result:

Proposition 1.1.4. *Being WQO is preserved under finite unions.*

Proof. If S and T are sets such that $S \cup T$ is not WQO, then $S \cup T$ contains a bad sequence. This sequence must have an infinite subsequence lying in one of S or T, and any infinite subsequence of a bad sequence is also a bad sequence, so at least one of S and T is not WQO. $\square$

Downward-closed sets and avoidance sets

A subset C of S is *downward-closed* if

$$x \in C \;\&\; y \leq x \Rightarrow y \in C.$$

Sometimes a downward-closed set is called a *class* or an *ideal*. We define the *basis* of C to be the minimal set of elements in $S \backslash C$, that is:

$$\mathcal{B}(C) = \{x \in S \backslash C \mid (\forall y \in S)(y < x \Rightarrow y \in C)\}.$$

If the set $\mathcal{B}(C)$ is finite then we say that C is *finitely-based*. Conversely, if $B \subseteq S$ is an anti-chain then we define the *avoidance set* of B to be the set $\mathrm{Av}(B)$ of elements in S which do not lie above any element in B, that is:

$$\mathrm{Av}(B) = \{x \in S \mid (\forall y \in B)(y \nleq x)\}.$$

We note the following elementary facts about these sets:

Proposition 1.1.5. *Let $C \subseteq S$ be a downward-closed set and let $X \subseteq S$ be an anti-chain. Then:*

(i) *The set $\mathcal{B}(C)$ is an anti-chain.*

(ii) *The set $\mathrm{Av}(B)$ is downward-closed.*

(iii) *We have $\mathcal{B}(\mathrm{Av}(B)) = B$.*

(iv) *If, in addition, the set S has no infinite descending chains, then $\mathrm{Av}(\mathcal{B}(C)) = C$.*

Proof. (i) We note that $\mathcal{B}(C) \cap C = \varnothing$. Now let $x, y \in \mathcal{B}(C)$ and suppose $x < y$. Then since $y \in \mathcal{B}(C)$ we have $x \in C$, a contradiction.

(ii) Let $x \in \mathrm{Av}(B)$ and let $y \leq x$. If $y \notin \mathrm{Av}(B)$ then there is some $z \in B$ with $z \leq y$. We then have $z \leq x$, but this cannot be the case since $x \in \mathrm{Av}(B)$.

(iii) ($\subseteq$) Let $x \in \mathcal{B}(\mathrm{Av}(B))$. Then $x \notin \mathrm{Av}(B)$, so there is some $y \in B$ with $y \leq x$. If $y < x$ then $y \in \mathrm{Av}(B)$, but this cannot be the case since $y \in B$. Hence $y = x$ and so $x \in B$.

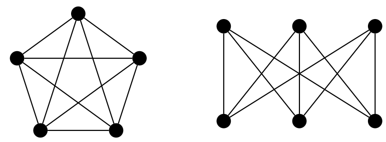
($\supseteq$) Let $x \in B$. Then $x \notin \mathrm{Av}(B)$. Since B is an anti-chain, if $y \in S$ is such that $y < x$ then there cannot be any $z \in X$ with $z \leq y$, since otherwise we would have $z < x$. Hence any such y belongs to $\mathrm{Av}(B)$, so $x \in B(\mathrm{Av}(B))$.

(iv) ($\subseteq$) We prove the contrapositive. Let $x \in S$ be such that $x \notin C$ and let $y \leq x$ be minimal such that $y \notin C$. Such an element exists since $\leq$ admits no infinite descending chains. Then for all $z < y$ we have $z \in C$, so $y \in \mathcal{B}(C)$ and hence $x \notin \mathrm{Av}(\mathcal{B}(C))$.

($\supseteq$) Again we prove the contrapositive. Let $s \in S$ be such that $s \notin \mathrm{Av}(\mathcal{B}(C))$. Then there is some $t \in \mathcal{B}(C)$ with $t \leq s$. Since $t \in \mathcal{B}(C)$ we have $t \notin C$, and since C is downward-closed this means $s \notin C$. □

Example 1.1.6. Wagner's Theorem [33] asserts that a graph is planar if and only if it does not contain either of K_5 and $K_{3,3}$ as a minor[2].

[2]This should not be confused with the more widely known Kuratowski's Theorem [25] asserting that a graph is planar if and only if it does not have a subgraph which is a subdivision of K_5 or $K_{3,3}$. In general this concept is distinct from having a graph as a minor, but in this case the two are equivalent.



K_5 $K_{3,3}$

Under the minor ordering, the set $B = \{K_5, K_{3,3}\}$ is an anti-chain, and the set $\mathcal{P}$ of planar graphs is downward-closed. Wagner's Theorem tells us that $\mathcal{P} = \mathrm{Av}(B)$.

Upward and downward closures

The *downward closure* of a set $T \subseteq S$ is the set down(T) of all elements in S which lie below some element of T, that is:

$$\mathrm{down}(T) = \{x \in S \mid (\exists y \in T)(x \leq y)\}.$$

Defined analogously, the *upward closure* of T is the set up(T) of elements in S which lie above some element of T, that is:

$$\mathrm{up}(T) = \{x \in S \mid (\exists y \in T)(y \leq x)\}.$$

We note that T is downward-closed if and only if it coincides with its downward closure, and that it is upward-closed if and only if it coincides with its upward closure. We also note that Av(T) is precisely the complement of up(T), i.e. we have

$$\mathrm{Av}(T) = S \setminus \mathrm{up}(T).$$

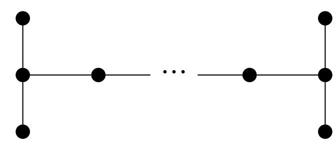
The WQO Problem

In the case that $\leq$ is not a well-quasi-ordering on S it will be of interest to ask when the avoidance set of a given set *is* WQO. We state this in isolation below.

Problem 1 (The WQO Problem). Let S be a set, let $\leq$ be an ordering on S and let $B \subseteq S$ be a finite anti-chain under $\leq$. Can one determine whether Av(B) is WQO under $\leq$?

The WQO Problem has been considered for a number of other orderings in combinatorics. In a recent survey article, Cherlin [7] comments on the significance of this problem and describes the contexts in which it has been studied. One result of note is the following theorem of Ding [10]:

Theorem 1.1.7 (Ding)**.** *A set of graphs which is downward-closed under the subgraph relation is WQO if and only if it contains finitely many cycles and double-ended forks, that is, graphs of the form:*



It follows that WQO Problem is decidable for the subgraph ordering, since if B is finite then $\mathrm{Av}(B)$ is WQO if and only if B contains a path. Other orderings for which the WQO Problem has been studied include the induced subgraph ordering on graphs [23], the involvement ordering on permutations [6], the minor ordering on matroids [16] and the embedding ordering on tournaments [8]. In each of these cases, the problem has only been solved for bases containing one or two elements. A recent result of Atminas et al. [3] asserts that the WQO Problem is decidable for the factor ordering on words. Moreover, they extend the solution to sets avoiding a given regular language. We will define and explore these concepts in more detail later.

Atomicity

Definition 1.1.8. Let S be a set, let $\leq$ be an ordering on S and let $C \subseteq S$ be a set which is downward-closed under $\leq$. We say C is *atomic* if it is not the union of two downward-closed proper subsets.

The following is a very general characterisation of atomicity, which has been adapted from [Mu02, Theorem 15]:

Proposition 1.1.9. *Let S be a countable set, let $\leq$ be an ordering on S and let $C \subseteq S$ be a set which is downward-closed under $\leq$. Then the following are equivalent:*

(i) *The set C is atomic.*

(ii) *For all $x, y \in C$ there exists some $z \in C$ such that $x \leq z$ and $y \leq z$.*

(iii) *There is a sequence of elements $z_1 \leq z_2 \leq \dots$ in C such that for all $x \in C$, there exists some i such that $x \leq z_i$.*

Proof. (i) ⇒ (ii): We prove the contrapositive. Suppose there are elements $x, y \in C$ such that for all $z \in C$, we have $x \nleq z$ or $y \nleq z$. Let $C_1 = C \cap \mathrm{Av}(x)$ and $C_2 = C \cap \mathrm{Av}(y)$. Then for all $z \in C$ we either have $z \in C_1$ or $z \in C_2$, so $C = C_1 \cup C_2$. The sets C_1 and C_2 are downward-closed, and they are proper subsets of C since $x \in C \backslash C_1$ and $y \in C \backslash C_2$, so C is not atomic.

(ii) ⇒ (iii): Let $x_1, x_2, \dots$ be an enumeration of C. Let $z_1 = x_1$ and for $n \geq 1$ let z_{n+1} be an element of C such that $x_{n+1} \leq z_{n+1}$ and $z_n \leq z_{n+1}$. The existence of such an element is guaranteed by condition (ii). The list $x_1, x_2, \dots$ contains every element of C and for each $i \geq 1$ we have $x_i \leq z_i$, so the sequence $z_1 \leq z_2 \leq \dots$ witnesses that condition (iii) is satisfied.

(iii) ⇒ (i): Let $z_1 \leq z_2 \leq \dots$ be as in condition (iii) and suppose, aiming for a contradiction, that C is not atomic. Let C_1 and C_2 be downward-closed proper subsets of C such that $C = C_1 \cup C_2$, and let $x \in C_1 \backslash C_2$ and $y \in C_2 \backslash C_1$. Let i and j be such that $x \leq z_i$ and $y \leq z_j$, and assume without loss of generality that $i \leq j$. Then $z_i \leq z_j$ and so x and y both lie below z_j. The element z belongs to one of C_1 or C_2, and since they are both downward-closed, one of C_1 or C_2 contains both x and y. This contradicts our selection of x and y. □

An element z such that $x \leq z$ and $y \leq z$ is called a *join* for x and y, and condition (ii) of Proposition 1.1.9 is referred to as the *join property*, or the *joint embedding property*. A sequence $z_1 \leq z_2 \leq \dots$ as described in condition (iii) of Proposition 1.1.9 is referred to as an *atomic sequence* for C.

Atomic sets are of interest when studying the WQO property as they are in some sense the building blocks of WQO downward-closed sets. We make this explicit below:

Proposition 1.1.10. *Let S be a countable set, let $\leq$ be an ordering on S and let $C \subseteq S$ be a set which is downward-closed under $\leq$. Then C is WQO if and only if it is a union of finitely many WQO atomic sets.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose that C is not the union of finitely many WQO atomic sets, and let $x_1, x_2, \ldots$ be an enumeration of C. Since C is downward-closed we have

$$C = \bigcup_{n=1}^{\infty} \mathrm{down}(x_n). \tag{1.1}$$

If $x, y \in \mathrm{down}(x_n)$ then $x \leq x_n$ and $y \leq x_n$, so x_n is a join for x and y. Hence each set $\mathrm{down}(x_n)$ satisfies the join property, and so is atomic. If any of the sets $\mathrm{down}(x_n)$ is not WQO then neither is C, and we are done. Suppose then that each set $\mathrm{down}(x_n)$ is WQO. This means that equation (1.1) gives C as a union of WQO atomic sets. We have $x_i \leq x_j$ if and only if $\mathrm{down}(x_i) \subseteq \mathrm{down}(x_j)$, so if J is the set

$$J = \{j \in \mathbb{N} \mid \forall i < j \mid x_i \nleq x_j\}$$

then we have

$$C = \bigcup_{j \in J} \mathrm{down}(x_j).$$

This again gives C as a union of WQO atomic sets, and so by our assumption the set J must be infinite. If its elements are $j_1 < j_2 < \ldots$ then $x_{j_1}, x_{j_2}, \ldots$ is a bad sequence, and so C is not WQO.

($\Leftarrow$) This follows from the fact that being WQO is preserved under finite unions, and the fact that a union of downward-closed sets is again downward-closed. □

The Atomicity Problem

Analogous to the WQO Problem is the following:

Problem 2 (The Atomicity Problem). Let S be a set, let $\leq$ be an ordering on S and let $B \subseteq S$ be a finite anti-chain under $\leq$. Can one determine whether $\mathrm{Av}(B)$ is atomic under $\leq$?

1.2 Words, automata and transducers

Words and languages

An *alphabet* is a finite set A of symbols called *letters*, and a *word* over A is a finite sequence of elements from A. For brevity we denote a word $(a_1, \dots, a_n)$ by $a_1 \dots a_n$. We let A^* denote the set of all words over A, including the empty word ε, and we let $A_\varepsilon = A \cup \{\varepsilon\}$. A *language over* A is any subset of A^*.

For $u \in A^*$ and $a \in A$ we let $|u|_a$ denote the number of occurrences of the letter a in the word u. As an example if $A = \{a, b, c\}$ and $u = babab$ then $|u|_a = 2$, $|u|_b = 3$ and $|u|_c = 0$. If $u = a_1 \dots a_n$ and $v = b_1 \dots b_m$ are words then their *concatenation* is the word

$$uv = a_1 \dots a_n b_1 \dots b_m.$$

If L and K are languages over A then we define $LK = \{uv \mid u \in L, v \in K\}$. We will also define $uL = \{uv \mid v \in L\}$ and $Lu = \{vu \mid v \in L\}$.

Automata and regular languages

Definition 1.2.1. A *deterministic finite state automaton* (DFA) is a 5-tuple $\mathcal{A} = (Q, A, \delta, q_0, F)$ where:

- Q is a finite set of *states*;
- A is a finite *alphabet*;
- $\delta : Q \times A \to Q$ is a partial function called the *transition function*;
- $q_0 \in Q$ is the *start state*;
- $F \subseteq Q$ is the set of *accept states*.

We extend δ to a partial function $\delta^* : Q \times A^* \to Q$ by setting $\delta^*(q, \varepsilon) = q$ and recursively setting

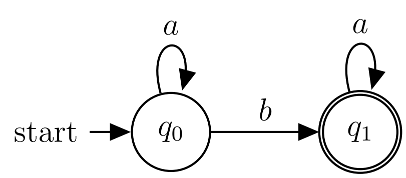
$$\delta^*(q, wa) = \delta(\delta^*(q, w), a)$$

for all $q \in Q$, $w \in A^*$ and $a \in A$. A word $w \in A^*$ is *accepted* by $\mathcal{A}$ if $\delta^*(q_0, w) \in F$, and the *language accepted by* $\mathcal{A}$ is the set of all words accepted by $\mathcal{A}$. We note that if, at any stage of reading a word, the next letter cannot be read from the current state, then $\mathcal{A}$ will reject the word. We represent a DFA $\mathcal{A}$ by a labelled directed graph called a *state diagram*, with an edge (q, a, p) whenever $\delta(q, a) = p$. We indicate the start state q_0 by an incoming edge labelled 'start' and we decorate each accept state with a double outline.

Example 1.2.2. Let $Q = \{q_0, q_1\}$, $A = \{a, b\}$, $F = \{q_1\}$ and let $\delta : Q \times A \to Q$ be the partial function given by

- $\delta(q_0, a) = q_0$;
- $\delta(q_0, b) = \delta(q_1, a) = q_1$.

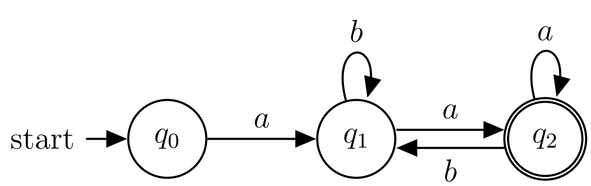
Then the DFA $\mathcal{A} = (Q, A, \delta, q_0, F)$ accepts the language of words $u \in A^*$ with $|u|_b = 1$. The state diagram for $\mathcal{A}$ is shown below.



Example 1.2.3. Let $Q = \{q_0, q_1, q_2\}$, $A = \{a, b\}$, $F = \{q_2\}$ and let $\delta : Q \times A \to Q$ be the partial function given by

- $\delta(q_0, a) = \delta(q_1, b) = \delta(q_2, b) = q_1$;
- $\delta(q_1, a) = \delta(q_2, a) = q_2$.

Then the DFA $\mathcal{A} = (Q, A, \delta, q_0, F)$ accepts the set of words in A^* of the form aua. The state diagram for $\mathcal{A}$ is shown below.



Definition 1.2.4. The class of languages accepted by DFAs is the class of *regular languages*. We let $\mathrm{Reg}(A)$ denote the set of regular languages over an alphabet A.

We will go on to prove various decidability results for regular languages. For these we will always assume that a given regular language is given via an automaton.

Another notion of an automaton is as follows, where use $\mathcal{P}(Q)$ to denote the power set of Q, i.e. the set of all subsets of Q:

Definition 1.2.5. A *non-deterministic finite state automaton* (NFA) is a 5-tuple $\mathcal{A} = (Q, A, \delta, q_0, F)$ where:

- Q is a finite set of *states*;
- A is a finite *alphabet*;
- $\delta : Q \times A_\varepsilon \to \mathcal{P}(Q)$ is a partial function called the *transition function*;
- $q_0 \in Q$ is the *start state*;
- $F \subseteq Q$ is the set of *accept states*.

We extend δ to a partial function $\delta^* : Q \times A^* \to \mathcal{P}(Q)$ by setting

$$\delta^*(q, \varepsilon) = \delta(q, \varepsilon) \cup \{q\}$$

and recursively setting

$$\delta^*(q, wa) = \bigcup_{p \in \delta^*(q,w)} \delta(p, a)$$

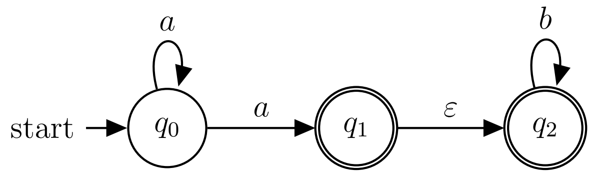
for all $q \in Q$, $w \in A^*$ and $a \in A$. We say a word $w \in A^*$ is *accepted* by $\mathcal{A}$ if $\delta^*(q_0, w)$ contains a state in F, and again we say that the *language accepted* by $\mathcal{A}$ is the set of all words accepted by $\mathcal{A}$. We note that if δ takes the value $\varnothing$ on a particular input then this is equivalent to δ being undefined on that input. We again represent a NFA $\mathcal{A}$ via a state diagram, this time with a labelled edge (q, x, p) whenever $p \in \delta(q, x)$. As with the deterministic case, we indicate the start state q_0 by an incoming edge labelled 'start' and decorate the accept states with a double outline.

Example 1.2.6. Let $Q = \{q_0, q_1, q_2\}$, $A = \{a, b\}$, $F = \{q_1, q_2\}$ and let $\delta : Q \times A_\varepsilon \to Q$ be the partial function given by

- $\delta(q_0, a) = \{q_0, q_1\}$;

- $\delta(q_1, \varepsilon) = \delta(q_2, b) = \{q_2\}$.

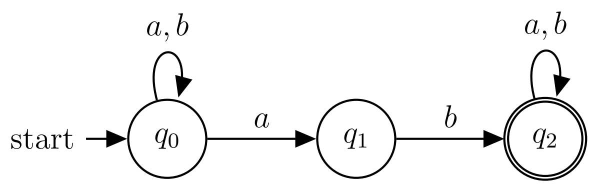
Then the NFA $\mathcal{A} = (Q, A, \delta, q_0, F)$ accepts the language of words of the form $a^i b^j$ where $i \geq 1$ and $j \geq 0$. The state diagram for $\mathcal{A}$ is shown below.



Example 1.2.7. Let $Q = \{q_0, q_1, q_2\}$, $A = \{a, b\}$, $F = \{q_2\}$ and let $\delta : Q \times A_\varepsilon \to \mathcal{P}(Q)$ be the partial function given by:

- $\delta(q_0, a) = \{q_0, q_1\}$;
- $\delta(q_0, b) = \{q_0\}$;
- $\delta(q_1, b) = \delta(q_2, a) = \delta(q_2, b) = \{q_2\}$.

Then the NFA $\mathcal{A} = (Q, A, \delta, q_0, F)$ accepts the language of words in A^* of the form $uabv$. The state diagram for $\mathcal{A}$ is shown below.



The class of NFAs was introduced by Rabin and Scott [30], as it was thought that some languages are better suited to their expressive power than to that of their deterministic counterparts. This is evident, for instance, with languages of the form shown in Example 1.2.7, where membership depends on the presence of some particular contiguous sequence of letters. In the same paper it is shown that the two varieties of automata are equivalent, that is:

Proposition 1.2.8. *The class of languages accepted by NFAs is precisely the class of regular languages. Moreover, there is an algorithm which takes as input an NFA and outputs a DFA accepting the same language.*

Regular expressions

Another way of describing regular languages is via *regular expressions*. The class of regular expressions over A is denoted by $\mathrm{Exp}(A)$ and defined recursively by:

- for each $u \in A^*$ we have $u \in \mathrm{Exp}(A)$;
- if $r, s \in \mathrm{Exp}(A)$ then $rs \in \mathrm{Exp}(A)$;
- if $r, s \in \mathrm{Exp}(A)$ then $r + s \in \mathrm{Exp}(A)$;
- if $r \in \mathrm{Exp}(A)$ then $r^* \in \mathrm{Exp}(A)$.

The *language corresponding to a regular expression* r is denoted by $L(r)$ and is defined recursively by:

- for each $u \in A^*$, we have $L(u) = \{u\}$;
- if $r, s \in \mathrm{Exp}(A)$ then $L(rs) = L(r)L(s)$;
- if $r, s \in \mathrm{Exp}(A)$ then $L(r + s) = L(r) \cup L(s)$;
- if $r \in \mathrm{Exp}(A)$ then $L(r^*) = \{r_1 \dots r_n \mid r_1, \dots, r_n \in L(r),\ n \geq 0\}$.

Example 1.2.9. We present some examples of regular expressions and their corresponding languages.

- If r is the regular expression $a^* + b^*$ then $L(r) = \{\varepsilon, a, b, aa, bb, \dots\}$.
- If r is the regular expression ab^*a then $L(r) = \{a, aba, abba, abbba, \dots\}$.
- The language accepted by the automaton in Example 1.2.2 corresponds to the regular expression a^*ba^*.
- The language accepted by the automaton in Example 1.2.3 corresponds to the regular expression $ab^*a(a^*bb^*a)^*$.

We note that there need not be a unique regular expression corresponding to a given language, since for instance we have $L(a^*) = L(a^* + a)$.

The following result of Kleene [22] states the equivalence of regular languages with regular expressions:

Theorem 1.2.10 (Kleene's Theorem). *A language $L \subseteq A^*$ belongs to* $\mathrm{Reg}(A)$ *if and only if it coincides with $L(r)$ for some $r \in \mathrm{Exp}(A)$.*

Closure and decidability results for regular languages

The following proposition is a list of standard closure properties for regular languages. For more details see, e.g., [17, Chapter 3].

Proposition 1.2.11. *The class of regular languages is closed under the following operations:*

- *finite unions;*
- *finite intersections;*
- *set difference;*
- *complementation;*
- *concatenation.*

Additionally, the following proposition is a list of relevant decidability results for regular languages (again, see [17, Chapter 3]):

Proposition 1.2.12. *It is decidable whether:*

- *a given regular language is empty;*
- *a given regular language is finite;*
- *two given regular languages coincide.*

A note on non-regular languages

Not every language is regular. The following lemma, which was first noted in [30], gives a necessary condition for an infinite language to be regular:

Lemma 1.2.13 (Pumping Lemma). *Let $L \subseteq A^*$ be an infinite regular language. Then there is a positive integer p such that each word $u \in L$ can be expressed as $u = xyz$ for some $x, y, z \in A^*$ such that $|xy| \leq p$, $|y| \geq 1$ and $xy^n z \in L$ for all $n \geq 1$.*

We use this to give an example of a language which is not regular.

Example 1.2.14. The language $L = \{a^n b^n \mid n \geq 0\}$ is not regular.

Proof. Suppose that L is regular and let p be as in the statement of the Pumping Lemma. We consider the word $u = a^p b^p \in L$. If x, y and z are as in the statement of the Pumping Lemma then y must be of the form a^m for some $m \geq 1$, but this would imply that $a^{p+m} b^p \in L$, which is not the case. $\square$

Although the language $\{a^n b^n \mid n \geq 0\}$ is not regular, it is an example of a *context-free language*. For more details on this and other classes of infinite languages the reader is referred to [17, Chapters 4, 9].

Finite state transducers and rational relations

Definition 1.2.15. A *finite state transducer*, or simply a *transducer*, is a 4-tuple $\mathcal{T} = (Q, A, \delta, q_I)$ where:

- Q is a finite set of *states*;
- A is a finite *alphabet*;
- $\delta : Q \times A_\varepsilon \to \mathcal{P}(Q \times A_\varepsilon)$ is a partial function called the *transition function*;
- $q_I \in Q$ is the *start state.*

We extend δ to a partial function $\delta^* : Q \times A^* \to \mathcal{P}(Q \times A^*)$ by recursively setting

$$\delta^*(q, wx) = \bigcup_{(p,u) \in \delta^*(q,w)} u\delta(p, x)$$

for all $q \in Q$, $w \in A^*$ and $x \in A_\varepsilon$. Corresponding to a transducer $\mathcal{T}$ is a relation $R_{\mathcal{T}}$ called the *relation generated by* $\mathcal{T}$, defined by

$$((u, v) \in R_{\mathcal{T}}) \Leftrightarrow (\exists s \in Q)((s, v) \in \delta^*(q_I, u)).$$

Intuitively, a transducer u as input and can output v. As with automata, we represent a transducer via a state diagram. If $(p, y) \in \delta(q, x)$ then we include an edge from q to p labelled by $x : y$. If P is a path in $\mathcal{T}$ consisting of edges

$$(p_0, x_1, y_1, p_1), (p_1, x_2, y_2, p_2), \dots, (p_{n-1}, x_n, y_n, p_n)$$

then we will describe P using the notation:

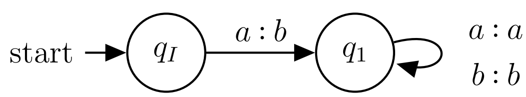
$$p_0 \xrightarrow{x_1 : y_1} p_1 \xrightarrow{x_2 : y_2} \cdots \xrightarrow{x_n : y_n} p_n.$$

Furthermore if $u = x_1 \dots x_n$ and $v = y_1 \dots y_n$ then we will say that P is labelled by (u, v).

Example 1.2.16. We define a transducer $\mathcal{T} = (Q, A, \delta, q_I)$ where $Q = \{q_I, q_1\}$, $A = \{a, b\}$, and $\delta : Q \times A_\varepsilon \to \mathcal{P}(Q \times A_\varepsilon)$ is the partial function given by

- $\delta(q_I, a) = \delta(q_1, b) = \{(q_1, b)\}$;
- $\delta(q_1, a) = \{(q_1, a)\}$.

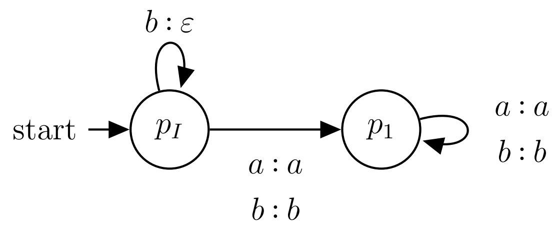
We have $(u, v) \in R_{\mathcal{T}}$ if and only if there is some $w \in A^*$ such that $u = aw$ and $v = bw$. The state diagram for $\mathcal{T}$ is shown below.



Example 1.2.17. We define a transducer $\mathcal{S} = (P, A, \sigma, p_I)$ where $P = \{p_I, p_1\}$, $A = \{a, b\}$ and $\sigma : P \times A_\varepsilon \to \mathcal{P}(P \times A_\varepsilon)$ is the partial function given by:

- $\sigma(p_I, a) = \{(p_1, a)\}$;
- $\sigma(p_I, b) = \{(p_I, \varepsilon), (p_1, b)\}$;
- $\sigma(p_1, a) = \{(p_1, a)\}$;
- $\sigma(p_1, b) = \{(p_1, b)\}$.

We have $(u, v) \in R_{\mathcal{S}}$ if and only if v can be obtained from u by deleting a prefix of the form b^i. The state diagram for $\mathcal{S}$ is shown below.



Other notions of transducers have been considered in the literature, for instance where there is no start state, where the edges can be labelled by words as well as symbols from A_ε, or where there is a set of accept states. In this thesis, however, we will almost exclusively consider transducers as specified in Definition 1.2.15, and so we will use this as our primary definition. In section 3.2 we will briefly use some alternative models, but this will be made explicit at the time.

Definition 1.2.18. The class of relations which can be generated by transducers is the class of *rational relations*.

It is always possible to determine an automaton or transducer uniquely from its state diagram. We will therefore frequently define one by simply showing its state diagram, or by describing the states and the edges present in the state diagram, without explicitly defining the transition function. In particular, it will be of use to define transducers simply by describing their transitions, which we will frequently do using the notation

$$q \xrightarrow{x : y} p.$$

Composition of transducers

Let $\mathcal{T} = (Q, A, \delta, q_I)$ and $\mathcal{S} = (P, A, \sigma, p_I)$ be two transducers over the same alphabet, generating relations $R_\mathcal{T}$ and $R_\mathcal{S}$, respectively. The composition of $\mathcal{T}$ and $\mathcal{S}$ is the transducer

$$\mathcal{T} \circ \mathcal{S} = (Q \times P, A, \Delta, (q_I, p_I))$$

where the transitions are as follows:

- $(q_1, p_1) \xrightarrow{x : z} (q_2, p_2)$ where there is some $y \in A_\varepsilon$ such that $q_1 \xrightarrow{x : y} q_2$ is transition in $\mathcal{T}$ and $p_1 \xrightarrow{y : z} p_2$ is a transition in $\mathcal{S}$;

- $(q_1, p) \xrightarrow{x:\varepsilon} (q_2, p)$ where $q_1 \xrightarrow{x:\varepsilon} q_2$ is a transition in $\mathcal{T}$;
- $(q, p_1) \xrightarrow{\varepsilon:z} (q, p_2)$ where $p_1 \xrightarrow{\varepsilon:z} p_2$ is a transition in $\mathcal{S}$.

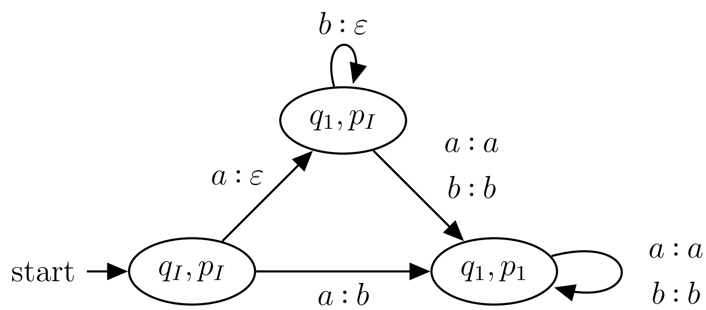
Informally, the transitions in $\mathcal{T} \circ \mathcal{S}$ are obtained by feeding the output of $\mathcal{T}$ into $\mathcal{S}$. This means that the relation generated by $\mathcal{T} \circ \mathcal{S}$ is the composition of $R_{\mathcal{T}}$ and $R_{\mathcal{S}}$, that is, the relation $R_{\mathcal{T}} \circ R_{\mathcal{S}}$ defined by

$$((u, w) \in R_{\mathcal{T}} \circ R_{\mathcal{S}}) \Leftrightarrow (\exists v \in A^*)((u, v) \in R_{\mathcal{T}} \ \& \ (v, w) \in R_{\mathcal{S}}).$$

For more details see [20, Chapter 3].

We remark that, drawing transducers of the form $\mathcal{T} \circ \mathcal{S}$, we will typically only show states which are accessible.

Example 1.2.19. Let $\mathcal{T}$ be the transducer from Example 1.2.16 and let $\mathcal{S}$ be the transducer from Example 1.2.17. Then $\mathcal{T} \circ \mathcal{S} = (Q \times P, A, \Delta, (q_I, p_I))$ is the transducer shown below.



1.3 Orderings on words

The subword ordering

Definition 1.3.1. The *subword* ordering on A^* is denoted by $\leq_w$ and defined by

$$(u \leq_w v) \Leftrightarrow (\exists a_1, \ldots, a_n \in A)(\exists v_0, v_1, \ldots, v_n \in A^*)(u = a_1 \ldots a_n \ \& \ v = v_0 a_1 v_1 \ldots a_n v_n).$$

Intuitively we have $u \leq_w v$ if v can be obtained from u by inserting words between its letters.

Example 1.3.2. We have $aaa \leq_w ababa$ and $this \leq_w thesis$.

The following is a celebrated result of Higman [15]:

Lemma 1.3.3 (Higman's Lemma)**.** *Let A be a finite alphabet. Then A^* is WQO under the subword ordering.*

One surprising consequence of Higman's Lemma, which was noted by Haines [14], is that the downward and upward closures of *any* language L are both regular. To see this for $\mathrm{down}(L)$ we let

$$B = \mathcal{B}(\mathrm{down}(L)).$$

The set B is an anti-chain, so by Higman's Lemma it must be finite. Since $\mathrm{down}(L) = \mathrm{Av}(B)$ we have

$$A^* \setminus \mathrm{down}(L) = \bigcup_{a_1 \ldots a_n \in B} A^* a_1 A^* \ldots a_n A^*.$$

Hence $A^* \setminus \mathrm{down}(L)$ is the union of finitely many regular languages, and so is itself regular. The fact that $\mathrm{down}(L)$ is regular follows from the fact that $\mathrm{Reg}(A)$ is closed under complementation. The proof for $\mathrm{up}(L)$ is similar.

The factor ordering

Definition 1.3.4. The *factor ordering* on A^* is denoted by $\leq_f$ and defined by:

$$(u \leq_f v) \Leftrightarrow (\exists \alpha, \beta \in A^*)(v = \alpha u \beta).$$

Intuitively we have $u \leq_f v$ if v can be obtained by inserting words at the beginning and end of u.

Example 1.3.5. We have $aaa \leq_f baaab$ and $the \leq_f mathematics$.

The factor ordering is not a WQO unless $|A| = 1$. To see this let a and b be distinct letters in A. Then the language L given by the regular expression ab^*a is an infinite anti-chain. Indeed, if $v \in L$ and $u <_f v$ then $|u|_a \leq 1$, so $u \notin L$.

Atminas et al. [3] give a solution to the WQO Problem for the factor ordering, for sets avoiding a given regular language. They first prove the following result:

Theorem 1.3.6 (Atminas et al.)**.** *Let L be a regular language. Then it is decidable whether L is WQO under the factor ordering.*

The avoidance set $\mathrm{Av}_f(L)$ of a regular language L under the factor ordering is again a regular language, since it is given by

$$\mathrm{Av}_f(L) = A^* \backslash A^* L A^*$$

and $\mathrm{Reg}(A)$ is closed under the operations of concatenation and complementation. Hence the following can be deduced:

Theorem 1.3.7 (Atminas et al.)**.** *Let L be a regular language. Then it is decidable whether $\mathrm{Av}_f(L)$ is WQO under the factor ordering.*

In their paper Atminas et al. give a useful method of constructing anti-chains under the factor ordering, which we will go on to describe. We start with the following definition:

Definition 1.3.8. Let γ, α and δ be words. We say γ is a *left extension of a power of* α if we can write $\gamma = \sigma\alpha^i$ where σ is a suffix of α. We say δ is a *right extension of a power of* α if we can write $\delta = \alpha^i\sigma$ where σ is a prefix of α.

Example 1.3.9. The word *abbabba* is a left extension of a power of *bba* and the word *bcabbcabbc* is a right extension of a power of *bcab*.

The anti-chain construction is then as follows:

Lemma 1.3.10 (Atminas et al.)**.** *Let γ, α and δ be words such that γ is not a left extension of a power of α and δ is not a right extension of a power of α. Then the set of words*

$$\{\gamma\alpha^i\delta \mid i = 1, 2, \dots\}$$

is an infinite anti-chain under the factor ordering.

Additionally they give the following result:

Lemma 1.3.11 (Atminas et al.). *Let γ, α and δ be words. Then the language*

$$\{\gamma^i \alpha \delta^j \mid i, j = 1, 2, \dots\}$$

is WQO under the factor ordering. Furthermore, the downward closure of such a language is also WQO.

The prefix ordering

Definition 1.3.12. The *prefix ordering* on A^* is denoted by $\leq_p$ and defined by:

$$(u \leq_p v) \Leftrightarrow (\exists \beta \in A^*)(v = u\beta).$$

Example 1.3.13. We have $ab \leq_p abcd$ and $math \leq_p mathematics$.

Since $u \leq_p v$ implies $u \leq_f v$, the prefix ordering is not a WQO. The language given by the regular expression a^*b is an example of a set which is WQO under the factor ordering but is an anti-chain under the prefix ordering.

The suffix ordering

The suffix ordering on A^* is defined analogously to the prefix ordering.

Definition 1.3.14. The suffix ordering on A^* is denoted by $\leq_s$ and defined by:

$$(u \leq_s v) \Leftrightarrow (\exists \alpha \in A^*)(v = \alpha u).$$

Example 1.3.15. We have $bba \leq_s ababba$ and $ample \leq_s example$.

The natural symmetry between the prefix and suffix orderings allows us to convert decidability problems for the suffix case to their counterparts in the prefix case. We first make the following definition:

Definition 1.3.16. Let $u \in A^*$ and write $u = a_1 \dots a_n$. We define $u^R = a_n \dots a_1$, and for $L \subseteq A^*$ we define

$$L^R = \{u^R \mid u \in L\}.$$

We then note the following:

Observation 1.3.17. *Let $u, v \in A^*$. Then $u \leq_p v$ if and only if $u^R \leq_s v^R$.*

As a consequence of this, a language L is (a chain, an anti-chain, WQO) under the prefix ordering if and only if the language L^R is (a chain, an anti-chain, WQO) under the suffix ordering. Furthermore, we note that the regularity of L implies the regularity of L^R:

Proposition 1.3.18. *Let $L \subseteq A^*$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a DFA accepting L. Let p_0 be a symbol not in Q and let $P = Q \cup \{p_0\}$. Define a partial function*

$$\sigma : P \times A_\varepsilon \to \mathcal{P}(P)$$

so that for each $q \in Q$ and $a \in A$ we have

$$\sigma(q, a) = \{p \in Q \mid \delta(p, a) = q\}$$

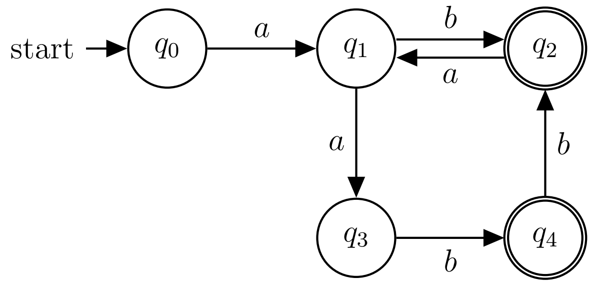
and so that $\sigma(p_0, \varepsilon) = F$. Then $B = (P, A, \sigma, p_0, \{q_0\})$ is an NFA accepting L^R.

Proof. Let $u \in L^R$ and write $u = v^R$ for some $v \in L$. Write $v = a_1 \dots a_n$ so that $u = a_n \dots a_1$, and for $i \in \{1, \dots, n\}$ let $q_i = \delta(q_{i-1}, a_i)$ so that $q_{i-1} \in \sigma(q_i, a_i)$. Finally note that $q_n \in F$. Then $\mathcal{B}$ contains the path

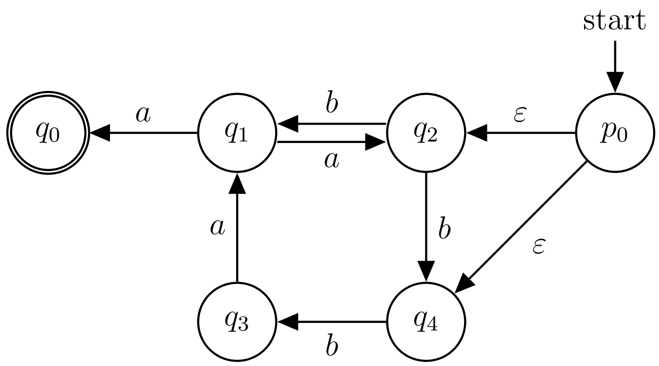
$$(p_0, \varepsilon, q_n)(q_n, a_n, q_{n-1}) \dots (q_1, a_1, q_0). \tag{1.2}$$

This path is labelled by v, and so v is accepted by $\mathcal{B}$. Conversely any path in $\mathcal{B}$ has the form (1.2) and so is labelled by a word in L^R. Hence the language accepted by $\mathcal{B}$ is precisely L^R. □

Example 1.3.19. Let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be the DFA shown below and let L be the language accepted by $\mathcal{A}$.



The NFA $\mathcal{B} = (P, A, \sigma, p_0, \{q_0\})$ accepting L^R, as constructed in the proof of Proposition 1.3.18, is shown below.



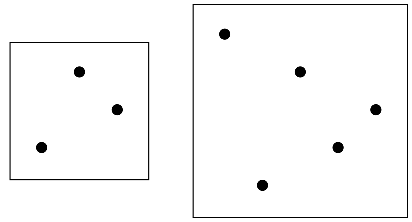
1.4 Permutations

For the purposes of this thesis, a *permutation* of length n is a sequence containing each element of the set $\{1, \dots, n\}$ exactly once. For brevity we will typically denote a permutation $(s_1, \dots, s_n)$ by $s_1 \dots s_n$.

Example 1.4.1. Some permutations of length 3 are 132 and 231. Some permutations of length 5 are 51423 are 12345.

At times it will be useful to represent a permutation $s_1 \dots s_n$ visually. We do so by plotting its points from left to right in a *permutation diagram*, where the point in position i is given height s_i.

Example 1.4.2. Below are permutation diagrams for 132 and 51423.



132 51423

Definition 1.4.3. Let $\sigma = s_1 \dots s_n$ and $\tau = t_1 \dots t_n$ be two sequences of distinct positive integers, both of the same length. We say σ and τ are *order isomorphic* if

$$s_i < s_j \Leftrightarrow t_i < t_j.$$

Example 1.4.4. The sequences 253 and 132 are order isomorphic. The sequences 524 and 213 are not, since the first has its largest term at the beginning and the second has its largest term at the end.

Definition 1.4.5. Let σ be a sequence of n distinct positive integers. The *reduction of* σ is the permutation $\rho(\sigma)$ of length n which is order isomorphic to σ.

Example 1.4.6. We have $\rho(253) = 132$ and $\rho(524) = 312$.

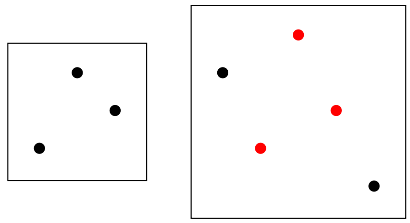
We now introduce the ordering which is to be the principal topic of our study of permutations. This will be the *consecutive involvement ordering*, which is an analogue of the factor ordering on words. For a recent survey of results on this ordering, the reader is invited to consult [12].

Definition 1.4.7. The *consecutive involvement* ordering on the set of permutations is denoted by $\leq$ and defined as follows. Let $\sigma = s_1 \dots s_n$ and $\tau = t_1 \dots t_m$ be two permutations, with $n \leq m$. Then $\sigma \leq \tau$ if there exists some $i \in \{1, \dots, m-n+1\}$ such that $\sigma = \rho(t_i \dots t_{i+n-1})$.

Example 1.4.8. We have $132 \leq 42531$ and $312 \leq 52413$.

It is easy to visualise the consecutive involvement ordering using permutation diagrams, as we illustrate in the next example.

Example 1.4.9. Below are permutation diagrams for $\sigma = 132$ and $\tau = 42531$. In the diagram for τ we have highlighted in red the points which witness the comparison $132 \leq 42531$.

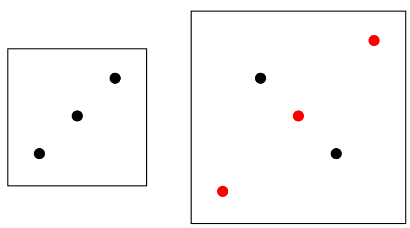


132 42531

Another ordering on permutations which we do not consider in this thesis, but which has been studied extensively in the literature, is the *involvement* ordering. This is an analogue of the subword ordering on words, and is defined as follows:

Definition 1.4.10. Let $\sigma = s_1 \dots s_n$ and $\tau = t_1 \dots t_m$ be permutations, with $n \leq m$. We say that σ is *involved* in τ if there are indices $i_1 < \dots < i_n \in \{1, \dots, m\}$ such that $\sigma = \rho(t_{i_1} \dots t_{i_n})$.

Example 1.4.11. The permutation 123 is involved in 14325. This is highlighted in the figure below.



123 15324

A survey of recent results on the involvement ordering can be found in [32]. For a discussion of the historical developments in the theory see [5] and [21], and for an account of anti-chains and the WQO property see [27].

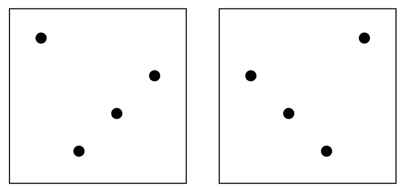
Symmetries preserving the consecutive involvement ordering

We now introduce some symmetries which preserve the consecutive involvement ordering.

Definition 1.4.12. Let $\sigma = s_1 \dots s_n$ be a permutation. The *reverse* of σ is the permutation σ^R of length n which has in position i the number s_{n+1-i}. For a set S of permutations we define

$$S^R = \{\sigma^R \mid \sigma \in S\}.$$

Geometrically, the permutation σ^R can be obtained by reflecting σ about a vertical line. As an example we have $4123^R = 3214$, which we illustrate below.

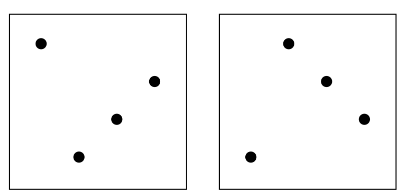


4123 $\qquad$ $4123^R = 3214$

Definition 1.4.13. Let $\sigma = s_1 \dots s_n$ be a permutation. The *complement* of σ is the permutation σ^C of length n which has in position i the number $n + 1 - s_i$. For a set S of permutations we define

$$S^C = \{\sigma^C \mid \sigma \in S\}.$$

Geometrically, the permutation σ^C can be obtained by reflecting σ about a horizontal axis. As an example we have $4123^C = 1432$, which we illustrate below.



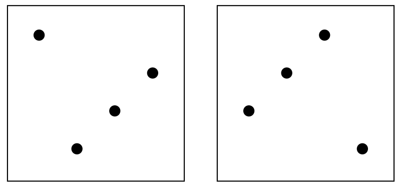
4123 $\qquad$ $4123^C = 1432$

We note without proof that the operations of reversal and complementation commute with one another, so that $(\sigma^R)^C = (\sigma^C)^R$ for all permutations σ Hence we may omit the brackets and refer to this permutation simply as σ^{RC}.

Definition 1.4.14. Let $\sigma = s_1 \dots s_n$ be a permutation. The *reverse complement* of σ is the permutation σ^{RC}. For a set S of permutations we define

$$S^{RC} = \{\sigma^{RC} \mid \sigma \in S\}.$$

Geometrically, the permutation σ^{RC} can be obtained by rotating σ by $180°$. As an example we have $4123^{RC} = 2341$, which we illustrate below.



4123 $\quad$ $4123^R = 2341$

We note without proof that the operations of reversal, complementation and reverse complementation all preserve the consecutive involvement ordering, that is:

Proposition 1.4.15. *Let σ, τ be permutations. Then the following are equivalent:*

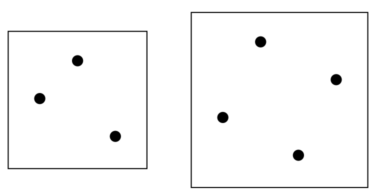
(i) $\sigma \leq \tau$;

(ii) $\sigma^R \leq \tau^R$;

(iii) $\sigma^C \leq \tau^C$;

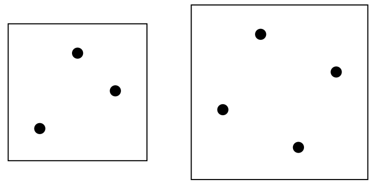
(iv) $\sigma^{RC} \leq \tau^{RC}$.

The involvement ordering on permutations is preserved under rotation by $90°$ clockwise, but this is not the case for the consecutive involvement ordering. Indeed, let $\sigma = 231$ and $\tau = 2413$, which are shown below.



$\sigma = 231$ $\quad$ $\tau = 2413$

We have $\sigma \leq \tau$, since $\sigma = \rho(241)$. By rotating these permutations clockwise by $90°$, we obtain the permutations $\sigma' = 132$ and $\tau' = \tau = 2413$, which are shown below.



$\sigma' = 231$ $\tau' = 2413$

We have $\sigma' \nleq \tau'$, so the operation does not preserve the ordering.

Operations on permutations

We now describe some binary operations which can be performed on pairs of permutations. The purpose of this will be to describe large permutations, or large sets of permutations, which will be constructed in some systematic way from a collection of smaller ones.

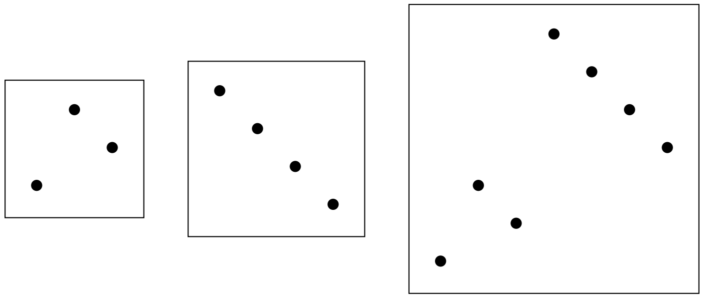
Notation 1.4.16. If σ is a permutation of length n and $i \in \{1, \ldots, n\}$ then we will write $(\sigma)_i$ to mean the point in position i of σ.

Definition 1.4.17. Let σ, τ be permutations of lengths n, m respectively. The *sum* of σ and τ is the permutation $\sigma \oplus \tau$ of length $n + m$ where

$$(\sigma \oplus \tau)_i = \begin{cases} \sigma_i & \text{for } i \in \{1, \ldots, n\} \\ \tau_i + n & \text{for } i \in \{n+1, \ldots, n+m\}. \end{cases}$$

Geometrically, the permutation $\sigma \oplus \tau$ can be obtained by placing a copy of τ above and to the right of σ.

Example 1.4.18. Let $\sigma = 132$ and $\tau = 4321$. Then $\sigma \oplus \tau = 1327654$. These permutations are illustrated below.



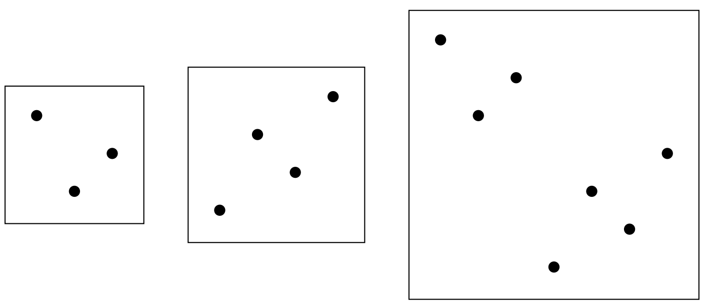
$\sigma = 132$ $\qquad$ $\tau = 4321$ $\qquad$ $\sigma \oplus \tau = 1327654$

Definition 1.4.19. Let σ, τ be permutations of lengths n, m respectively. The *skew sum* of σ and τ is the permutation $\sigma \ominus \tau$ of length $n + m$ where

$$(\sigma \ominus \tau)_i = \begin{cases} \sigma_i + m & \text{for } i \in \{1, \dots, n\} \\ \tau_i & \text{for } i \in \{n+1, \dots, n+m\}. \end{cases}$$

Geometrically, the permutation $\sigma \ominus \tau$ can be obtained by placing a copy of τ below and to the right of σ.

Example 1.4.20. Let $\sigma = 312$ and $\tau = 1324$. Then $\sigma \ominus \tau = 7561324$. These permutations are shown below.



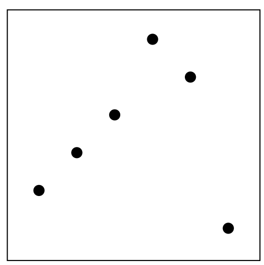
$\sigma = 312$ $\qquad$ $\tau = 1324$ $\qquad$ $\sigma \ominus \tau = 7561324$

We note without proof that the sum and skew sum operations are both associative, so that $(\sigma \oplus \tau) \oplus \pi = \sigma \oplus (\tau \oplus \pi)$ and $(\sigma \ominus \tau) \ominus \pi = \sigma \ominus (\tau \ominus \pi)$ for all permutations σ, τ, π. Hence we may omit the brackets and refer to these permutations simply as $\sigma \oplus \tau \oplus \pi$ and $\sigma \ominus \tau \ominus \pi$ respectively. However, brackets must typically be included in expressions containing both the sum and skew sum operations, since for instance $\sigma \ominus (\tau \oplus \pi)$ may be different from $(\sigma \ominus \tau) \oplus \pi$. Indeed, we have $1 \ominus (1 \oplus 1) = 312$ and $(1 \ominus 1) \oplus 1 = 213$.

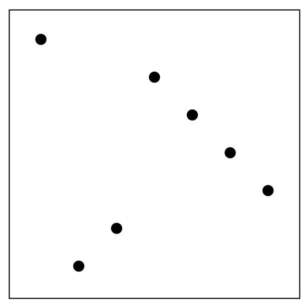
An *ascent* is a permutation of the form $1 \dots n$ and a *descent* is a permutation of the form $n \dots 1$. We will go on to construct certain permutations as sums or skew sums of various ascents and descents, as well as the trivial permutation 1.

Notation 1.4.21. We will let α_n denote the ascent of length n and δ_n denote the descent of length n.

As an example of constructing permutations from ascents, descent and the trivial permutation we have $(\alpha_3 \oplus \delta_2) \ominus 1 = 234651$ and $1 \ominus (\alpha_2 \oplus \delta_4) = 7126543$. These are illustrated below.



$(\alpha_3 \oplus \delta_2) \ominus 1$



$1 \ominus (\alpha_2 \oplus \delta_4)$

Chapter 2

Orderings on words

2.1 The prefix and suffix orderings

The prefix ordering

We recall that the prefix ordering $\leq_p$ on A^* is defined by:

$$(u \leq_p v) \Leftrightarrow (\exists \beta \in A^*)(v = u\beta).$$

Our aim in this section is to prove certain order-theoretic facts about regular languages under the prefix and suffix orderings. We will make use of a particular type of DFA called a *reduced* DFA, which we define below.

Definition 2.1.1. We say a DFA is *reduced* if each of its states is accessible and admits a path to an accept state.

We note without proof that any DFA can be converted into a reduced DFA by removing states which are inaccessible or which do not admit a path to an accept state, and so every regular language is accepted by a reduced DFA. As an example, the two automata in Figure 2.1 both accept the language given by the regular expression ab^*, with the one on the right being reduced.

We refer to the number of edges leaving a state as the *out-degree* of that state. Up

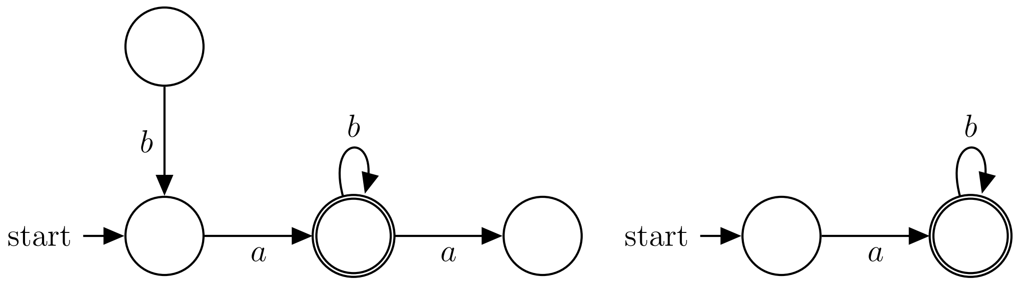


Figure 2.1: Two DFAs accepting the same language. The one on the right is reduced.

until this point, when drawing state diagrams we have typically represented two edges (q, a, p) and (q, b, p) between the same pair of states by a single arrow labelled by both a and b. In the results which follow we will, of course, consider the edges (q, a, p) and (q, b, p) to be distinct as long as the letters a and b are distinct.

Our first result is a classification of the regular languages which are *prefix chains*, i.e. chains under the prefix ordering.

Theorem 2.1.2. *Let $L \subseteq A^*$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a reduced DFA accepting L. Then L is a prefix chain if and only if every state in $\mathcal{A}$ has out-degree at most* 1.

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose there is a state $q \in Q$ with out-degree at least 2, so that there are distinct letters $a, b \in A$ such that $\delta(q, a)$ and $\delta(q, b)$ are both defined. Let $u \in A^*$ be such that $\delta^*(q_0, u) = q$ and let $v, w \in A^*$ be such that $\delta^*(q_0, uav)$ and $\delta^*(q_0, ubw)$ are both in F. The existence of these words is guaranteed by the fact that $\mathcal{A}$ is reduced. The words uav and ubw are both in L but are incomparable under the prefix ordering since they differ in position $|u| + 1$, and so L is not a prefix chain.

($\Leftarrow$) Again we prove the contrapositive. Suppose that L is not a prefix chain, so that there are words $u, v \in L$ which are incomparable under the prefix ordering. Suppose u and v share a maximal common prefix w, so that we can write $u = wav'$ and $v = wbv'$ where $a \neq b$. If we let $q = \delta^*(q_0, w)$ then $\delta(q, a)$ and $\delta(q, b)$ are both defined, so q has out-degree at least 2. $\square$

The above theorem tells us that a reduced DFA accepts a prefix chain if and only if it consists of a single (possible trivial) path from the start state into a simple cycle. This is sketched in Figure 2.2.

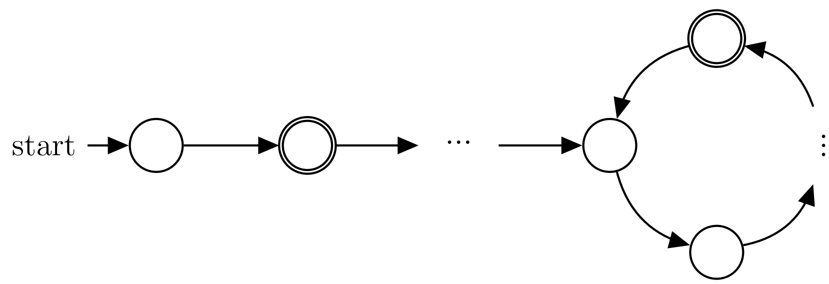


Figure 2.2: An automaton accepting a prefix chain.

Our next result gives a classification of the regular languages which are *prefix anti-chains*, i.e. anti-chains under the prefix ordering.

Theorem 2.1.3. *Let $L \subseteq A^*$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a reduced DFA accepting L. Then the language L is a prefix anti-chain if and only if every accept state in $\mathcal{A}$ has out-degree* 0*.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose there is some accept state q with out-degree at least 1. Let $a \in A$ be such that $\delta(q, a)$ is defined, let $p = \delta(q, a)$ and let $u, v \in A^*$ be such that $\delta(q_0, u) = q$ and $\delta^*(p, v) \in F$. The existence of these words is guaranteed by the fact that $\mathcal{A}$ is reduced. Then $u, uav \in L$ and $u <_p uav$, so L is not a prefix anti-chain.

($\Leftarrow$) Again we prove the contrapositive. Suppose that L is not a prefix anti-chain, so that there are words $u, v \in L$ with $u <_p v$. Write $v = uav'$ where $a \in A$ and let $q = \delta^*(q_0, u)$. Then q is an accept state and $\delta(q, a)$ is defined, so q has out-degree at least 1. $\square$

Our next goal is to show it is decidable whether a given regular language is WQO under the prefix ordering. We first make the following definition:

Definition 2.1.4. Let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a DFA. A state $q \in Q$ is a *loop-state* if there is a non-empty word w such that $\delta^*(q, w) = q$.

We then have the following result:

Theorem 2.1.5. *Let $L \subseteq A^*$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a reduced DFA accepting L. Then L is WQO under the prefix ordering if and only if every loop-state in Q has out-degree* 1.

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose there is a loop-state $q \in Q$ whose out-degree is not 1. Since q is a loop-state its out-degree cannot be 0, so must be at least 2. Let w be a non-empty word such that $\delta^*(q, w) = q$ and write $w = aw'$ where $a \in A$. Let b be a letter distinct from a such that $\delta(q, b)$ is defined and let $p = \delta(q, b)$. Then let $u, v \in A^*$ be such that $\delta^*(q_0, u) = q$ and $\delta^*(p, v) \in F$. The existence of these words is guaranteed by the face that $\mathcal{A}$ is reduced. We will prove that the set $K \subseteq L$ given by the regular expression uw^*bv is a prefix anti-chain. Let $x, y \in K$ and suppose $x <_p y$. Write $x = uw^i bv$ and $y = uw^j bv$ where i, j are indices with $j > i \geq 0$. The words x and y must agree in position $|u| + i|w| + 1$, but in this position the word x has the letter b and the word y has the letter a, a contradiction.

($\Leftarrow$) Let K be an infinite subset of L. In order to prove that L is WQO under the prefix ordering it suffices to prove that K contains an infinite chain. Since K is infinite, there is a word u such that $\delta^*(q_0, u)$ is a loop state and such that infinitely many words in K begin with u. Let H be this set of words, that is, let $H = K \cap uA^*$. Our aim is to prove that H is a prefix chain. Let $w \in A^*$ be a non-empty word such that $\delta^*(q, w) = q$. Since q has out-degree 1, the word w is unique, and so every word in H can be written as $uw^i v$ for some prefix v of w. Let $x = uw^i v_1$ and $y = uw^j v_2$ be two words in H with $|x| \leq |y|$. Then $v_1 \leq_p w^{j-i} v_2$ and so $x \leq_p y$. $\square$

The above theorem tells us that the language accepted by a reduced DFA $\mathcal{A}$ is WQO under the prefix ordering if and only if every cycle in $\mathcal{A}$ has no exit. This is sketched in Figure 2.3.

The conditions presented in Theorems 2.1.2, 2.1.3 and 2.1.5 are decidable, and we summarise this as follows:

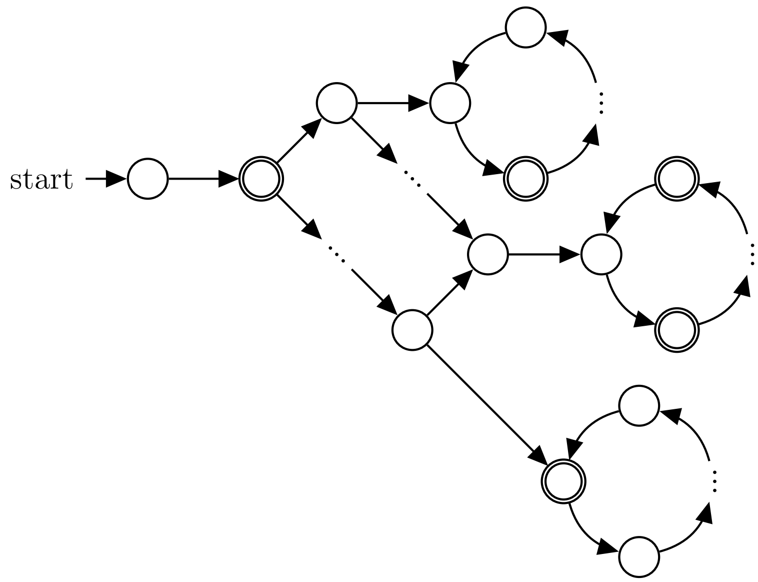


Figure 2.3: An automaton for a language which is WQO under the prefix ordering.

Corollary 2.1.6. *Let $L \subseteq A^*$ be a regular language. Then for the prefix ordering it is decidable whether L is:*

- *a chain;*
- *an anti-chain;*
- *WQO.*

We will use this to show that these properties are decidable for the avoidance set of a given regular language. We first specify some notation:

Notation 2.1.7. Let $L \subseteq A^*$. We let $\mathrm{up}_p(L)$ and $\mathrm{Av}_p(L)$ respectively denote the upward closure and avoidance set of L under the prefix ordering.

We then have:

Lemma 2.1.8. *Let $L \subseteq A^*$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a DFA accepting L. Let p be a symbol not in Q and let $P = Q \cup \{p\}$. Define a partial function*

$$\sigma : P \times (A \cup \varepsilon) \to \mathcal{P}(P)$$

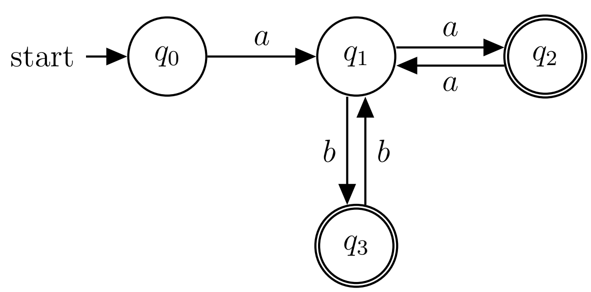
by

- $\sigma(q,a) = \{\delta(q,a)\}$ *for all* $q \in Q$ *and* $a \in A$;
- $\sigma(q,\varepsilon) = \{p\}$ *for all* $q \in F$;
- $\sigma(p,a) = \{p\}$ *for all* $a \in A$.

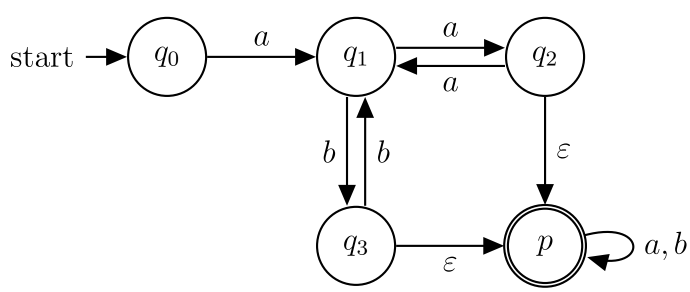
Then $\mathcal{B} = (P, A, \sigma, q_0, \{p\})$ *is an NFA accepting* $\mathrm{up}_p(L)$.

Proof. A word u is accepted by $\mathcal{B}$ if and only if there is a path from q_0 to an accept state in $\mathcal{A}$ labelled by a prefix of w, which is the case if and only if u belongs to $\mathrm{up}_p(L)$. Hence the language accepted by $\mathcal{B}$ is precisely $\mathrm{up}_p(L)$. $\square$

Example 2.1.9. Let $A = \{a, b\}$ and let $L \subseteq A^*$ be the regular language accepted by the automaton shown below.



Then an automaton for $\mathrm{up}_p(L)$, as constructed in the proof of Lemma 2.1.8, is:



We can use the previous result to state:

Lemma 2.1.10. *Let* $L \subseteq A^*$ *be a regular language. Then* $\mathrm{Av}_p(L)$ *is a regular language.*

Proof. This follows from Lemma 2.1.8, and the facts that $\mathrm{Av}_p(L) = A^* \setminus \mathrm{up}_p(L)$ and that $\mathrm{Reg}(A)$ is closed under the operation of set difference. $\square$

We can then state the following as a special case of Corollary 2.1.6:

Corollary 2.1.11. *Let $L \subseteq A^*$ be a regular language. Then for the prefix ordering it is decidable whether $\mathrm{Av}_p(L)$ is:*

- *a chain;*
- *an anti-chain;*
- *WQO.*

Atomicity

The following theorem gives a characterisation of atomicity in the prefix ordering:

Theorem 2.1.12. *Let $C \subseteq A^*$ be a set which is downward-closed under the prefix ordering. Then the following are equivalent:*

(i) *The set C is atomic under the prefix ordering.*

(ii) *The set C is a prefix chain.*

(iii) *The set C coincides with the set of finite prefixes of some right-infinite word w.*

Proof. We prove the implications (i) $\Rightarrow$ (iii), (iii) $\Rightarrow$ (ii), and (ii) $\Rightarrow$ (i).

(i) $\Rightarrow$ (iii): Suppose that C is atomic and let $v_1 \leq_p v_2 \leq_p \dots$ be an atomic sequence for C, so that every word in C is a prefix of some v_i. By removing duplicates and adding intermediate words as necessary we may assume that each word v_i satisfies $|v_i| = i$. Write $v_1 = a_1$ and for $i \geq 1$ let a_{i+1} be the letter such that $v_{i+1} = v_i a_{i+1}$. Then C coincides with the set of finite prefixes of the right-infinite word $w = a_1 a_2 \dots$ obtained by successively concatenating the letters a_i.

(iii) $\Rightarrow$ (ii): Suppose C coincides with the set of finite prefixes of some right-infinite word $w = a_1 a_2 \dots$ and let $u, v \in C$. Let i and j be such that $u = a_1 \dots a_i$ and $v = a_1 \dots a_j$. If $i \leq j$ then $u \leq_p v$ and if $j \leq i$ then $v \leq_p u$, so C is a prefix chain.

(ii) $\Rightarrow$ (i): If C is a prefix chain then we can list its elements as $v_1, v_2, \ldots$ where $v_i \leq v_j$ whenever $i \leq j$. Hence the sequence $v_1 \leq_p v_2 \leq_p \ldots$ is an atomic sequence for C, and so C is atomic. □

Taking this together with Corollary 2.1.11, we then have:

Corollary 2.1.13. *Let $L \subseteq A^*$ be a regular language. Then it is decidable whether $\mathrm{Av}_p(L)$ is atomic.*

The suffix ordering

We recall that the suffix ordering $\leq_s$ on A^* is defined by

$$(u \leq_s v) \Leftrightarrow (\exists \alpha \in A^*)(v = \alpha u).$$

A language $L \subseteq A^*$ is (a chain, an anti-chain, WQO, atomic) under the suffix ordering if and only if the language L^R is (a chain, an anti-chain, WQO, atomic) under the prefix ordering. As a consequence of this, together with Corollary 2.1.6 and Proposition 1.3.18, we then have the following result:

Corollary 2.1.14. *Let $L \subseteq A^*$ be a regular language. Then for the suffix ordering on A^* it is decidable whether L is:*

- *a chain;*
- *an anti-chain;*
- *WQO.*

Letting $\mathrm{Av}_s(L)$ denote the avoidance set of a language L under the suffix ordering, as a consequence of Corollary 2.1.11 we have:

Corollary 2.1.15. *Let $L \subseteq A^*$ be a regular language. Then it is decidable whether $\mathrm{Av}_s(L)$ is WQO.*

Finally, as a consequence of Corollary 2.1.13 we have:

Corollary 2.1.16. *Let $L \subseteq A^*$ be a regular language. Then it is decidable whether $\mathrm{Av}_s(L)$ is atomic.*

2.2 The factor ordering

In this section we show that for the factor ordering on words, it is decidable whether a given finitely-based set C is atomic. Our approach will be to introduce a graph $G(C)$, called the *factor graph* for C, such that each sufficiently long word corresponds uniquely to a path in $G(C)$. The problem of deciding atomicity for C with then essentially be reduced to deciding atomicity in the contiguous subpath ordering on the set of paths in $G(C)$, and performing a small analysis on a finite set of short words in C.

2.2.1 Factor graphs

Notation 2.2.1. Throughout this section we will let $\mathrm{Av}_f(B)$ denote the avoidance set of a set $B \subseteq A^*$ under the factor ordering.

Definition 2.2.2. Let $u \in A^*$ and write $u = a_1 \dots a_n$, and let k be a positive integer with $k \leq n$. We define $\mathrm{seq}_k(u)$ to be the sequence of words $\mathrm{seq}_k(u) = (u_1, \dots, u_{n-k+1})$ where each $u_i = a_i \dots a_{i+k-1}$.

The sequence $\mathrm{seq}_k(u)$ can be thought of as the sequence of consecutive length k factors of the word u.

Example 2.2.3. Let $u = abcbc$. Then we have $\mathrm{seq}_2(u) = (ab, bc, cb, bc)$ and $\mathrm{seq}_3(u) = (abc, bcb, cbc)$.

We would also like to take a sequence S which coincides with $\mathrm{seq}_k(u)$ for some u, and recover the word u. First we introduce some notation:

Notation 2.2.4. If $u = a_1 \dots a_n$ is a word then we let $u^P = a_1 \dots a_{n-1}$ denote the largest proper prefix of u and let $u^S = a_2 \dots a_n$ denote the largest proper suffix of u.

We then make the following definition:

Definition 2.2.5. Let $S = (u_1, \dots, u_m)$ be a sequence of words of the same length k. We say S is an *overlapping k-sequence* if $u_i^S = u_{i+1}^P$ for all $i \in \{1, \dots, m-1\}$.

Example 2.2.6. The sequence (abb, bbc, bca) is a overlapping 3-sequence, and the sequence $(aaba, abac)$ is an overlapping 4-sequence.

We can construct a word from an overlapping k-sequence as defined below:

Definition 2.2.7. Let $S = (u_1, \dots, u_m)$ be an overlapping k-sequence and write each word $u_i = a_i \dots a_{i+k-1}$. Then we define the word $w(S) = a_1 \dots a_{m+k-1}$.

Example 2.2.8. We have $w(abb, bbc, bca) = abbca$ and $w(aaba, abac) = aabac$.

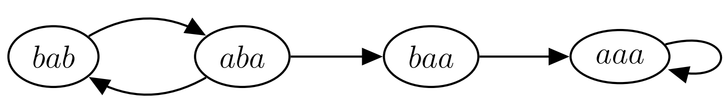
Notation 2.2.9. Let C be a finitely-based set of words and $k \geq 1$. We let C_k denote the set of words in C of length k. We also let $C_{\geq k}$ denote the set of words in C of length at least k, and make analogous definitions for the sets $C_{>k}$, $C_{\leq k}$ and $C_{<k}$.

We now define the notion of a factor graph corresponding to a finitely-based class.

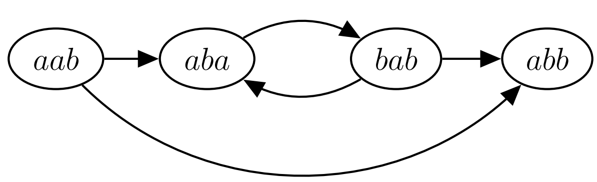
Definition 2.2.10. Let $B \subseteq A^*$ be finite, let $k = \max_{v \in B} |v|$ and let $C = \mathrm{Av}_f(B)$. Then the *factor graph* for C is the directed graph $G(C)$ which has vertex set C_k and edge set $\{(u, v) \mid u^S = v^P\}$.

We remark that the factor graph is essentially a specific induced subgraph of the de Bruijn graph [9] on A^k.

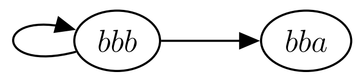
Example 2.2.11. Let $A = \{a, b\}$, $B = \{bb, aab\}$ and $C = \mathrm{Av}_f(B)$. Then $\max_{v \in B} |v| = 3$ and $C_3 = \{bab, aba, baa, aaa\}$. The factor graph $G(C)$ is shown below.



Example 2.2.12. Let $A = \{a, b\}$, $B = \{aaa, baa, bba, bbb\}$ and $C = \mathrm{Av}_f(B)$. Then $\max_{v \in B} |v| = 3$ and $C_3 = \{aab, abb, aba, bab\}$. The factor graph $G(C)$ is shown below.



Example 2.2.13. Let $A = \{a, b\}$, $B = \{aa, aba, abb, bab\}$ and $C = \mathrm{Av}_f(B)$. Then $\max_{v \in B} |v| = 3$ and $C_3 = \{bbb, bba\}$. The factor graph $G(C)$ is shown below.



We now show how paths in $G(C)$ correspond to words in C. If $P = (u_1, \ldots, u_n)$ is a path in $G(C)$ then $u_i^S = u_{i+1}^P$ for all $i \in \{1, \ldots, n-1\}$. Hence the path P is an overlapping k-sequence and the word $w(P)$ is defined. We prove the following:

Proposition 2.2.14. *Let $B \subseteq A^*$ be finite, let $k = \max_{v \in B} |v|$ and let $C = \mathrm{Av}_f(B)$. Then:*

(i) *If $u \in C_{\geq k}$ then $\mathrm{seq}_k(u)$ is a path in $G(C)$.*

(ii) *If P is a path in $G(C)$ then $w(P) \in C_{\geq k}$.*

Proof. (i) Write $u = a_1 \ldots a_n$ and let $\mathrm{seq}_k(u) = (u_1, \ldots, u_{n-k+1})$ where each $u_i = a_i \ldots a_{i+k-1}$. The set C is downward-closed, so for each $i \in \{1, \ldots, n-k+1\}$ we have $u_i \in C$. In particular we have $u_i \in C_k$ since $|u_i| = k$. Furthermore for each $i \in \{1, \ldots, n-k\}$ we have $u_i = a_i \ldots a_{i+n-1}$ and $u_{i+1} = a_{i+1} \ldots a_{i+n}$, so

$$u_i^S = a_{i+1} \ldots a_{i+n-1} = u_{i+1}^P.$$

Hence (u_i, u_{i+1}) is an edge in G for each $i \in \{1, \ldots, n-k\}$ and so $\mathrm{seq}_k(u)$ is a path in $G(C)$.

(ii) Let $P = (u_1, \ldots, u_m)$ be a path in $G(C)$ and write each $u_i = a_i \ldots a_{i+k-1}$, so that $w(P) = a_1 \ldots a_{m-k+1}$. If $v \in B$ is such that $v \leq_f w(P)$ then $|v| \leq k$, and so $v \leq_f u_i$ for some i. However, this cannot be the case since each $u_i \in C$ and $C = \mathrm{Av}_f(B)$. $\square$

We wish to show that if $u, v \in C_{\geq k}$ with $u \leq_f v$ then this is reflected in $G(C)$. We first introduce some notation:

Notation 2.2.15. If G is a directed graph then we let $\mathcal{P}(G)$ denote the set of paths in G.

We then make the following definition:

Definition 2.2.16. Let G be a directed graph. Let $P, Q \in \mathcal{P}(G)$ and write $P = (u_1, \ldots, u_n)$ and $Q = (v_1, \ldots, v_m)$, with $n \leq m$. We say that P is a *contiguous subpath* of Q if there is some $i \in \{1, \ldots, m - n + 1\}$ such that $P = (v_i, \ldots, v_{i+n-1})$. We denote this by $P \leq Q$.

The contiguous subpath relation $\leq$ is an ordering on $\mathcal{P}(G)$, since it is really just the factor ordering on a specific subset of $V(G)^*$. We then have the following:

Proposition 2.2.17. *Let $B \subseteq A^*$ be finite, let $k = \max_{v \in B} |v|$ and let $C = \mathrm{Av}_f(B)$. Then:*

(i) *If $u, v \in C_{\geq k}$ with $u \leq_f v$ then $\mathrm{seq}_k(u) \leq \mathrm{seq}_k(v)$.*

(ii) *If $P, Q \in \mathcal{P}(G)$ with $P \leq Q$ then $w(P) \leq_f w(Q)$.*

Proof. (i) Write $u = a_1 \ldots a_n$ and let $\mathrm{seq}_k(u) = (u_1, \ldots, u_{n-k+1})$ where each $u_i = a_i \ldots a_{i+k-1}$, then write $v = b_1 \ldots b_m$ and let $\mathrm{seq}_k(v) = (v_1, \ldots, v_{m-k+1})$ where each $v_i = b_i \ldots b_{i+k-1}$. Let j be such that $u = b_{j+1} \ldots b_{j+n}$. Then for each $i \in \{1, \ldots, n+k-1\}$ we can write

$$u_i = a_i \ldots a_{i+k-1} = b_{j+i} \ldots b_{j+i+k-1} = v_{j+i}$$

and so

$$\mathrm{seq}_k(u) = (u_1, \ldots, u_{n+k-1}) = (v_{j+1}, \ldots, v_{j+n+k-1}) \leq \mathrm{seq}_k(v).$$

(ii) Let $P = (u_1, \ldots, u_n)$ and write each $u_i = a_i \ldots a_{i+k-1}$ so that $w(P) = a_1 \ldots a_{n+k-1}$, then let $Q = (v_1, \ldots, v_m)$ and write each $v_i = b_i \ldots b_{i+k-1}$ so that $w(Q) = b_1 \ldots b_{m+k-1}$. Let j be such that $P = (v_{j+1}, \ldots, v_{j+n})$. Then for each $i \in \{1, \ldots, m\}$ we have $u_i = v_{j+i}$ and so

$$a_i \ldots a_{i+k-1} = b_{j+i} \ldots b_{j+i+k-1}.$$

This means that

$$w(P) = a_1 \ldots a_{n+k-1} = b_{j+1} \ldots b_{j+n+k-1} \leq_f w(Q).$$

$\square$

2.2.2 Deciding atomicity in the contiguous subpath ordering

In this subsection we show that atomicity is decidable for the contiguous subpath ordering on $\mathcal{P}(G)$. In the next subsection we will show that this implies the decidability of atomicity for the factor ordering on words.

Notation 2.2.18. If G is a directed graph and $u, v \in V(G)$ are such that there is a path in G from u to v, then we denote this by $u \to^* v$.

Definition 2.2.19. A directed graph G is *connected* if for all $u, v \in V(G)$ we have $u \to^* v$ or $v \to^* u$, and *strongly connected* if for all $u, v \in V(G)$ we have both $u \to^* v$ and $v \to^* u$.

Sometimes the word 'connected' is used to mean that G is connected when considered as an undirected graph, so that for instance the graph $\bullet\!\longrightarrow\!\bullet\!\longleftarrow\!\bullet$ would be called connected. We will have no need to single out graphs with this property, and so will exclusively use the word 'connected' as in Definition 2.2.19.

Definition 2.2.20. Let P and Q be paths such that the last vertex of P is the first vertex of Q, and write $P = (u_1, \ldots, u_n)$ and $Q = (u_n, v_1, \ldots, v_m)$. We define the *concatenation* of P and Q to be the path

$$PQ = (u_1, \ldots, u_n, v_1, \ldots, v_m).$$

We then note the following elementary facts relating the atomicity of $\mathcal{P}(G)$ to the (strong) connectedness of G:

Lemma 2.2.21. *Let G be a directed graph. Then:*

(i) *If $\mathcal{P}(G)$ is atomic under $\leq$ then G is connected.*

(ii) *If G is strongly connected then $\mathcal{P}(G)$ is atomic.*

Proof. (i) Suppose $\mathcal{P}(G)$ is atomic and let $u, v \in V(G)$. Let $P = (u_1, \ldots, u_n)$ be a join for the trivial paths (u) and (v), so that $(u) \leq P$ and $(v) \leq P$. Let i, j be such that $u = u_i$ and $v = u_j$. If $i < j$ then $u \to^* v$, and if $i > j$ then $v \to^* u$.

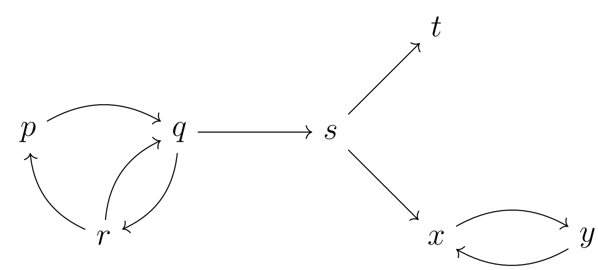
(ii) Suppose G is strongly connected and let $P = (u_1, \ldots, u_n)$ and $Q = (v_1, \ldots, v_m)$ be paths in G. Since G is strongly connected we can find a path R from u_n to v_1. Let $S = PRQ$. Then $P \le S$ and $Q \le S$, so S is a join for P and Q. □

The remainder of this subsection will focus on conditions under which $\mathcal{P}(G)$ is atomic when G is connected but not strongly connected.

Definition 2.2.22. Let G be a directed graph. We define an equivalence relation $\sim$ on $V(G)$ by $u \sim v$ if $u \to^* v$ and $v \to^* u$. The equivalence classes of $\sim$ are called the *strongly connected components* of G. A component C is *trivial* if $|C| = 1$ and *non-trivial* otherwise.

We will refer to the strongly connected components of G simply as *components* since we will not need to distinguish from any other meaning of this word.

Example 2.2.23. Let G be the graph shown below.



The components of G are $\{p, q, r\}$, $\{s\}$, $\{t\}$ and $\{x, y\}$. The components $\{s\}$ and $\{t\}$ are trivial, and the components $\{p, q, r\}$ and $\{x, y\}$ are non-trivial.

Definition 2.2.24. We define a relation $\to$ on the components of G by $C_1 \to C_2$ if $C_1 \ne C_2$ and if there are vertices $p_1 \in C_1$ and $p_2 \in C_2$ such that $p_1 \to^* p_2$.

If G is connected then any two distinct components of G are comparable under the relation $\to$, so we can list the components of G as

$$C_1 \to \cdots \to C_N.$$

Definition 2.2.25. Let G be a directed graph and let C be a component of G. An *entrance* of C is an edge (q, p) where $q \notin C$ and $p \in C$, and an *exit* of C is an edge (p, q) where $p \in C$ and $q \notin C$.

If G is connected and its components are given by $C_1 \to \cdots \to C_N$, then the components with entrances are exactly $C_2, \ldots, C_N$ and the components with exits are exactly $C_1, \ldots, C_{n-1}$.

Our aim is to prove that if G is connected but not strongly connected and $\mathcal{P}(G)$ is atomic then G must have a very specific form. This will involve a sequence of lemmas which, when taken together, will dictate the form that G must have if $\mathcal{P}(G)$ is to be atomic. We first prove the following:

Lemma 2.2.26. *Let G be a directed graph which is connected but not strongly connected and suppose $\mathcal{P}(G)$ is atomic. If C is a non-trivial component of G then C does not have both an entrance and an exit.*

Proof. Suppose that C has both an entrance $E = (q, p_1)$ and an exit $F = (p_2, r)$. Let P be a path from p_1 to p_2 and let Q be a non-trivial path from p_2 to itself. We note that all the vertices in P and Q belong to the component C. Let $S = EPF$ and $T = EPQF$, noting that each of these paths contain precisely one occurrence of each of q and r. Furthermore note that S is strictly shorter than T since the path Q is non-trivial. Suppose, aiming for a contradiction, that $R = (u_1, \ldots, u_n)$ is a join for S and T, so that $S \leq R$ and $T \leq R$. Let i, j, k, l be such that $S = (u_i, \ldots, u_j)$ and $T = (u_k, \ldots, u_l)$. We note that that $i < j$ and $k < l$, so at least one of the following comparisons must be true:

(i) $i = k$;
(ii) $j = l$;
(iii) $j < k$;
(iv) $l < i$;
(v) $i < k < j$;
(vi) $k < i < l$.

If $i = k$ or $j = l$ then S is a proper subpath of T, but this cannot be the case since T contains the vertices q and r only once each. If $j < k$ then R has the contiguous subpath $(u_j, \ldots, u_k)$, but this cannot be the case since $u_j = r$, $u_k = q$ and $r \not\to^* q$. Similarly if $l < i$ then R has the contiguous subpath $(u_l, \ldots, u_i)$, but this cannot be the case since $u_l = r$, $u_i = q$ and $r \not\to^* q$. If $i < k < j$ then u_k is a vertex of P, but this cannot be the case since $u_k = q$ and $q \notin C$. Finally if $k < i < l$ then u_i is a vertex of Q, but this cannot be the case since $u_i = q$ and $q \notin C$. Hence there is no join for S and T and so the set $\mathcal{P}(G)$ is not atomic. $\square$

From this we can immediately deduce the following result:

Lemma 2.2.27. *Let G be a directed graph which is connected but not strongly connected and let $C_1 \to \cdots \to C_N$ be the components of G. If $\mathcal{P}(G)$ is atomic then the components $C_2, \ldots, C_{N-1}$ are trivial.*

Next we show that the components of G must be arranged linearly.

Lemma 2.2.28. *Let G be a directed graph which is connected but not strongly connected, and suppose $\mathcal{P}(G)$ is atomic. If C is a component of G then C has at most one exit and at most one entrance.*

Proof. We show only that the C has at most one exit, as that the proof that C has at most one entrance will be analogous. Suppose that $E = (p_1, q)$ and $F = (p_2, r)$ are exits of C and that $S = (u_1, \ldots, u_n)$ is a join for E and F, so that $E \leq S$ and $F \leq S$. Let i, j be such that $E = (u_i, u_{i+1})$ and $F = (u_j, u_{j+1})$. If $i < j$ then $q \to^* p_2$ and if $i > j$ then $r \to^* p_1$. Neither of these are true, so $i = j$ and hence $E = F$. $\square$

The remaining lemmas in this subsection will show that the non-trivial components of G must be cyclic. We first define this formally:

Definition 2.2.29. Let G be a directed graph and let C be a component of G. We say C is *cyclic* either if C is trivial, or if for each $p \in C$ there is a unique vertex $q \in C$ such that $(q, p) \in E(G)$ and a unique vertex $r \in C$ such that $(p, r) \in E(G)$.

If a non-trivial component C is not cyclic then it either has a vertex with two outgoing edges in C or it has a vertex with two incoming edges in C. In fact we can use the pigeonhole principle to show that each of these implies the other, since the number of outgoing edges and incoming edges within C must be equal. Hence we have:

Lemma 2.2.30. *Let G be a directed graph and let C be a non-trivial component of G which is not cyclic. Then there are vertices $p, q, r \in C$ with q, r distinct such that $(p, q), (p, r) \in E(G)$.*

To show that the non-trivial components of G must be cyclic if $\mathcal{P}(G)$ is to be atomic, we first characterise cyclic components in terms of cycles at a single vertex.

Definition 2.2.31. A cycle $(p, p_1 \dots, p_n, p)$ is called *p-simple* if $p \notin \{p_1, \dots, p_n\}$.

We then have:

Lemma 2.2.32. *Let G be a directed graph, let C be a component of G and let $p \in C$. Suppose C contains a unique p-simple cycle P. Then P contains every vertex of C.*

Proof. We prove the contrapositive. Write $P = (p, p_1, \dots, p_n, p)$ and suppose there is some vertex $q \in C$ which is not in P. Let $S = (p, s_1, \dots, s_m, q)$ and $T = (q, t_1, \dots, t_k, p)$ be paths such that $p \notin \{s_1, \dots, s_m, t_1, \dots, t_k\}$. Then ST is a p-simple cycle which contains q and is therefore different from P. □

The following lemma characterises cyclic components in terms of p-simple cycles:

Lemma 2.2.33. *Let G be a directed graph and let C be a component of G. Suppose there is some $p \in C$ such that C contains a unique p-simple cycle P. Then C is cyclic.*

Proof. Write $P = (p, p_1, \dots, p_n, p)$ and note that $C = \{p, p_1, \dots, p_n\}$ by Lemma 2.2.32. For notational convenience we write $p_0 = p_{n+1} = p$. If C is not cyclic then it is non-trivial, and one of its vertices has two outgoing edges in C by Lemma 2.2.30. Hence there is some $i \in \{0, \dots, n\}$ such that $(p_i, p_k) \in E(G)$ for some $k \neq i + 1$. Then the cycle

$$(p, p_1, \dots, p_i, p_k, \dots, p_n, p)$$

is a p-simple cycle which is different from P. □

We now present our lemma on the cyclic nature of non-trivial components.

Lemma 2.2.34. *Let G be a directed graph which is connected but not strongly connected, and suppose $\mathcal{P}(G)$ is atomic. If C is a non-trivial component of G then C is cyclic.*

Proof. Suppose the components of G are $C_1 \to \cdots \to C_N$. By Lemma 2.2.26 the components $C_2, \dots, C_{N-1}$ are trivial. We will show only that if C_1 is non-trivial then it cyclic, as the argument for C_N will be analogous. By Lemma 2.2.28 the component C_1 has only one exit $E = (p, r)$. We show that C_1 contains a unique p-simple cycle, and it will follow from Lemma 2.2.33 that it is cyclic. Suppose C contains two p-simple cycles $P = (p, s_1, \dots, s_n, p)$ and $Q = (p, t_1, \dots, t_m, p)$, and let $S = PE$ and $T = QE$. Let $R = (u_1, \dots, u_M)$ be a join for S and T, so that $S \leq R$ and $T \leq R$. Let i, j, k, l be such that $S = (u_i, \dots, u_j)$ and $T = (u_k, \dots, u_l)$. If $j < l$ then R contains the contiguous subpath $(u_j, \dots, u_{l-1})$, but this cannot be the case since $u_j = r$, $u_{l-1} = p$ and $r \not\to^* p$. An analogous argument discredits the case that $j > l$, so we can assume that $j = l$. If $i < k$ then $u_k \in \{s_1, \dots, s_n\}$, but this cannot be the case since $u_k = p$ and P is p-simple. Similarly if $i > k$ then $u_i \in \{t_1, \dots, t_m\}$, but this cannot be the case since $u_i = p$ and Q is p-simple. Hence we have both $i = k$ and $j = l$, so $S = T$. □

By combining Lemmas 2.2.26 to 2.2.34 we can make significant deductions about the form which G must have is $\mathcal{P}(G)$ is atomic. In particular if the components of G are $C_1 \to \cdots \to C_N$ then we must have the following:

(i) The component C_1 is cyclic and has exactly one exit.

(ii) The components $C_2, \dots, C_{N-1}$ are trivial and have exactly one entrance and one exit.

(iii) The component C_N is cyclic and has exactly one entrance.

Hence the graph G must be a *bicycle*, which we define below.

Definition 2.2.35. A *bicycle* is a directed graph B consisting of two simple cycles $S = (s_n, s_1, \dots, s_n)$ and $E = (e_1, \dots, e_m, e_1)$, and a path $P = (s_n, p_1, \dots, p_l, e_1)$ from S to E. We will describe B as an ordered triple $B = (S, P, E)$.

We illustrate a bicycle below.

$$\begin{tikzcd}
& s_{n-1} \arrow[dr, bend left] & & & & & & e_2 \arrow[dr, bend left] & \\
\vdots \arrow[ur, bend left] & & s_n \arrow[r] \arrow[dl, bend left] & p_1 \arrow[r] & \cdots \arrow[r] & p_l \arrow[r] & e_1 \arrow[ur, bend left] & & \vdots \arrow[dl, bend left] \\
& s_1 \arrow[ul, bend left] & & & & & & e_m \arrow[ul, bend left] &
\end{tikzcd}$$

We remark that it is decidable whether a given directed graph is a bicycle, since, for instance, one can check that there are at most two vertices with out-degree 2

Our main theorem on atomicity in $\mathcal{P}(G)$ is as follows:

Theorem 2.2.36. *Let G be a directed graph which is connected but not strongly connected. Then $\mathcal{P}(G)$ is atomic under the contiguous subpath ordering if and only if G is a bicycle.*

Proof. ($\Rightarrow$) This implication is a combination of Lemmas 2.2.26 to 2.2.34.

($\Leftarrow$) Suppose G is a bicycle and write $G = (S, P, E)$ where $S = (s_n, s_1, \dots, s_n)$, $P = (s_n, p_1, \dots, p_l, e_1)$ and $E = (e_1, \dots, e_m, e_1)$. Then every path in G a contiguous subpath of a path of the form $S^i P E^i$. Hence the sequence

$$SPE \leq S^2PE^2 \leq S^3PE^3 \leq \dots$$

is an atomic sequence for $\mathcal{P}(G)$, and so $\mathcal{P}(G)$ is atomic. □

Corollary 2.2.37. *Let G be a directed graph. Then it is decidable whether $\mathcal{P}(G)$ is atomic under the contiguous subpath ordering.*

2.2.3 Deciding atomicity in the factor ordering

In this subsection we use our result on the contiguous subpath ordering to show that atomicity is decidable for the factor ordering on words.

Theorem 2.2.38. *Let $B \subseteq A^*$ be finite, let $k = \max_{v \in B} |v|$ and let $C = \mathrm{Av}_f(B)$. Then C is atomic if and only if:*

(i) *The graph $G(C)$ is either strongly connected or is a bicycle.*

(ii) *For each word $u \in C_{<k}$ there is a word $v \in C_k$ with $u \leq_f v$.*

Proof. ($\Rightarrow$) We prove the contrapositive. If condition (i) does not hold then $C_{\geq k}$ does not satisfy the join property, and so neither does C. If condition (ii) does not hold then there is a word $u \in C_{<k}$ such that $u \nleq_f v$ for all $v \in C_k$. We will fix a word $w \in C_k$ and show that u has no join with w. Indeed, if $z \in C$ is such that $u \leq_f z$ and $v \leq_f z$ then $|z| \geq k$, and so z has a factor $v \in C_k$ such that $u \leq_f v$, which is a contradiction.

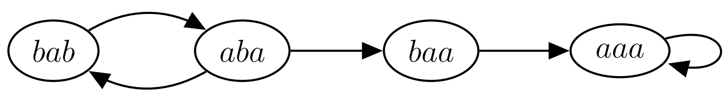
($\Leftarrow$) Let $u, v \in C$. Since condition (ii) holds, there are words $u', v' \in C_{\geq k}$ such that $u \leq_f u'$ and $v \leq_f v'$. Since condition (i) holds the set, $C_{\geq k}$ satisfies the join property, so there is a word $w \in C$ such that $u' \leq_f w$ and $v' \leq_f w$. Hence by transitivity we have $u \leq_f w$ and $v \leq_f w$, so C satisfies the join property and is atomic. $\square$

Corollary 2.2.39. *Let $B \subseteq A^*$ be finite and let $C = \mathrm{Av}_f(B)$. Then it is decidable whether C is atomic.*

Proof. Condition (i) of Theorem 2.2.38 is certainly decidable, and so is condition (ii) since there are only finitely many words in each of the sets $C_{<k}$ and C_k. $\square$

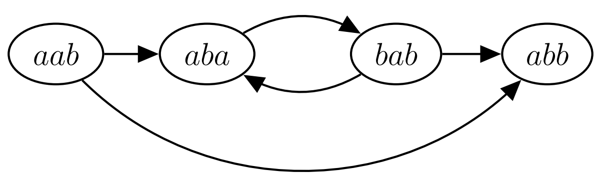
We conclude the present subsection with a number of examples of deciding atomicity.

Example 2.2.40. Returning to Example 2.2.11, we let $A = \{a, b\}$, $B = \{bb, aab\}$ and $C = \mathrm{Av}_f(B)$. The factor graph $G = G(C)$ is shown below.



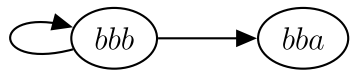
The graph $G(C)$ is a bicycle and so the set $\mathcal{P}(G)$ is atomic under the contiguous subpath ordering. The set $C_{<3}$ consists of the words aa, ab and ba, each of which is a factor of a word in C_3, since $aa \leq_f baa$ and the words $ab, ba \leq_f aba$. Hence the set C is atomic under the factor ordering by Theorem 2.2.38.

Example 2.2.41. Returning to Example 2.2.12, we let $A = \{a, b\}$, $B = \{aaa, baa, bba, bbb\}$ and $C = \mathrm{Av}_f(B)$. The factor graph $G = G(C)$ is shown below.



The graph $G(C)$ is not a bicycle and so $\mathcal{P}(G)$ is not atomic under the contiguous subpath ordering by Theorem 2.2.36. Hence C is not atomic under the factor ordering.

Example 2.2.42. As in Example 2.2.13, we let $A = \{a, b\}$, $B = \{aa, aba, abb, bab\}$ and $C = \mathrm{Av}_f(B)$. The factor graph $G = G(C)$ is shown below.



The graph G is a bicycle and so $\mathcal{P}(G)$ is atomic under the contiguous subpath ordering by Theorem 2.2.36. However, the word ab belongs to $C_{<3}$ and there is no vertex in $G(C)$ with ab as a factor, so C is not atomic by Theorem 2.2.38.

2.2.4 WQO in the factor ordering

In this subsection we will prove that it is decidable whether the set of paths in a given directed graph is WQO under the consecutive subpath ordering. We will then use this to give a solution to the WQO Problem for the factor ordering, which is a special case of Theorem 1.3.7.

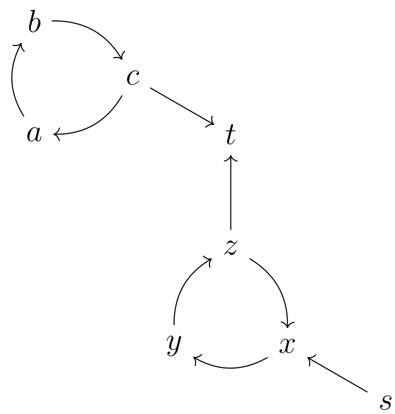
Definition 2.2.43. Let $G, H_1, \ldots, H_n$ be directed graphs. We say that G is a *union* of $H_1, \ldots, H_n$ if every vertex of G belongs to some H_i and every path of G belongs to some H_i, that is if

$$V(G) = \bigcup_{i=1}^{n} V(H_i)$$

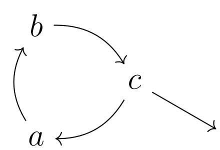
and

$$\mathcal{P}(G) = \bigcup_{i=1}^{n} \mathcal{P}(H_i).$$

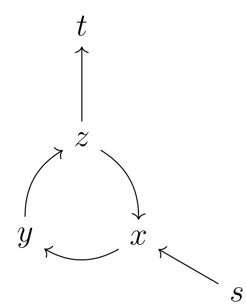
Example 2.2.44. Let G be the directed graph shown below.



Then G is a union of the directed graphs H_1 and H_2 shown below.



H_1



H_2

Definition 2.2.45. An *in-out cycle* in a directed graph is a cycle which has both an entrance and an exit.

We then have:

Proposition 2.2.46. *Let G be a directed graph. The following are equivalent:*

(i) *The graph G has no in-out cycles.*

(ii) *The graph G is a union of bicycles.*

(iii) *The set $\mathcal{P}(G)$ is WQO under the contiguous subpath ordering.*

Proof. (i) $\Rightarrow$ (ii): If G has no in-out cycles then G consists of a collection of 'start cycles' which have no entrances, 'end-cycles' which have no exists, and a collection of paths from certain starts cycles to certain end cycles. If $S = (s_n, s_1, \dots, s_n)$ is a start cycle, $E = (e_1, \dots, e_m, e_1)$ is an end cycle and $P = (s_n, p_1, \dots, p_l, e_1)$ is a path from S to E then together these form a bicycle $B = (S, P, E)$. Moreover, every path in G is a path in such a bicycle, so G is therefore a union of them.

(ii) $\Rightarrow$ (iii): Suppose that G is a union of bicycles $B_1, \dots, B_n$, so that

$$\mathcal{P}(G) = \bigcup_{i=1}^{n} \mathcal{P}(B_i).$$

In order to prove that $\mathcal{P}(G)$ is WQO, it suffices to prove that each set $\mathcal{P}(B_i)$ is WQO. Fix a bicycle $B_i = (S, P, E)$ and write $S = (s_n, s_1, \dots, s_n)$, $P = (s_n, p_1, \dots, p_l, e_1)$ and $E = (e_1, \dots, e_m, e_1)$. Then $\mathcal{P}(B_i)$ is precisely the downward closure of the set

$$\{S^i P E^j \mid i, j \geq 1\}$$

which is WQO by Lemma 1.3.11.

(iii) $\Rightarrow$ (i): We prove the contrapositive. Suppose G has an in-out cycle $P = (p_1, \dots, p_n, p_1)$ which has an entrance $E = (q, p_i)$ and an exit $F = (p_j, r)$. Let Q be the path from p_i to p_j along the cycle P, and let S be the path from p_j to p_i along the cycle P. We claim that the set

$$X = \{EQ(SQ)^i F \mid i \geq 1\}$$

is an infinite anti-chain. Indeed, if T is a path with $T < EQ(SQ)^i F$ then either T does not contain q or does not contain r, and so $T \notin X$. □

Corollary 2.2.47. *Let C be a set of words which is finitely-based under the factor ordering. Then C is WQO if and only if $G(C)$ is a union of bicycles.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose that $G(C)$ is not a union of bicycles. Then by Proposition 2.2.46, the set of paths in $G(C)$ is not WQO, and so neither is C.

($\Leftarrow$) Again we prove the contrapositive. Suppose that C contains an infinite anti-chain X and let $k = \max_{v \in \mathcal{B}(C)} |v|$. Then X has an infinite subset Y where every element has length at least k. The set Y is also an infinite anti-chain, and every word in Y corresponds to a path in $G(C)$, so the set of paths in $G(C)$ is not WQO. Hence $G(C)$ is not a bicycle by Proposition 2.2.46. $\square$

Chapter 3

Rational orderings on words

In this chapter we discussion *rational* orderings on words, that is, ones which can be generated by transducers.

3.1 Insertion relations

In this section we discuss the class of *insertion relations*. This work is based largely on the paper [26] written by the author, in which this class was first introduced.

3.1.1 Definitions and basic properties

Definition 3.1.1. An *insertion transducer* is a transducer $\mathcal{T} = (Q, A, \delta, q_I)$ whose transitions are exactly as follows:

(i) For each state $q \in Q$ and for each letter $a \in A$, there is exactly one state $p \in Q$ such that $q \xrightarrow{a:a} p$ is a transition in $\mathcal{T}$.

(ii) For each state $q \in Q$ there is a set of letters $A(q) \subseteq A$ such that $q \xrightarrow{\varepsilon:a} q$ is a transition in $\mathcal{T}$ for each $a \in A(q)$.

A transition of type (i) described above can be regarded as $\mathcal{T}$ *copying* a letter of the input word, while a transition of type (ii) can be regarded as $\mathcal{T}$ *inserting* a letter into the input word. Intuitively, an insertion transducer copies an input word u and can insert certain letters into u, with the set of insertable letter depending on the state which $\mathcal{T}$ is in.

Definition 3.1.2. The relation generated by an insertion transducer $\mathcal{T}$ is called an *insertion relation*, and is denoted by $\leq_{\mathcal{T}}$. If an insertion relation is an ordering, then it is called an *insertion ordering*.

An insertion relation $\leq_{\mathcal{T}}$ is always reflexive since $\mathcal{T}$ can copy any word, and it is always anti-symmetric since $u <_{\mathcal{T}} v$ implies $|u| < |v|$. Hence we have the following:

Observation 3.1.3. *An insertion relation is an insertion ordering if and only if it is transitive.*

Since $u <_{\mathcal{T}} v$ implies $|u| < |v|$, an insertion relation cannot admit any infinite descending chains. Hence we have the following:

Observation 3.1.4. *An insertion ordering is a WQO if and only if it admits no infinite anti-chains.*

We now introduce some notation which we will use throughout the section.

Notation 3.1.5. Let $T = (Q, A, \delta, q_I)$ be an insertion transducer, let $q \in Q$ and let $u \in A^*$. Let p be the unique state such that $\mathcal{T}$ has a path from q to p labelled by (u, u). Then we will write $p = q \cdot u$.

Notation 3.1.6. Let $T = (Q, A, \delta, q_I)$ be an insertion transducer and let $q \in Q$. We let $W(q)$ denote the set of words u such that $\mathcal{T}$ has a path from q_I to q labelled by (u, u). That is:

$$W(q) = \{u \in A^* \mid q_I \cdot u = q\}.$$

If $u \in W(q)$ then we may write $A(u)$ in place of $A(q)$.

We note the following:

Observation 3.1.7. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer and let $q \in Q$.*

Then the language $W(q)$ is regular, and is accepted by the DFA

$$\mathcal{A} = (Q, A, \delta', q_I, \{q\})$$

where $\delta'(p,a) = s$ whenever $p \xrightarrow{a:a} s$ is a transition in $\mathcal{T}$.

As stated earlier, an insertion transducer $\mathcal{T}$ inserts letters into a word, with the set of letters available for insertion depending on the state which $\mathcal{T}$ is in. With this in mind, we can characterise the insertion relation $\leq_{\mathcal{T}}$ generated by $\mathcal{T}$ as follows:

Observation 3.1.8. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer, let $u, v \in A^*$ and write $u = a_1 \dots a_n$. Then $u \leq_{\mathcal{T}} v$ if and only if there are words*

$$v_0 \in A(\varepsilon)^*, v_1 \in A(a_1)^*, v_2 \in A(a_1a_2)^*, \dots, v_n \in A(u)^*$$

such that $v = v_0a_1v_1a_2v_2 \dots a_nv_n$.

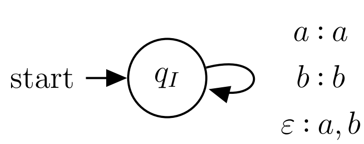
Examples of insertion relations

We demonstrate that some word orderings from the literature are in fact insertion relations, and then present some new examples.

Example 3.1.9. The subword ordering on A^* is an insertion relation. Indeed, it is generated by the insertion transducer

$$\mathcal{T}_w(A) = (\{q_I\}, A, \delta, q_I)$$

whose copy transitions are given by $q_I \xrightarrow{a:a} q_I$ for all $a \in A$ and where $A(q_I) = A$. We show $\mathcal{T}_w(a,b)$ below.



Example 3.1.10. Aichinger et al. [1] introduced what they call the *embedding* ordering. This ordering is denoted by $\leq_E$ and is defined as follows. Let $u, v \in A^*$ and

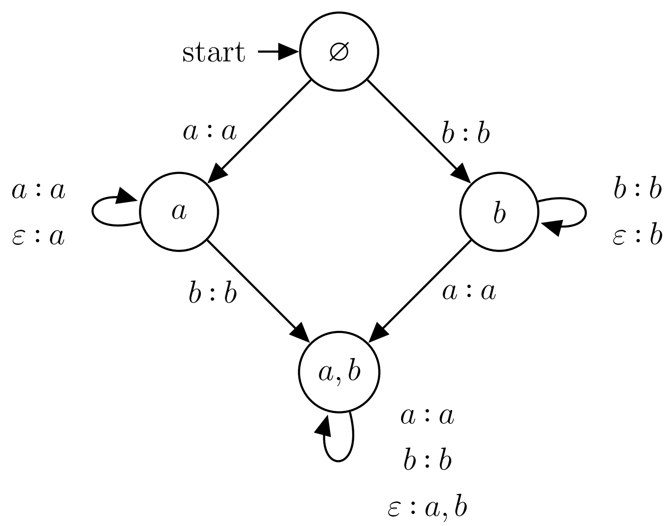
write $u = a_1 \dots a_n$. Then $u \leq_E v$ if there are words

$$v_1 \in \{a_1\}^*, v_2 \in \{a_1, a_2\}^*, \dots, v_n \in \{a_1, \dots, a_n\}^*$$

such that $v = a_1 v_1 \dots a_n v_n$. Intuitively we have $u \leq_E v$ if v can be obtained from u by inserting letters after their first occurrence in u. For instance we have $ab \leq_E aba$, but $ab \nleq_E bab$. In their paper, Aichinger et al. showed that the embedding ordering is a WQO. We show that the embedding ordering is an insertion relation. Indeed, it is generated by the insertion transducer

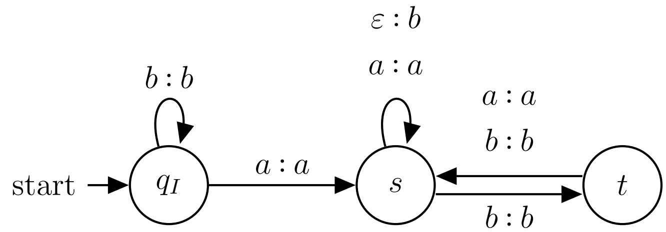
$$\mathcal{T}_E(A) = (\mathcal{P}(A), A, \delta, \varnothing)$$

whose copy transitions are given by $S \xrightarrow{a:a} S \cup \{a\}$ for each $S \subseteq A$ and each $a \in A$, and where $A(S) = S$ for each $S \subseteq A$. For instance, the transducer $\mathcal{T}_E(a, b)$ is given by

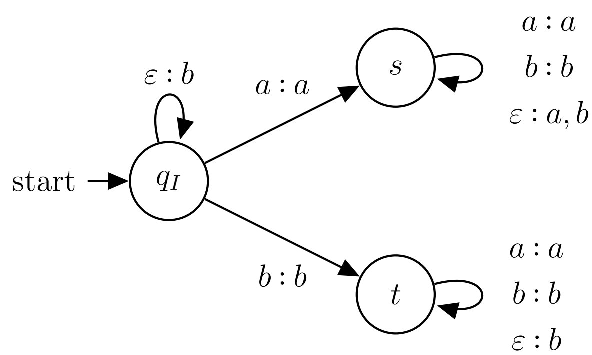


We now present some new examples of insertion relations.

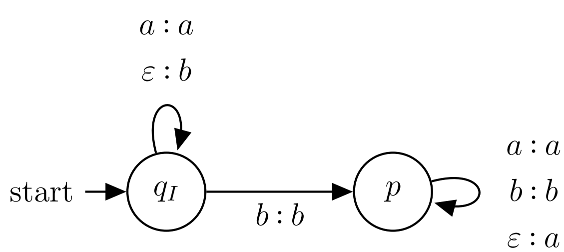
Example 3.1.11. The transducer $\mathcal{T}$ shown below is an insertion transducer over the alphabet $\{a, b\}$. We have $A(q_I) = A(t) = \varnothing$ and $A(s) = \{b\}$. As an instance of the insertion relation $\leq_{\mathcal{T}}$ generated by $\mathcal{T}$, we have $ba \leq_{\mathcal{T}} bab$.



Example 3.1.12. The transducer $\mathcal{T}$ shown below is an insertion transducer over the alphabet $\{a, b\}$. We have $A(q_I) = A(t) = \{b\}$ and $A(s) = \{a, b\}$. As an instance of the insertion relation $\leq_{\mathcal{T}}$ generated by $\mathcal{T}$, we have $a \leq_{\mathcal{T}} bbab$.



Example 3.1.13. The transducer $\mathcal{T}$ shown below is an insertion transducer over the alphabet $\{a, b\}$. We have $A(q_I) = \{b\}$ and $A(p) = \{a\}$. As an instance of the insertion relation $\leq_{\mathcal{T}}$ generated by $\mathcal{T}$, we have $b \leq_{\mathcal{T}} ba$.



3.1.2 Deciding transitivity

An insertion relation need not be an ordering. To see this, we consider the insertion transducer $\mathcal{T}$ from Example 3.1.13. We have $\varepsilon \leq_{\mathcal{T}} b \leq_{\mathcal{T}} ba$, but $\varepsilon \nleq_{\mathcal{T}} ba$ since $a \notin A(\varepsilon)$. We therefore devote this subsection to investigating conditions under which a given insertion relation is an ordering, with the end result being that the property is decidable.

Composition of insertion transducers

In general, it is not decidable whether a given rational relation is transitive [19]. In order to decide whether a given insertion transducer $\mathcal{T}$ generates an ordering, we will consider the composition of $\mathcal{T}$ with itself. We denote this transducer by $\mathcal{T}^2$ and the relation it generates by $\leq^2_{\mathcal{T}}$. Since the relation $\leq_{\mathcal{T}}$ is reflexive, it will be transitive if and only if $\mathcal{T}$ and $\mathcal{T}^2$ are equivalent, meaning that the relations $\leq_{\mathcal{T}}$ and $\leq^2_{\mathcal{T}}$ coincide. Equivalence of transducers is not decidable in general [13], but we will show that it is in this case.

When $\mathcal{T}$ is an insertion transducer, the transitions in $\mathcal{T}^2$ are exactly those of the following forms:

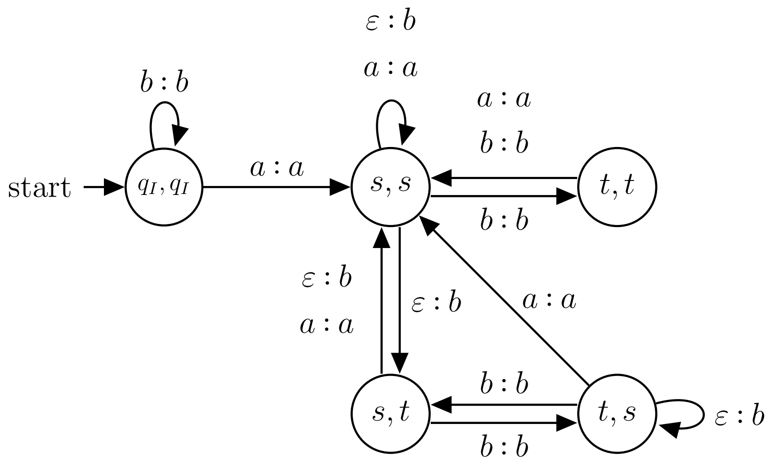
(i) $(q_1, p_1) \xrightarrow{a\,:\,a} (q_2, p_2)$ where $q_1 \xrightarrow{a\,:\,a} q_2$ and $p_1 \xrightarrow{a\,:\,a} p_2$ are transitions in $\mathcal{T}$;

(ii) $(q, p_1) \xrightarrow{\varepsilon\,:\,a} (q, p_2)$ where $a \in A(q)$ and where $p_1 \xrightarrow{a\,:\,a} p_2$ is a transition in $\mathcal{T}$;

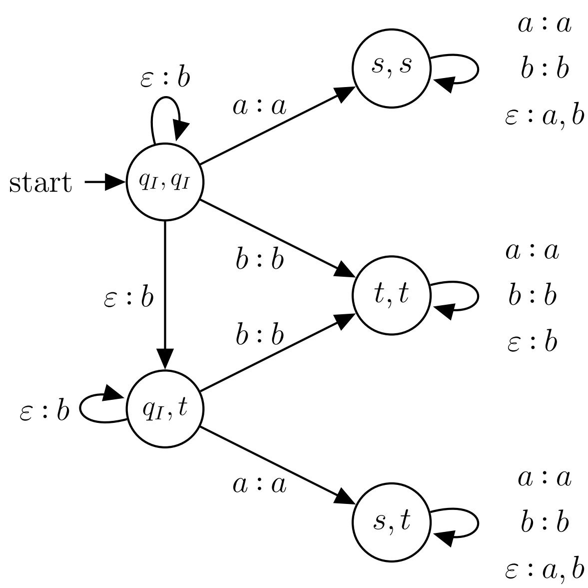
(iii) $(q, p) \xrightarrow{\varepsilon\,:\,a} (q, p)$ where $a \in A(p)$.

Definition 3.1.14. We refer to the above transitions as type (i), (ii) and (iii) transitions respectively.

Example 3.1.15. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.11. Then $\mathcal{T}^2$ is given by:



Example 3.1.16. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.12. Then $\mathcal{T}^2$ is given by:



The Insertion Path Condition (IPC)

In order to show that transitivity is decidable, we introduce a condition which an insertion transducer may satisfy, called the *Insertion Path Condition* (IPC). We will go on to show that this condition is equivalent to transitivity, and that it is decidable.

Definition 3.1.17. Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer and consider the transducer $\mathcal{T}^2$. Let P be a path in $\mathcal{T}^2$ and suppose P is labelled by (u, v). The *in-path* of P is the path in $\mathcal{T}$ labelled by (u, u), and we denote it by $\mathrm{in}(P)$. The *out-path* of P is the path in $\mathcal{T}$ labelled by (v, v), and we denote it by $\mathrm{out}(P)$.

Example 3.1.18. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.11. The transducer $\mathcal{T}^2$ was shown in Example 3.1.15. Let P be the path in $\mathcal{T}^2$ given by

$$(q_I, q_I) \xrightarrow{a\,:\,a} (s, s) \xrightarrow{\varepsilon\,:\,b} (s, t) \xrightarrow{b\,:\,b} (t, s).$$

The path P is labelled by (ab, abb), and so $\mathrm{in}(P)$ is the path in $\mathcal{T}$ labelled by (ab, ab). This is given by

$$q_I \xrightarrow{a\,:\,a} s \xrightarrow{b\,:\,b} t.$$

Likewise, $\mathrm{out}(P)$ is the path in $\mathcal{T}$ labelled by (abb, abb). This is given by

$$q_I \xrightarrow{a\,:\,a} s \xrightarrow{b\,:\,b} t \xrightarrow{b\,:\,b} s.$$

Definition 3.1.19. Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer and let $b \in A$. Let P be a path in $\mathcal{T}$ containing only copy transitions, given by

$$q_I = q_0 \xrightarrow{a_1\,:\,a_1} q_1 \xrightarrow{a_2\,:\,a_2} \cdots \xrightarrow{a_n\,:\,a_n} q_n.$$

We say that P is a *b-insertion path* if there is some $i \in \{0, \ldots, n-1\}$ such that $b \in A(q_i)$ and $a_{i+1} = a_{i+2} = \cdots = a_n = b$. The largest such i is the *b-insertion length* of the path P.

Example 3.1.20. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.11, and let P be the path in $\mathcal{T}$ given by

$$q_I \xrightarrow{b\,:\,b} q_I \xrightarrow{a\,:\,a} s \xrightarrow{b\,:\,b} t.$$

The final transition of P is given by $s \xrightarrow{b\,:\,b} t$, and we have $b \in A(s)$, so P is a b-insertion path. Its b-insertion length is 2.

Our condition is then as follows:

Insertion Path Condition (IPC): For each state (q,p) accessible in $\mathcal{T}^2$ such that there is a letter $b \in A(p) \backslash A(q)$, and for each path P in $\mathcal{T}^2$ from (q_I, q_I) to (q,p), the path $\mathrm{in}(P)$ is a b-insertion path.

Auxiliary results for deciding transitivity

Lemma 3.1.21. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer and let P be a path in $\mathcal{T}^2$ from (q_I, q_I) to some state (q,p). Let $u \in A^*$ be such that* $\mathrm{in}(P)$ *is labelled by (u,u). Then there is a word $v \in W(p)$ with $u \leq_{\mathcal{T}} v$.*

Proof. Let P' be the path obtained from P by deleting all type (iii) transitions. Since type (iii) transitions do not change the state which $\mathcal{T}$ is in, the path P' is also from (q_I, q_I) to (q,p). If $u = a_1 \dots a_n$ then there are words $v_0, v_1, \dots, v_n$ such that P' is labelled by

$$(\varepsilon, v_0)(a_1, a_1)(\varepsilon, v_1) \dots (a_n, a_n)(\varepsilon, v_n).$$

Since P' has no type (iii) transitions, each subpath (ε, v_i) consists of a (possibly empty) sequence of type (ii) transitions. Hence by putting $v = v_0 a_1 v_1 \dots a_n v_n$ we see that $u \leq_{\mathcal{T}} v$ and $v \in W(p)$. □

Lemma 3.1.22. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer satisfying IPC. Let $v \in A(q_I)^*$ and let $p \in Q$ be such that $v \in W(p)$. Then $A(p) \subseteq A(q_I)$.*

Proof. Since $v \in A(q_I)^*$ and $v \in W(p)$, there is a path P in $\mathcal{T}^2$ from (q_I, q_I) to (q_I, p) labelled by (ε, v). If there is a letter $b \in A(p) \backslash A(q_I)$ then $\mathrm{in}(P)$ is a b-insertion path, but this cannot be the case since a b-insertion path has at least one transition labelled by (b,b) and $\mathrm{in}(P)$ has no transitions. □

Lemma 3.1.23. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer satisfying IPC, and let (q,p) be an accessible state in $\mathcal{T}^2$. Then either $A(p) \subseteq A(q)$ or $|A(p) \backslash A(q)| = 1$.*

Proof. Suppose that $A(p) \nsubseteq A(q)$ and that there are letters $b, c \in A(p) \backslash A(q)$. We show that $b = c$. Let P be a path in $\mathcal{T}^2$ from (q_I, q_I) to (q,p) and suppose $\mathrm{in}(P)$ is given by

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_n : a_n} q_n = q.$$

Then $\mathrm{in}(P)$ is both a b-insertion path and a c-insertion path, so $a_n = b = c$. □

Lemma 3.1.24. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer satisfying IPC. Let (q, p) be an accessible state in $\mathcal{T}^2$ and suppose there is a letter $b \in A(p)\backslash A(q)$. Let P be a path in $\mathcal{T}^2$ from (q_I, q_I) to (q, p). Suppose* $\mathrm{in}(P)$ *is given by*

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_n : a_n} q_n = q$$

and has insertion length i. Then $A(q_n) \subseteq A(q_{n-1}) \subseteq \cdots \subseteq A(q_{i+1})$.

Proof. Since i is the b-insertion length of $\mathrm{in}(P)$ we have $a_{i+1} = a_{i+2} = \cdots = a_n = b$. Let $k \in \{i+1, \ldots, n-1\}$ and suppose there is a letter $c \in A(q_{k+1})\backslash A(q_k)$. Then $\mathcal{T}^2$ contains the path P' given by

$$(q_I, q_I) = (q_0, q_0) \xrightarrow{a_1 : a_1} \cdots \xrightarrow{a_i : a_i} (q_i, q_i) \xrightarrow{\varepsilon : b} (q_i, q_{i+1}) \xrightarrow{b : b} \cdots \xrightarrow{b : b} (q_k, q_{k+1})$$

with $\mathrm{in}(P')$ being given by

$$q_I = q_0 \xrightarrow{a_1 : a_1} \cdots \xrightarrow{a_i : a_i} q_i \xrightarrow{b : b} \cdots \xrightarrow{b : b} q_k.$$

We have $c \in A(q_{k+1})\backslash A(q_k)$, so the path $\mathrm{in}(P')$ must be a c-insertion path, and hence $c = b$. But i is the b-insertion length of $\mathrm{in}(P)$ and $i < k + 1$, so $b \notin A(q_{k+1})$, a contradiction. □

Notation 3.1.25. We let $\mathrm{Alph}(u)$ denote the set of letters appearing in a word u.

In the following lemma we consider an insertion transducer $\mathcal{T}$ satisfying IPC. If $\mathcal{T}^2$ has a transition labelled by (ε, b) starting at a state (q, p) and $b \notin A(q)$, then we have $b \in A(p)\backslash A(q)$. Hence if P is a path in $\mathcal{T}^2$ from (q_I, q_I) to (q, p) then $\mathrm{in}(P)$ is a b-insertion path. Furthermore if $w_{i+1}, w_{i+2}, \ldots, w_n \in A^*$ with $b \in \mathrm{Alph}(w_n)$ then the word $w = w_{i+1} b w_{i+2} \ldots b w_n$ has at least $n - i$ occurrences of the letter b. Hence we can write $w = x b y_{i+1} b y_{i+2} \ldots b y_n$ where $y_{i+1}, y_{i+2}, \ldots, y_n \in A^*$ are such that $b \notin \mathrm{Alph}(y_{i+1} y_{i+2} \ldots y_n)$.

Lemma 3.1.26. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer satisfying IPC. Let P be a path in $\mathcal{T}^2$ and suppose P is labelled by (u, w). Write $u = a_1 \dots a_n$ and suppose* $\mathrm{in}(P)$ *is given by*

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_n : a_n} q_n.$$

Let $w_0, w_1, \dots, w_n$ be words such that $w = w_0 a_1 w_1 \dots a_n w_n$ and such that P is labelled by

$$(\varepsilon, w_0)(a_1, a_1)(\varepsilon, w_1) \dots (a_n, a_n)(\varepsilon, w_n).$$

Suppose there is a letter $b \in \mathrm{Alph}(w_n) \backslash A(q_n)$. Let i be the b-insertion length of $\mathrm{in}(P)$ *and let $x, y_{i+1}, \dots, y_n$ be words such that $b \notin \mathrm{Alph}(y_{i+1} \dots y_n)$ and such that*

$$w_{i+1} b w_{i+2} \dots b w_n = x b y_{i+1} b y_{i+2} \dots b y_n.$$

Then:

(i) $\mathrm{Alph}(w_n) \backslash \{b\} \subseteq A(q_n)$;

(ii) $y_k \in A(q_k)^*$ *for* $k \in \{i+1, \dots, n\}$;

(iii) $x \in (A(q_{i+1}) \cup \{b\})^*$.

Proof. (i): In a similar argument to the proof of Lemma 3.1.23, if there is a letter $c \in \mathrm{Alph}(w_n) \backslash A(q_n)$ then $\mathrm{in}(P)$ must be both a b-insertion path and a c-insertion path, so $b = c$.

(ii): We note that $\mathrm{Alph}(w_n) \backslash \{b\} \subseteq A(q_n)$ by (i). The suffix w_n has at least one occurrence of the letter b, while the suffix y_n has none. Hence y_n must be a factor of w_n, and since $b \notin \mathrm{Alph}(y_n)$ we see that $y_n \in A(q_n)^*$. Similarly, the suffix $w_{n-1} b w_n$ has at least two occurrences of the letter b, while the suffix $y_{n-1} b y_n$ has only one. Hence y_{n-1} must be a factor of either w_{n-1} or w_n, and since $b \notin \mathrm{Alph}(y_{n-1})$ we see that $y_{n-1} \in A(q_{n-1})^* \cup A(q_n)^*$. Hence $y_{n-1} \in A(q_{n-1})^*$ by Lemma 3.1.24. Continuing in this fashion we obtain the desired result.

(iii): From (i), (ii) and Lemma 3.1.24 we see that $x b y_{i+1} b y_{i+2} \dots b y_n \in (A(q_{i+1}) \cup \{b\})^*$, and in particular $x \in (A(q_{i+1}) \cup \{b\})^*$. □

Transitivity characterisation

We now show that transitivity is equivalent to satisfying IPC.

Theorem 3.1.27. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer generating an insertion relation $\leq_{\mathcal{T}}$. Then $\leq_{\mathcal{T}}$ is transitive if and only if $\mathcal{T}$ satisfies IPC.*

Proof. ($\Rightarrow$) Suppose that $\leq_{\mathcal{T}}$ is transitive, but that $\mathcal{T}$ does not satisfy IPC. Then there is a path P in $\mathcal{T}^2$ from (q_I, q_I) to some state (q, p) such that there is a letter $b \in A(p)\backslash A(q)$ and such that in(P) is not a b-insertion path. Suppose in(P) is given by

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_i : a_i} q_i \xrightarrow{a : a} s \underbrace{\xrightarrow{b : b} \cdots \xrightarrow{b : b}}_{b^n : b^n} q_n = q$$

where:

(i) for every state r on the path from s to q labelled by (b^n, b^n) we have $b \notin A(r)$;

(ii) $a \neq b$.

Put $u = a_1 \dots a_i$. By Lemma 3.1.21 there is a word $v \in W(p)$ such that $uab^n \leq_{\mathcal{T}} v$. Since $b \in A(p)$ we have $v \leq_{\mathcal{T}} vb^{n+1}$, and since $\leq_{\mathcal{T}}$ is transitive we have $uab^n \leq_{\mathcal{T}} vb^{n+1}$. By (i), if S is a path in $\mathcal{T}$ starting at q_I and labelled by (uab^n, vb^{n+1}) then the last n transitions of S must all be labelled by (b, b). By removing these transitions we see there is also a path in $\mathcal{T}$ starting at q_I and labelled by (ua, vb), and so $ua \leq_{\mathcal{T}} vb$. The last transition of this path must be labelled by (b, b) since $b \notin A(s)$, but this cannot be the case by (ii).

($\Leftarrow$) Suppose IPC holds, and let $u, v, w \in A^*$ be such that $u \leq_{\mathcal{T}} v \leq_{\mathcal{T}} w$. Then $u \leq^2_{\mathcal{T}} w$, and so there is a path P_1 in $\mathcal{T}^2$ starting at (q_I, q_I) and labelled by (u, w). We will show that there is a path in $\mathcal{T}$ starting at q_I and labelled by (u, w), and so $u \leq_{\mathcal{T}} w$. Write $u = a_1 \dots a_n$ and let $w_0, w_1, \dots, w_n$ be words such that $w = w_0 a_1 w_1 \dots a_n w_n$ and such that P_1 is labelled by

$$(\varepsilon, w_0)(a_1, a_1)(\varepsilon, w_1) \dots (a_n, a_n)(\varepsilon, w_n).$$

Suppose $\mathrm{in}(P_1)$ is given by

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_n : a_n} q_n.$$

We proceed by induction on the length n of the word u. For $n = 0$ we have $u = \varepsilon$ and $w = w_0$. We then have $w_0 \in A(q_I)^*$ by Lemma 3.1.22, and so $u \leq_{\mathcal{T}} w$. Now let $n \geq 1$ and suppose the result holds for all words of length at most $n - 1$.

We first consider the case where $w_n \in A(q_n)^*$. If we let $u' = a_1 \dots a_{n-1}$ and

$$w' = w_0 a_1 w_1 \dots a_{n-1} w_{n-1},$$

then by induction we have $u' \leq_{\mathcal{T}} w'$. Hence $\mathcal{T}$ has a path from q_I to q_{n-1} labelled by (u', w'). Since $w_n \in A(q_n)^*$, we see that $\mathcal{T}$ has a path from q_{n-1} to q_n labelled by $(a_n, a_n w_n)$. By concatenating these paths we see that $\mathcal{T}$ has a path starting at q_I labelled by (u, w), and so $u \leq_{\mathcal{T}} w$.

We now consider the case where there is a letter $b_1 \in \mathrm{Alph}(w_n) \backslash A(q_n)$. Since $\mathcal{T}$ satisfies IPC, we see that $\mathrm{in}(P_1)$ is a b_1-insertion path. Let i_1 be its b_1-insertion length, so that $b_1 \in A(q_{i_1})$ and $a_{i_1+1} = a_{i_1+2} = \cdots = a_n = b_1$. From Lemma 3.1.26 (i) we have

$$\mathrm{Alph}(w_n) \backslash \{b_1\} \subseteq A(q_n).$$

Let $x_1, y_{i_1+1}, y_{i_1+2}, \dots, y_n$ be words such that $b_1 \notin \mathrm{Alph}(y_{i_1+1} y_{i_1+2} \dots y_n)$ and such that

$$w_{i_1+1} b_1 w_{i_1+2} \dots b_1 w_n = x_1 b_1 y_{i_1+1} b_1 y_{i_1+2} \dots b_1 y_n.$$

Then from Lemma 3.1.26 (ii) we have $y_k \in A(q_k)^*$ for $k \in \{i_1 + 1, \dots, n\}$. We now consider whether the word x_1 belongs to $A(q_{i_1})^*$. We first consider the case where this holds. If we let $u' = a_1 \dots a_{i_1}$ and $w' = w_0 a_1 w_1 \dots a_{i_1} w_{i_1}$ then by induction we have $u' \leq_{\mathcal{T}} w'$, and so $\mathcal{T}$ has a path from q_I to q_{i_1} labelled by (u', w'). Since $b_1 x_1 \in A(q_{i_1})^*$ we see that $\mathcal{T}$ has a loop at q_{i_1} labelled by $(\varepsilon, b_1 x_1)$. Furthermore, since $y_k \in A(q_k)^*$ for $k \in \{i_1 + 1, \dots, n\}$ we see that $\mathcal{T}$ has a path from q_{i_1} to q_n labelled by

$$(b_1, b_1)(\varepsilon, y_{i_1+1})(b_1, b_1)(\varepsilon, y_{i_1+2}) \dots (b_1, b_1)(\varepsilon, y_n). \quad (\dagger)$$

By concatenating these paths we see that $\mathcal{T}$ has a path starting at q_I and labelled by (u, w), so $u \leq_{\mathcal{T}} w$.

We now consider the case where there is a letter $b_2 \in \mathrm{Alph}(x_1) \backslash A(q_{i_1})$. From Lemma 3.1.26 (iii) we have $b_2 \in A(q_{i+1}) \cup \{b_1\}$, and since $b_1 \in A(q_{i_1})$ we see that $b_2 \in A(q_{i_1+1}) \backslash A(q_{i_1})$. The transducer $\mathcal{T}^2$ contains the path P_2 given by

$$(q_I, q_I) = (q_0, q_0) \xrightarrow{a_1 : a_1} (q_1, q_1) \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_{i_1} : a_{i_1}} (q_{i_1}, q_{i_1}) \xrightarrow{\varepsilon : b_1} (q_{i_1}, q_{i_1+1})$$

with $\mathrm{in}(P_2)$ being given by

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_{i_1} : a_{i_1}} q_{i_1}.$$

Since $b_2 \in A(q_{i_1+1}) \backslash A(q_{i_1})$, we see that $\mathrm{in}(P_2)$ must be a b_2-insertion path. Let i_2 be its b_2-insertion length, so that $b_2 \in A(q_{i_2})$ and $a_{i_2+1} = a_{i_2+2} = \cdots = a_{i_1} = b_2$. Let $x_2, y_{i_2+1}, y_{i_2+2}, \ldots, y_{i_1}$ be words such that $b_2 \notin \mathrm{Alph}(y_{i_2+1} y_{i_2+2} \ldots y_{i_1})$ and such that

$$w_{i_2+1} b_2 w_{i_2+2} \ldots b_2 w_{i_1} b_1 x_1 = x_2 b_2 y_{i_2+1} b_2 y_{i_2+2} \ldots b_2 y_{i_1}.$$

Then from Lemma 3.1.26 (ii) we have $y_k \in A(q_k)^*$ for $k \in \{i_2 + 1, \ldots, i_1\}$. We must now consider whether or not the word x_2 belongs to $A(q_{i_2})^*$.

Continuing in this fashion we obtain a finite strictly decreasing sequence of indices i_k, and a finite sequence of words x_k, such that the following conditions hold for $k \geq 2$:

(i) There is a letter $b_k \in \mathrm{Alph}(x_{k-1}) \backslash A(q_{i_{k-1}})^*$.

(ii) The index i_k is the b_k-insertion length of the path from q_0 to $q_{i_{k-1}}$ labelled by $(a_1 \dots a_{i_{k-1}}, a_1 \dots a_{i_{k-1}})$.

(iii) We can write

$$w_{i_k+1} b_k w_{i_k+2} \dots b_k w_{i_{k-1}} b_{k-1} x_{k-1} = x_k b_k y_{i_k+1} b_k y_{i_k+2} \dots b_k y_{i_{k-1}}$$

for some words $y_{i_k+1}, y_{i_k+2}, \dots, y_{i_{k-1}}$ such that $b_k \notin \mathrm{Alph}(y_{i_k+1} y_{i_k+2} \dots y_{i_{k-1}})$.

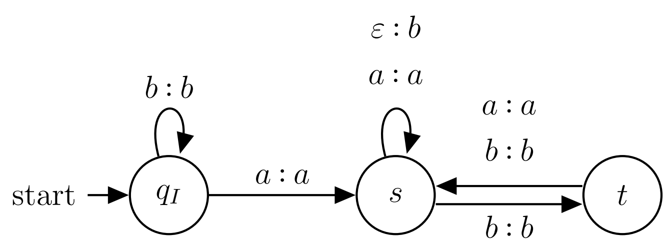
(iv) For $j \in \{i_k + 1, \dots, i_{k-1}\}$ we have $y_j \in \mathrm{Alph}(q_j)^*$.

We claim that for some m we have $x_m \in A(q_{i_m})^*$. This is since each $i_k > i_{k+1}$, and if $i_m = 0$ then $w_0 \in A(q_I)^*$ by Lemma 3.1.22. If we let $u' = a_1 \dots a_{i_m}$ and $w' = w_0 a_1 w_1 \dots a_{i_m} w_{i_m}$ then by induction we have $u' \leq_{\mathcal{T}} w'$, and so $\mathcal{T}$ has a path from q_I to q_{i_m} labelled by (u', w'). Since $b_m x_m \in A(q_{i_m})^*$, we see that $\mathcal{T}$ has a loop at q_{i_m} labelled by $(\varepsilon, b_m x_m)$. For $k \in \{2, \dots, m\}$ we see that $\mathcal{T}$ has a path from q_{i_k} to $q_{i_{k-1}}$ labelled by

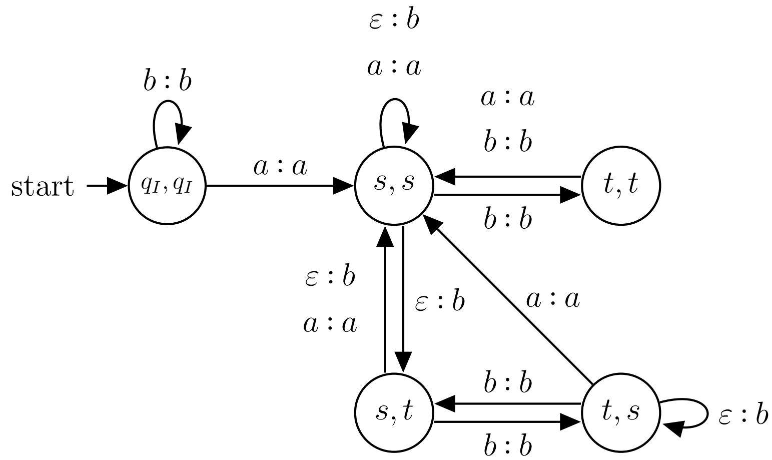
$$(b_k, b_k)(\varepsilon, y_{i_k+1})(b_k, b_k)(\varepsilon, y_{i_k+2}) \dots (b_k, b_k)(\varepsilon, y_{i_{k-1}}).$$

By concatenating these paths, together with the path from q_{i_1} to q_n labelled by (†), we see that $\mathcal{T}$ has a path starting at q_I and labelled by (u, w), so $u \leq_{\mathcal{T}} w$. □

Example 3.1.28. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.11. This was given by:



and $\mathcal{T}^2$ was given by:



We show that $\mathcal{T}$ satisfies IPC. To do this we consider each state (q, p) accessible in $\mathcal{T}^2$ such that $A(p) \backslash A(q)$ is non empty. The only such state is (t, s), where we have $A(s) \backslash A(t) = \{b\}$. Let P be a path in $\mathcal{T}^2$ from (q_I, q_I) to (t, s). The only transition in $\mathcal{T}$ which ends at t is

$$s \xrightarrow{b:b} t,$$

and so this must be the final transition of $\mathrm{in}(P)$. We have $b \in A(s)$, and so $\mathrm{in}(P)$ is a b-insertion path. Hence $\mathcal{T}$ satisfies IPC, and so $\leq_{\mathcal{T}}$ is transitive by Theorem 3.1.27.

As a corollary to Theorem 3.1.27, we note the following sufficient condition for transitivity:

Corollary 3.1.29. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer. If for every state (q, p) accessible in $\mathcal{T}^2$ we have $A(p) \subseteq A(q)$, then the insertion relation $\leq_{\mathcal{T}}$ is transitive.*

Proof. Assume that for each state (q, p) accessible in $\mathcal{T}^2$ we have $A(p) \subseteq A(q)$. Then there is no state (q, p) accessible in $\mathcal{T}^2$ such that $A(p) \backslash A(q)$ is non-empty, and so $\mathcal{T}$ satisfies IPC. Hence $\leq_{\mathcal{T}}$ is transitive by Theorem 3.1.27. $\square$

Deciding IPC

Our goal is now to show that IPC is decidable by introducing a bounded version.

Notation 3.1.30. If P is a path in a transducer then we denote the first and last states of P by $s(P)$ and $t(P)$ respectively.

Definition 3.1.31. Let P and S be paths with $s(P) = s(S)$ and $t(P) = t(S)$. We say that S is a *subpath* of P if S can be obtained from P by deleting some number of transitions.

We emphasise that this differs from the definition of a contiguous subpath introduced in the previous chapter, in that we no longer require the containment to be contiguous and that we do require the two paths to have the same end vertices. We also note that if P is a loop, meaning that $s(P) = t(P)$, then one of its subpaths will be the trivial path at $s(P)$ which has no transitions.

Definition 3.1.32. A path is *simple* if it contains each of its states at most once each, and is *semi-simple* if it contains each of its states at most twice each.

Observation 3.1.33. *Any path has a simple subpath.*

Lemma 3.1.34. *Let P_1 and P_2 be simple paths with $t(P_1) = s(P_2)$. Then the path P_1P_2 obtained by concatenating P_1 and P_2 is semi-simple.*

Proof. If P_1P_2 contains a state 3 times then one of P_1 and P_2 must contain that state twice by the pigeonhole principle. $\square$

Lemma 3.1.35. *Let P be a path in an insertion transducer given by*

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_n : a_n} q_n$$

and let e be a transition $q_{i-1} \xrightarrow{a_i : a_i} q_i$ in P. Then P has a semi-simple subpath containing the transition e.

Proof. From Observation 3.1.33 we see that the path

$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_i : a_i} q_i$$

has a simple subpath P_1 containing e. Also from Observation 3.1.33 the path

$$q_i \xrightarrow{a_{i+1} : a_{i+1}} q_{i+1} \xrightarrow{a_{i+2} : a_{i+2}} \cdots \xrightarrow{a_n : a_n} q_n$$

has a simple subpath P_2. The path P_1P_2 is a subpath of P containing e, and it is semi-simple by Lemma 3.1.34. $\square$

We now present our bounded version of IPC.

Bounded Insertion Path Condition (BIPC): For each state (q,p) accessible in $\mathcal{T}^2$ such that there is a letter $b \in A(p)\backslash A(q)$, and for every semi-simple path P in $\mathcal{T}^2$ from (q_I,q_I) to (q,p), the path $\text{in}(P)$ is a b-insertion path.

We note that BIPC is decidable since a transducer has only finitely many semi-simple paths.

Theorem 3.1.36. *An insertion transducer $\mathcal{T} = (Q,A,\delta,q_I)$ satisfies IPC if and only if it satisfies BIPC.*

Proof. ($\Rightarrow$) This implication is trivial, since BIPC is just IPC applied to a specific set of paths.

($\Leftarrow$) Suppose that IPC does not hold. Then $\mathcal{T}^2$ contains a path P from (q_I,q_I) to some state (q,p) such there is a letter $b \in A(p)\backslash A(q)$, and such that $\text{in}(P)$ is not a b-insertion path. Suppose that $\text{in}(P)$ is given by

$$q_I = q_0 \xrightarrow{a_1:a_1} q_1 \xrightarrow{a_2:a_2} \cdots \xrightarrow{a_i:a_i} q_i \xrightarrow{a:a} \underbrace{s \xrightarrow{b:b} \cdots \xrightarrow{b:b} p}_{b^n:b^n}$$

such that $a \neq b$, and such that $b \notin A(r)$ for any state r on the path from s to p labelled by (b^n,b^n). By Lemma 3.1.35, the path P has a semi-simple subpath P' such that $\text{in}(P')$ contains the transition given by $q_i \xrightarrow{a:a} s$. Hence $\text{in}(P')$ is not a b-insertion path, and so $\mathcal{T}$ does not satisfy BIPC. $\square$

Corollary 3.1.37. *Let $\mathcal{T} = (Q,A,\delta,q_I)$ be an insertion transducer generating an insertion relation $\leq_{\mathcal{T}}$. Then it is decidable whether $\leq_{\mathcal{T}}$ is an ordering.*

Proof. This is seen by combining Theorems 3.1.27 and 3.1.36, together with Observation 3.1.3. $\square$

3.1.3 Left-most insertion relations

Let $\mathcal{T}$ be an insertion transducer. As is the case with many substructure relations throughout mathematics, there may be more than one way to witness a comparison $u \leq_{\mathcal{T}} v$. That is, there may be more than one path in $\mathcal{T}$ labelled by (u, v). We would like to consider insertion relations which elicit a canonical way in which to view each comparison $u \leq_{\mathcal{T}} v$, and to this end we introduce the class of *left-most* insertion relations.

Definition 3.1.38. Let $\mathcal{T}$ be an insertion transducer generating an insertion relation $\leq_{\mathcal{T}}$. The relation $\leq_{\mathcal{T}}$ is *left-most* if the following condition holds: for all $u, v \in A^*$ such that $u \leq_{\mathcal{T}} v$, the transducer $\mathcal{T}$ contains a path labelled by

$$(\varepsilon, v_0)(a_1, a_1)(\varepsilon, v_1) \dots (a_n, a_n)(\varepsilon, v_n)$$

where $u = a_1 \dots a_n$, $v = v_0 a_1 v_1 \dots a_n v_n$ and each $a_i \notin \mathrm{Alph}(v_{i-1})$. Such a path is called the *left-most path* of the comparison $u \leq_{\mathcal{T}} v$.

Intuitively, if $\leq_{\mathcal{T}}$ is left-most then $\mathcal{T}$ can output v on input u in such a way that it always copies the first successive (left-most) occurrence of each letter of u within the word v. We note that both the subword ordering and the embedding ordering of Aichinger et al. are left-most.

Notation 3.1.39. If an insertion transducer contains a path starting at a state q labelled by (u, v) then we write $u \preccurlyeq_q v$.

Lemma 3.1.40. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer generating a left-most insertion relation $\leq_{\mathcal{T}}$. Let $u, v, w \in A^*$ be such that $uv \leq_{\mathcal{T}} uw$, and let $q \in Q$ be such that $u \in W(q)$. Then $v \preccurlyeq_q w$.*

Proof. Write $u = a_1 \dots a_n$ and $v = b_1 \dots b_m$. Suppose the left-most path of the comparison $uv \leq_{\mathcal{T}} uw$ is labelled by

$$(\varepsilon, w_0)(a_1, a_1)(\varepsilon, w_1) \dots (a_n, a_n)(\varepsilon, w_n)(b_1, b_1)(\varepsilon, w_{n+1}) \dots (b_m, b_m)(\varepsilon, w_{n+m}),$$

so that

$$uw = w_0 a_1 w_1 \dots a_n w_n b_1 w_{n+1} \dots b_m w_{n+m}.$$

We have $a_1 \notin \mathrm{Alph}(w_0)$, but the first letter of uw is a_1, so $w_0 = \varepsilon$. Similarly we have $a_2 \notin \mathrm{Alph}(w_1)$, but the second letter of uw is a_2, so $w_1 = \varepsilon$. Continuing in this fashion we see that $w_i = \varepsilon$ for all $i \in \{0, \dots, n-1\}$, and so we can write

$$uw = a_1 \dots a_n w_n b_1 w_{n+1} \dots b_m w_{n+m}$$

where each $w_{n+j} \in A(q \cdot b_1 \dots b_j)^*$. Hence $\mathcal{T}$ contains a path starting at q labelled by

$$(\varepsilon, w_n)(b_1, b_1)(\varepsilon, w_{n+1}) \dots (b_m, b_m)(\varepsilon, w_{n+m}).$$

By concatenating these labels we see that this path is labelled by (v, w), so $v \preccurlyeq_q w$. □

Lemma 3.1.41. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer, let $q \in Q$ and let $v, w \in A^*$ be such that $v \preccurlyeq_q w$. Suppose there is a loop at q labelled by (z, z) for some $z \in A^*$. Then $zv \preccurlyeq_q zw$.*

Proof. Write $v = b_1 \dots b_m$ and suppose $\mathcal{T}$ has a path starting at q labelled by

$$(\varepsilon, w_0)(b_1, b_1)(\varepsilon, w_1) \dots (b_m, b_m)(\varepsilon, w_m)$$

such that $w = w_0 b_1 w_1 \dots b_m w_m$. By concatenating the loop at q labelled by (z, z) together with this path, we see that $\mathcal{T}$ contains a path starting at q labelled by

$$(z, z)(\varepsilon, w_0)(b_1, b_1)(\varepsilon, w_1) \dots (b_m, b_m)(\varepsilon, w_m).$$

By concatenating these labels we see that this path is labelled by (zv, zw), so we have $zv \preccurlyeq_q zw$. □

Lemma 3.1.42. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer generating an insertion relation $\leq_{\mathcal{T}}$. Let $u, v, w \in A^*$ and $q \in Q$ be such that $u \in W(q)$ and $v \preccurlyeq_q w$. Then $uv \leq_{\mathcal{T}} uw$.*

Proof. Since $u \in W(q)$, there is a path in $\mathcal{T}$ from q_I to q labelled by (u, u). Since

$v \preccurlyeq_q w$, there is a path in $\mathcal{T}$ starting at q labelled by (v, w). By concatenating these paths, we see that there is a path in $\mathcal{T}$ starting at q_I labelled by (uv, uw), so $uv \leq_{\mathcal{T}} uw$. $\square$

We now give a result characterising the insertion transducers generating left-most insertion relations.

Theorem 3.1.43. *Let $\leq_{\mathcal{T}}$ be an insertion relation. Then $\leq_{\mathcal{T}}$ is left-most if and only if for each $q \in Q$ and for each $a \in A(q)$ we have $A(q) \subseteq A(q \cdot a)$.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose that there is a state $q \in Q$ and letters $a, b \in A(q)$ such that $b \notin A(q \cdot a)$. Let $w \in W(q)$, so that $wa \in W(q \cdot a)$. Then $\mathcal{T}$ contains a path starting at q_I labelled by

$$(w, w)(\varepsilon, ab)(a, a)$$

and so $wa \leq_{\mathcal{T}} waba$. But $\varepsilon \not\preccurlyeq_{q \cdot a} ba$ since $b \notin A(q \cdot a)$, and so by Lemma 3.1.40 we see that $\leq_{\mathcal{T}}$ is not left-most.

($\Leftarrow$) Assume the stated condition holds and let $u, v \in A^*$ be such that $u \leq_{\mathcal{T}} v$. Write $u = a_1 \dots a_n$ and $v = v_0 a_1 v_1 \dots a_n v_n$ where each $v_i \in A(a_1 \dots a_i)^*$. Let i be minimal such that $a_i \in \mathrm{Alph}(v_{i-1})$, write $v_{i-1} = v'_{i-1} a_i x$ where $a_i \notin \mathrm{Alph}(x)$, and let $v'_i = x a_i v_i$. Since $a_i \in \mathrm{Alph}(v_{i-1})$ we have $a_i \in A(a_1 \dots a_{i-1})$, and so by our assumption we have $A(a_1 \dots a_{i-1}) \subseteq A(a_1 \dots a_i)$. Hence $x a_i \in A(a_1 \dots a_i)^*$, and so $v'_i \in A(a_1 \dots a_i)^*$. Therefore $\mathcal{T}$ contains a path starting at q_I which is labelled by

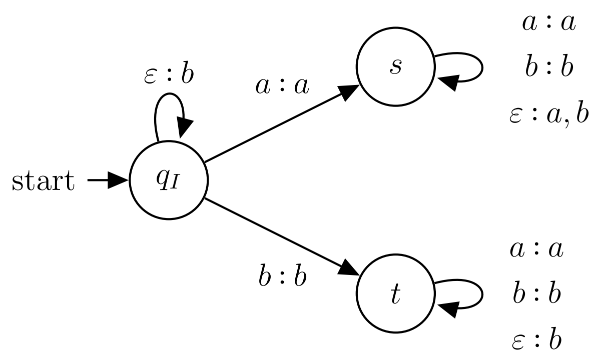
$$(\varepsilon, v_0)(a_1, a_1)(\varepsilon, v_1) \dots (a_{i-1}, a_{i-1})(\varepsilon, v'_{i-1})(a_i, a_i)(\varepsilon, v'_i) \dots (a_n, a_n)(\varepsilon, v_n).$$

By concatenating these labels we see that this path is again labelled by (u, v). Repeating this process for each successive offending letter a_j will yield a left-most path for the comparison $u \leq_{\mathcal{T}} v$, and so the relation $\leq_{\mathcal{T}}$ is left-most. $\square$

Since an insertion transducer has only finitely many states q and each alphabet $A(q)$ is finite, we can immediately deduce the following:

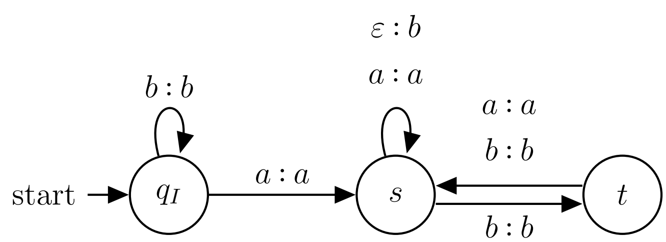
Corollary 3.1.44. *It is decidable whether a given insertion relation is left-most.*

Example 3.1.45. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.12. This was given by:



We will show that for $q \in Q$ and for each $c \in A(q)$ we have $A(q) \subseteq A(q \cdot c)$. It will then follow from Theorem 3.1.43 that $\leq_{\mathcal{T}}$ is left-most. We have $A(q_I) = \{b\}$, $q_I \cdot b = t$ and $A(t) = \{b\}$, so $A(q_I) \subseteq A(t)$. For $q \in \{s, t\}$ and $c \in A(q)$ we have $q \cdot c = q$, so $A(q) \subseteq A(q \cdot c)$. Hence the stated property is satisfied and so $\leq_{\mathcal{T}}$ is left-most.

Example 3.1.46. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.11. This was given by:



We show that the insertion relation $\leq_{\mathcal{T}}$ is not left-most. We have $b \in A(s)$ and $s \cdot b = t$, but $b \notin A(t)$ so $A(s) \nsubseteq A(t)$. Hence $\leq_{\mathcal{T}}$ is not left-most by Theorem 3.1.43.

Our next result gives a simple transitivity classification for the class of left-most insertion relations:

Theorem 3.1.47. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer generating a left-most insertion relation $\leq_{\mathcal{T}}$. Then $\leq_{\mathcal{T}}$ is transitive if and only if for each state (q, p) accessible in $\mathcal{T}^2$ we have $A(p) \subseteq A(q)$.*

Proof. (⇒) Assume that $\leq_{\mathcal{T}}$ is transitive, and that there is a state (q,p) which is accessible in $\mathcal{T}^2$ such that there is a letter $b \in A(p)\backslash A(q)$. Let P be a path in $\mathcal{T}^2$ from (q_I, q_I) to (q,p) and suppose that $\mathrm{in}(P)$ is given by

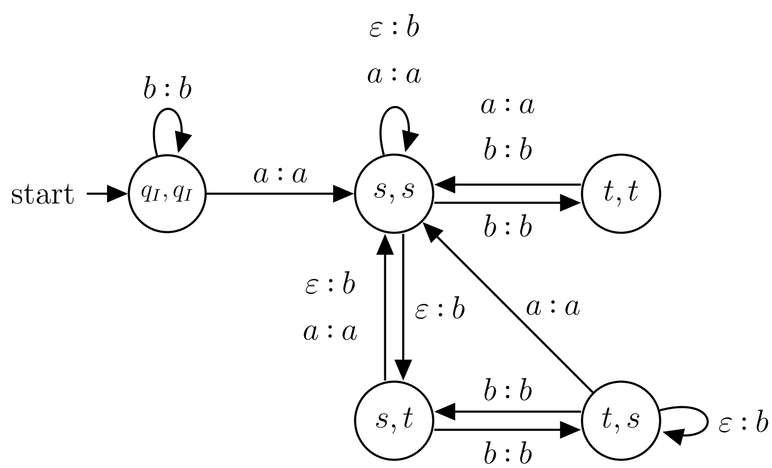
$$q_I = q_0 \xrightarrow{a_1 : a_1} q_1 \xrightarrow{a_2 : a_2} \cdots \xrightarrow{a_n : a_n} q_n = q.$$

By Theorem 3.1.27, the path $\mathrm{in}(P)$ is a b-insertion path. Let i be its b-insertion length, so that $b \in A(q_i)$ and $a_{i+1} = a_{i+2} = \cdots = a_n = b$. Then $u \leq_{\mathcal{T}} ub$, and since $\leq_{\mathcal{T}}$ is left-most we have $\varepsilon \preccurlyeq_q b$ by Lemma 3.1.40. This means that $b \in A(q)$, contradicting our assumption.

(⇐) This follows from Corollary 3.1.29. □

We note that the condition stated in Theorem 3.1.47 is decidable since $\mathcal{T}^2$ contains only finitely many states.

Example 3.1.48. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.12. It was shown in Example 3.1.45 that $\leq_{\mathcal{T}}$ is left-most. The transducer $\mathcal{T}^2$ was given by:



We show that for each state (q,p) accessible in $\mathcal{T}^2$, we have $A(p) \subseteq A(q)$. Clearly this holds for the states (q_I, q_I), (s,s) and (t,t). The other states we need to consider are (q_I, t) and (s,t). For the state (q_I, t) we have $A(t) = \{b\} = A(q_I)$, and for the state (s,t) we have $A(t) = \{b\} \subseteq \{a,b\} = A(q_I)$. Hence $\leq_{\mathcal{T}}$ is transitive by Theorem 3.1.47.

Deciding WQO for left-most insertion orderings

Our goal is now to show that it is decidable whether a given left-most insertion ordering is a WQO. We first present the following theorem. The reverse direction of its proof is a variation of the 'minimal bad sequence' argument introduced by Nash-Williams [28].

Theorem 3.1.49. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer generating a left-most insertion ordering $\leq_{\mathcal{T}}$. Then $\leq_{\mathcal{T}}$ is a WQO if and only if for each $q \in Q$ and for each loop at q labelled by (w, w) for some $w \in A^*$, we have $w \in A(q)^*$.*

Proof. ($\Rightarrow$) Suppose that there is a state $q \in Q$ with a loop at q labelled by (w, w) for some $w \in A^*$, and that $w \notin A(q)^*$. Let $v \in W(q)$. In order to show that $\leq_{\mathcal{T}}$ is not a WQO, we show that the language $L = \{vw^n \mid n \geq 1\}$ is an anti-chain. Let $x, y \in L$ be such that $x <_{\mathcal{T}} y$, and write $x = vw^i$ and $y = vw^j$. Since the relation $\leq_{\mathcal{T}}$ is left-most, we have $\varepsilon \preccurlyeq_q w^{j-i}$ by Lemma 3.1.40, but $w \notin A(q)^*$ so this is a contradiction.

($\Leftarrow$) Suppose that the stated condition holds, and that $\leq_{\mathcal{T}}$ is not a WQO. By Proposition 1.1.3, there are sequences from A^* which are bad under $\leq_{\mathcal{T}}$. Let u_1 be a word such that there is a bad sequence beginning with u_1 but no bad sequence beginning with u_1' for any word u_1' with $u_1' <_{\mathcal{T}} u_1$. Next, let u_2 be a word such that there is a bad sequence beginning with u_1, u_2 but no bad sequence beginning with u_1, u_2' for any word u_2' with $u_2' <_{\mathcal{T}} u_2$. Continuing in this fashion *ad infinitum* we obtain a bad sequence S given by

$$u_1, u_2, \ldots$$

which is *minimal* in the sense that, for each $i \geq 1$, there is no bad sequence starting with $u_1, u_2, \ldots, u_{i-1}, u_i'$ for any word u_i' with $u_i' <_{\mathcal{T}} u_i$. Next suppose that $\mathcal{T}$ has n states. Then there is a word x of length $n + 1$ such that the sequence S has a subsequence

$$u_{i_1}, u_{i_2}, \ldots$$

where each word starts with x. The transducer $\mathcal{T}$ only has n states, so the path in $\mathcal{T}$ starting at q_I labelled by (x, x) must visit some state q twice. Hence we can write $x = ywz$ where $y \in W(q)$ and where $\mathcal{T}$ has a loop at q labelled by (w, w). By our assumption, we then have $w \in A(q)^*$. For $k \geq 1$ write $u_{i_k} = ywz_k$ and let $u'_{i_k} = yz_k$. Since $w \in A(q)^*$ and $w \neq \varepsilon$, we have $u'_{i_k} <_{\mathcal{T}} u_{i_k}$ for each $k \geq 1$. Our aim is to show that the sequence S' given by

$$u_1, u_2, \ldots, u_{i_1 - 1}, u'_{i_1}, u'_{i_2}, \ldots$$

is bad. This will contradict the minimality of the sequence S, since $u'_{i_1} <_{\mathcal{T}} u_{i_1}$. Since S is a bad sequence we have $u_i \nleq_{\mathcal{T}} u_j$ for $i < j < i_1$. We cannot have $u_i \leq_{\mathcal{T}} u'_{i_k}$ for some $i < i_1$ and $k \geq 1$, as otherwise we would have $u_i \leq_{\mathcal{T}} u_{i_k}$ by transitivity. Now suppose that $u'_{i_k} \leq_{\mathcal{T}} u'_{i_l}$ for some $k \leq l$, meaning that $yz_k \leq_{\mathcal{T}} yz_l$. The ordering $\leq_{\mathcal{T}}$ is left-most, so we have $z_k \preccurlyeq_q z_l$ by Lemma 3.1.40. The transducer $\mathcal{T}$ has a loop at q labelled by (w, w), and so $wz_k \preccurlyeq_q wz_l$ by Lemma 3.1.41. Hence $ywz_k \leq_{\mathcal{T}} ywz_l$ by Lemma 3.1.42. This means that $u_{i_k} \leq_{\mathcal{T}} u_{i_l}$, and so $k = l$. Hence the sequence S' is bad, giving us the desired contradiction. □

The next proposition shows that the condition in Theorem 3.1.49 is decidable.

Proposition 3.1.50. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer. The following are equivalent:*

(i) *For each $q \in Q$ and for each loop at q labelled by (w, w) for some $w \in A^*$, we have $w \in A(q)^*$.*

(ii) *For each copy transition $s \xrightarrow{a\,:\,a} t$ in $\mathcal{T}$, and for each state $q \in Q$ such that there is a path from q to s and a path from t to q, we have $a \in A(q)$.*

Proof. (i) ⇒ (ii): Let $s \xrightarrow{a\,:\,a} t$ be a copy transition in $\mathcal{T}$ and let $q \in Q$ be such that there is a path from q to s labelled by (u, u) and a path from t to q labelled by (v, v). Then there is a loop at q labelled by (uav, uav), and so $uav \in A(q)^*$ by our assumption. In particular we have $a \in A(q)$.

(ii) ⇒ (i): Let $q \in Q$ and suppose there is a loop P at q labelled by (w, w) for some $w \in A^*$. Write $w = a_1 \ldots a_n$ and suppose that P is given by

$$q = q_0 \xrightarrow{a_1\,:\,a_1} q_1 \xrightarrow{a_2\,:\,a_2} \cdots \xrightarrow{a_n\,:\,a_n} q_n = q. \qquad (\ddagger)$$

In order to show that $w \in A(q)^*$, we show that each letter a_i belongs to $A(q)$. For each $i \geq 1$, transducer $\mathcal{T}$ contains the transition $q_{i-1} \xrightarrow{a_i : a_i} q_i$, and by (‡) there is a path from q to q_{i-1} and a path from q_i to q. Hence by our assumption we have $a_i \in A(q)$. □

Condition (ii) of Proposition 3.1.50 is decidable since a transducer contains only finitely many transitions, so we have the following corollary:

Corollary 3.1.51. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer generating a left-most insertion relation $\leq_{\mathcal{T}}$. Then it is decidable whether $\leq_{\mathcal{T}}$ is a WQO.*

Proof. This can be seen by combining Theorem 3.1.49 and Proposition 3.1.50. □

The following result provides an interesting class of insertion relations which are WQOs. Both the subword ordering and the embedding ordering belong to this class.

Proposition 3.1.52. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer such that the following two conditions hold:*

(i) *Every loop in $\mathcal{T}$ contains only one state.*

(ii) *For each $q \in Q$ we have $A(q) = \{a \in A \mid q \cdot a = q\}$.*

Then the insertion relation $\leq_{\mathcal{T}}$ is a left-most WQO.

Proof. We first show that $\leq_{\mathcal{T}}$ is left-most. Let $q \in Q$ and let $a \in A(q)$. Then by (ii) we have $q \cdot a = q$, so in particular we have $A(q) \subseteq A(q \cdot a)$. Hence $\leq_{\mathcal{T}}$ is left-most by Theorem 3.1.43. Next we show that $\leq_{\mathcal{T}}$ is transitive. Let (q, p) be an accessible state in $\mathcal{T}^2$. We claim that $q = p$. Suppose instead that $q \neq p$, and let P be a path in $\mathcal{T}^2$ from (q_I, q_I) to (q, p). Then P must include a type (ii) transition $(s, s) \xrightarrow{\varepsilon : a} (s, t)$ where $a \in A(s)$, $s \cdot a = t$ and $s \neq t$. But $a \in A(s)$ implies that $s \cdot a = s$ by (ii). Hence we have $q = p$, and so $A(p) \subseteq A(q)$. Hence $\leq_{\mathcal{T}}$ is transitive by Theorem 3.1.47. Finally we show that $\leq_{\mathcal{T}}$ is a WQO. Let $q \in Q$ and let $s \xrightarrow{a : a} t$ be a transition in $\mathcal{T}$ such that there is a path from q to s and a path from t to q. Then $\mathcal{T}$ has a loop containing the states q, s and t, so by (i) we have $q = s = t$. Hence $q \cdot a = q$, and so $a \in A(q)$. Hence $\leq_{\mathcal{T}}$ is a well-quasi-ordering by Theorem 3.1.49. □

Finally we demonstrate that there exist non-trivial left-most insertion orderings which are not WQOs. Indeed, consider the insertion transducer from Example 3.1.12. It was shown in Example 3.1.48 that the insertion relation $\leq_{\mathcal{T}}$ is an ordering. The transducer $\mathcal{T}$ has a loop at t labelled by (a, a) but $a \notin A(t)^*$, so $\leq_{\mathcal{T}}$ is not WQO by Theorem 3.1.49.

Concluding remarks and further questions

We leave the following as open questions:

Question 1: Is it decidable whether a given insertion transducer generates a WQO?

Question 2: Let $\mathcal{T}$ be an insertion transducer, let $L \subseteq A^*$ be a regular language and let $\mathrm{Av}_{\mathcal{T}}(L)$ denote the avoidance set of L under $\leq_{\mathcal{T}}$. Is it decidable whether $\mathrm{Av}_{\mathcal{T}}(L)$ is WQO under $\leq_{\mathcal{T}}$?

In addressing Question 2 we can note the following result:

Proposition 3.1.53. *Let $\mathcal{T} = (Q, A, \delta, q_0)$ be an insertion transducer, let $L \subseteq A^*$ be a regular language and let $\mathcal{A} = (P, A, \sigma, p_0, F)$ be a DFA accepting L. Then the upward closure $\mathrm{up}_{\mathcal{T}}(L)$ of L under the insertion relation $\leq_{\mathcal{T}}$ is a regular language, accepted by the NFA*

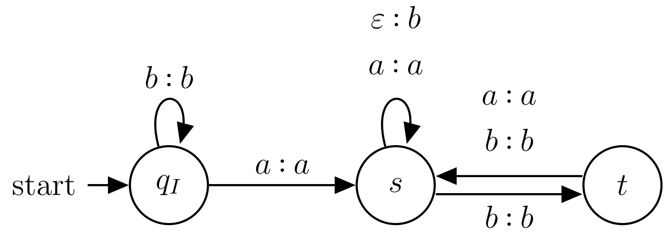
$$\mathcal{B} = (Q \times P, A, \rho, (q_0, p_0), Q \times F)$$

where the partial function $\rho : (Q \times P) \times A \to \mathcal{P}(Q \times P)$ is given by

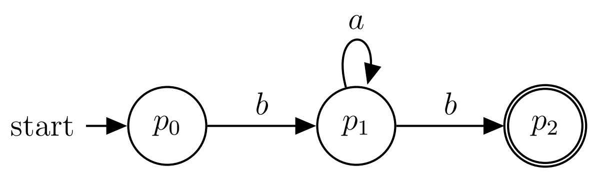
$$\rho((q,p),a) = \begin{cases} \{(q \cdot a, \delta(p,a)), (q,p)\} & \text{if } a \in A(q) \\ \{(q \cdot a, \delta(p,a))\} & \text{if } a \notin A(q). \end{cases}$$

Proof. A word v is accepted by $\mathcal{B}$ if and only if there is a path from (q_0, p_0) to a state $(q, p) \in Q \times F$. This is the case if and only if we can write $v = v_0 a_1 v_1 \dots a_n v_n$ where $a_1 \dots a_n \in L$ and where each $v_i \in A(a_1 \dots a_i)^*$, which is the case if and only if $v \in \mathrm{up}_{\mathcal{T}}(L)$. ☐

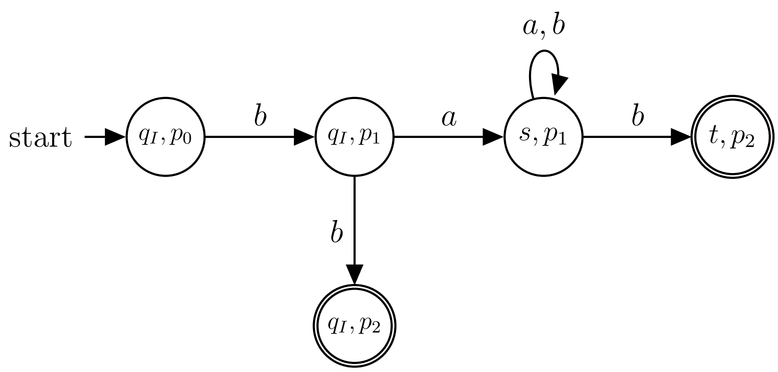
Example 3.1.54. Let $\mathcal{T}$ be the insertion transducer from Example 3.1.11. This transducer is shown below.



Let $\mathcal{A}$ be the DFA shown below and let L be the language accepted by $\mathcal{A}$. It is given by the regular expression ba^*b.



The NFA $\mathcal{B}$ accepting the upward closure of L under $\leq_{\mathcal{T}}$, whose construction is described in Proposition 3.1.53, is shown below. We note, for instance, that $\mathcal{B}$ accepts the word *babb* while $\mathcal{A}$ does not.



Corollary 3.1.55. *Let $\mathcal{T}$ be an insertion transducer generating an insertion ordering $\leq_{\mathcal{T}}$ and let $L \subseteq A^*$ be a regular language. Then the set $\mathrm{Av}_{\mathcal{T}}(L)$ is a regular language.*

Proof. This comes from Proposition 3.1.53 and the fact that $\mathrm{Av}_{\mathcal{T}}(L) = A^* \setminus \mathrm{up}_{\mathcal{T}}(L)$, together with the fact that $\mathrm{Reg}(A)$ is closed under the operation of set difference. $\square$

Hence Question 2 can be solved if we can find a positive answer to following question:

Question 3: Let $\mathcal{T}$ be an insertion transducer generating an insertion ordering $\leq_{\mathcal{T}}$. Is it decidable whether a given regular language is WQO under $\leq_{\mathcal{T}}$?

3.2 Word-insertion relations

In this section we introduce a class of rational relations which generalise insertion relations, called *word-insertion relations*. These relations will be generated by transducers where at each state there is a set of words which may be inserted, rather than a set of letters. Our current definition of a transducer does not allow for transitions to be labelled by words, so we introduce a more general notion of a transducer.

Definition 3.2.1. A *word-labelled transducer* is a 4-tuple $\mathcal{T} = (Q, A, \delta, q_0)$ where:

- Q is a finite set of *states*;
- A is a finite *alphabet*;
- $\delta : Q \times A_\varepsilon \to \mathcal{P}(Q \times A^*)$ is a partial function called the *transition function*;
- $q_I \in Q$ is the *start state*.

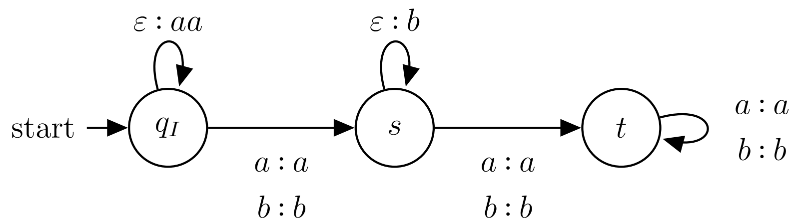
The class of word-labelled transducers generating our new class of relations is then defined as follows:

Definition 3.2.2. An *word-insertion transducer* is a word-labelled transducer $\mathcal{T}$ whose transitions are exactly as follows:

- For each state $q \in Q$ and for each letter $a \in A$, there is exactly one state $p \in Q$ such that $q \xrightarrow{a:a} p$ is a transition in $\mathcal{T}$.
- For each state $q \in Q$ there is a set of words $S(q) \subseteq A^*$ such that $q \xrightarrow{\varepsilon:w} q$ is a transition in $\mathcal{T}$ for each $w \in S(q)$.

Definition 3.2.3. Let $\mathcal{T}$ be a word-insertion transducer. The relation generated by $\mathcal{T}$ is denoted by $\leq_{\mathcal{T}}$ and is called a *word-insertion relation*.

Example 3.2.4. Below is an example of a word-insertion transducer $\mathcal{T}$. We have $S(q_I) = \{aa\}$, $S(s) = \{b\}$ and $S(t) = \varnothing$. As an instance of the word-insertion relation $\leq_{\mathcal{T}}$ generated by $\mathcal{T}$, we have $ba \leq_{\mathcal{T}} aabba$.



Transitivity of word-insertion relations

As with insertion relations, a word-insertion relation $\leq_{\mathcal{T}}$ is an ordering if and only if it is transitive. In order to investigate whether $\leq_{\mathcal{T}}$ is transitive, we would like to find a transducer generating the relation $\leq_{\mathcal{T}}^2$. The relation $\leq_{\mathcal{T}}$ will then be transitive if and only if this new transducer is equivalent to $\mathcal{T}$. In order to capture $\leq_{\mathcal{T}}^2$ we will first need to introduce another more general notion of a transducer, which we define below.

Definition 3.2.5. An *accept state transducer* (AST) is a 5-tuple $\mathcal{T} = (Q, A, \delta, F, q_I)$ where:

- Q is a finite set of *states*;
- A is a finite *alphabet*;
- $\delta : Q \times A_\varepsilon \to \mathcal{P}(Q \times A_\varepsilon)$ is a partial function called the *transition function*;
- $F \subseteq Q$ is the set of *accept states*;
- $q_I \in Q$ is the *start state*.

If $\mathcal{T}$ is an AST then the relation $R_{\mathcal{T}}$ generated by $\mathcal{T}$ is defined by $(u, v) \in R_{\mathcal{T}}$ if and only if there is a path from (q_I, q_I) to an accept state labelled by (u, v).

Now let $\mathcal{T}$ be a word-insertion transducer. In order to build a transducer generating $\leq_{\mathcal{T}}^2$ we will first convert $\mathcal{T}$ into an equivalent AST, denoted by $\overline{\mathcal{T}}$. We will construct $\overline{\mathcal{T}}$

from $\mathcal{T}$ by introducing a set of 'dummy' states corresponding to each state q, which, intuitively, allow us to insert words from $S(q)$ one letter at a time. The original states are then set to be accept states, and the new dummy states as non-accept states. We make this definition formal below.

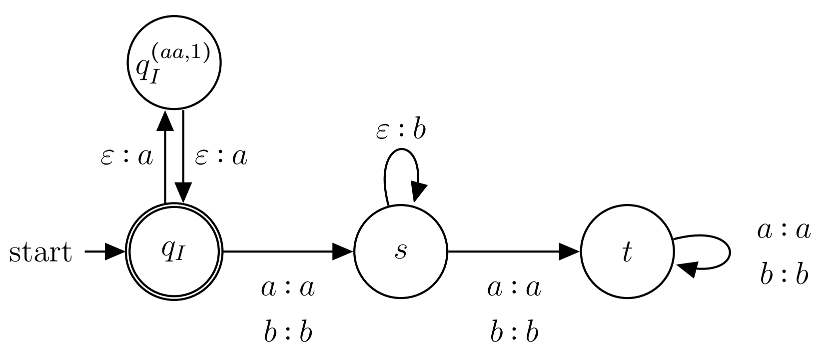
Definition 3.2.6. Let $\mathcal{T} = (Q, A, \delta, q_0)$ be a word-insertion transducer. The *AST corresponding to* $\mathcal{T}$ is denoted by $\overline{\mathcal{T}}$ and defined as follows. For each $q \in Q$ and each $w \in S(q)$ we write $w = a_1 \dots a_n$ and introduce n states $q^{(w,1)}, \dots, q^{(w,n)}$. By convention we will also refer to the state $q^{(w,n)}$ as $q^{(w,0)}$ or simply q. We then let $P_{q,w} = \{q^{(w,1)}, \dots, q^{(w,n)}\}$ and let

$$P = \bigcup_{q \in Q, w \in S(q)} P_{q,w}.$$

We then set $\overline{T} = (P, A, \sigma, Q, q_0)$ where the transitions are exactly as follows:

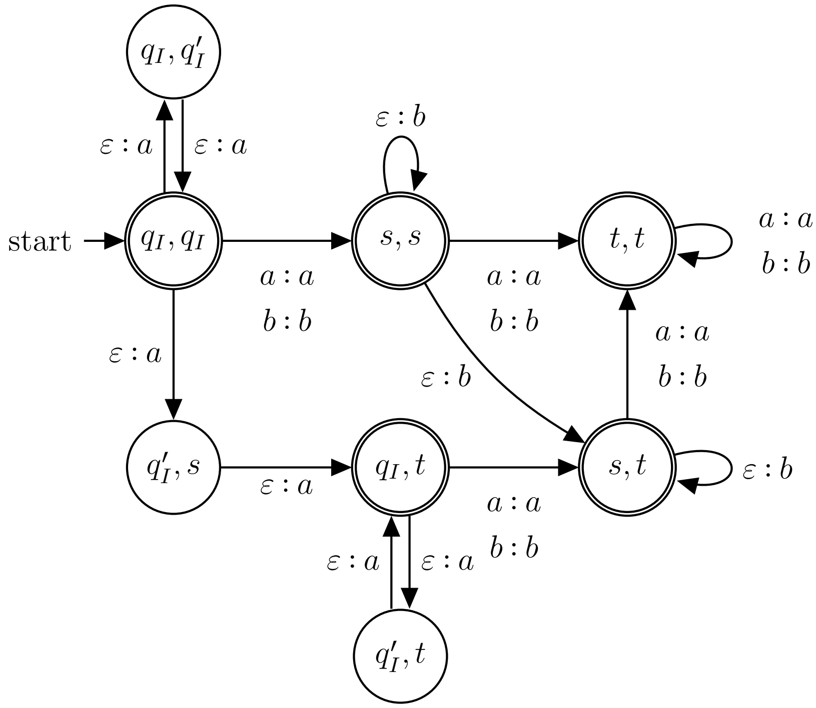
- $q \xrightarrow{a:a} p$ for each copy transition $q \xrightarrow{a:a} p$ in $\mathcal{T}$;
- $q^{(w,i-1)} \xrightarrow{\varepsilon : a_i} q^{(w,i)}$ for each $q \in Q$, $w = a_1 \dots a_n \in S(q)$ and $i \in \{1, \dots, n\}$.

Example 3.2.7. Let $\mathcal{T}$ be the word-insertion transducer from Example 3.2.4. Then $\overline{T}$ is given by:

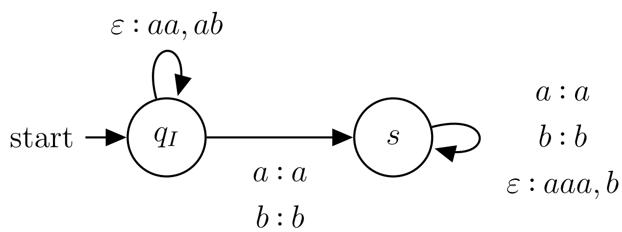


We can construct $(\overline{\mathcal{T}})^2$ from $\overline{\mathcal{T}}$ in the usual way, with the convention that (q, p) is an accept state if and only if both q and p are accept states.

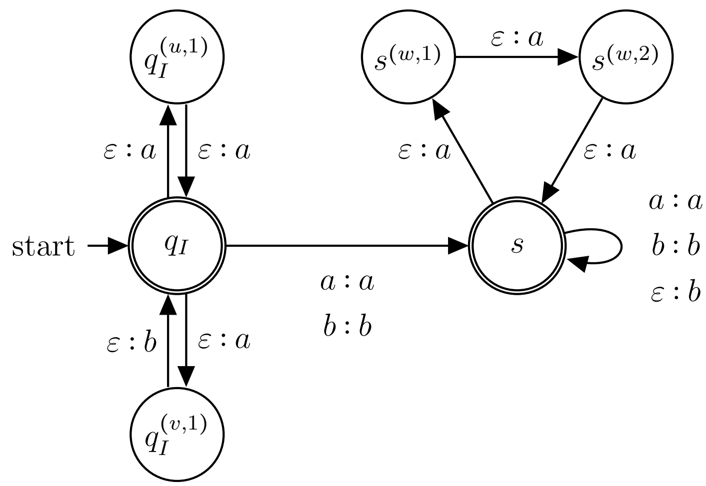
Example 3.2.8. Let $\mathcal{T}$ be the transducer from Example 3.2.4, and for convenience write q_I' in place of $q_I^{(aa,1)}$. Then $\overline{\mathcal{T}}^2$ is given by:



Example 3.2.9. Let $\mathcal{T}$ be the word-insertion transducer shown below.



For convenience we write $u = aa$, $v = ab$ and $w = aaa$. The transducer $\overline{\mathcal{T}}$ is given by:

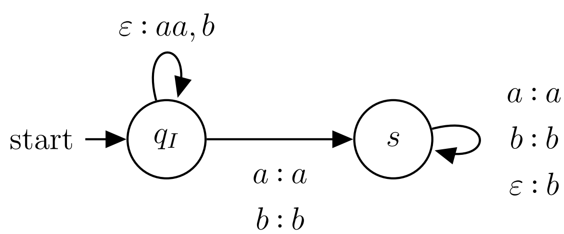


We recall the following result concerning insertion transducers:

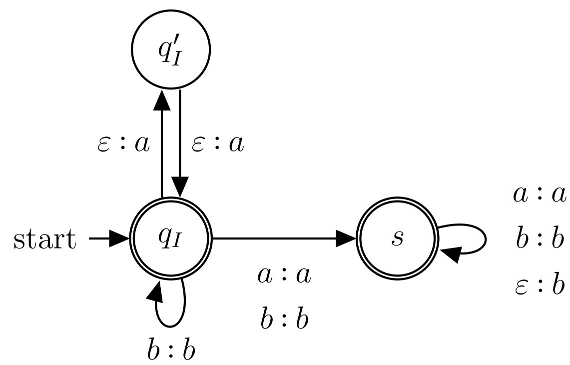
Corollary 3.1.29. *Let $\mathcal{T} = (Q, A, \delta, q_I)$ be an insertion transducer. If for every state (q, p) accessible in $\mathcal{T}^2$ we have $A(p) \subseteq A(q)$, then the insertion relation $\leq_{\mathcal{T}}$ is transitive.*

The next example shows that the equivalent statement for word-insertion transducers, with $S(q)$ and $S(p)$ in place of $A(q)$ and $A(p)$ respectively, does not hold in general.

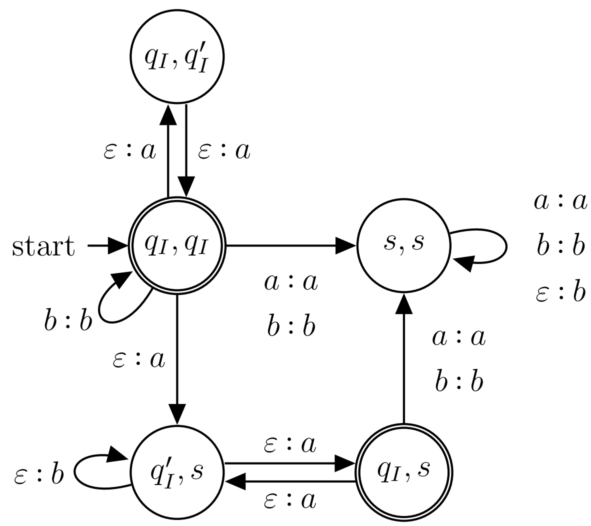
Example 3.2.10. Let $\mathcal{T}$ be the word-insertion transducer shown below. We have $S(q_I) = \{aa, b\}$ and $S(s) = \{b\}$.



The transducer $\overline{\mathcal{T}}$ is show below. For convenience we write q_I' in place of $q_I^{(aa,1)}$.



The transducer $(\overline{\mathcal{T}})^2$ is then given by:



For each state $(q^{(u,n)}, p^{(v,m)})$ accessible in $(\overline{\mathcal{T}})^2$, we have $S(p) \subseteq S(q)$. However, the relation $\leq_{\mathcal{T}}$ is not transitive, as we have $\varepsilon \leq_{\mathcal{T}} aa \leq_{\mathcal{T}} aba$, but $\varepsilon \nleq_{\mathcal{T}} aba$ since $aba \notin S(0)^*$.

A sufficient condition for transitivity

Our goal is now to develop a sufficient condition for a given word-insertion relation to be transitive.

Definition 3.2.11. Let $u, \alpha, \beta \in A^*$. We say that the ordered pair (α, β) is a *parsing* of the word u if $u = \alpha\beta$.

Example 3.2.12. Some parsings of $aabb$ are $(aa, bb), (a, abb)$ and $(\varepsilon, aabb)$. Some parsings of abc are $(a, bc), (ab, c)$ and (abc, ε).

Definition 3.2.13. Let $L, K \subseteq A^*$ be languages. We say L is *closed under insertion from* K if for all words $u \in L$, $v \in K$ and for all parsings (α, β) of u we have $\alpha v \beta \in L$.

We then have:

Theorem 3.2.14. *Let $\mathcal{T}$ be a word-insertion transducer. Suppose that for each state $(q^{(w,n)}, p)$ accessible in $(\overline{\mathcal{T}})^2$, the language $S(q)^*$ is closed under insertion from $S(p)$. Then the word-insertion relation $\leq_{\mathcal{T}}$ is transitive.*

Proof. Let $u, v, w \in A^*$ be such that $u \leq_{\mathcal{T}} v \leq_{\mathcal{T}} w$. Write $u = a_1 \dots a_n$ and for $i \in \{0, 1, \dots, n\}$ let q_i be the state which $\mathcal{T}$ enters after reading $a_1 \dots a_i$. Then write $v = v_0 a_1 v_1 \dots a_n v_n$ where each $v_i \in S(q_i)^*$. For each $i \in \{0, \dots, n\}$ such that $v_i \neq \varepsilon$ write

$$v_i = x_{i0} x_{i1} \dots x_{im_i}$$

where each $x_{ij} \in S(q_i)$. Then for each $i \in \{0, 1, \dots, n\}$ and $j \in \{0, \dots, m_i\}$ write

$$x_{ij} = b_{ij1} b_{ij2} \dots b_{ijr_{ij}}$$

where each $b_{ijk} \in A$. Finally for each $i \in \{0, \dots, n\}$, $j \in \{0, \dots m_i\}$ and $k \in \{1, \dots, r_{ij}\}$ let

$$y_{ijk} = v_0 a_1 v_1 \dots a_i x_{i1} x_{i2} \dots x_{i(j-1)} b_{ij1} b_{ij2} \dots b_{ijk}$$

and let p_{ijk} be the state with $\mathcal{T}$ enters after reading y_{ijk}. Then $(\overline{\mathcal{T}})^2$ has a path from (q_0, q_0) to $({q_i}^{(x_{ij}, k)}, p_{ijk})$ labelled by $(a_1 \dots a_i, y_{ijk})$, and so by our assumption the set $S(q_i)^*$ is closed under insertion from $S(p_{ijk})$. Now write

$$w = \gamma_0 a_1 \gamma_1 \dots a_n \gamma_n$$

where each γ_i is

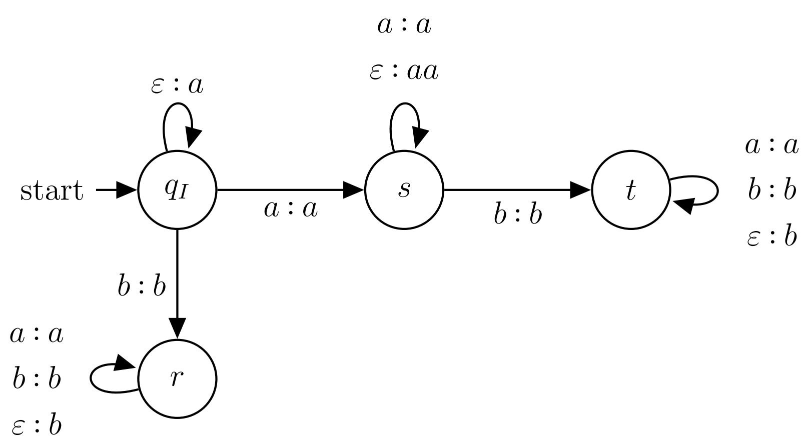
$$\gamma_i = \delta_{i0} \delta_{i1} \dots \delta_{im_1},$$

where each δ_{ij} is

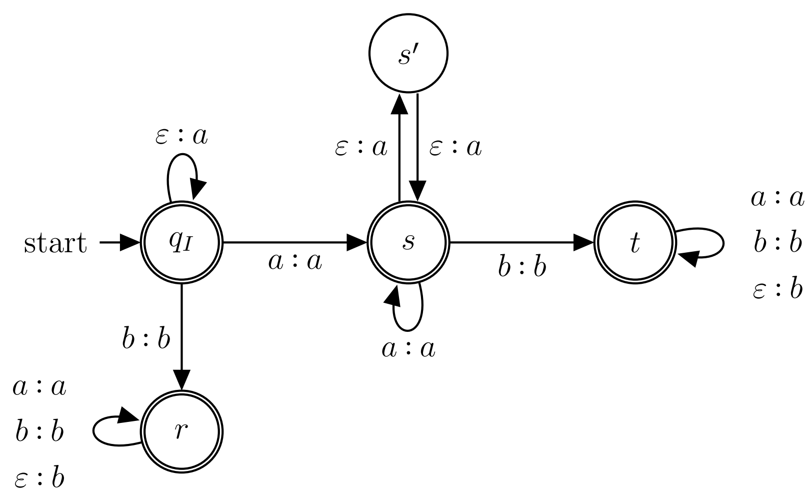
$$\delta_{ij} = z_{ij0} b_{ij1} z_{ij1} \dots b_{ijr_{ij}} z_{ijr_{ij}}$$

and where each $z_{ijk} \in S(p_{ijk})^*$. Then for each $i \in \{0, \ldots, n\}$, $j \in \{0, \ldots, m_i\}$ and $k \in \{1, \ldots, r_{ij}\}$ the set $S(q_i)^*$ is closed under insertion from $S(p_{ijk})$ and so $\delta_{ij} \in S(q_i)^*$. Hence for each i we have $\gamma_i \in S(q_i)^*$, and so $u \leq_{\mathcal{T}} w$. $\square$

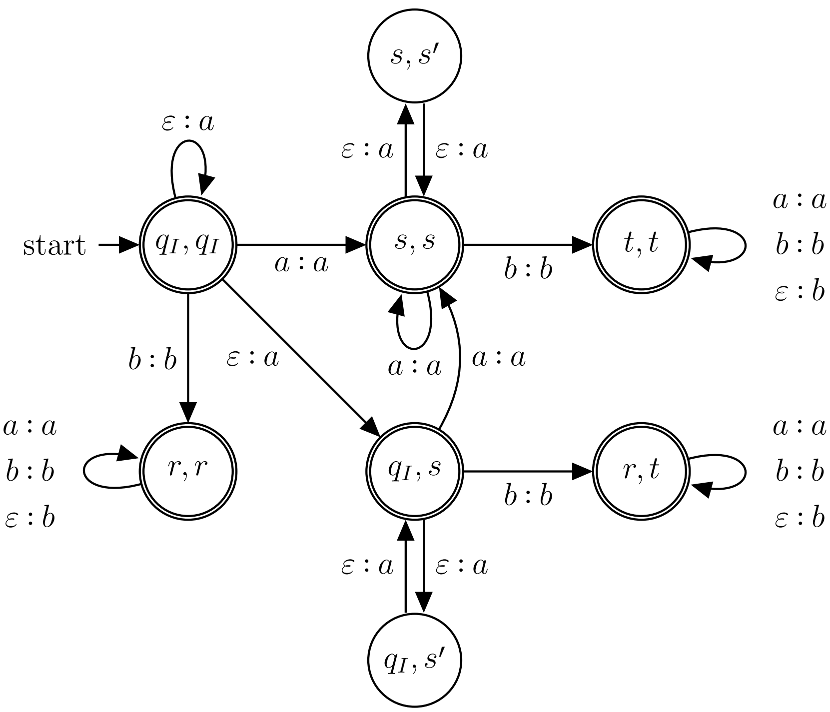
Example 3.2.15. Let $\mathcal{T}$ be the word-insertion transducer shown below.



The transducer $\overline{\mathcal{T}}$ is shown below. For convenience we have written s' in place of $s^{(aa,1)}$.



The transducer $(\overline{\mathcal{T}})^2$ is then given by:



The states we must consider are (q_I, q_I), (q_I, s), (s, s), (t, t), (r, r), and (r, t). We have $S(q_I) = \{a\}$, $S(s) = \{aa\}$ and $S(t) = S(r) = \{b\}$, so for every state of the form (q, q) we see that $S(q)^*$ is closed under insertion from $S(q)$. Hence we must consider the pairs (q_I, s) and (r, t). For the first case we see that $S(q_I)^* = \{a\}^*$ is closed under insertion from $S(s) = \{aa\}$, and for the second case we see that $S(r)^* = \{b\}$ is closed under insertion from $S(t) = \{b\}$. Hence $\mathcal{T}$ satisfies the condition of Theorem 3.2.14, so the word-insertion relation $\leq_{\mathcal{T}}$ is transitive.

Our goal is now to show that the condition stated in Theorem 3.2.14 is decidable. We start with the following:

Definition 3.2.16. Let $L \subseteq A^*$. We define the *K-insertion language* of L to be

$$L_K = \{\alpha v \beta \mid v \in K \text{ and } (\alpha, \beta) \text{ is a parsing of a word in } L\}.$$

We then note the following, which we state without proof:

Lemma 3.2.17. *Let $L, K \subseteq A^*$ with $\varepsilon \in L$. Then L is closed under insertion from K if and only if $L = L_K$.*

We then have the following:

Proposition 3.2.18. *Let $L \subseteq A^*$ be a regular language. Then the language L_K is regular.*

Proof. Let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a DFA accepting L and let $\mathcal{B} = (P, A, \sigma, p_0, G)$ be a DFA accepting K. Our aim is to construct an NFA $\mathcal{C} = (R, A, \Delta, q_0', F'')$ accepting L_K. The state set will be

$$R = Q' \cup (Q \times P) \cup Q''$$

where $Q' = \{q' \mid q \in Q\}$ and $Q'' = \{q'' \mid q \in Q\}$ are two copies of Q, and the set of accept states will be $F'' = \{q'' \mid q \in F\}$. The transitions in $\mathcal{C}$ will be exactly as follows:

- (q_1', a, q_2') for each transition (q_1, a, q_1) in $\mathcal{A}$;
- $(q', \varepsilon, (q, p_0))$ for every $q \in Q$;
- $((q, p_1), a, (q, p_2))$ for each $q \in Q$ and each transition (p_1, a, p_2) in $\mathcal{B}$;
- $((q, p), \varepsilon, q'')$ for every $q \in Q$ and $p \in G$.
- (q_1'', a, q_2'') for each transition (q_1, a, q_2) in $\mathcal{A}$;

The automaton $\mathcal{C}$ accepts a word u if and only if u can be written as $u = \alpha v \beta$ where:

- there is path in from q_0' to some state q' labelled by α;
- there is a path from (q, p_0) to (q, p) for some accept state $p \in G$ labelled by v;
- there is a path from q'' to an accept state s'' labelled by β.

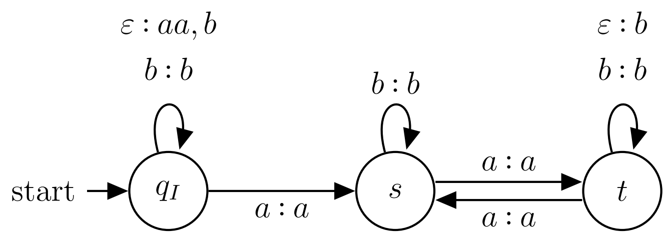
This is the case if and only if there are paths from q_0 to q, from p_0 to p and from q to s labelled by α, v and β respectively. This in turn is the case if and only if $\alpha\beta \in L$ and $v \in K$ are in L, meaning that the word $u = \alpha v \beta$ belong to L_K. □

Corollary 3.2.19. *Let $L, K \subseteq A^*$ be regular languages. Then it is decidable whether L is closed under insertion from K.*

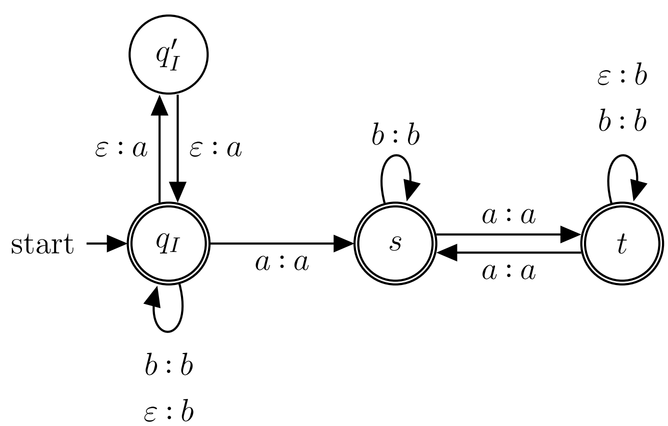
Proof. This follows from Lemma 3.2.17, Proposition 3.2.18 and the fact that equality is decidable for regular languages. □

Hence, in particular, the condition stated in Theorem 3.2.14 is decidable. The next example, however, shows that this condition is not necessary for a word-insertion relation to be transitive.

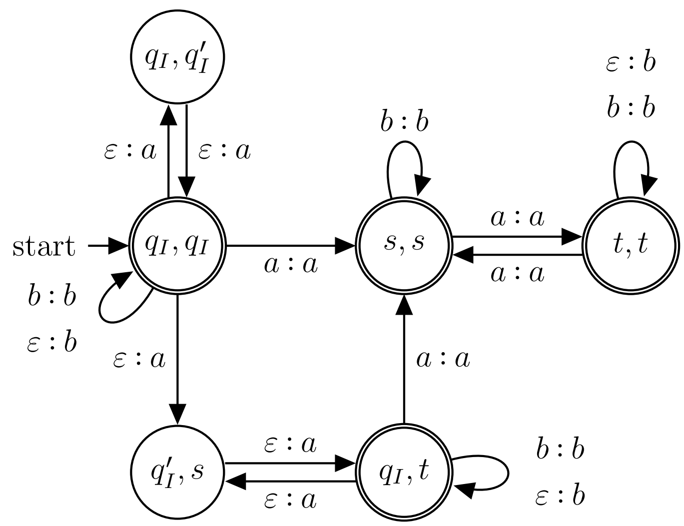
Example 3.2.20. Let $\mathcal{T}$ be the word-insertion transducer shown below. We have $S(q_I) = \{aa, b\}$, $S(s) = \varnothing$ and $S(t) = \{b\}$.



The transducer $\overline{\mathcal{T}}$ is shown below. For convenience we have written q_I' in place of $q_I^{(aa,1)}$.



The transducer $(\overline{\mathcal{T}})^2$ is then given by:



The state (q_I, t) is accessible in $(\overline{\mathcal{T}})^2$, but the set $S(q_I)^*$ is not closed under insertion from $S(t)$, since $aa \in S(q_I)^*$ and $b \in S(t)$ but $aba \notin S(q_I)^*$. Hence $\mathcal{T}$ does not satisfy the condition of Theorem 3.2.14. Now, the states (q_I, q_I), (q_I, q_I'), (s, s) and (t, t) behave like their counterparts in $\overline{\mathcal{T}}$, and by inspection any path in $(\overline{\mathcal{T}})^2$ through the states (q_I', t) and (q_I, t) can be simulated by the states q_I and q_I' in $\overline{\mathcal{T}}$. Hence, given a path in $(\overline{\mathcal{T}})^2$, there is a path in $\mathcal{T}$ with the same label, and so $\leq_{\mathcal{T}}$ is transitive.

Further remarks

While Theorem 3.2.14 appears to be a nice analogue of Corollary 3.1.29, the condition of being closed under insertion from another set is actually quite restrictive, especially since the sets $S(q)$ and $S(p)$ are finite. There are some non-trivial instances, for example:

- The set $\{a\}^*$ is closed under insertion from the set $\{aa\}$.
- More generally, if $n, m \geq 1$ then $\{a^n\}^*$ is closed under insertion from $\{a^{nm}\}$.
- The set $\{aa, ab, ba, bb\}^*$ is closed under insertion from the set $\{aa, bb\}$.
- More generally, if $X \subseteq A^n$ for some $n \geq 1$ then $(A^n)^*$ is closed under insertion from X.

The condition is certainly decidable, but it opens up the following questions:

- what are some (necessary, sufficient) combinatorial conditions on finite sets of words S, T for S^* to be closed under insertion from T?
- is it perhaps possible to classify (in some way) the finite sets satisfying this property?

It would also be of interest to see if a better condition than that of Theorem 3.2.14 can be formulated, i.e. a tighter sufficient condition or a condition which is both sufficient and necessary.

Chapter 4

Other families of orderings on words

4.1 The class of L-subword relations

We recall the definition of the *subword* ordering $\leq_w$ on A^*:

$$(u \leq_w v) \Leftrightarrow (\exists a_1, \ldots, a_n \in A)(\exists v_0, v_1, \ldots, v_n \in A^*)(u = a_1 \ldots a_n \ \&\ v = v_0 a_1 v_1 \ldots a_n v_n).$$

We consider generalising this ordering by replacing the language A^* in the above definition by some fixed language L. This gives rise to the class of L-*subword relations*, as defined below.

Definition 4.1.1. Let $L \subseteq A^*$. The L-*subword* relation on A^* is denoted by $\leq_L$ and defined by:

$$(u \leq_L v) \Leftrightarrow (\exists a_1, \ldots, a_n \in A)(\exists v_0, v_1, \ldots, v_n \in L)(u = a_1 \ldots a_n \ \&\ v = v_0 a_1 v_1 \ldots a_n v_n).$$

Example 4.1.2. Let $A = \{a, b, c\}$ and let $L \subseteq A^*$ be the language given by the regular expression $(aa)^*$. Then $bbb \leq_L baabaab$ and $abc \leq_L abaaaac$.

Our next example shows that not every relation in this class is an ordering.

Example 4.1.3. Let $A = \{a, b\}$ and let $L \subseteq A^*$ be the language given by the regular expression $(ab)^*$. Then $\varepsilon \leq_L ab \leq_L aabb$, but $\varepsilon \nleq_L aabb$ since $aabb \notin L$. Hence the relation $\leq_L$ is not transitive, and so it is not an ordering.

Closure under insertion

In the previous example we saw that $\leq_L$ will fail to be transitive if it is possible to insert a word $u \in L$ into a word $v \in L$ and obtain a word $w \notin L$, since then we will have $\varepsilon \leq_L v \leq_L w$ but $\varepsilon \nleq_L w$. We will go on to show that $\leq_L$ is transitive if and only if L does not exhibit this phenomenon.

We recall that a *parsing* of a word u is an ordered pair (α, β) such that $u = \alpha\beta$.

Definition 4.1.4. Let $L \subseteq A^*$. We say that L is *closed under insertion* if for all words $u, v \in L$ and for all parsings (α, β) of u we have $\alpha v \beta \in L$.

Example 4.1.5. We now show some examples of languages which are closed under insertion, and of some which are not.

- If $a \in A$ and $n \geq 1$ then the language given by the regular expression $(a^n)^*$ is closed under insertion.
- By contrast, if u is a word which is not of the form a^n then the language given by the regular expression u^* is not closed under insertion.
- The language of words $u \in \{a, b\}^*$ such that $|u|_a$ is divisible by 2 and $|u|_b$ is divisible by 3 is closed under insertion.
- By contrast, the language of words $u \in \{a, b\}^*$ such that $|u|_a$ is divisible by 2 and $|u|_b \equiv 1 \pmod 3$ is not closed under insertion.

Deciding whether an L-insertion relation is an ordering

Our goal is now to show that it is decidable whether a given L-insertion relation is an ordering.

Lemma 4.1.6. *Let $L \subseteq A^*$ be a language which is closed under insertion. Let $u \in L$ and let $v \in A^*$ be such that $u \leq_L v$. Then $v \in L$.*

Proof. This comes from repeatedly applying the fact that L is closed under insertion. Write $u = a_1 \ldots a_n$ and write $v = v_0 a_1 v_1 \ldots a_n v_n$ where each $v_i \in L$. The pair $(\varepsilon, a_1 \ldots a_n)$ is a parsing of the word u, so we have $v_0 a_1 \ldots a_n \in L$. Likewise, the pair

$(v_0a_1, a_2 \dots a_n)$ is a parsing of the word $v_0a_1 \dots a_n$, so we have $v_0a_1v_1a_2 \dots a_n \in L$. Continuing in this fashion we arrive at the desired result. □

The following theorem characterises the languages $L \subseteq A^*$ such that the L-subword relation $\leq_L$ is an ordering:

Theorem 4.1.7. *Let $L \subseteq A^*$. Then the L-subword relation $\leq_L$ is an ordering if and only if L is closed under insertion and contains ε.*

Proof. ($\Rightarrow$) We prove the contrapositive. First suppose that $\varepsilon \notin L$ and let $u \in A^*$. Then $u \nleq_L u$ and so $\leq_L$ is not reflexive. Next suppose that L is not closed under insertion, so that there exist words $u, v \in L$ and a parsing (α, β) of u such that $\alpha v \beta \notin L$. Then $\varepsilon \leq_L u \leq_L \alpha v \beta$, but $\varepsilon \nleq_L \alpha v \beta$ and so $\leq_L$ is not transitive.

($\Leftarrow$) It is clear that $\leq_L$ is reflexive since $\varepsilon \in L$ and that it is anti-symmetric since $u <_L v$ implies $|u| < |v|$. To see that $\leq_L$ is transitive let $u, v, w \in A^*$ be such that $u \leq_L v \leq_L w$. Write $u = a_1 \dots a_n$ and write $v = v_0a_1v_1 \dots a_nv_n$ where each $v_i \in L$. Next write $v = b_1 \dots b_m$ and let $i_1 < \cdots < i_n$ be such that:

- $v_0 = b_1 \dots b_{i_1 - 1}$;
- $a_k = b_{i_k}$ for $k \in \{1, \dots, n\}$;
- $v_k = b_{i_k+1} \dots b_{i_{k+1}-1}$ for $k \in \{1, \dots, n-1\}$;
- $v_n = b_{i_n+1} \dots b_m$.

This is illustrated below:

$$v = \underbrace{b_1 \dots b_{i_1-1}}_{v_0} a_1 \underbrace{b_{i_1+1} \dots b_{i_2-1}}_{v_1} \dots a_n \underbrace{b_{i_n+1} \dots b_m}_{v_n}.$$

Next write $w = w_0b_1w_1 \dots b_mw_m$ where each $w_i \in L$. Let:

- $x_0 = w_0b_1w_1 \dots b_{i_1-1}w_{i_1-1}$;
- $x_k = w_{i_k}b_{i_k+1}w_{i_k+1} \dots b_{i_{k+1}-1}w_{i_{k+1}-1}$ for $k \in \{1, \dots, n-1\}$;
- $x_n = w_{i_n}b_{i_n+1}w_{i_n+1} \dots b_mw_m$.

For each $i \in \{0, \ldots, n\}$ we have $v_i \in L$ and $v_i \leq_L x_i$, and so each $x_i \in L$ by Lemma 4.1.6. We have $w = x_0 a_1 x_1 \ldots a_n x_n$ and so $u \leq_L w$. $\square$

From Corollary 3.2.19 in the previous chapter, it is decidable whether a regular language is closed under insertion from another, and so in particular it is decidable whether a regular language is closed under insertion from itself. Hence we have the following:

Corollary 4.1.8. *Let $L \subseteq A^*$ be a regular language. Then it is decidable whether the L-subword relation $\leq_L$ is an ordering.*

We leave the problem of deciding WQO for L-subword relations as an open question.

4.2 The class of (k, l)-factor orderings

We recall the definition of the factor ordering $\leq_f$ on A^*:

$$(u \leq_f v) \Leftrightarrow (\exists \alpha, \beta \in A^*)(v = \alpha u \beta).$$

We introduce a variation on the factor ordering which permits one to insert letters amongst a fixed length prefix and suffix of a word.

Definition 4.2.1. Let $k, l \geq 1$. The (k, l)-*factor* ordering on A^* is denoted by $\leq_{k,l}$ and defined as follows. Let $u, v \in A^*$. If $|u| \leq k + l$ then $u \leq_{k,l} v$ precisely if $u \leq_w v$. If $|u| > k + l$ then write $u = xu'y$ where $|x| = k$ and $|y| = l$. Then $u \leq_{k,l} v$ if there exist words $\alpha, \beta \in A^*$ such that $x \leq_w \alpha, y \leq_w \beta$ and $v = \alpha u' \beta$.

Example 4.2.2. Let $A = \{a, e, h, r, s, t\}$ and consider the $(1, 2)$-factor ordering $\leq_{1,2}$ on A^*. Consider the word $tears \in A^*$. We have $t \leq_w th$ and $rs \leq_w tres$, so $tears \leq_{1,2} theatres$.

Intuitively we have $u \leq_{k,l} v$ if we can obtain v by inserting letters amongst the first k and last l letters of u. The (k, l)-factor orderings seek to bridge the gap between the subword and factor orderings, in the sense that they behave like the subword ordering near the beginning and end of a word and like the factor ordering elsewhere.

We now establish that each (k,l)-factor ordering is indeed an ordering.

Theorem 4.2.3. *Let $k, l \geq 1$. Then the (k,l)-factor ordering $\leq_{k,l}$ is an ordering on A^*.*

Proof. It is immediate that $\leq_{k,l}$ is reflexive and anti-symmetric, so it suffices to show that it is transitive. Let $u, v, w \in A^*$ be such that $u \leq_{k,l} v \leq_{k,l} w$. First suppose that $|u| \leq k+l$. Since $\leq_w$ contains $\leq_{k,l}$ we have $u \leq_w v$ and $v \leq_w w$, hence we have $u \leq_w w$ by the transitivity of $\leq_w$. Since $|u| \leq k+l$ we then have $u \leq_{k,l} w$. Now suppose $|u| > k+l$ and write $u = xu'y$ where $|x| = k$ and $|y| = l$. Let $\alpha, \beta \in A^*$ be such that $x \leq_w \alpha$, $y \leq_w \beta$ and $v = \alpha u' \beta$, and note that $|\alpha| \geq k$ and $|\beta| \geq l$. Next write $v = zv't$ where $|z| = k$ and $|t| = l$. We have $v = zv't = \alpha u' \beta$, and so z is a prefix of α, say $\alpha = z\alpha'$, and t is a suffix of β, say $\beta = \beta' t$. We have

$$v = \alpha u' \beta = z\alpha' u' \beta' t = zv't,$$

so $v' = \alpha' u' \beta'$. Now let $\gamma, \delta \in A^*$ be such that $z \leq_w \gamma$, $t \leq_w \delta$ and $w = \gamma v' \delta$. Since $z \leq_w \gamma$ we have $z\alpha' \leq_w \gamma\alpha'$, and since $t \leq_w \delta$ we have $\beta' t \leq_w \beta' \delta$. That is, we have $\alpha \leq_w \gamma\alpha'$ and $\beta \leq_w \beta'\delta$. Since $x \leq_w \alpha$ and $y \leq_w \beta$, we then have $x \leq_w \gamma\alpha'$ and $y \leq_w \beta'\delta$. We have

$$w = \gamma v' \delta = \gamma \alpha' u' \beta' \delta,$$

and so $u \leq_{k,l} w$. ☐

The next example shows that there are sets which are anti-chains under $\leq_f$ but are WQO under $\leq_{k,l}$.

Example 4.2.4. Let $A = \{a, b\}$ and consider the $(1,0)$-factor ordering $\leq_{1,0}$ on A^*. Then the set $S = \{ab^i a \mid i \geq 1\}$ is a chain under $\leq_{1,0}$. In particular, S is WQO under $\leq_{1,0}$.

Proof. Let $ab^i a$, $ab^j a \in S$ with $i \leq j$. Then $|a| = 1$ and $a \leq_w ab^{j-i}$, so $ab^i a \leq_{1,0} ab^j a$. ☐

However, we have the following:

Proposition 4.2.5. *Let $k, l \geq 1$. Then $\leq_{k,l}$ is not a WQO on A^*.*

Proof. We show that the set

$$S = \{a^{k+1}b^i a^{l+1} \mid i \geq 1\}$$

is an anti-chain under $\leq_{k,l}$. Suppose $u, v \in S$ are such that $u \leq_{k,l} v$, and write $u = a^{k+1}b^i a^{l+1}$ and $v = a^{k+1}b^j a^{l+1}$ for some $i, j \geq 0$. Let $\alpha, \beta \in A^*$ be such that $a^k \leq_w \alpha$, $a^l \leq_w \beta$ and $v = \alpha a b^i a \beta$. Then $\alpha = a^k$ and $\beta = a^l$, so $i = j$. □

4.2.1 Deciding WQO

The following lemma will assist us in characterising certain anti-chains under (k, l)-factor orderings:

Lemma 4.2.6. *Let $k, l \geq 1$ and let $u, v, x, y \in A^*$ with $|x| = k$ and $|y| = l$. Then $xuy \leq_{k,l} xvy$ if and only if $u \leq_f v$.*

Proof. (⇒) Suppose $xuy \leq_{k,l} xvy$. Let $\alpha, \beta \in A^*$ be such that $x \leq_w \alpha$, $y \leq_w \beta$ and $xvy = \alpha u \beta$. Then x is a prefix of α, say $\alpha = x\alpha'$, and y is a suffix of β, say $\beta = \beta' y$, and so $xvy = x\alpha' u \beta' y$. Hence $v = \alpha' u \beta'$, and so $u \leq_f v$.

(⇐) Suppose $u \leq_f v$, and let $\alpha, \beta \in A^*$ be such that $v = \alpha u \beta$. Then $x \leq_w x\alpha$ and $y \leq_w \beta y$, so

$$xuy \leq_{k,l} x\alpha u \beta y = xvy.$$

□

From this we can immediately derive the following:

Corollary 4.2.7. *Let $K \subseteq A^*$, let $k, l \geq 1$ and let $x, y \in A^*$ with $|x| = k$ and $|y| = l$. Then the language xKy is an anti-chain under $\leq_{k,l}$ if and only if the language K is an anti-chain under $\leq_f$.*

We now introduce a type of language derived from a given language L, which will assist in identifying anti-chains of the kind described in the above corollary.

Definition 4.2.8. Let $L \subseteq A^*$ and let $u, v \in A^*$. We define

$$u^{-1}Lv^{-1} = \{w \in A^* \mid uwv \in L\}.$$

Example 4.2.9. Let $L = \{ab^ia \mid i \geq 1\}$. Then $a^{-1}La^{-1} = \{b^i \mid i \geq 1\}$.

It is important to note that the languages L and $u(u^{-1}Lv^{-1})v$ need not coincide. For instance, if $L = \{aaa, b\}$ then $a^{-1}La^{-1} = \{a\}$, and so $a(a^{-1}La^{-1})a = \{aaa\} \neq L$. However, we can state the following:

Lemma 4.2.10. *Let $L \subseteq A^*$ and suppose there are words $u, v \in A^*$ such that $L \subseteq uA^*v$. Then $u(u^{-1}Lv^{-1})v = L$.*

Proof. We certainly have $u(u^{-1}Lv^{-1})v \subseteq L$. Now if $w \in L$ then we can write $w = uw'v$ for some $w' \in A^*$. In particular we have $w' \in u^{-1}Lv^{-1}$, and so $w \in u(u^{-1}Lv^{-1})v$. □

The next proposition shows that when L is regular, so too is every language of the form $u^{-1}Lv^{-1}$.

Proposition 4.2.11. *Let $L \subseteq A^*$ be a regular language accepted by a deterministic finite state automaton $\mathcal{A} = (Q, A, \delta, q_0, F)$, and let $u, v \in A^*$. Let $p = \delta^*(q_0, u)$ and let*

$$G = \{q \in Q \mid \delta^*(q, v) \in F\}.$$

Then the language $u^{-1}Lv^{-1}$ is regular, and is accepted by the deterministic finite state automaton $\mathcal{B} = (Q, A, \delta, p, G)$.

Proof. We show that $w \in u^{-1}Lv^{-1}$ if and only if w is accepted by $\mathcal{B}$. We have $w \in u^{-1}Lv^{-1}$ if and only if $uwv \in L$, which is true if and only if $\delta^*(q_0, uwv) \in F$. We have

$$\delta^*(q_0, uwv) = \delta^*(\delta^*(q_0, u), wv) = \delta^*(p, wv)$$

and so $\delta^*(q_0, uwv) \in F$ if and only if $\delta^*(p, wv) \in F$. We have

$$\delta^*(p, wv) = \delta^*(\delta^*(p, w), v)$$

and so $\delta^*(p, wv) \in F$ if and only if $\delta^*(p, w) \in G$. This is the case if and only if w is accepted by $\mathcal{B}$. $\square$

Finally we have the following result:

Lemma 4.2.12. *Let $K, L \subseteq A^*$ and let $u, v \in A^*$. Then $K \subseteq u^{-1}Lv^{-1}$ if and only if $uKv \subseteq L$.*

Proof. We have $K \subseteq u^{-1}Lv^{-1}$ if and only if for each $w \in K$ we have $w \in u^{-1}Lv^{-1}$. This is the case if and only if for each $w \in K$ we have $uwv \in L$, which is equivalent to saying that $uKv \subseteq L$. $\square$

We now characterise the sets which are WQO under (k,l)-factor orderings.

Theorem 4.2.13. *Let $L \subseteq A^*$ and let $k, l \geq 1$. Then L is WQO under $\leq_{k,l}$ if and only if $u^{-1}Lv^{-1}$ is WQO under $\leq_f$ for every $u, v \in A^*$ with $|u| = k$ and $|v| = l$.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose there are words $u, v \in A^*$ with $|u| = k$ and $|v| = l$ such that $u^{-1}Lv^{-1}$ is not WQO under $\leq_f$. Then $u^{-1}Lv^{-1}$ contains some infinite set K which is an anti-chain under $\leq_f$, and by Corollary 4.2.7 the set uKv is an infinite anti-chain under $\leq_{k,l}$. We have $K \subseteq u^{-1}Lv^{-1}$, and so by Lemma 4.2.12 we have $uKv \subseteq L$. Hence L contains an infinite anti-chain under $\leq_{k,l}$ and so is not WQO under $\leq_{k,l}$.

($\Leftarrow$) Again we prove the contrapositive. Suppose L is not WQO under $\leq_{k,l}$ so that it contains some infinite set K which is an anti-chain under $\leq_{k,l}$. Since K is infinite, there are words $u, v \in A^*$ with $|u| = k$ and $|v| = l$ such that infinitely many words in K have the form uxv for some $x \in A^*$. Let H be this set of words, that is, let

$$H = K \cap uA^*v.$$

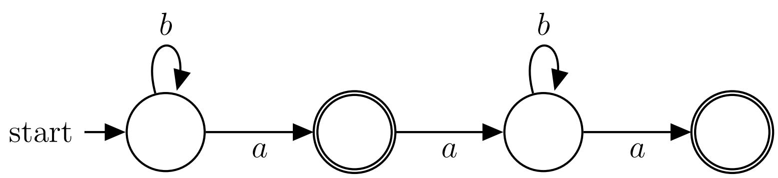
Furthermore let $G = u^{-1}Hv^{-1}$. We have $H \subseteq uA^*v$, and so by Lemma 4.2.10 we have $H = uGv$. The set K is an infinite anti-chain under $\leq_{k,l}$ and hence so is the set H. Hence G is an infinite anti-chain under $\leq_f$ by Corollary 4.2.7. We have $uGv = H \subseteq L$, and so by Lemma 4.2.12 we have $G \subseteq u^{-1}Lv^{-1}$. Hence $u^{-1}Lv^{-1}$ contains an infinite anti-chain under $\leq_f$ and so is not WQO under $\leq_f$. $\square$

As a corollary we then have:

Corollary 4.2.14. *Let $L \subseteq A^*$ be a regular language and let $k, l \geq 1$. Then it is decidable whether L is WQO under $\leq_{k,l}$.*

Proof. By Theorem 4.2.13, the language L is WQO under $\leq_{k,l}$ if and only if $u^{-1}Lv^{-1}$ is WQO under $\leq_f$ for every $u, v \in A^*$ with $|u| = k$ and $|v| = l$. By Proposition 4.2.11 each $u^{-1}Lv^{-1}$ is regular, and so by Theorem 1.3.6 it is decidable whether each $u^{-1}Lv^{-1}$ is WQO under $\leq_f$. There are only finitely many choices for u and v, and so the result follows. □

Example 4.2.15. Let L be the language accepted by the DFA shown below.



We show that L is WQO under the $(2,1)$-factor ordering $\leq_{2,1}$. To do this we consider each language $u^{-1}Lv^{-1}$ where $|u| = 2$ and $|v| = 1$ and show that it is WQO under the factor ordering. Such a language is empty unless $v = a$ and u is one of aa, ba and bb, so we must consider the languages $L_1 = (aa)^{-1}La^{-1}$, $L_2 = (ba)^{-1}La^{-1}$ and $L_3 = (bb)^{-1}La$. The language L_1 is given by the regular expression b^*, meaning that it is a chain under $\leq_f$ and hence is WQO. Likewise, L_2 is given by ab^* and so is also a chain. Finally, L_3 is given by $b^* + b^*aab^*$, and so is a union of L_1 together with the language $L_4 = L(b^*aab^*)$. The language L_4 is WQO by Lemma 1.3.11, meaning that L_3 is the union of two WQO sets and so is itself WQO.

We now show that the WQO Problem is decidable for (k,l)-factor orderings, for sets avoiding a given regular language. Throughout we will let $\mathrm{Av}_{k,l}(L)$ denote the avoidance set of a language L under the (k,l)-factor ordering. For a word $u \in A^*$ we will let S_u denote the upward closure of u under the subword ordering, and for a set $K \subseteq A^*$ we will let

$$S_K = \bigcup_{u \in K} S_u.$$

We first note the following result:

Lemma 4.2.16. *Let $K \subseteq A^*$ be a finite set. Then S_K is a regular language.*

Proof. For each word $u = a_1 \dots a_n$ in K the set S_u is given by $S_u = A^* a_1 A^* \dots a_n A^*$ and so is a regular language. Hence S_K is a union of finitely many regular languages and so is itself regular. □

We then have the following theorem:

Theorem 4.2.17. *Let $L \subseteq A^*$, let $k, l \geq 1$ and let K be the set of words in L of length at most $k + l$. Then*

$$A^* \setminus \mathrm{Av}_{k,l}(L) = S_K \cup \Big(\bigcup_{|u|=k, |v|=l} S_u (u^{-1} L v^{-1}) S_v \Big).$$

Proof. (⊆) Let y be a word in $A^* \setminus \mathrm{Av}_{k,l}(L)$. Then there is a word $x \in L$ with $x \leq_{k,l} y$. If $|x| < k + l$ then we have $x \in K$, and so $y \in S_K$. Now suppose $|x| > k + l$ and write $x = uzv$ where $|u| = k$ and $|v| = l$. Then z belongs to the set $u^{-1} L v^{-1}$. Now let $\alpha, \beta \in A^*$ be such that $u \leq_w \alpha$, $v \leq_w \beta$ and $y = \alpha z \beta$. Then $\alpha \in S_u$ and $\beta \in S_v$, so

$$y \in S_u (u^{-1} L v^{-1}) S_v.$$

(⊇) Let y be a word belonging to the set on the right hand side. If $y \in S_K$ then there is some $x \in K$ such that $x \leq_w y$. Since $x \in K$ we have $|x| \leq k + l$ and so $x \leq_{k,l} y$. Hence $y \in A^* \setminus \mathrm{Av}(L)$. Now suppose y belongs to

$$S_u (u^{-1} L v^{-1}) S_v$$

for some $u, v \in A^*$ with $|u| = k$ and $|v| = l$. Write $y = \alpha z \beta$ where $\alpha \in S_u$, $z \in u^{-1} L v^{-1}$ and $\beta \in S_v$. Then $u \leq_w \alpha$, $uzv \in L$ and $v \leq_w \beta$. We then have $uzv \leq_{k,l} y$, and since $uzv \in L$ we have $y \in A^* \setminus \mathrm{Av}(L)$. □

Lemma 4.2.18. *Let $L \subseteq A^*$ be a regular language and let $k, l \geq 1$. Then the language $\mathrm{Av}_{k,l}(L)$ is regular.*

Proof. We have shown that

$$A^*\backslash \mathrm{Av}_{k,l}(L) = S_K \cup \Big(\bigcup_{|u|=k,|v|=l} S_u(u^{-1}Lv^{-1})S_v \Big).$$

The set $\mathrm{Reg}(A)$ is closed under complementation, so to prove that $\mathrm{Av}_{k,l}(L)$ is regular it suffices to show the regularity of $A^*\backslash \mathrm{Av}_{k,l}(L)$. The set S_K is regular by Lemma 4.2.16, and the sets

$$S_u(u^{-1}Lv^{-1})S_v$$

are regular by Proposition 4.2.11 together with the fact that $\mathrm{Reg}(A)$ is closed under concatenation. There are only finitely many choices for u and v, so $A^*\backslash \mathrm{Av}(L)$ is a union of finitely many regular languages, meaning that it is itself a regular language. □

Finally we have:

Corollary 4.2.19. *Let $L \subseteq A^*$ be a regular language and let $k, l \geq 1$. Then it is decidable whether $\mathrm{Av}_{k,l}(L)$ is WQO under $\leq_{k,l}$.*

Proof. This is seen by combining Theorem 4.2.14 and Lemma 4.2.18. □

4.3 The class of I-factor orderings

Definition 4.3.1. For a finite set $I \subseteq A^*$ we define the *insertion closure* of I, denoted by $\mathrm{cl}(I)$, to be the smallest subset of A^* which contains I and is closed under insertion. By convention, we stipulate that $\varepsilon \in \mathrm{cl}(I)$.

Ehrenfeucht et al. [11] introduced a class of orderings $\leq_I$ on A^*, which are defined as follows:

Definition 4.3.2. Let $I \subseteq A^*$ be a finite set. The ordering $\leq_I$ on A^* is defined by

$$(u \leq_I v) \Leftrightarrow (\exists a_1, \ldots, a_n \in A)(\exists v_0, v_1, \ldots, v_n \in \mathrm{cl}(I))(u = a_1 \ldots a_n \;\&\; v = v_0 a_1 v_1 \ldots a_n v_n).$$

Intuitively, we have $u \leq_I v$ if v can be obtained from u by inserting words from $\mathrm{cl}(I)$. Another way to view this is that v can be obtained from u by repeated insertions of words from I.

Example 4.3.3. Let $A = \{a, b, c\}$ and $I = \{aa, bb\}$. Then $aa \leq_I abba$ and $ccc \leq_I cabbacaac$.

In their paper Ehrenfeucht et al. also make the following definition[1]:

Definition 4.3.4. Let $I \subseteq A^*$. We say I is *factor unavoidable* in A^* if only finitely many words in A^* do not have a factor in I, and *factor avoidable* otherwise.

Example 4.3.5. The set $I = \{aa, ab, bb\}$ is factor unavoidable in $\{a, b\}^*$ since the only words without a factor in I are ε, a, b and ba. The set $J = \{aa, bb\}$ is factor avoidable since every word of the form $(ab)^i$ has no factor in J.

It is decidable whether a given finite (indeed, regular) set $I \subseteq A^*$ is factor unavoidable. To see this we note that I is factor unavoidable if and only if the set $\mathrm{Av}_f(I)$ is finite. This language is regular and therefore it is decidable whether it is finite.

In their paper Ehrenfeucht et al. prove the following remarkable result[2]:

Theorem 4.3.6 (Ehrenfeucht et al.). *Let $I \subseteq A^*$ be finite. Then the following are equivalent:*

(i) *The set I is factor unavoidable in* $\mathrm{Alph}(I)^*$.

(ii) *The set* $\mathrm{cl}(I)$ *is a regular language.*

(iii) *The ordering $\leq_I$ is a WQO on* $\mathrm{Alph}(I)^*$.

Our goal in this section is to develop a class of orderings on A^* which extend the class introduced by Ehrenfeucht et al. and also encompass aspects of the factor ordering. These orderings are defined as follows:

[1] In their paper, they use the word *subword* to mean what we mean by *factor*.

[2] We state this as a single theorem. In their paper, it is stated disjointedly as Theorems 4.8 and 4.12.

Definition 4.3.7. Let $I \subseteq A^*$. The *I-factor ordering* on A^* is denoted by $\trianglelefteq_I$ and defined by:

$$(u \trianglelefteq_I v) \Leftrightarrow (\exists \alpha, u', \beta \in A^*)(u \leq_I u' \ \& \ v = \alpha u' \beta).$$

Informally, the ordering $\trianglelefteq_I$ behaves like $\leq_f$ on the exterior of a word and like $\leq_I$ on the interior, i.e. $u \trianglelefteq_I v$ if we can obtain v from u by inserting words from A^* at the start and end of u, and inserting words from $\mathrm{cl}(I)$ in-between the letters of u.

Example 4.3.8. Let $A = \{a, b, x, y\}$ and $I = \{aa, bb\}$. We have $aa \leq_I abba$, so $aa \trianglelefteq_I xyabbax$.

The following proposition, which we state without proof, provides two alternative characterisations of this class of relations:

Proposition 4.3.9. *Let $I \subseteq A^*$ be a finite set, let $u, v \in A^*$ and write $u = a_1 \dots a_n$. Then the following are equivalent:*

(i) *We have $u \trianglelefteq_I v$.*

(ii) *There is a word $u' \in A^*$ such that $u \leq_I u'$ and $u' \leq_f v$.*

(iii) *There are words $v_1, \dots, v_{n-1} \in \mathrm{cl}(I)$ and $v_0, v_n \in A^*$ such that $v = v_0 a_1 v_1 \dots a_n v_n$.*

We wish to show that each relation $\trianglelefteq_I$ is an ordering. First we establish the following lemma concerning the orderings $\leq_I$ of Ehrenfeucht et al.:

Lemma 4.3.10. *Let $I \subseteq A^*$ be a finite set and let $u, v \in A^*$ with $u \leq_I v$. Suppose we can write $u = xyz$ for some $x, y, z \in A^*$. Then we can write $v = x'y'z'$ for some $x', y', z' \in A^*$ such that $x \leq_I x'$, $y \leq_I y'$ and $z \leq_I z'$.*

Proof. Write $u = a_1 \dots a_n$ and write $v = v_0 a_1 v_1 \dots a_n v_n$ where each $v_i \in \mathrm{cl}(I)$. Let i, j be such that $x = a_1 \dots a_i$, $y = a_{i+1} \dots a_j$ and $z = a_{j+1} \dots a_n$. Let $x' = v_0 a_1 v_1 \dots a_i v_i$, $y' = a_{i+1} v_{i+1} \dots a_j v_j$ and $z' = a_{j+1} v_{j+1} \dots a_n v_n$. Then $v = x'y'z'$, and we have $x \leq_i x'$, $y \leq_I y'$ and $z \leq_i z'$. □

We then have:

Proposition 4.3.11. *Let $I \subseteq A^*$ be a finite set. Then the relation $\trianglelefteq_I$ is an ordering on A^*.*

Proof. It is clear that the relation $\trianglelefteq_I$ is reflexive and anti-symmetric, so we show it is transitive. Let $u, v, w \in A^*$ be such that $u \trianglelefteq_I v \trianglelefteq_I w$. By Proposition 4.3.9, we can write $v = \alpha u' \beta$ where $u \leq_I u'$ and we can write $w = \gamma v' \delta$ where $v \leq_I v'$. From Lemma 4.3.10 we can write $v = \alpha' u'' \beta'$ where $\alpha \leq_I \alpha'$, $u' \leq_I u''$ and $\beta \leq_I \beta'$. We then have $u \leq_I u' \leq_I u''$ so $u \leq_I u''$, and $w = \gamma \alpha' u'' \beta' \delta$ so $u \trianglelefteq_I w$. □

WQO under $\trianglelefteq_I$

Each ordering $\trianglelefteq_I$ contains the orderings $\leq_f$ and $\leq_I$ and incorporates elements of them both. Interestingly, there are certain sets which are anti-chains under each of these orderings but WQO under $\trianglelefteq_I$. We first note the following:

Observation 4.3.12. *Let $I \subseteq A^*$ be a finite set and let $u, v \in A^*$ with $u <_I v$. Then v has a factor in I.*

We then have:

Example 4.3.13. Let $I = \{aa, bb\}$. Then:

(i) The set S given by the regular expression $a(bb)^*a$ is an anti-chain under $\leq_f$ but is a chain under $\trianglelefteq_I$.

(ii) The set T given by the regular expression $(ab)^*$ is an anti-chain under $\leq_I$ but is a chain under $\trianglelefteq_I$.

Proof. (i) It is clear that S is an anti-chain under $\leq_f$. Now let $u, v \in S$ with $|u| \leq |v|$ and write $u = ab^{2i}a$ and $v = ab^{2j}a$. Then $v = ab^{2i}b^{2(j-i)}a$ and so $u \trianglelefteq_I v$.

(ii) No word in T has a factor in I, so T is an anti-chain under $\leq_I$ by Observation 4.3.12. It is clear that T is a chain under $\leq_f$. □

We will go on to show that, despite this, the ordering $\trianglelefteq_I$ will be WQO precisely when $\leq_I$ is.

Lemma 4.3.14. *Let $I \subseteq A^*$ be a finite set which is factor unavoidable in A^*. Then there exists a word $u \in A^*$ such that every power of u has no factor in I.*

Proof. Since I is factor unavoidable, its avoidance set $\mathrm{Av}_f(I)$ under the factor ordering is infinite. Furthermore, since I is finite, the set $\mathrm{Av}_f(I)$ is regular, so let

$\mathcal{A} = (Q, A, \delta, q_0, F)$ be a reduced DFA accepting $\mathrm{Av}_f(I)$. The set $\mathrm{Av}_f(I)$ is infinite so $\mathcal{A}$ must have a loop-state, that is, there must be a state $q \in Q$ and a non-empty word $u \in A^*$ such that $\delta^*(q, u) = q$. Since $\mathcal{A}$ is reduced we can select words v and w such that $\delta^*(q_0, v) = q$ and $\delta^*(q, w) \in F$. Then for all $i \geq 0$ we have

$$\delta^*(q_0, vu^i w) \in F$$

and so $vu^i w \in \mathrm{Av}_f(I)$. For each i the word u^i is a factor of $vu^i w$, and since $\mathrm{Av}_f(I)$ is downward-closed under the factor ordering this means that u^i belongs to $\mathrm{Av}_f(I)$. $\square$

For the following theorem we recall that a word γ is said to be a *left extension of a power of* α if we can write $\gamma = \sigma\alpha^i$ where σ is a suffix of α, and that δ is a *right extension of a power of* α if we can write $\delta = \alpha^i\sigma$ where σ is a prefix of α.

Theorem 4.3.15. *Let $I \subseteq A^*$ be regular. Then the ordering $\trianglelefteq_I$ is a WQO if and only if I is factor unavoidable in A^*.*

Proof. ($\Rightarrow$) If I is factor unavoidable in A^* then by Lemma 4.3.14 there is a word u such that every power of u has no factor in I. Write $u = a_1 \dots a_n$, and let $c \in A \backslash \{a_n\}$ and $d \in A \backslash \{a_1\}$. Then c is not a left extension of a power of u and d is not a right extension of a power of u, so by Lemma 1.3.10 the set

$$S = \{cu^i d \mid i = 1, 2, \dots\}$$

is an infinite anti-chain under the factor ordering. Our aim is to show that S is also an anti-chain under $\trianglelefteq_I$. Suppose instead that there are words $v, w \in S$ with $v \trianglelefteq_I w$ and write $v = cu^i d$ and $w = cu^j d$. Write $u^i = b_1 \dots b_m$ and write

$$w = zcw_0 b_1 w_1 \dots b_m w_m dt$$

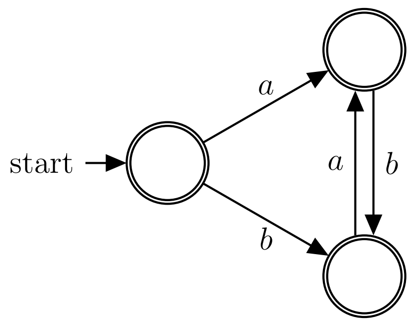
where $z, t \in A^*$ and each $w_i \in \mathrm{cl}(I)$. Since $|zc|, |dt| \geq 1$ we see that $w_0 b_1 w_1 \dots b_m w_m$ is a factor of u^j, but u^j has no factor in I and so each $w_i = \varepsilon$. Hence $v = w$.

($\Leftarrow$) Suppose that I is factor unavoidable in A^*. By Theorem 4.3.6, the ordering $\leq_I$ is a WQO, and since $\trianglelefteq_I$ contains $\leq_I$ we see that $\trianglelefteq_I$ is also a WQO. $\square$

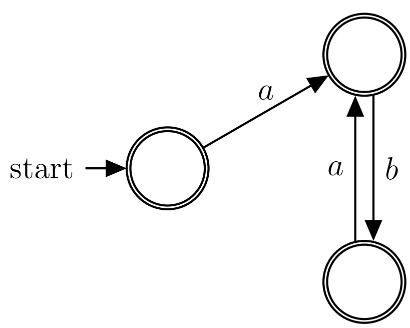
This leads us to considering that nature of anti-chains under $\trianglelefteq_I$. Our study is motivated by the following example:

Example 4.3.16. Let $I = \{aa, bb\}$ and let $S \subseteq \{a, b\}^*$ be an infinite set with no factor in I. Then S is WQO under the ordering $\trianglelefteq_I$.

Proof. In fact we can show that this holds as long as $|S| \geq 3$. The set S has no factor in I and so S is contained in the avoidance set $\mathrm{Av}_f(I)$ of I under the factor ordering. The set $\mathrm{Av}_f(I)$ is regular and is accepted by the automaton $\mathcal{A}$ shown below:



Let u, v and w be distinct non-empty words in S. Two of these words must start with the same letter, and we may assume without loss of generality that these words are u and v and that this letter is a. Furthermore we may assume that $|u| \leq |v|$. The words u and v are both accepted by the automaton $\mathcal{B}$ shown below:



We see that each state of $\mathcal{B}$ has out-degree 1, and so the language accepted by it is a chain under the prefix ordering by Theorem 2.1.2. This means u is a prefix of v and hence a factor of v, and so $u \trianglelefteq_I v$. This shows that if S contains an ant-chain under $\trianglelefteq_I$ then that anti-chain contains at most 2 non-empty words, so S is WQO. $\square$

The above result does not hold for every choice of I. Indeed if $I = \{aa\}$ then the set ab^*a has no factor in I and is an anti-chain under $\trianglelefteq_I$. This leads to the following more general statement:

Proposition 4.3.17. *Let $I \subseteq A^*$ be a finite set and suppose that the ordering $\leq_I$ is not WQO. Then there exists an infinite set $X \subseteq A^*$ which is an anti-chain under $\leq_I$, but which is WQO under $\trianglelefteq_I$.*

Proof. By Theorem 4.3.6 the set I is factor unavoidable in A^*. By Lemma 4.3.14 there exists a word $u \in A^*$ such that every power of u has no factor in I, and so the set

$$S = \{u^i \mid i = 1, 2, \dots\}$$

is an anti-chain under $\leq_I$. The set S is a chain under the factor ordering, and so it is also a chain under the ordering $\trianglelefteq_I$. Hence S is WQO under $\trianglelefteq_I$. $\square$

Chapter 5

The consecutive involvement ordering on permutations

5.1 Atomicity in the consecutive involvement ordering

In this section we show it is decidable whether a given finitely-based set C of permutations is atomic. Our approach will be similar to that for the factor ordering on words, in that we will employ a graph $G(C)$ whose paths correspond to all sufficiently long permutations in C. A key difference from the word case will be that a given path need not correspond to a unique permutation in C. In fact, we will show that if $G(C)$ is not strongly connected then any occurrence of this 'non-uniqueness' phenomenon will imply that C is not atomic. We will therefore explore this property in detail and show that it is decidable whether a given graph $G(C)$ exhibits it. Deciding atomicity will then essentially boil down to testing for this condition and performing the same analysis on $G(C)$ as we did in the word case.

We remark that much of the framework employed in this chapter, particularly around ambiguous sequences and graphs of permutations, has already been considered by Avgustinovich and Kitaev [4]. To help with the material which will follow, we will introduce it here under our own terminology.

5.1.1 Overlapping k-sequences and ambiguity

We begin by making some definitions, which are analogous to those used for words.

Definition 5.1.1. Let $\sigma = s_1 \dots s_n$ be a permutation and let $k \in \{1, \dots, n\}$. We define $\mathrm{seq}_k(\sigma)$ to be the sequence of permutations $\mathrm{seq}_k(\sigma) = (\sigma_1, \dots, \sigma_{n-k+1})$ where each $\sigma_i = \rho(s_i \dots s_{i+k-1})$.

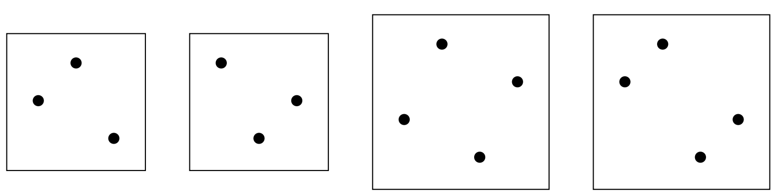
The sequence $\mathrm{seq}_k(\sigma)$ can be thought of as the sequence of consecutive length k permutations which are involved in σ.

Example 5.1.2. Let $\sigma = 34512$. Then $\mathrm{seq}_3(\sigma) = (123, 231, 312)$ and $\mathrm{seq}_2(\sigma) = (12, 12, 21, 12)$.

Notation 5.1.3. If $\sigma = s_1 \dots s_n$ is a permutation then we let $\sigma^P = \rho(s_1 \dots s_{n-1})$ and $\sigma^S = \rho(s_2 \dots s_n)$.

Definition 5.1.4. Let $S = (\sigma_1, \dots, \sigma_n)$ be a sequence of permutations of the same length k. We say S is an *overlapping k-sequence* if it satisfies $\sigma_i^S = \sigma_{i+1}^P$ for all $i \in \{1, \dots, n-1\}$.

For words it was always possible to take an overlapping k-sequence and recover a unique word, but for permutations this is not always be the case. Indeed, let $S = (231, 312)$, $\sigma = 2413$ and $\tau = 3412$. Then σ and τ are distinct but we have $\mathrm{seq}_k(\sigma) = \mathrm{seq}_k(\tau) = S$. We show each of these permutations below.



231 312 $\sigma = 2413$ $\tau = 3412$

It therefore makes sense to introduce the following definition:

Definition 5.1.5. Let S be an overlapping k-sequence. We say S is *unambiguous* if there is a unique permutation σ satisfying $\mathrm{seq}_k(\sigma) = S$. Otherwise we say S is *ambiguous*.

We will explore this concept in further detail later on. For now we turn our attention to defining a graph corresponding to a finitely-based class.

5.1.2 Consecutive involvement graphs

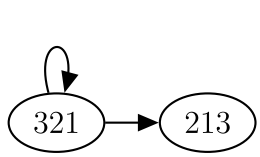
We will use the following notation, which is identical to that used for words:

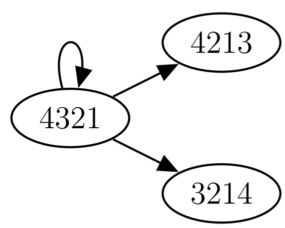
Notation 5.1.6. If C is a finitely-based set of permutations then we let C_k denote the set of permutations in C of length k. We also let $C_{\geq k}$ denote the set of permutations in C of length at least k, and make analogous definitions for the sets $C_{>k}, C_{\leq k}$ and $C_{<k}$.

We then make the following definition:

Definition 5.1.7. Let B be a finite set of permutations, let $k = \max_{\sigma \in B} |\sigma|$, let $C = \mathrm{Av}(B)$ and let l be an integer with $l \geq k$. Then the *l-consecutive involvement graph* for C is the directed graph $G_l(C)$ which has vertex set $V_l = C_l$ and edge set $E_l = \{(\sigma, \tau) \mid \sigma^S = \tau^P\}$. In the case that $l = k$ we will omit the subscript and simply refer to this graph as $G(C)$.

Example 5.1.8. Let $B = \{123, 132, 231, 312\}$ and $C = \mathrm{Av}(B)$. Then $\max_{\sigma \in B} |\sigma| = 3$, and we have $C_3 = \{213, 321\}$ and $C_4 = \{4321, 3214, 4213\}$. We show the graphs $G(C) = G_3(C)$ and $G_4(C)$ below.





As with words, all sufficiently long permutations in C correspond to paths in $G_l(C)$. We state without proof the following result, which is analogous to Proposition 2.2.14 for words:

Proposition 5.1.9. *Let B be a finite set of permutations, let $k = \max_{\sigma \in B} |\sigma|$, let $C = \mathrm{Av}(B)$ and let $l \geq k$. Then*

(i) *If* $\sigma \in C_{\geq l}$ *then* $\mathrm{seq}_l(\sigma)$ *is a path in* $G_l(C)$*.*

(ii) *If* S *is a path in* $G_l(C)$ *and* σ *is a permutation with* $\mathrm{seq}_l(\sigma) = S$ *then* $\sigma \in C_{\geq l}$*.*

When considering words we observed the following: if in $G(C)$ we have $S \leq T$ then in C we have $w(S) \leq_f w(T)$. However, for permutations the equivalent implication does not always hold. For instance, if we consider the graph $G(C)$ from Example 5.1.8 then $S = (321, 213)$ and $T = (321, 321, 213)$ are paths in $G(C)$ with $S \leq T$. However, $\alpha = 4213$ and $\beta = 43215$ are permutations in C with $\mathrm{seq}_3(\alpha) = S$ and $\mathrm{seq}_3(\beta) = T$, but $\alpha \nleq \beta$. However, we can state the following:

Lemma 5.1.10. *If* σ, *and* τ *are permutations with* $\sigma \leq \tau$ *then* $\mathrm{seq}_k(\sigma) \leq \mathrm{seq}_k(\tau)$*.*

Proof. Write $\sigma = s_1 \dots s_n$ and $\tau = t_1 \dots t_m$ and let j be such that $\rho(t_j \dots t_{j+n-1}) = \sigma$. Write $\mathrm{seq}_k(\sigma) = (\sigma_1, \dots, \sigma_{n-k+1})$ where each $\sigma_i = \rho(s_i \dots s_{i+l-1})$ and write $\mathrm{seq}_k(\tau) = (\tau_1, \dots, \tau_{m-k+1})$ where each $\tau_i = \rho(t_i \dots t_{i+k-1})$. Then we have:

$$(\tau_j, \dots, \tau_{j+n-k}) = (\rho(t_j, \dots, t_{j+k-1}), \dots, \rho(t_{j+n-k} \dots t_{j+n-k})).$$

The sequence on the right-hand side is just $(\rho(s_1 \dots s_k), \dots, \rho(s_{n-k+1} \dots s_n))$, and we have

$$(\rho(s_1 \dots s_k), \dots, \rho(s_{n-k+1} \dots s_n)) = (\sigma_1, \dots, \sigma_{n-k+1}) = \mathrm{seq}_k(\sigma),$$

so $\mathrm{seq}_k(\sigma) \leq \mathrm{seq}_k(\tau)$. □

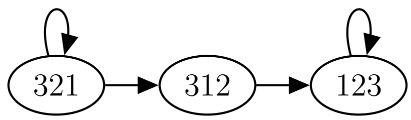
An alternative construction for l-consecutive involvement graphs

Once we have constructed $G(C)$, we can determine each graph $G_{k+l-1}(C)$ without considering the sets B or C. To do this we let $\mathcal{P}_l(C)$ denote the set of length l paths in $G(C)$ and for $S = (\sigma_1, \dots, \sigma_l) \in \mathcal{P}_l(C)$ we let $V(S)$ be the set of all permutations τ of length $k + l - 1$ satisfying $\mathrm{seq}_k(\tau) = S$. We then define the vertex set of $G_{k+l-1}(C)$ to be

$$V_{k+l-1} = \bigcup_{S \in \mathcal{P}_l(C)} V(S)$$

and construct the edge set E_{k+l-1} as follows: if $S = (\sigma_1, \dots, \sigma_l)$ and $T = (\tau_1, \dots, \tau_l)$ are paths of length l in $G(C)$ satisfying $(\sigma_2, \dots, \sigma_l) = (\tau_1, \dots, \tau_{l-1})$ then we add an edge from every vertex in $V(S)$ to every vertex in $V(T)$.

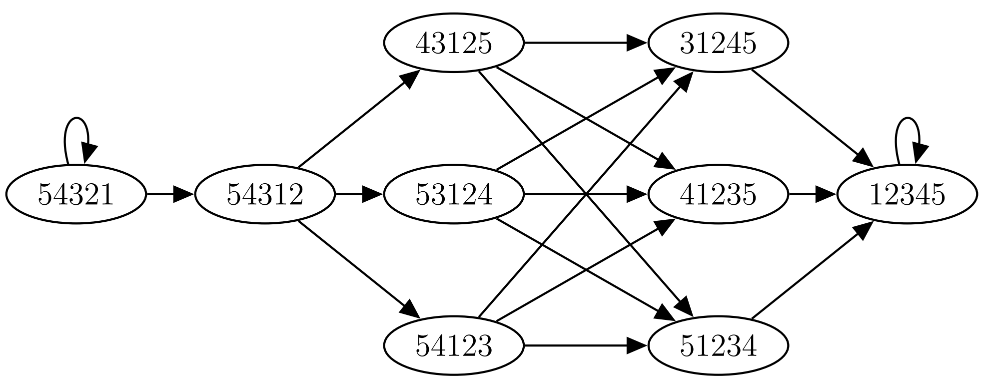
Example 5.1.11. Let $B = \{132, 213, 231\}$ and let $C = \mathrm{Av}(B)$. Then $\max_{\sigma \in B} |\sigma| = 3$ and $C_3 = \{123, 132, 213\}$. The graph $G(C)$ is shown below.



We will construct the graph $G_5(C)$, whose vertices correspond to length 3 paths in $G(C)$. The table below shows the paths S of length 3 in $G(C)$ and the corresponding vertex sets $V(S)$ in $G_5(C)$.

S	Members of $V(S)$
$(321, 321, 321)$	54321
$(321, 321, 312)$	54312
$(321, 312, 123)$	43125, 53124, 54123
$(312, 123, 123)$	31245, 41235, 51234
$(123, 123, 123)$	12345

The graph $G_5(C)$ is shown below.



With this construction in mind we can observe the following:

Lemma 5.1.12. *Suppose $G(C)$ is strongly connected, let $k = \max_{\sigma \in B} |\sigma|$ and let $l \geq 1$. Then $G_{k+l-1}(C)$ is strongly connected.*

Proof. Let α and β be vertices of $G_{k+l-1}(C)$ and let $S = (\sigma_1, \ldots, \sigma_l)$ and $T = (\tau_1, \ldots, \tau_l)$ be such that $\mathrm{seq}_k(\alpha) = S$ and $\mathrm{seq}_k(\beta) = T$, so that $\alpha \in V(S)$ and $\beta \in V(T)$.

Since G is strongly connected there is a path in $G(C)$ from σ_l to τ_1, say

$$P = (\sigma_l, \sigma_{l+1}, \dots, \sigma_{l+n}, \tau_1).$$

For $i \in \{1, \dots, n\}$ let $R_i = (\sigma_{i+1}, \dots, \sigma_{i+l})$ and select a permutation $\gamma_i \in V(R_i)$. Then $(\alpha, \gamma_1, \dots, \gamma_n, \beta)$ is a path in $G_{k+l-1}(C)$ from α to β. □

Lemma 5.1.13. *Let $k = \max_{\sigma \in B} |\sigma|$ and suppose $G(C)$ is strongly connected. Then $C_{\geq k}$ satisfies the join property.*

Proof. We first note that in a graph which is strongly connected, there are arbitrarily long paths containing any given vertex. Now let $\alpha, \beta \in C_{\geq k}$ and let $S = \mathrm{seq}_k(\alpha)$ and $T = \mathrm{seq}_k(\beta)$. Let $n = |S|$ and $m = |T|$ and assume without loss of generality that $n \leq m$. By Lemma 5.1.12 the graphs $G_{k+n-1}(C)$ and $G_{k+m-1}(C)$ are strongly connected. The permutation α is a vertex in $G_{k+n-1}(C)$, and since $G_{k+n-1}(C)$ is strongly connected it has a path R of length $m - n$ containing α. Let γ be a permutation satisfying $\mathrm{seq}_k(\gamma) = R$. Then γ is a vertex of $G_{k+m-1}(C)$ and $\alpha \leq \gamma$. Since $G_{k+m-1}(C)$ is strongly connected, it has a path Q from β to γ. Now let δ be a permutation satisfying $\mathrm{seq}_k(\delta) = Q$. Then we have $\alpha \leq \delta$ and $\beta \leq \delta$, so δ is a join for α and β. □

We will therefore once again focus on graphs which are connected but not strongly connected.

5.1.3 Deciding ambiguity

In this subsection we show that it is decidable whether a given graph $G(C)$ has ambiguous paths. We will go on to show that this is pivotal to deciding whether C is atomic. We begin with the following:

Lemma 5.1.14. *Let S and T be overlapping k-sequences such that $S \leq T$ and suppose T is unambiguous. Then S is unambiguous.*

Proof. We will prove that if $S = (\sigma_1, \dots, \sigma_n)$ is ambiguous then $S' = (\sigma_1, \dots, \sigma_n, \sigma_{n+1})$ is ambiguous. It will follow by symmetry that a sequence of the form $(\sigma_0, \sigma_1 \dots, \sigma_n)$

is ambiguous. If $S \le T$ then T can be constructed from S by some sequence

$$S = S_0 < S_1 < \cdots < S_N = T$$

obtained by appending elements in this way, so this will mean T is ambiguous.

Suppose $\alpha = s_1 \dots s_m$ and $\beta = t_1 \dots t_m$ are two distinct permutations such that $\mathrm{seq}_k(\alpha) = \mathrm{seq}_k(\beta)$, and write $\sigma_{n+1} = r_1 \dots r_k$. First suppose that $r_k = k$. Let $s_{m+1} = t_{m+1} = k+1$ and let $\alpha' = s_1 \dots s_{m+1}$ and $\beta' = t_1 \dots t_{m+1}$. Then α' and β' are distinct since α and β are distinct, and we have $\mathrm{seq}_k(\alpha') = \mathrm{seq}_k(\alpha)' = S'$, so S' is ambiguous. Next suppose $r_k < k$ and let $j \in \{1, \dots, k-1\}$ be such that $r_j = r_k + 1$. Now let $p_1 \dots p_m$ be the sequence given by

$$p_i = \begin{cases} s_i + 1 & \text{if } s_i \ge s_{n+j-1} \\ s_i & \text{if } s_i < s_{n+j-1} \end{cases}$$

and let $p_{m+1} = s_{n+j-1}$. Also let $q_1 \dots q_m$ be the sequence given by

$$q_i = \begin{cases} t_i + 1 & \text{if } t_i \ge t_{n+j-1} \\ t_i & \text{if } t_i < t_{n+j-1} \end{cases}$$

and let $q_{m+1} = t_{n+l-1}$. Then $\alpha' = p_1 \dots p_{m+1}$ and $\beta' = q_1 \dots q_{m+1}$ are distinct permutations and we have $\mathrm{seq}_k(\alpha') = \mathrm{seq}_k(\beta) = S'$, so S' is ambiguous. $\square$

With this in mind we can make the following definition:

Definition 5.1.15. Let $S = (\sigma_1, \dots, \sigma_n)$ be an overlapping k-sequence. We say S is a *minimal ambiguous k-sequence* if it is ambiguous and the sequences $(\sigma_1, \dots, \sigma_{n-1})$ and $(\sigma_2, \dots, \sigma_n)$ are unambiguous.

Example 5.1.16. Let $S = (132, 312, 231)$. Then S is ambiguous since

$$\mathrm{seq}_3(15342) = \mathrm{seq}_3(25341) = S,$$

and S is minimal since $\sigma = 1423$ is unique such that $\mathrm{seq}_k(\sigma) = (132, 312)$ and $\tau = 4231$ is unique such that $\mathrm{seq}_k(\tau) = (312, 231)$.

The reason for considering minimal ambiguous sequences is the following:

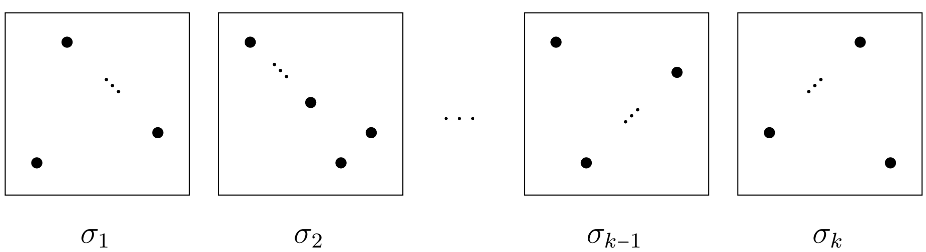
Observation 5.1.17. *A graph $G(C)$ has ambiguous paths if and only if it has minimal ambiguous paths.*

We can therefore reduce the problem of deciding whether $G(C)$ has ambiguous paths to that of deciding whether $G(C)$ has minimal ambiguous paths. Our approach will be to show that, for a fixed k, there is a bound on the length of minimal ambiguous k-sequences. This would not be worthwhile if, for instance, every possible minimal ambiguous sequence had length at most 3. We therefore describe how minimal ambiguous sequences can be arbitrarily long, in the sense that, for any given n, there exists a k such that there are minimal ambiguous k-sequences of length n.

Indeed, we can select $n = k$. Define a sequence of permutation $S = (\sigma_1, \ldots, \sigma_k)$ by

$$\sigma_i = \begin{cases} 1 \oplus \delta_{k-1} & \text{if } i = 1 \\ \delta_{k-i} \ominus \alpha_i & \text{if } i \in \{2, \ldots, k-1\} \\ \alpha_{k-1} \ominus 1 & \text{if } i = k. \end{cases}$$

This sequence is sketched below.

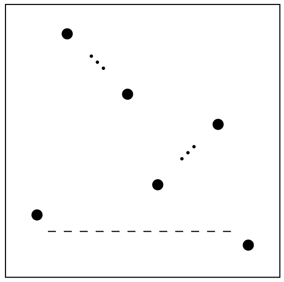
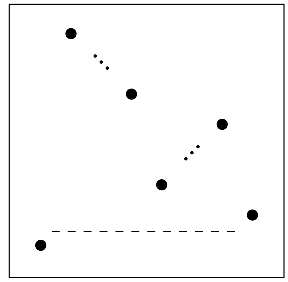


The sequence S is ambiguous since the permutations

$$\sigma = (1 \oplus (\delta_{k-1} \ominus \alpha_{k-1})) \ominus 1,$$

$$\tau = 1 \oplus (\delta_{k-1} \ominus \alpha_{k-1} \ominus 1)$$

satisfy $\mathrm{seq}_k(\sigma) = \mathrm{seq}_k(\tau) = S$. These are sketched below.

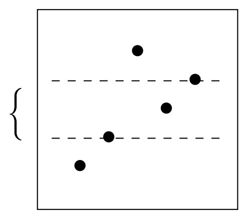
 

$\sigma = (1 \oplus (\delta_{k-1} \ominus \alpha_{k-1})) \ominus 1 \qquad \tau = 1 \oplus (\delta_{k-1} \ominus \alpha_{k-1} \ominus 1)$

Finally, S is a minimal ambiguous sequence since the sequence $S' = (\sigma_1, \ldots, \sigma_{k-1})$ uniquely determines the permutation $1 \oplus (\delta_{k-1} \ominus \alpha_{k-1})$ and $S'' = (\sigma_2, \ldots, \sigma_k)$ uniquely determines the permutation $\delta_{k-1} \ominus \alpha_{k-1} \ominus 1$.

Intervals

We will consider intervals of values in a permutation. Geometrically, we mean intervals on the vertical axis, as shown below.

The interval (2,4) 

1 2 5 3 4

For our purposes, the *intervals* present in a permutation of length k will be all the ordered pairs of the form $I = (x, y)$ where $0 \leq x < y \leq k + 1$. The integers x and y are said to be the *endpoints* of the interval I. We emphasise that, for our purposes, an interval is an ordered pair rather than a set of numbers, so that the intervals $(0, 1)$ and $(2, 3)$ are considered distinct despite the fact that

$$\{n \in \mathbb{N} \mid 0 < n < 1\} = \{n \in \mathbb{N} \mid 2 < n < 3\} = \varnothing.$$

An interval (x, y) is said to *contain* n if $x < n < y$. We say an interval is *non-empty* if it contains some n and *empty* otherwise. We will say that an interval (x, y) is a *sub-interval* of an interval (z,t) if $z \leq x$ and $y \leq t$. Two intervals are said to *overlap* if they are equal or if one contains at least one endpoint of the other.

Lemma 5.1.18. *Suppose that I and J are intervals and that there is an integer n contained in both I and J. Then the intervals I and J overlap.*

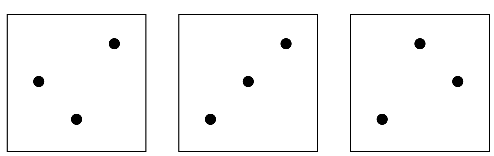
Proof. We will suppose that the intervals I and J are distinct and prove that one contains at least one endpoint of the other. Write $I = (x,y)$ and $J = (z,t)$ and assume without loss of generality that $x < z$. Since n is contained in both I and J we have $x < n < y$ and $z < n < t$, and so $x < z < y$. Hence z is contained in I. $\square$

We now develop some definitions pertaining to overlapping k-sequences.

Definition 5.1.19. Let $S = (\sigma_1, \dots, \sigma_n)$ be an overlapping k-sequence and let $\alpha = a_1 \dots a_m$ be a permutation such that $\mathrm{seq}_k(\alpha) = S$. Let $i \in \{1, \dots, n\}$ and write $\sigma_i = s_1 \dots s_k$. For $j \in \{1, \dots, k\}$ we define the *α-representative* of s_j to be the point $s'_j = a_{i+j-1}$. By convention we also define $0' = 0$ and $k + 1' = m + 1$. If $I = (x,y)$ is an interval in σ_i then we define the α-representative of I to be $I_\alpha = (x', y')$.

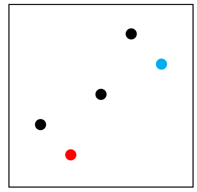
Intuitively, the point in position j of s_i will 'manifest' as the point in position $i+j-1$ of α. We note that a particular point of α will typically be the α-representative of several points, for instance, the second point of α will be the α-representative of the second point of σ_1 and of the first point of σ_2.

Example 5.1.20. Let $S = (\sigma_1, \sigma_2, \sigma_3)$ where $\sigma_1 = 213, \sigma_2 = 123$ and $\sigma_3 = 132$. These are illustrated below.



σ_1 σ_2 σ_3

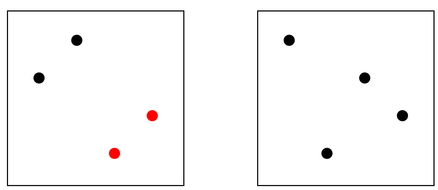
Let $\alpha = 21354$, so that $\mathrm{seq}_3(\alpha) = S$. Then the α-representative of $(\sigma_2)_1$ is $(\alpha)_2$ and the α-representative of $(\sigma_3)_3$ is $(\alpha)_5$. These points are highlighted below in red and blue respectively.



Definition 5.1.21. Let $\sigma = s_1 \dots s_k$ and $\tau = t_1 \dots t_k$ be two permutations of the same length satisfying $\sigma^S = \tau^P$. If $t_k \neq 1$ then we let i be such that $t_i = t_k - 1$, and if $t_k \neq k$ then we let j be such that $t_j = t_k + 1$. We then define the *right-insertion interval* for the pair (σ, τ) to be the interval (s_{i+1}, s_{j+1}) in σ. By convention if $t_k = 1$ then we instead define this interval to be $(0, s_{j+1})$ and if $t_k = k$ then we define it to be $(s_{i+1}, k+1)$.

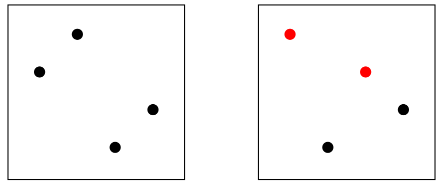
Intuitively, one can take a permutation whose last k points are order isomorphic to σ and 'extend' it to one whose last k points are order isomorphic to τ by inserting a point anywhere in the interval (s_{i+1}, s_{j+1}).

Example 5.1.22. Let $\sigma = 3412$ and $\tau = 4132$. Then the right-insertion interval for the pair (σ, τ) is the interval (1,2) in σ. We highlight its endpoints in red below.



Definition 5.1.23. Let $\sigma = s_1 \dots s_k$ and $\tau = t_1 \dots t_k$ be two permutations of the same length satisfying $\sigma^S = \tau^P$. If $s_k \neq 1$ then we let i be such that $s_i = s_k - 1$, and if $s_k \neq k$ then we let j be such that $s_j = s_1 + 1$. We then define the *left-insertion interval* for the pair (σ, τ) to be the interval (t_{i-1}, t_{j-1}) in τ. By convention if $t_k = 1$ then we instead define this interval to be $(0, t_{j-1})$ and if $t_k = k$ then we define it to be $(t_{i-1}, k+1)$.

Example 5.1.24. Again let $\sigma = 3412$ and $\tau = 4132$. Then the left-insertion interval for the pair (σ, τ) is the interval (3,4) in σ. We highlight its endpoints in red below.



Results

We now present our results pertaining to ambiguity. The end goal will be to show that the length of minimal ambiguous paths in a given graph is bounded, from which we will deduce that it is decidable if a given graph has ambiguous paths.

Lemma 5.1.25. *Let $S = (\sigma_1, \ldots, \sigma_n)$ be an overlapping k-sequence such that the sequence $S' = (\sigma_1, \ldots, \sigma_{n-1})$ is unambiguous. Let $\alpha = a_1 \ldots a_m$ be the unique permutation such that $\mathrm{seq}_k(\alpha) = S'$ and let J be the right-insertion interval for the pair (σ_{n-1}, σ_n). Then S is ambiguous if and only if J_α is non-empty.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose that J_α is empty and that β and δ are permutations such that $\mathrm{seq}_k(\beta) = \mathrm{seq}_k(\delta) = S$. We will prove that β and δ are order isomorphic and hence equal.

Write $\beta = b_1 \ldots b_{m+1}$ and $\delta = c_1 \ldots c_{m+1}$. W note that the sequences $\beta' = b_1 \ldots b_m$ and $\delta' = c_1 \ldots c_m$ are order isomorphic. This is because $S(\rho(\beta')) = S(\rho(\delta')) = S'$, and since S' is unambiguous we have $\rho(\beta') = \rho(\delta') = \alpha$. Hence for $i, j \in \{1, \ldots m\}$ we have $b_i < b_j$ if and only if $c_i < c_j$, so we just have to prove that $b_i < b_{m+1}$ if and only if $c_i < c_{m+1}$.

Write $J_\beta = (x, y)$ and $J_\delta = (z, t)$. Since J_α is empty and $\rho(\beta') = \rho(\delta') = \alpha$, we see that b_{m+1} is the only point in β between x and y and that c_{m+1} is the only point in δ between z and t. Hence we have

$$b_i < b_{m+1} \Leftrightarrow b_i \le x \Leftrightarrow c_i \le z \Leftrightarrow c_i < c_{m+1}.$$

($\Leftarrow$) Suppose J_α contains the point a_j. Let $b_1 \dots b_m$ be the sequence given by

$$b_i = \begin{cases} a_i + 1 & \text{if } a_i > a_j \\ a_i & \text{if } a_i \le a_j \end{cases}$$

then let $b_{m+1} = a_j + 1$. Also let $c_1 \dots c_m$ be the sequence given by

$$c_i = \begin{cases} a_i + 1 & \text{if } a_i \ge a_j \\ a_i & \text{if } a_i < a_j \end{cases}$$

and let $c_{m+1} = a_j$. Then $\beta = b_1 \dots b_{m+1}$ and $\delta = c_1 \dots c_{m+1}$ are two distinct permutations and we have $\mathrm{seq}_k(\beta) = \mathrm{seq}_k(\delta) = S$, so S is ambiguous. □

By symmetry with Lemma 5.1.25 we can immediately observe the following result, which we state without proof:

Lemma 5.1.26. *Let $S = (\sigma_1, \dots, \sigma_n)$ be an overlapping k-sequence such that the sequence $S'' = (\sigma_2, \dots, \sigma_n)$ is unambiguous. Let $\alpha = a_1 \dots a_m$ be the unique permutation such that $\mathrm{seq}_k(\alpha) = S''$ and let I be the left-insertion interval for the pair (σ_1, σ_2). Then S is ambiguous if and only if I_α is non-empty.*

The key insight for deciding ambiguity is the following:

Theorem 5.1.27. *Let $S = (\sigma_1, \dots, \sigma_n)$ be an overlapping k-sequence with $n \ge 3$ such that $S' = (\sigma_1, \dots, \sigma_{n-1})$ and $S'' = (\sigma_2, \dots, \sigma_n)$ are unambiguous. Let δ be the unique permutation such that $\mathrm{seq}_k(\delta) = (\sigma_2, \dots, \sigma_{n-1})$. Let I be the left-insertion interval for the pair (σ_1, σ_2) and let J be the right-insertion interval for the pair (σ_{n-1}, σ_n). Then S is ambiguous if and only if $I_\delta = J_\delta$.*

Proof. Throughout we will let $\alpha = a_1 \dots a_m$ be the unique permutation such that $\mathrm{seq}_k(\alpha) = S'$ and $\beta = b_1 \dots b_m$ be the unique permutation such that $\mathrm{seq}_k(\beta) = S''$.

($\Rightarrow$) If S is ambiguous then by Lemma 5.1.25 the interval J_α is non-empty. Since S'' is unambiguous we see that J_δ is empty, and so the only point which can be in J_α is a_1. The intervals I_α and J_α therefore both contain a_1 and so they must overlap, and since a_1 is the only point in J_α we see that J_α is a sub-interval of I_α. Hence also

J_δ is a sub-interval of I_δ. Similarly, by Lemma 5.1.26 the interval I_β is non-empty. Since S' is unambiguous we see that I_δ is empty, and so the only point which can be in I_β is b_m. This means that the intervals I_β and J_β both contain b_m and so they must overlap, and since b_m is the only point in I_β we see that J_β is a sub-interval of I_β. Hence also J_δ is contained in I_δ. The intervals I_δ and J_δ therefore contain each other and so are equal.

($\Leftarrow$) Since $I_\delta = J_\delta$ we see that J_α contains the point a_1. Hence S is ambiguous by Lemma 5.1.25. $\square$

This leads to the following result, which gives the desired bound on the length of minimal ambiguous k-sequences:

Corollary 5.1.28. *If $S = (\sigma_1, \ldots, \sigma_n)$ is a minimal ambiguous overlapping k-sequence then $n \leq k$.*

Proof. If $n = 2$ then the result certainly holds. If $n \geq 3$ then let $\delta = c_1 \ldots c_{n+k-3}$ be the unique permutation such that $\mathrm{seq}_k(\delta) = (\sigma_2, \ldots, \sigma_{n-1})$. Let I be the left-insertion interval for the pair (σ_1, σ_2) and J be the right-insertion interval for the pair (σ_{n-1}, σ_n). Since S is a minimal ambiguous sequence, we have $I_\delta = J_\delta$ by Theorem 5.1.27. Let K denote the interval $I_\delta = J_\delta$. At least one of the endpoints of K is a point[1] in δ, say the point c_i. Since $K = I_\delta$, the point c_i is the δ-representative of a point in σ_2. Moreover, since I is the left-insertion interval for the pair (σ_1, σ_2), the point c_i cannot be the δ-representative of the final point of σ_2, and so the number of points to the left of c_i is at most $k-2$. Similarly since $K = J_\delta$, the point c_i is the δ-representative of a point in σ_{n-1}. Since J is the right-insertion interval for the pair (σ_{n-1}, n), the point c_i cannot be the δ-representative of the first point of σ_{n-1}, and so the number of points to the right of c_i is at most $k-2$. Hence the number of points in δ is at most

$$(k-2) + (k-2) + 1 = 2k - 3.$$

We know there are exactly $n + k - 3$ points in δ and so $n + k - 3 \leq 2k - 3$, meaning that $n \leq k$. $\square$

[1]As opposed to being either 0 or $|\delta| + 1$.

Finally we have:

Corollary 5.1.29. *It is decidable whether $G(C)$ has any ambiguous paths.*

Proof. The graph $G(C)$ has ambiguous paths if and only if it has minimal ambiguous paths. Minimal ambiguous paths in $G(C)$ have bounded length by Corollary 5.1.28, so there are only finitely paths which can possibly fall into this category. We can test whether any given path is ambiguous, so the property is decidable. $\square$

5.1.4 Deciding atomicity

We recall the following theorem from Chapter 2:

Theorem 2.2.36. *Let G be a directed graph which is connected but not strongly connected. Then $\mathcal{P}(G)$ is atomic under the contiguous subpath ordering if and only if G is a bicycle.*

We will therefore explore the impact of ambiguous paths in $G(C)$ when $G(C)$ is a bicycle.

Definition 5.1.30. A path $(p, p_1, \ldots, p_n, q)$ is called $(p,q)-$*simple* if $p, q \notin \{p_1, \ldots, p_n\}$.

We note the following about bicycles:

Observation 5.1.31. *Let G be a bicycle and let p and q be any two vertices of G. Then there is at most one (p,q)-simple path in G.*

We can use this to prove the following:

Lemma 5.1.32. *Suppose that $G(C)$ is a bicycle and has an ambiguous path of length $l \geq 1$. Then the set of paths in the graph $G_{k+l-1}(C)$ is not atomic.*

Proof. We prove that $G_{k+l-1}(C)$ is not a bicycle, and the result will follow from Theorem 2.2.36. Suppose the ambiguous path in question is given by $S = (\sigma_1, \ldots, \sigma_l)$. Then in $G_{k+l-1}(C)$ there are at least two vertices α and β in the set $V(S)$. If in $G(C)$ there is no vertex τ such that (τ, σ_1) is an edge, or if there is no vertex π such that (σ_l, π) is an edge, then there is no path in G_{k+l-1} from α to β or from β to α. Hence the graph $G_{k+l-1}(C)$ is not connected and so is not a bicycle. Suppose then (τ, σ_1)

and (σ_l, π) are edges in $G(C)$. In $G_{k+l-1}(C)$, fix a vertex $\gamma \in V(\tau, \sigma_1, \ldots, \sigma_{l-1})$ and a vertex $\delta \in V(\sigma_2, \ldots, \sigma_l, \pi)$. Then there are two (γ, δ)-simple paths in $G_{k+l-1}(C)$, one which goes via α and one which goes via β. Hence $G_{k+l-1}(C)$ is not a bicycle by Observation 5.1.31. □

Lemma 5.1.33. *Let $k = \max_{\sigma \in B} |\sigma|$, let $l \geq k$ and suppose that S and T are paths in $G_l(C)$ which have no join and that σ and τ are permutations satisfying $\mathrm{seq}_l(\sigma) = S$ and $\mathrm{seq}_l(\tau) = T$. Then σ and τ have no join in C.*

Proof. We prove the contrapositive. Suppose there is a permutation $\pi \in C$ which is a join for σ and τ, so that $\sigma \leq \pi$ and $\tau \leq \pi$. Since $\pi \in C$ we see that $\mathrm{seq}_l(\pi)$ is a path in $G_l(C)$, and so by Lemma 5.1.10 we have $S \leq P$ and $T \leq P$. Hence P is a join for S and T. □

Lemma 5.1.34. *Suppose $G(C)$ has no ambiguous paths and that S and T are paths in $G(C)$ satisfying $S \leq T$. If σ and τ are permutations satisfying $\mathrm{seq}_k(\sigma) = S$ and $\mathrm{seq}_k(\tau) = T$ then $\sigma \leq \tau$.*

Proof. Write $\sigma = s_1 \ldots s_n$ and $\tau = t_1 \ldots t_m$, and write $T = (\tau_1, \ldots, \tau_{m-k+1})$ so that each $\tau_i = \rho(t_i \ldots t_{i+k-1})$. Let j be such that $S = (\tau_j, \ldots, \tau_{j+n-1})$. Then $(\tau_j, \ldots, \tau_{j+n-1}) = \mathrm{seq}_k(t_j \ldots t_{j+n-1})$, and since $G(C)$ has no ambiguous paths we have $\rho(t_j \ldots t_{j+n-1}) = \sigma$, so $\sigma \leq \tau$. □

This gives us:

Lemma 5.1.35. *Suppose that $G(C)$ is a bicycle. Then set $C_{\geq k}$ satisfies the join property if and only if $G(C)$ has no ambiguous paths.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose that $G(C)$ has an ambiguous path of length l. By Lemma 5.1.32 the set of paths in the graph G_{k+l-1} is not atomic, and so there are paths S and T in this graph which have no join. If σ and τ are permutations in $C_{\geq k}$ satisfying $\mathrm{seq}_{k+l-1}(\sigma) = S$ and $\mathrm{seq}_{k+l-1}(\tau) = T$ then by Lemma 5.1.33 the permutations σ and τ have no join in $C_{\geq k}$.

($\Leftarrow$) Let $\sigma, \tau \in C_{\geq k}$ and let $S = \mathrm{seq}_k(\sigma)$ and $T = \mathrm{seq}_k(\tau)$. Since $G(C)$ is a bicycle, the paths S and T have a join P, so that $S \leq P$ and $T \leq P$. Since $G(C)$ has no ambiguous paths we have $\sigma \leq \pi$ and $\tau \leq \pi$ by Lemma 5.1.34. Hence π is a join for σ and τ. $\square$

Finally we have our main theorem:

Theorem 5.1.36. *Let B be a finite set of permutations, $C = \mathrm{Av}(B)$ and $k = \max_{\sigma \in B} |\sigma|$. Then C is atomic if and only if:*

(i) *The graph $G(C)$ is either strongly connected, or is a bicycle and has no ambiguous paths.*

(ii) *For each permutation $\sigma \in C_{<k}$ there is a permutation $\tau \in C_k$ with $\sigma \leq \tau$.*

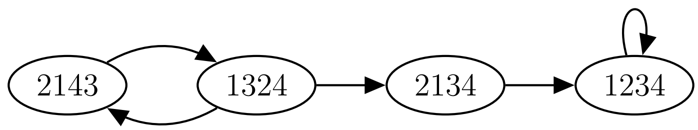
Proof. ($\Rightarrow$) We prove the contrapositive. First suppose condition (i) does not hold. Then by Lemma 5.1.35 the set $C_{\geq k}$ does not satisfy the join property, and so neither does C. Now suppose condition (ii) does not hold. Then there is a permutation $\sigma \in C_{<k}$ such that for each $\tau \in C_k$ we have $\sigma \nleq \tau$. Fix a permutation $\tau \in C_k$ and suppose that σ and τ have a join $\pi \in C$, so that $\sigma \leq \pi$. Then $|\pi| \geq k$ and so there is a permutation $\tau' \in C_k$ such that $\sigma \leq \tau'$ and $\tau' \leq \pi$, which is a contradiction as $\tau' \in C_k$.

($\Leftarrow$) Let $\sigma, \tau \in C$. By condition (ii), there are permutations $\sigma', \tau' \in C_{\geq k}$ such that $\sigma \leq \sigma'$ and $\tau \leq \tau'$. By condition (i) together with Lemmas 5.1.13 and 5.1.35, the set $C_{\geq k}$ satisfies the join property. Hence we can find a join π in $C_{\geq k}$ for σ' and τ', so that $\sigma' \leq \pi$ and $\tau' \leq \pi$. We have $\sigma \leq \sigma' \leq \pi$ and $\tau \leq \tau' \leq \pi$, so $\sigma \leq \pi$ and $\tau \leq \pi$ by transitivity. Hence C satisfies the join property and is atomic. $\square$

Corollary 5.1.37. *Let B be a finite set of permutations. Then it is decidable whether $\mathrm{Av}(B)$ is atomic.*

Proof. The set $\mathrm{Av}(B)$ is atomic if and only if the two conditions in Theorem 5.1.36 hold. By Lemmas 5.1.13 and 5.1.35, condition (i) is the case if and only if $G(C)$ is either strongly connected, or is a bicycle and has no ambiguous paths. It is certainly decidable whether a graph is strongly connected and whether it is a bicycle, and by Corollary 5.1.29 it is decidable whether $G(C)$ has ambiguous paths. Condition (ii) is decidable since there are only finitely many permutations in $C_{\leq k}$, so the result follows. $\square$

Example 5.1.38. Let $B = \{231, 312, 321, 1243, 2134, 3142\}$ and let $C = \mathrm{Av}(B)$. Then $\max_{\sigma \in B} |\sigma| = 4$ and $C_4 = \{1234, 1324, 2134, 2143\}$. The graph $G(C)$ is shown below.



We will show that C is atomic. The graph $G(C)$ is clearly a bicycle. By Lemma 5.1.28, any minimal ambiguous paths in $G(C)$ have length at most 4. The table below lists the paths S of length 4 in $G(C)$ and the corresponding permutation sets $V(S)$. For ease of legibility we have written $\alpha = 2143, \beta = 1324, \gamma = 2134$ and $\delta = 1234$.

S	Members of $V(S)$
$(\alpha, \beta, \alpha, \beta)$	2143657
$(\alpha, \beta, \gamma, \delta)$	2143567
$(\beta, \alpha, \beta, \alpha)$	1325476
$(\beta, \alpha, \beta, \gamma)$	1324657
$(\beta, \gamma, \delta, \delta)$	1324567
$(\gamma, \delta, \delta, \delta)$	2134567
$(\delta, \delta, \delta, \delta)$	1234567

We see from this table that every path of length 4 in $G(C)$ is unambiguous, so $G(C)$ has no minimal ambiguous paths and hence no ambiguous paths. Furthermore, for every permutation σ in the set $C_{<4} = \{1, 12, 21, 123, 213\}$ we have $\sigma \leq 2134$. Hence C is atomic by Theorem 5.1.36.

5.2 WQO in the consecutive involvement ordering

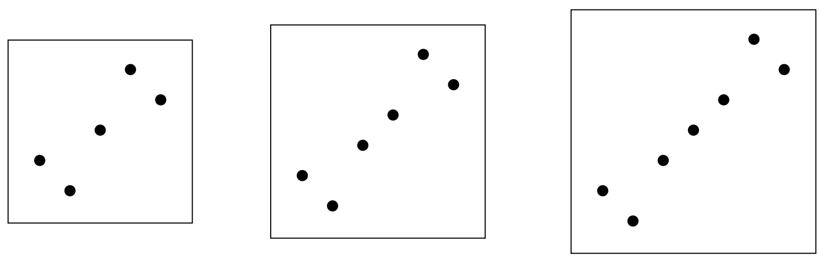
5.2.1 Examples of anti-chains

In the factor ordering on words we observed the following method of constructing an anti-chain: suppose α and β are words and that S is an infinite collection of words such that for all $w \in S$ we have $\alpha \nleq_f \alpha^S w\beta$ and $\beta \nleq_f \alpha w \beta^P$. Then the set $\alpha S \beta$ is an infinite anti-chain under $\leq_f$. Naturally there is a similar construction for the consecutive involvement ordering on permutations, as our next two examples show.

Example 5.2.1. For $n \geq 1$ let

$$\sigma_n = \delta_2 \oplus \alpha_n \oplus \delta_2,$$

and let $M = \{\sigma_n \mid n \geq 1\}$. We show the permutations σ_1, σ_2 and σ_3 below.



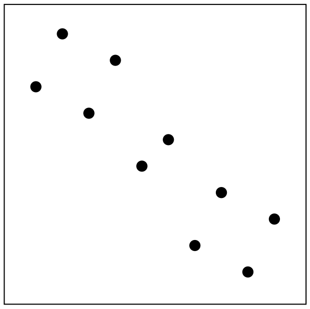
σ_1 $\quad$ σ_2 $\quad$ σ_3

We claim that the set M is an infinite anti-chain. To see this we note that every permutation σ_n in M satisfies both $213 \leq \sigma_n$ and $231 \leq \sigma_n$. If $\tau < \sigma_n$ then either $\tau \leq \sigma_n^P$ or $\tau \leq \sigma_n^S$. In the former case we see that $213 \nleq \tau$ and in the latter we see that $231 \nleq \tau$, meaning that in either case we have $\tau \notin M$.

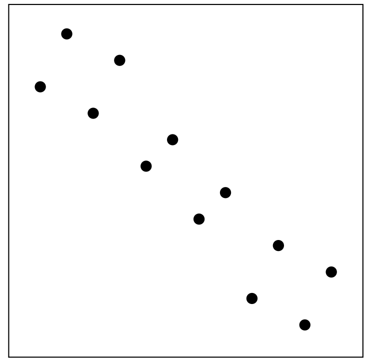
Example 5.2.2. For $n \geq 1$ let

$$\sigma_n = 2413 \ominus \underbrace{21 \ominus 21 \ominus \cdots \ominus 21}_{n \text{ copies of } 21} \ominus 2413,$$

and let $T = \{\sigma_n \mid n \geq 1\}$. We show the permutations τ_1 and τ_2 below.



τ_1



τ_2

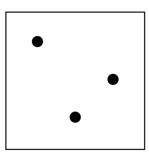
We claim that the set T is an anti-chain. Indeed, if $n \geq 1$ and $\tau < \sigma_n$ then either $35241 \nleq \tau$ or $52413 \nleq \tau$, and so $\tau \notin T$.

For the consecutive involvement ordering, there is another type of anti-chain construction. The incomparability of the permutations built from this method depends on the relationship between their points, and so the construction has no analogue in the factor ordering on words. We will explore this concept in greater detail later in the following section, and for now present a simple example.

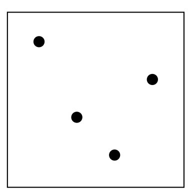
Example 5.2.3. For $n \geq 1$ let

$$\sigma_n = 1 \ominus (\delta_n \oplus 1)$$

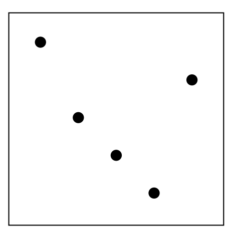
and let $J = \{\sigma_n \mid n \geq 1\}$. We show the permutations σ_1, σ_2 and σ_3 below.



σ_1



σ_2



σ_3

We claim that the set J is an anti-chain. Indeed, if $n \geq 1$ and $\tau < \sigma_n$ then either τ is a descent or the final point of τ lies above every other point, and in either case we have $\tau \notin J$.

5.2.2 Deciding WQO

We recall the following result from Chapter 2:

Proposition 2.2.46. *Let G be a directed graph. The following are equivalent:*

(i) *The graph G has no in-out cycles.*

(ii) *The graph G is a union of bicycles.*

(iii) *The set $\mathcal{P}(G)$ is WQO under the contiguous subpath ordering.*

Our approach for deciding WQO for the consecutive involvement ordering will be as follows. Let C be a finitely-based set of permutations and suppose $G(C)$ can be expressed as a union of bicycles $B_1, \ldots, B_n$. Let P_i denote the set of permutations in C arising from each B_i, so that

$$C = \bigcup_{i=1}^{n} P_i.$$

Then C will be WQO if and only if each P_i is WQO. Our approach will therefore be to develop a procedure to determine whether each P_i is WQO, by considering each bicycle B_i. This will rely on an analysis of the ambiguous paths in B_i, and an understanding of the structure of permutations arising from cycles.

Ambiguous cycles

Our first step will be to prove that if $G(C)$ has a cycle containing an ambiguous path (an *ambiguous cycle*) then C is not WQO.

Notation 5.2.4. Hereafter, if S is a cycle with edges

$$(\sigma_1, \sigma_2), \ldots, (\sigma_{n-1}, \sigma_n), (\sigma_n, \sigma_1)$$

then we will describe S using the notation $S = (\sigma_1, \ldots, \sigma_n)$, rather than the notation $S = (\sigma_1, \ldots, \sigma_n, \sigma_1)$ which we used before.

Definition 5.2.5. Let $S = (s_1, \ldots, s_n)$ be a sequence, let $i \in \{1, \ldots, n\}$ and let $l \geq 1$. We define $S_{i,l}$ to be the sequence of length l whose first element is s_i and whose remaining elements are the successive elements of S in cyclic order.

Example 5.2.6. Let $S = (a, b, c, d)$. Then $S_{3,8}$ starts with the 3^{rd} element of S, i.e. c, and has length 8, so is given by $S_{3,8} = (c, d, a, b, c, d, a, b)$.

We note the following without proof:

Lemma 5.2.7. *Suppose $G(C)$ contains a cycle $S = (\sigma_1, \dots, \sigma_n)$. Then for every $l \geq 1$ and for every choice of permutations*

$$\tau_1 \in V(S_{1,l}), \dots, \tau_n \in V(S_{n,l}),$$

there is a cycle in G_{k+l-1} given by $(\tau_1, \dots, \tau_n)$.

Example 5.2.8. Suppose $G(C)$ contains the cycle $S = (\sigma_1, \sigma_2, \sigma_3)$ where $\sigma_1 = 3412$, $\sigma_2 = 3124$ and $\sigma_3 = 2341$. Then $S_{1,3} = (\sigma_1, \sigma_2, \sigma_3)$, $S_{2,3} = (\sigma_2, \sigma_3, \sigma_1)$ and $S_{3,3} = (\sigma_3, \sigma_1, \sigma_2)$. We have:

- $V(S_{1,3}) = \{\tau_1\} = \{452361\}$,
- $V(S_{2,3}) = \{\tau_2, \tau_2'\} = \{524613, 534612\}$,
- $V(S_{3,3}) = \{\tau_3, \tau_3'\} = \{245136, 345126\}$,

so G_6 will contain the cycles (τ_1, τ_2, τ_3), $(\tau_1, \tau_2', \tau_3)$, $(\tau_1, \tau_2, \tau_3')$ and $(\tau_1, \tau_2', \tau_3')$.

We then have:

Lemma 5.2.9. *Suppose $G(C)$ contains an ambiguous cycle. Then C is not WQO.*

Proof. Suppose that the cycle in question is given by $S = (\sigma_1, \dots, \sigma_n)$ and let P be an ambiguous path starting and ending on S. Suppose that P starts on σ_i and has length l, so that $P = S_{i,l}$. Then in $G_{k+l-1}(C)$ there are at least two vertices $\alpha, \beta \in V(P)$. Select vertices

$$\tau_1, \dots, \tau_{i-1}, \tau_{i+1}, \dots, \tau_n$$

in G_{k+l-1} where each $\tau_j \in V(S_{j,l})$. By Lemma 5.2.7 we see that

$$(\tau_1, \dots, \tau_{i-1}, \alpha, \tau_{i+1}, \dots, \tau_n)$$

is a cycle in $G_{k+l-1}(C)$. This cycle has an entrance (β, τ_{i+1}) and an exit (τ_{i-1}, β), so is an in-out cycle. Hence by Proposition 2.2.46 the set of paths in $G_{k+l-1}(C)$ is not WQO, and so neither is C. □

From this point on, we will therefore focus on graphs $G(C)$ whose cycles do not admit any ambiguous paths.

Bicycles and juxtapositions

We discussed bicycles in Chapter 2 and in the previous section of the current chapter. Since we have changed our notation for cycles we will also introduce an adjusted definition of a bicycle.

Definition 5.2.10. A *bicycle* is a directed graph B consisting of two simple cycles $S = (\sigma_1, \ldots, \sigma_n)$ and $E = (\pi_1, \ldots, \pi_m)$, and a path $P = (\sigma_n, \tau_1, \ldots, \tau_l, \pi_1)$ from S to E. We will describe B as an ordered triple $B = (S, P, E)$.

Below is an illustration of a bicycle as described above.

$$\begin{tikzcd}
 & \sigma_{n-1} \arrow[dr, bend left] & & & & & & \pi_2 \arrow[dr, bend left] & \\
\vdots \arrow[ur, bend left] & & \sigma_n \arrow[r] \arrow[dl, bend left] & \tau_1 \arrow[r] & \cdots \arrow[r] & \tau_l \arrow[r] & \pi_1 \arrow[ur, bend left] & & \vdots \arrow[dl, bend left] \\
 & \sigma_1 \arrow[ul, bend left] & & & & & & \pi_m \arrow[ul, bend left] &
\end{tikzcd}$$

Hererafter, if α, β are permutations with $\alpha \le \beta$ then we will say α is a *factor* of β. If β can be expressed as $\beta = \alpha'\gamma$ where $\rho(\alpha') = \alpha$ then we will say α is a *prefix* of β, and if β can be expressed as $\beta = \gamma\alpha'$ where $\rho(\alpha') = \alpha$ then we will say α is a *suffix* of β.

We now give some motivation for the work which is to follow. Suppose that $G(C)$ has no ambiguous cycles and let $B = (S, P, E)$ be a bicycle in $G(C)$. Then we can find permutations α and γ such that every permutation arising from B is a factor of a permutation of the form

$$\alpha_i \ldots \alpha_1 \beta \gamma_1 \ldots \gamma_j$$

where $\alpha_i, \ldots, \alpha_1$ and $\gamma_1, \ldots, \gamma_j$ are sequences satisfying $\rho(\alpha_i) = \cdots = \rho(\alpha_1) = \alpha$ and

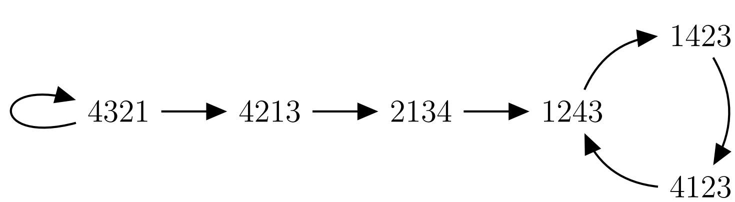
$\rho(\gamma_1) = \cdots = \rho(\gamma_j) = \gamma$. Moreover, any sequence $\gamma_1 \ldots \gamma_j$ as described above can be constructed by juxtaposing some number of copies of γ next to one another, according to some fixed rule. This rule will take the form of a permutation $\phi = \gamma_1\gamma_2$ where γ_1 and γ_2 are sequences satisfying $\rho(\gamma_1) = \rho(\gamma_2) = \gamma$. This will enable us to construct every such sequence $\gamma_1 \ldots \gamma_j$, because γ corresponds to a path in a cycle of $G(C)$, and there will only be one way to juxtapose two copies of γ in C, since $G(C)$ has no ambiguous cycles. Similarly, to construct sequences $\alpha_i \ldots \alpha_1$ as described above, we will use a particular permutation $\theta = \alpha_2\alpha_1$ where α_1 and α_2 are sequences satisfying $\rho(\alpha_2) = \rho(\alpha_1) = \alpha$.

Specifically, suppose that $S = (\sigma_1, \ldots, \sigma_n)$ and $E = (\pi_1, \ldots, \pi_m)$, and let k be the maximum length of a basis element for C. In order to determine the permutations γ and ϕ corresponding to E we will let j be minimal such that $mj \geq k$ and

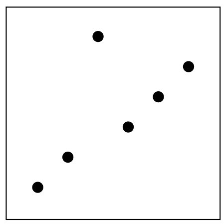
$$2mj \geq l + k - 1.$$

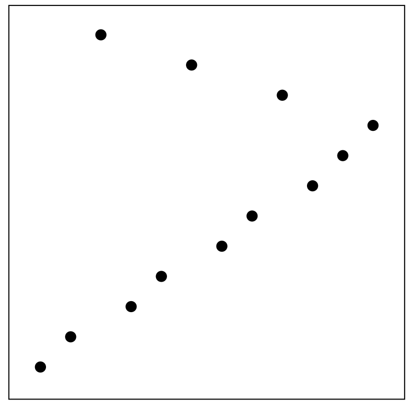
We will then select γ to be the length mj permutation whose path starts at π_1 and ϕ to be the length $2mj$ permutation whose path starts at π_1. The restriction $2mj \geq l+k-1$ is chosen so that ϕ will have every permutation from E as a factor, and the restriction $mj \geq k$ is chosen so that γ will have π_1 as a prefix. The permutation γ will be called the *generator* for the cycle E, and ϕ will be called the *γ-juxtaposition rule*. Likewise, in order to determine the permutations α and θ corresponding to the cycle S we will let i be minimal such that $ni \geq k$ and $2ni \geq n + k - 1$. We will then select α to be the length ni permutation whose path ends at σ_n and θ to be the length $2ni$ permutation whose path ends at σ_n. The permutation α will be called the generator for the cycle S and θ will be called the α-juxtaposition rule.

Example 5.2.11. Let $B = (S, P, E)$ where $S = (4321)$, $P = (4321, 4213, 2134, 1243)$ and $E = (1243, 1423, 4123)$. We show B below.



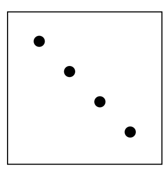
We have $k = 4$ and the length of E is $m = 3$, so the smallest j such that $mj \geq k$ and $2mj \geq m + k - 1$ is $j = 2$. Hence the generator for E is $\gamma = 126345$ and the γ-juxtaposition rule is $\phi = (1, 2, 12, 3, 4, 11, 5, 6, 10, 7, 8, 9)$. These are shown below.

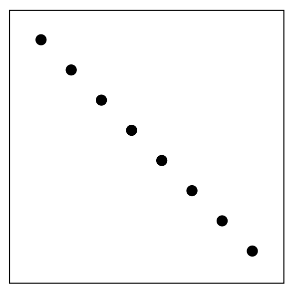




$\gamma = 126345$ $\phi = (1, 2, 12, 3, 4, 11, 5, 6, 10, 7, 8, 9)$

The length of S is $n = 1$, so the smallest i such that $ni \geq k$ and $2ni \geq n + k - 1$ is $i = 1$. Hence the generator for S is $\alpha = 4321$ and the α-juxtaposition rule is $\theta = 87654321$. These are shown below.





$\alpha = 4321$ $\theta = 87654321$

Intervals

In order to understand the structure of large permutations arising from a given bicycle $B = (S, P, E)$, we will consider the relative positioning of intervals within the two copies of γ in ϕ, and within the two copies of α in θ. We will largely prove results for the permutations γ and ϕ arising from E and simply state the corresponding results for the permutations α and ϕ arising from S, as the arguments will be analogous.

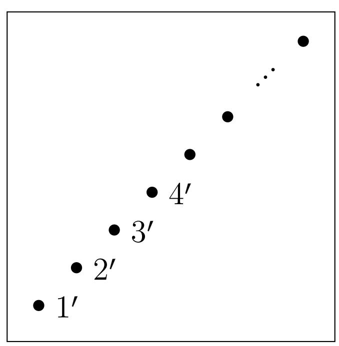
Definition 5.2.12. Let $B = (S, P, E)$ be a bicycle and write $S = (\sigma_1, \dots, \sigma_n)$ and $E = (\pi_1, \dots, \pi_m)$. A minimal interval I of π_1 is *unbounded* if for every $l \geq 1$ there is a permutation δ whose path starts at π_1 such that the δ-representative of the first copy of I contains at least l points. Likewise, a minimal interval I of σ_n is unbounded if for every $l \geq 1$ there is a permutation δ whose path ends at σ_n such that δ-representative of the final copy of I contains at least l points.

Since π_1 is a prefix of γ, we can extend the above definition to minimal intervals of γ, where a minimal interval I of γ is unbounded if and only if the corresponding interval of π_1 is unbounded. We can also extend this definition to α in an analogous fashion.

Example 5.2.13. Let $S = (3241, 2314)$, $P = (2314, 2134, 1234)$ and $E = (1234)$ and consider the bicycle $B = (S, P, E)$. This is shown below.

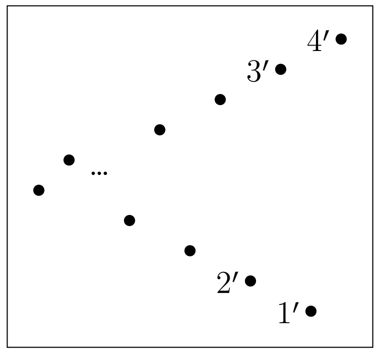
$$\begin{tikzcd} 3241 \arrow[r, bend left] & 2314 \arrow[l, bend left] \arrow[r] & 2134 \arrow[r] & 1234 \arrow[loop above] \end{tikzcd}$$

We first consider the minimal intervals of $\pi_1 = 1234$. If δ is a permutation whose path begins at π_1 then δ is an ascent α_n for some $n \geq 4$. We sketch a typical δ below, where the δ-representatives of the points from π_1 are highlighted.



We can see that for any such δ, the δ-representatives of the intervals $(0,1)$, $(1,2)$, $(2,3)$ and $(3,4)$ in π_1 will be empty. On the other hand, for any $n \geq 1$ there is clearly a choice of δ such that the δ-representative of the interval $(4,5)$ in π_1 contains at least n points, namely $\delta = \alpha_{m+4}$, so the interval $(4,5)$ is unbounded.

We now consider the minimal intervals of $\sigma_2 = (2314)$. Any permutation δ whose path ends at σ_2 is a suffix of a permutation of the form shown in the sketch below, where we have highlighted the representatives of the points from σ_2.



We can see that for any such δ, the δ-representatives of the intervals $(0,1)$, $(1,2)$, $(3,4)$ and $(4,5)$ in σ_2 will be empty. On the other hand, for any $n \geq 1$ there is clearly a choice of δ such that the δ-representative of the interval $(2,3)$ in σ_2 contains at least n points, namely the permutation arising from $\lceil \frac{n}{2} \rceil$ iterations of S, so the interval $(2,3)$ is unbounded.

Definition 5.2.14. Let $I = (x,y)$ and $J = (z,t)$ be distinct intervals. We say I *contains* J if $x \leq z$ and $t \leq y$, and denote this by $I > J$. We say I and J have a *partial overlap* if either $x < z < y < t$ or $z < x < t < y$, and denote this by $I \sim J$. We will let $\gtrsim$ denote the union of the relations $>$ and $\sim$, so that $I \gtrsim J$ if I contains or has a partial overlap with J.

Observation 5.2.15. *If $I = (x,y)$ and $J = (z,t)$ then $I \gtrsim J$ if and only if $x < t$ and $z < y$.*

In the material which follows, for each minimal interval $I = (x,y)$ of γ we will let $I_1 = (x_1, y_1)$ denote the corresponding interval in γ_1 and $I_2 = (x_2, y_2)$ denote the corresponding interval in γ_2.

Lemma 5.2.16. *Suppose that B has no ambiguous cycles and let I be a minimal interval of γ. Then $I_1 \not\gtrsim I_2$.*

Proof. Suppose instead that $I_1 \sim I_2$. We will consider only the case where $x_1 < x_2 < y_1 < y_2$, as the argument for the case where $x_2 < x_1 < y_2 < y_1$ will be analogous. Let $\delta = \gamma_1'\gamma_2'\gamma_3$ be a permutation in C consisting of three copies of γ, so that $\rho(\gamma_1'\gamma_2') = \rho(\gamma_2'\gamma_3) = \phi$. Let x_1', y_1' denote the copies of x, y in γ_1', let x_2', y_2' denote the copies of x, y in γ_2' and let x_3, y_3 denote the copies of x, y in γ_3. Then $x_1 < x_2 < y_1 < y_2$ implies $x_2' < x_3 < y_2' < y_3$, but then x_3 can be chosen to go above or below y_1', as illustrated in Figure 5.1. This yields two different choices for a permutation consisting of three copies of γ, contradicting the assumption that $G(C)$ has no ambiguous cycles. □

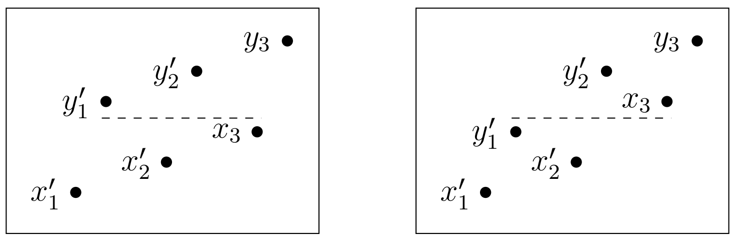


Figure 5.1: A partial overlap leading to an ambiguous cycle.

Definition 5.2.17. Let $I = (x, y)$ and $J = (z, t)$ be intervals. We say that I *lies below* J if $y \leq z$, and that I *lies above* J if $x \geq t$.

Lemma 5.2.18. *Let I, J be minimal intervals of γ such that $I_1 \not\gtrsim I_2$ and $I_1 \gtrsim J_2$. Then $J_1 \not\gtrsim J_2$.*

Proof. Write $I = (x, y)$ and $J = (z, t)$. We have $I_1 \gtrsim J_2$, so by Observation 5.2.15 we have $z_2 < y_1$. We will consider only the case where I lies below J, as the argument for the case where I lies above J will be analogous. Since I lies below J we have $y \leq z$. We will then also have $y_1 \leq z_1$, and combining this with $z_2 < y_1$ we get $z_2 < z_1$, so $J_1 \not\gtrsim J_2$. □

Lemma 5.2.19. *Let I, J, K be minimal intervals of γ such that $I_1 \gtrsim J_2$ and $J_1 \gtrsim K_2$. If I lies below (respectively, above) J then J lies below (respectively, above) K.*

Proof. We will consider only the case where I lies below J, as the argument for the case where I lies above J will be analogous. Write $I = (x, y)$, $J = (z, t)$ and $K = (a, b)$. Using the same argument as in the proof for Lemma 5.2.18, we have

$z_2 < z_1$. By Lemma 5.2.16 we have $J_1 \not\sim J_2$, so $z_2 < z_1$ implies $t_2 < z_1$. Since $J_1 \gtrsim K_2$, we have $z_1 < b_2$ by Observation 5.2.15. Combining $t_2 < z_1$ and $z_1 < b_2$ we get $t_2 < b_2$, so also $t < b$. Since a is directly below b in γ we get $t \leq a$, so J lies below K. $\square$

Definition 5.2.20. Let $A = (I^{(1)}, \dots, I^{(n)})$ be a sequence of intervals. We say A is *increasing* if $I^{(i)}$ lies below $I^{(i+1)}$ for all $i \in \{1, \dots, n-1\}$, and *decreasing* if $I^{(i)}$ lies above $I^{(i+1)}$ for all $i \in \{1, \dots, n-1\}$. We say A is *monotone* if it is either increasing or decreasing.

Lemma 5.2.21. *Let $A = (I^{(1)}, \dots, I^{(n)})$ be a sequence of minimal intervals from γ. Suppose that $I_1^{(1)} \not\gtrsim I_2^{(1)}$ and that $I_1^{(i)} \gtrsim I_2^{(i+1)}$ for all $i \in \{1, \dots, n-1\}$. Then the sequence A is monotone.*

Proof. We proceed by induction on n. If $n = 2$ then the result holds from Lemma 5.2.16 and if $n = 3$ then the result holds from Lemma 5.2.19. Now let $n \geq 4$ and suppose the result holds for all such sequences of length $n-1$. Clearly $B = (I^{(1)}, \dots, I^{(n-1)})$ is such a sequence. From Lemma 5.2.18 we have that $I_1^{(2)} \not\gtrsim I_2^{(2)}$, so $C = (I^{(2)}, \dots, I^{(n)})$ is also such a sequence. By induction the sequences B and C are both monotone. If $I^{(2)}$ lies below $I^{(3)}$ then B and C are both increasing, so A is also increasing. Similarly, if $I^{(2)}$ lies above $I^{(3)}$ then B and C are both decreasing, so A is also decreasing. Hence in either case A is monotone. $\square$

Lemma 5.2.22. *Let I be a minimal interval of γ. Then I is unbounded if and only if $I_1 > I_2$.*

Proof. ($\Rightarrow$) We prove the contrapositive. Suppose that $I_1 \not\gtrsim I_2$. Let δ be a permutation whose path starts at π_1 and let I' denote the δ-representative of I. Then every point in I' arises from a sequence $A = (I^{(1)}, \dots, I^{(n)})$ of minimal intervals from γ where $I^{(1)} = I$ and where $I_1^{(i)} \gtrsim I_2^{(i+1)}$ for all $i \in \{1, \dots, n-1\}$. We have $I_1 \not\gtrsim I_2$, so by Lemma 5.2.21 every such sequence is monotone, and therefore has bounded length. Hence there is a bound on the number of such sequences, and therefore a bound on the number of points which lie in I'.

($\Leftarrow$) Suppose that $I_1 > I_2$. Then for any n it is possible to construct a permutation δ consisting of n copies of γ given by $\gamma_1', \dots, \gamma_n'$ where each γ_i' has a copy of I given by I_i'. We will then have $I_1' > \dots > I_n'$ so the interval I_1' contains at least n points. $\square$

Definition 5.2.23. Let $B = (S, P, E)$ be a bicycle, write $E = (\pi_1, \ldots, \pi_m)$ and let I be a minimal interval of π_1. Suppose there is a path $Q = (\beta_1, \ldots, \beta_l, \pi_1)$ and a permutation δ corresponding to Q such that δ_I overlaps with the δ-representative of the right-insertion interval for the pair (β_1, β_2), or for (β_1, π_1) if $l = 1$. Then we say that I is *inserted into on the left*, and that Q is a *left-insertion path* for I.

We note that since the length of a minimal ambiguous path at most k, a minimal interval of π_1 will be inserted into on the left if and only if it has a left-insertion path of length at most k. We also make the following analogous definition:

Definition 5.2.24. Let $B = (S, P, E)$ be a bicycle, write $S = (\sigma_1, \ldots, \sigma_n)$ and let I be a minimal interval of σ_n. Suppose there is a path $Q = (\sigma_n, \beta_1, \ldots, \beta_l)$ and a permutation δ corresponding to Q such that δ_I overlaps with the δ representative of the left-insertion interval for the pair (β_{l-1}, β_l), or for (σ_n, β_1) if $l = 1$. Then we say that I is *inserted into on the right*, and that Q is a *right-insertion path* for I.

Again, a minimal interval of σ_n will be inserted into on the right if and only it has a right-insertion path of length at most k.

Lemma 5.2.25. *Let $B = (S, P, E)$ be a bicycle and write $E = (\pi_1, \ldots, \pi_m)$. Suppose π_1 has an unbounded interval which is inserted into on the left. Then C contains an infinite anti-chain.*

Proof. Let I' be the interval in question, and let I denote the corresponding interval in γ. Since I' is unbounded, so too is I. Let $Q = (\beta_1, \ldots, \beta_l, \pi_1)$ be a left-insertion path for I'. Let $\tau_1, \tau_2, \ldots$ be an infinite sequence of permutations such that each τ_i traverses the path $Q' = (\beta_2, \ldots, \beta_l)$ and then features i copies of γ. Then each τ_i contains i nested copies of the interval I which we can label as

$$I_{i,1} > I_{i,2} > \cdots > I_{i,i}.$$

We will consider only the case where $I \neq (k, k+1)$, as the argument for the case where $I = (k, k+1)$ will be analogous, since then $I \neq (0, 1)$. For each i let τ'_i be the permutation obtained from τ_i by inserting a point directly above the second endpoint of $I_{i,i}$. This is sketched in Figure 5.2. We will show that the sequence $\tau'_1, \tau'_2, \ldots$ is an infinite anti-chain. Suppose instead that $\tau'_i < \tau'_j$ for some i, j. Then by Lemma 5.1.10 we have $\mathrm{seq}_k(\tau'_i) < \mathrm{seq}_k(\tau'_j)$. Both $\mathrm{seq}_k(\tau'_i)$ and $\mathrm{seq}_k(\tau'_j)$ have the sequence Q' as a prefix, and moreover this is the only occurrence of Q' in either of these sequences, so $\mathrm{seq}_k(\tau'_i)$ must be a prefix of $\mathrm{seq}_k(\tau'_j)$. Hence, as permutations, τ'_i must also be a prefix of τ'_j. However, the first point of τ'_i lies above the second endpoint p of $I_{i,i}$ whereas the first point of τ'_j lies below the second endpoint of q of $I_{j,i}$. Write $\tau'_i = s_1 \ldots s_N$ and $\tau'_j = t_1 \ldots t_M$ and let l be such that $s_l = p$ and $t_l = q$. Then $s_1 > s_l$ and $t_1 < t_l$, so τ'_i is not a prefix of τ'_j, which is a contradiction. $\square$

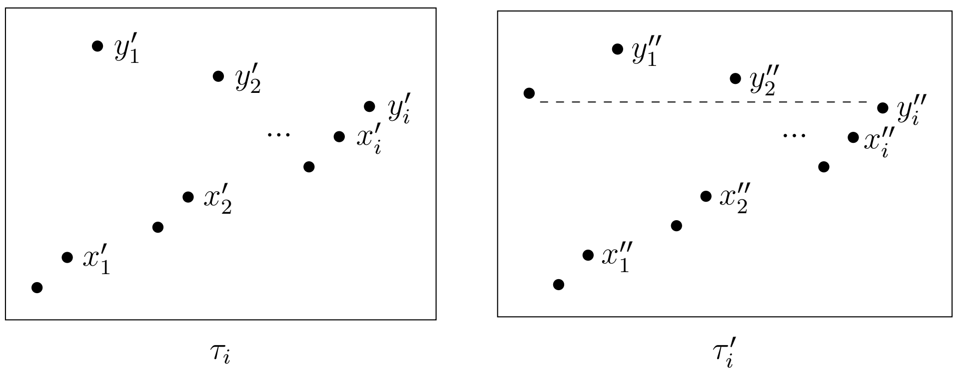


Figure 5.2: Inserting a point into τ_i to construct τ'_i.

By symmetry with the previous lemma we can state the following:

Lemma 5.2.26. *Let $B = (S, P, E)$ be a bicycle and write $S = (\sigma_1, \ldots, \sigma_n)$. Suppose σ_n has an unbounded interval which is inserted into on the right. Then C contains an infinite anti-chain.*

In the work which follows we will be considering permutations of the form

$$\sigma = \alpha'_m \ldots \alpha'_1 \beta \gamma'_1 \ldots \gamma'_n$$

arising from a particular bicycle B. We wish to be able to refer to the positions of σ in terms of the various copies of α and γ, so that we can refer to a point in position p and know that it means, say, the point in position 2 of γ'_3. We can then make statements along the lines of "for all positions p and for all permutations σ arising from B..." without referring to the various permutations separately. We will therefore adopt the convention of referring to the positions of σ as follows:

- The point in position j of α_i' will be said to be in position $(-i, j)$ of σ.
- The point in position j of β will be said to be in position $(0, j)$ of σ.
- The point in position j of γ_i' will be said to be in position (i, j) of σ.

These positions will then be ordered in lexicographic order, i.e. if $p = (i_1, j_1)$ and $q = (i_2, j_2)$ are positions then $p < q$ either if $i_1 < i_2$ or if $i_1 = i_2$ and $j_1 < j_2$.

Lemma 5.2.27. *Let $B = (S, P, E)$ be a bicycle and let γ be the generator for E. Suppose that the first permutation in E has no unbounded interval which is inserted into on the left. Then there is an integer $N \geq 1$ such that for every point x of γ, for every position $p < (1, 1)$ and for every permutation σ arising from B, either $(\sigma)_p < x_n$ for all $n \geq N$ or $(\sigma)_p > x_n$ for all $n \geq N$.*

Proof. We prove the contrapositive. Suppose that for every $N \geq 1$ there is a point x of γ, a position $p < (1, 1)$ and a permutation σ arising from B such that either $x_N < (\sigma)_p < x_{N+1}$ or $x_N > (\sigma)_p > x_{N+1}$. Then one of the minimal intervals of γ having x as an endpoint can contain arbitrarily many copies of x and is hence unbounded, and is inserted into on the left by the point in position p. $\square$

Intuitively, the above lemma tells us that given a particular position p and a point x in γ, after a certain point all copies of x will be below the point in position p or they will all be above it. By symmetry we can state the following:

Lemma 5.2.28. *Let $B = (S, P, E)$ be a bicycle and let α be the generator for S. Suppose that the last permutation in S has no unbounded interval which is inserted into on the right. Then there is an integer $M \geq 1$ such that for every point x of α, for position $p > (-1, k)$ and for every permutation σ arising from B, either $(\sigma)_p < x_m$ for all $m \geq M$ or $(\sigma)_p > x_m$ for all $m \geq M$.*

In the proof of the following lemma we will write $\sigma \cong \tau$ to mean that σ and τ are order isomorphic.

Lemma 5.2.29. *Suppose that $G(C)$ has no ambiguous cycles and let $B = (S, P, E)$ be a bicycle in $G(C)$. Suppose that the first permutation in E has no unbounded* *interval which is inserted into on the left and that the last permutation in S has no unbounded interval which is inserted into on the right. Then the set of permutations arising from B is WQO.*

Proof. Let X be an infinite anti-chain arising from B, and let N and M be as in the statements of Lemmas 5.2.27 and 5.2.28 respectively. There are three cases to consider, which are:

(i) There are permutations in X with arbitrarily many copies of both α and γ.

(ii) There is a bound on the number of copies of α which a permutation in X has.

(iii) There is a bound on the number of copies of γ which a permutation in X has.

We will consider only case (i) as the others will be analogous. Since any infinite subset of X will also be an infinite anti-chain, we can assume that every permutation in X has at least N copies of α and at least M copies of γ. Every permutation $\sigma \in X$ has a factor of the form

$$\alpha'_M \dots \alpha'_1 \beta \gamma'_1 \dots \gamma'_N$$

where $\rho(\alpha'_M) = \dots = \rho(\alpha_1) = \alpha$ and $\rho(\gamma'_1) = \dots = \rho(\gamma'_N) = \gamma$. We will call this the *core factor* of σ, and denote it by σ^C. Since σ^C is a finite sequence of a fixed length, we may assume that $\sigma^C \cong \tau^C$ for all $\sigma, \tau \in X$. Write $X = \{\sigma_1, \sigma_2, \dots\}$ and for $i \geq 1$ write

$$\sigma_i = \delta_i {\sigma_i}^C \varepsilon_i$$

Neither of the sequences $|\delta_1|, |\delta_2|, \dots$ or $|\varepsilon_1|, |\varepsilon_2|, \dots$ can be strictly decreasing, so we can find i, j such that $|\delta_i| < |\delta_j|$ and $|\varepsilon_i| < |\varepsilon_j|$. Our goal is prove that $\sigma_i < \sigma_j$, and so X is not an anti-chain. Let δ'_j denote the length $|\delta_i|$ suffix of δ_j, let ε'_j denote the length $|\varepsilon_i|$ prefix of δ_j and let

$$\sigma'_j = \delta'_j \sigma_j^C \varepsilon'_j.$$

We will prove that $\sigma_i \cong {\sigma_j}'$. For each point a of σ_i we will let a' denote the point in the corresponding position of σ_j. Hence we wish to prove that $a < b$ if and only if $a' < b'$. We note that $\delta_i \cong \delta'_j$ since S is unambiguous, that $\sigma_i^C \cong \sigma_j^C$ by our choice of X and that $\varepsilon_i \cong \varepsilon'_j$ since E is unambiguous. Hence if a, b are both in δ_i, both in σ_i^C

or both in ε_i then $a < b$ if and only if $a' < b'$. Now suppose that a is in δ_i and b is in σ_i^C. Let x be the point of α such that $a = x_m$ for some $m \geq M$. We than have:

$$a < b \Leftrightarrow x_M < b \Leftrightarrow x'_M < b' \Leftrightarrow a' < b'.$$

The argument for the case where a is in σ_i^C and b is in ε_i is analogous. Finally suppose that a is in δ_i and b is in ε_i. Let x be the point of α such that $a = x_m$ for some $m \geq M$ and let y be the point of γ such that $b = y_n$ for some $n \geq N$. Then let p be the position of x and q be the position of y. Then $p < (1,1)$ and $q > (-1,k)$ so the points in positions p and q are in a fixed position relative to one another for every permutation arising from B, hence $a < b$ if and only if $a' < b'$. □

Below is our main theorem on WQO for the consecutive involvement ordering.

Theorem 5.2.30. *Let C be a set of permutations which is finitely-based under the consecutive involvement ordering. Then C is WQO if and only if:*

(i) *The graph $G(C)$ has no in-out cycles.*

(ii) *The graph $G(C)$ has no ambiguous cycles.*

(iii) *For every bicycle $B = (S, P, E)$ in $G(C)$, the last permutation in S has no unbounded interval which is inserted into on the right, and the first permutation in E has no unbounded interval which is inserted into on the left.*

Proof. This is a combination of Lemmas 5.2.9, 5.2.25, 5.2.26 and 5.2.29, together with Proposition 2.2.46. □

Corollary 5.2.31. *Let C be a set of permutations which is finitely-based under the consecutive involvement ordering. Then it is decidable whether C is WQO.*

Proof. It is certainly decidable whether $G(C)$ has no in-out cycles, and it is decidable whether $G(C)$ has ambiguous cycles by Corollary 5.1.29. It is decidable whether a given interval is unbounded by Lemma 5.2.22, and whether it is inserted into since minimal ambiguous sequences have bounded length. Hence the result follows from Theorem 5.2.30. □

Bibliography

[1] E. Aichinger, P. Mayr, and R. McKenzie. On the Number of Finite Algebraic Structures. *Journal of the European Mathematical Society,* 16(8):1673–1686, 2013.

[2] M. Atkinson, M. Murphy and N. Ruškuc. Partially well-ordered closed sets of permutations. *Order,* (19):101–113, 2002.

[3] A. Atminas, V. Lozin and M. Moshkov. Deciding WQO for Factorial Languages. *Lecture Notes in Computer Science,* 7810:68–79, 2013.

[4] S. Avgustinovich and S. Kitaev. On uniquely k-determined permutations. *Discrete Math,* 308:1500-1507, 2008.

[5] M. Bóna. Combinatorics of Permutations. CRC Press, 2004.

[6] R. Brignall, N. Ruškuc and V. Vatter. Simple permutations: decidability and unavoidable substructures. *Theoretical Computer Science,* 391:150–163, 2008.

[7] G. Cherlin. Forbidden substructures and combinatorial dichotomies: WQO and universality. *Discrete Mathematics,* 311(15):1534–1584, 2011.

[8] G. Cherlin and B. Latka. Minimal antichains in well-founded quasi-orders with an application to tournaments. *Journal of Combinatorial Theory Series B,* 80(2):258–276, 2000.

[9] N. de Bruijn. A Combinatorial Problem. *Koninklijke Nederlandse Akademie V. Wetenschappen,* 49:758-764, 1946.

[10] G. Ding. Subgraphs and well-quasi-ordering. *Journal of Graph Theory,* 16(5):489-502, 1992.

[11] A. Ehrenfeucht, D. Haussler and G. Rozenberg. On Regularity of Context-Free Languages. *Theoretical Computer Science,* 27:311–332, 1983.

[12] S. Elizalde. A survey of consecutive patterns in permutations. *Recent trends in combinatorics,* IMA Volume in Mathematics and its Applications, (159):601–618, 2016.

[13] T. Griffiths. The Unsolvability of the Equivalence Problem for Λ-Free Nondeterministic Generalized Machines. *Journal of the Association for Computing Machinery,* 15:409–413, 1968.

[14] L. Haines. On free monoids partially ordered by embedding. *Journal of Combinatorial Theory,* 6:94–98, 1969.

[15] G. Higman. Ordering by divisibility in abstract algebras. *Proceedings of the London Mathematical Society,* 2(7):326–336, 1952.

[16] N. Hine and J. Oxley. When excluding one matroid prevents infinite antichains. *Advances in Applied Mathematics,* 45(1):74–76, 2010.

[17] J. Hopcroft and J. Ullman. Introduction to Automata Theory, Languages, and Computation. Addison-Wesley, 1979.

[18] S. Huczynska and N. Ruškuc. Well quasi-order in combinatorics: embeddings and homomorphisms. *Surveys in Combinatorics,* London Mathematical Society Lecture Series Notes, 424:261–293, 2015.

[19] J. Johnson. Rational Equivalence Relations. *Theoretical Computer Science,*. 47:37–60, 1986.

[20] D. Jurafsky and J. Martin. *Speech and Language Processing.* Prentice Hall, 2000.

[21] S. Kitaev. *Patterns in Permutations and Words.* Monographs in Theoretical Computer Science, Springer, 2011.

[22] S. Kleene. Representation of Events in Nerve Nets and Finite Automata. *Automata Studies,* 3–42, 1951.

[23] N. Korpelainen and V. Lozin. Two forbidden induced subgraphs and well-quasi-ordering. *Discrete Mathematics,* 311:1813–1822, 2011.

[24] J. Kruskal. Well-quasi-ordering, the Tree theorem, and Vazsonyi's conjecture. *Transactions of the American Mathematical Society,* 95(2):210–255, 1960.

[25] K. Kuratowski. Sur le problème des courbes gauches en topologie. Fundamenta Mathematicae, 15:271–283, 1930.

[26] M. McDevitt. A Class of Rational Relations Generalising the Subword Order. *Journal of Automata, Languages and Combinatorics,* 23(4):361–386, 2018.

[27] M. Murphy. Restricted permutations, anti-chains, atomic sets and stack sorting (PhD thesis). University of St Andrews, 2002.

[28] C. Nash-Williams. On Well-Quasi-Ordering Finite Trees. *Mathematical Proceedings of the Cambridge Philosophical Society,* 59(4):833–835, 1963.

[29] A. Nerode. Linear Automaton Transformations. *Proceedings of the American Mathematical Society* 9:541–544, 1958.

[30] M. Rabin and D. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development,* 3(2):114–125, 1959.

[31] N. Robertson and P. Seymour. Graph Minors XX: Wagner's conjecture. *Journal of Combinatorial Theory Series B,* 92(2):325–357, 2004.

[32] E. Steingrìmsson. Some open problems on permutation patterns. *Surveys in Combinatorics,* London Mathematical Society Lecture Notes Series, 409:239–263, 2013.

[33] K. Wagner. Über eine Eigenschaft der ebenen Komplexe. *Mathematische Annalen,*s 114:570–590, 1937.