

THÈSE DE DOCTORAT

Soutenue à AMU — Aix-Marseille Université
le 12 juin 2026 par

Raphaël HENRY

Complexité et minimalité pour les substitutions et les mots
lisses

Discipline

Mathématiques

École doctorale

ED 184 Mathématiques et Informatique

Laboratoire/Partenaires de recherche

Institut de Mathématiques de Marseille (I2M)



INSTITUT
de MATHÉMATIQUES
de MARSEILLE

Composition du jury

| | |
|--|-----------------------|
| Irène MARCOVICI PR, Univ. Rouen Normandie | Rapporteuse |
| Reem YASSAWI Senior lecturer, Queen Mary Univ. of London | Rapporteuse |
| Fabien DURAND PR, Univ. Picardie Jules Verne | Président du jury |
| Emmanuel JEANDEL PR, Univ. de Lorraine | Examineur |
| Julien LEROY Chargé de cours, Univ. de Liège | Examineur |
| Michaël RAO DR, ENS de Lyon | Examineur |
| Nicolas BÉDARIDE MCF, Aix Marseille Univ. | Directeur de thèse |
| Étienne MOUTOT CR, Univ. Grenoble Alpes | Co-directeur de thèse |

Affidavit

Je soussigné, Raphaël HENRY, déclare par la présente que le travail présenté dans ce manuscrit est mon propre travail, réalisé sous la direction scientifique de Nicolas BÉDARIDE et Étienne MOUTOT, dans le respect des principes d'honnêteté, d'intégrité et de responsabilité inhérents à la mission de recherche. Les travaux de recherche et la rédaction de ce manuscrit ont été réalisés dans le respect à la fois de la charte nationale de déontologie des métiers de la recherche et de la charte AMU relative à la lutte contre le plagiat.

Ce travail n'a pas été précédemment soumis en France ou à l'étranger dans une version identique ou similaire à un organisme examinateur.

Fait à Marseille le 14/04/2026.



Cette œuvre est mise à disposition selon les termes de la [Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 4.0 International](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Liste de publications et participation aux conférences

Liste des publications et/ou brevets réalisées dans le cadre du projet de thèse:

1. J. CASSAIGNE et R. HENRY. “ The complexity of smooth words over binary alphabets ”. Preprint: arxiv.org/abs/2603.10733 (2026) (soumis pour publication)
2. R. HENRY. “ On the minimal components of substitution subshifts ”. Publié dans Theoretical Computer Science : doi.org/10.1016/j.tcs.2025.115517 (2025)
3. R. HENRY. “ Morphic Sequences : Complexity and Decidability ”. Publié dans Electronic Proceedings in Theoretical Computer Science : doi.org/10.4204/eptcs.403.24 (2024)

Participation aux conférences et écoles d’été au cours de la période de thèse:

1. Dyadisc 9, juin 2026, Saint-Valery-sur-Somme, France (Exposé : *Generalized smooth sequences*)
2. Combinatorics on Words workshop, mai 2026, Janov nad Nisou, République Tchèque (Exposé : *Generalized smooth sequences*)
3. Journées Combinatoires de Bordeaux, février 2026, Bordeaux, France
4. École-conférence de Combinatoire, d’Automates et de Théorie des Nombres (CANT), septembre 2025, Marseille, France
5. Dyadisc 8, juillet 2025, Amiens, France (Exposé : *On the minimal components of substitution subshifts*)
6. WORDS, juin 2025, Nancy, France
7. EJCIFM, juin 2025, Caen, France (Exposé : *La séquence de Kolakoski: plongée dans la combinatoire des mots*)

8. Combinatorics on Words workshop, mai 2025, Janov nad Nisou, République Tchèque (Exposé : *On the minimal components of substitution subshifts*)
9. Journées du GDR IFM, mars 2025, Bordeaux, France (Présentation de poster : *On the minimal components of substitution subshifts*)
10. Dyadisc 7, décembre 2024, Amiens, France
11. Journées Montoises d'informatique théorique, juin 2024, Nice, France (Exposé : *On the minimal components of substitution subshifts*)
12. Groupe de travail Porkroll, juillet 2024, Porquerolles, France (Exposé : *On the minimal components of substitution subshifts*)
13. GASCOM, juin 2024, Bordeaux, France (Exposé : *Morphic sequences: complexity and decidability*)
14. Journées SDA2, mai 2024, Orléans, France (Exposé : *Morphic sequences: complexity and decidability*)
15. Journées Combinatoires de Bordeaux, février 2024, Bordeaux, France
16. Mois thématique "Mathématiques discrètes et informatique : mots, complexité, dynamique, groupes", février 2024, Marseille, France

Résumé et mots clés

La dynamique symbolique et la combinatoire des mots sont deux approches complémentaires pour évaluer la complexité des mots infinis, ces derniers étant souvent des versions discrètes de processus plus complexes. Cette thèse explore deux notions centrales pour décrire les mots infinis : la *complexité des facteurs* est la fonction qui compte les mots finis de chaque longueur, et la *réurrence uniforme/minimalité* est une propriété qui contrôle la répartition des occurrences de chaque facteur.

Les *substitutions* sont une des principales manières de générer des mots infinis et ont donné naissance à une théorie prolifique qui est au cœur de la dynamique symbolique. En nous intéressant d'abord au problème du calcul de la complexité d'une suite morphique donnée, nous présentons une revue des résultats de Pansiot et de Devyatov qui décrivent les classes de complexité possibles. Ensuite, en nous fondant sur la caractérisation des sous-shifts substitutifs minimaux donnée récemment par Shimomura, nous étudions à quel point un sous-shift peut être non-minimal. Pour cela nous caractérisons et comptons les composantes minimales de ces sous-shifts.

Au-delà du cadre substitutif, la *suite de Oldenburger-Kolakoski* pose des questions notoirement difficiles à propos de sa récurrence, de sa complexité et de ses fréquences. En considérant la généralisation que sont les *suites lisses* sur des alphabets de deux entiers, nous présentons nos contributions à deux problèmes sur ce sujet. En premier nous étudions la complexité du langage des *mots lisses*, dont la conjecture est qu'elle croît comme $\Theta(n^\rho)$ où ρ dépend de l'alphabet. Nous prouvons la borne inférieure dans le cas général et la borne supérieure quand les deux lettres sont des entiers pairs, et nous améliorons la borne supérieure quand les deux lettres sont des entiers impairs. En second nous étudions les suites lisses sur les alphabets pairs et impairs, qui ont l'avantage d'avoir une représentation S-adique. Nous prouvons que ces suites sont uniformément récurrents, et, avec des conditions sur l'alphabet, qu'ils ont une complexité linéaire et sont uniquement ergodiques.

Mots clés : Dynamique symbolique, combinatoire des mots, complexité des facteurs, minimalité, substitutions, mots lisses

Abstract and keywords

Symbolic dynamics and combinatorics on words are two complementary approaches to evaluate how complex infinite words are, where infinite words are often seen as discrete representations of more complex processes. This thesis explores two major notions to describe infinite words: the *factor complexity* is the function that counts the number of finite words of each length, and *uniform recurrence/minimality* is the property that controls how occurrences of finite words are spread.

Substitutions are one of the primary ways to generate infinite words and gave birth to a fruitful theory in symbolic dynamics. Driven by the problem of computing the complexity class of a given morphic sequence, we begin by presenting a review of the results of Pansiot and Devyatov that describe the possible complexity classes. Then, inspired by the recent characterization of minimal substitution subshifts by Shimomura, we investigate how non-minimal a substitution subshift can be. To do so we characterize and count the minimal components of these subshifts.

Beyond the substitutive framework, the famous *Oldenburger-Kolakoski sequence* raises difficult questions about its recurrence, its factor complexity and its frequencies. By considering a generalization called *smooth sequences* over 2-integer alphabets, we present contributions to two connected problems. We first study the factor complexity of the language of *smooth words*, which is conjectured to grow like $\Theta(n^\rho)$ where ρ depends on the alphabet. We prove the lower bound in the general case and the upper bound when the two letters are even integers, and we improve the known upper bound when the two letters are odd integers. Then we study smooth sequences over even and odd alphabets, which happen to have an S-adic representation. We prove that these sequences are uniformly recurrent, and, under conditions on the alphabet, that they have linear factor complexity and are uniquely ergodic.

Keywords: Symbolic dynamics, combinatorics on words, factor complexity, minimality, substitutions, smooth words

Remerciements

En premier je remercie Nicolas et Étienne, qui ont été de super directeurs de thèse, pour la curiosité et la liberté scientifique, et pour m'avoir fait confiance pendant ces trois années. Je suis très fier de ce que j'ai appris avec vous et du travail accompli, j'espère que vous l'êtes aussi.

Je remercie Julien Cassaigne, pour m'avoir encadré en stage avec Étienne, puis pour avoir été présent tout au long de ma thèse et pour avoir écrit deux articles ensemble. C'est une grande chance d'avoir pu travailler avec toi comme cela.

Je remercie grandement Irène et Reem d'avoir pris le temps de rapporter ma thèse aussi rigoureusement, ainsi que Fabien, Emmanuel, Julien et Michaël pour avoir accepté de faire partie du jury. C'est un honneur et un plaisir d'avoir un tel jury.

Je veux remercier ici les chercheurs/collègues dont j'ai croisé la route et qui ont démontré que la communauté que j'ai intégrée est exceptionnelle, tant scientifiquement qu'humainement.

Je pense d'abord à Michaël, qui, partageant le virus de Kolakoski, m'a fait envisager de faire de la recherche et d'étudier sérieusement la combinatoire des mots. Je remercie aussi Julien Leroy et Sébastien Labbé pour m'avoir fait confiance très tôt pour candidater en thèse avec eux, votre soulagement quand j'ai obtenu la thèse à Marseille m'a vraiment touché.

Je salue les liégeois qui m'ont toujours accueilli chaleureusement. Notamment Pierre, je me rappellerai toujours du soir où tu m'as embarqué avec un ami au cinéma alors que je débarquais à Liège le jour même, ça me fait sourire de penser que je vais soutenir ma thèse avant toi ! Je salue aussi Julien, Manon et son énergie, et Savinien que j'ai mieux appris à connaître à Janov.

Je salue Srečko, qui est devenu une sorte de parrain scientifique pour moi. Je remercie énormément Sébastien pour m'avoir fait bénéficier de la bourse d'échange avec Prague, ça a été une super expérience. I also thank Lubka et Zuzana for inviting me and welcoming me at Prague and Janov. Finally, I thank Reem and Dan for receiving me in London and Milton Keynes and for the useful discussions.

C'est une chance d'avoir fait ma thèse à l'I2M, un laboratoire incroyable où j'ai rencontré tant de personnes intéressantes, amusantes, et des amis proches.

Je commence par Firas, Yenni et Nikolai, j'ai vraiment grandi avec vous pendant deux ans. On a partagé tellement de bons moments entre les bibimbaps, les parties de dame de pique endiablées (le fameux Shai Hulud !), le babyfoot et les fléchettes. Je repense surtout au séjour à Porquerolles en 2024, avec Virgile aussi, et à notre expédition en Tunisie pour le mariage de Firas. J'ajoute évidemment Naoufal, Juliette et Rahim qui se sont beaucoup joints à nous.

Je salue mes deux co-bureaux Thomas et Bernhard, et plus généralement Guillaume,

Lucas, Pierre, Rémi, Yohan, Ira, Joffrey, Félix, Aurore, Thibaut, Clément et le reste des doctorants/post-doctorants/ATER de Saint-Charles. C'est un bonheur d'avoir des collègues de travail aussi sympas, le travail incluant évidemment le baby-foot... Je fais aussi un coucou aux Luminyens, dont Adrien, Willem, Axel et Siméon. Je remercie aussi les gestionnaires de l'I2M et tout particulièrement Kaina et Évelyne pour nous rendre la vie plus facile.

J'ai une pensée pour Marius et Marc, pour toutes les heures qu'on a passées à jouer aux cartes. Big up au groupe de vulgarisation, Marc, Sarah, Rémi, Anita, Simon et Jeanne, vous m'avez accepté dans votre groupe improbable alors que je n'aurai toujours pas fait cette formation... Je salue aussi mes camarades de M1 à Paris, Arthur, Juliette et Rayane.

Quentin, c'est fou tout ce qu'on a vécu en même temps, toutes les passions qu'on a partagées, dont les maths qui font qu'on est collègues maintenant... Pendant la thèse tu étais un modèle pour moi, et tu continueras de l'être à Toulouse. Mais par-dessus tout tu es mon meilleur ami, je suis trop fier de ce qu'on est devenus et que ça continue.

Vincent, depuis le GEC on en a fait du chemin, et c'est pas fini. Encore une étape importante qu'on passe ensemble à Marseille, cette fois c'est pour moi !

Je remercie mes parents de tout mon cœur. Merci pour votre soutien, vos conseils, votre tendresse, notre relation est si précieuse. Je fais plein de bisous à Jojo et Susu, les compères de toujours. Et je n'oublie pas Tissa !!

Gros bisous aussi à Mamilyne, ma meilleure supportrice, ton affection est un super-pouvoir.

Je dédie cette thèse à la chaîne Youtube de vulgarisation mathématique "Numberphile", sans laquelle cette thèse n'existerait pas.

Don't know where I'm in five
But I'm young and alive

Scarlet Pleasure

Contents

| | |
|---|-----------|
| Affidavit | 2 |
| Liste de publications et participation aux conférences | 3 |
| Résumé et mots clés | 5 |
| Abstract and keywords | 6 |
| Remerciements | 7 |
| Contents | 9 |
| 1. Introduction | 13 |
| 2. Notation and basic concepts | 20 |
| 2.1. Words | 20 |
| 2.1.1. Finite words | 20 |
| 2.1.2. Infinite sequences | 21 |
| 2.1.3. Subshifts | 21 |
| 2.2. Factor complexity | 22 |
| 2.2.1. Asymptotic behavior | 22 |
| 2.2.2. Low factor complexity | 23 |
| 2.3. Uniform recurrence/minimality | 23 |
| 2.3.1. Recurrence | 23 |
| 2.3.2. Minimality | 23 |
| I. Substitutions | 25 |
| 3. Substitutions properties | 26 |
| 3.1. Iterable morphisms and substitutions | 26 |
| 3.2. Growth rate of letters | 27 |
| 3.3. Tame morphisms | 28 |
| 4. Factor complexity of morphic sequences | 30 |
| 4.1. Preliminaries | 32 |
| 4.1.1. D0L-systems | 32 |
| 4.1.2. Morphic sequences | 32 |
| 4.1.3. Complexity classes of purely morphic sequences | 33 |

| | | |
|------------|---|-----------|
| 4.1.4. | Complexity classes of morphic sequences | 34 |
| 4.2. | Factor complexity of purely morphic sequences | 35 |
| 4.2.1. | Classification of morphisms | 35 |
| 4.2.2. | Pansiot's theorem | 38 |
| 4.3. | Factor complexity of morphic sequences | 40 |
| 4.3.1. | Classification of k -blocks | 41 |
| 4.3.2. | Devyatov's theorem | 44 |
| 4.4. | Open questions | 46 |
| 4.4.1. | Improved complexity classes | 46 |
| 4.4.2. | New complexity classes | 46 |
| 5. | Minimal components of substitution subshifts | 48 |
| 5.1. | Preliminaries | 51 |
| 5.1.1. | Substitution subshifts | 51 |
| 5.1.2. | Minimal substitution subshifts | 51 |
| 5.1.3. | Minimal components | 53 |
| 5.1.4. | New results | 54 |
| 5.2. | Tame minimal components | 57 |
| 5.2.1. | Dynamics of alphabets | 57 |
| 5.2.2. | Proof of Theorem 5.1.24 (i) | 60 |
| 5.3. | Wild minimal components | 61 |
| 5.3.1. | Maximal bounded factors | 61 |
| 5.3.2. | Proof of Theorem 5.1.24 (ii) | 70 |
| 5.4. | Dynamics of minimal components | 71 |
| 5.4.1. | Dynamics of tame minimal components | 71 |
| 5.4.2. | Dynamics of wild minimal components | 72 |
| 5.5. | Counting minimal components | 73 |
| 5.5.1. | Computing bounded letters | 73 |
| 5.5.2. | Computing minimal components | 74 |
| 5.5.3. | Upper bounds for minimal components | 74 |
| 5.6. | Minimal components on two letters | 76 |
| 5.6.1. | Growing substitutions | 76 |
| 5.6.2. | Non-growing substitutions | 78 |
| 5.7. | Open questions | 79 |
| II. | Smooth sequences | 80 |
| 6. | More Kolakoski sequences | 81 |
| 6.1. | Smooth sequences | 82 |
| 6.1.1. | Conjugacy to the full-shift | 83 |
| 6.1.2. | Fixed points | 84 |
| 6.1.3. | Dichotomy | 86 |
| 6.2. | Factor complexity and bispecial words | 87 |

| | |
|--|------------|
| 7. Complexity of smooth words | 88 |
| 7.1. Preliminaries | 90 |
| 7.1.1. Smooth words | 90 |
| 7.1.2. New results | 94 |
| 7.2. Factors of smooth sequences | 94 |
| 7.2.1. r -smooth words | 95 |
| 7.2.2. Prefixes and factors of smooth sequences | 95 |
| 7.3. Bispecial smooth words | 96 |
| 7.3.1. Finite primitives | 96 |
| 7.3.2. Reduction of bispecial smooth words | 97 |
| 7.3.3. Families of bispecial smooth words | 98 |
| 7.3.4. Computation of the complexity | 99 |
| 7.4. Bounds of the complexity | 100 |
| 7.4.1. The lower bound over any alphabet | 101 |
| 7.4.2. An upper bound from ℓ_i | 103 |
| 7.4.3. The upper bound over even alphabets | 103 |
| 7.4.4. An upper bound over odd alphabets | 104 |
| 8. Smooth sequences over even and odd alphabets | 107 |
| 8.1. Preliminaries | 108 |
| 8.1.1. S -adic smooth sequences | 108 |
| 8.1.2. New results | 110 |
| 8.2. Occurrences in smooth sequences | 111 |
| 8.2.1. Occurrences in derivatives | 111 |
| 8.2.2. Almost-short words | 112 |
| 8.2.3. Complementation of factors | 113 |
| 8.2.4. Uniform recurrence | 114 |
| 8.3. Factor complexity over even alphabets | 114 |
| 8.3.1. Forbidden and mandatory factors | 114 |
| 8.3.2. Bispecial factors | 115 |
| 8.3.3. Computation of the factor complexity | 119 |
| 8.3.4. Unique ergodicity | 120 |
| 8.4. Factor complexity over odd alphabets | 121 |
| 8.4.1. Forbidden and mandatory factors | 121 |
| 8.4.2. Bispecial factors | 123 |
| 8.4.3. Computation of factor complexity | 124 |
| 8.4.4. Unique ergodicity | 125 |
| Bibliography | 126 |
| APPENDIX | 133 |
| A. Examples of morphisms for purely morphic sequences | 134 |
| A.1. Growing morphisms | 134 |

Contents

| | |
|-----------------------------------|-----|
| A.2. Tame morphisms | 134 |
| A.3. Wild morphisms | 135 |
| A.4. Periodic morphisms | 135 |

1. Introduction

I've had my ups and downs, my fair share of bumpy roads and heavy winds. That's what made me what I am today. Now I stand before you.

JCVD

The study of infinite words, or (symbolic) sequences, is at a crossroad between dynamical systems and theoretical computer science, incarnated respectively in *symbolic dynamics* and *combinatorics on words*. As such, it finds its roots in both fields and raises important theoretical questions as well as computational and decidability problems. One of the main concerns about sequences over finite alphabets is to evaluate how complex they are, and several notions were introduced to achieve that. In this thesis we are interested in two of these notions that are *factor complexity* and *minimality*, the first describes the quantity of information in sequences and the second is a homogeneity property of sequences. We present contributions in two different frameworks. The first one is *substitutive* sequences, they appear in many contexts and gave birth to a powerful theory that is still being developed today. The second one is *smooth* sequences, they can be seen as a specific generalization of substitutive sequences and they raise interesting questions related to the famous Oldenburger-Kolakoski sequence.

One way to obtain sequences over finite alphabets is to apply a discrete coding to complex systems. The aim is to study a simplified representation of the system, and the fundamental question is: how the properties of the original system can be recovered from the properties of the sequence? More generally, a valuable resource is the online encyclopedia OEIS [[Inc26](#)] which gathers every known integer sequence. In particular, the digit expansion of any number provides a symbolic sequence, the most popular example being

$$\pi = 3.141592653589793238462643383279502884197169399375105820974944\dots$$

A basic result that links the arithmetic properties of numbers to the combinatorial properties of their digit expansion is that a number is rational if and only if its digits are eventually periodic. Symbolic sequences also arise from discrete time dynamical systems, of the form (X, T) where X is a compact topological space on which acts a continuous bijective transformation $T : X \rightarrow X$. Given a point $x_0 \in X$, the sequence $(x_n)_{n \geq 0} := (T^n(x_0))_{n \geq 0}$ is the *orbit* of x_0 . Then, by taking a finite partition of the space

X and associating a symbol to each region, the orbit of a point provides a symbolic sequence. A simple but interesting example of this process is the coding of a rotation of the circle, where the space X is the circle of perimeter 1 and the transformation T is a rotation (see Figure 1.1).

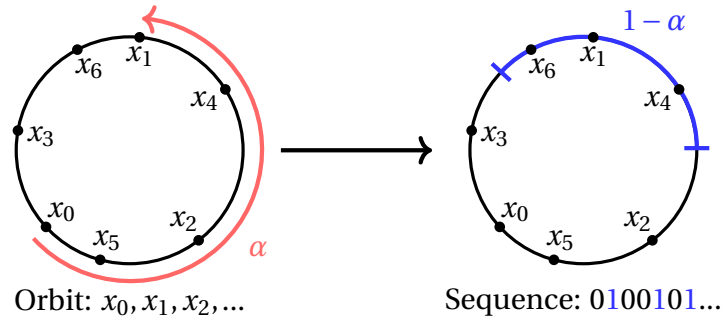


Figure 1.1.: A binary coding of a trajectory from a rotation of angle $\alpha = 1/\varphi$ where φ is the golden ratio. The partition associates the letter 0 to the black region and the letter 1 to the blue region.

Another way to obtain sequences over finite alphabets is to consider simple rules that generate discrete configurations. The aim is to understand how complex behaviors can emerge from these simple rules, as in nature for example, and the fundamental question is: how complex can the generated configurations be? Famous examples are the Game of Life [BCG82] and Langton's ant [Lan86] in dimension two. More general classes of simple rules are *cellular automata* or *substitutions* in dimension one.

Substitutions are maps that replace each letter by a finite word (on a possibly different alphabet) independantly of the surrounding letters, they are *morphisms* on words. Despite their simplicity, substitutions arise from more complex systems. One of their first usage was in *DOL-systems*, which are part of the larger class of *L-systems* introduced by Lindenmayer to study the development of filamentous organisms [RS76]. Substitutions also arise in numeration systems, coding of billiard trajectories, automata (see [AS03]), computer graphics or crystallography. Therefore, substitutions are an important part of symbolic dynamics, for a general background see [Que87], [DP22] or the upcoming book [BDP26]. The simplest setting is to consider a single iterable substitution, meaning that it maps to the same alphabet, which generates three types of symbolic systems. Firstly, to such a substitution is associated a canonical language, that is, the set of finite words that appear in the iterated image of any letter. DOL-systems also define a larger class of languages from substitutions. Secondly, a fixed point of an iterable substitution is a *purely morphic sequence*. One of the most emblematic examples is the *Fibonacci word* (A003849 in the OEIS [Inc26]), fixed by the substitution $\sigma : 0 \mapsto 01, 1 \mapsto 0$

$$x_{Fibo} = \underbrace{0}_{01} \underbrace{1}_0 \underbrace{0}_{01} \underbrace{0}_{01} \underbrace{1}_0 \underbrace{0}_{01} \underbrace{1}_0 \underbrace{0}_{01} \underbrace{0}_{01} \underbrace{1}_0 \underbrace{0}_{01} \underbrace{0}_{01} \underbrace{1}_0 \underbrace{0}_{01} \underbrace{1}_0 \dots = x_{Fibo}$$

Note that x_{Fibo} is also the sequence obtained from the rotation of the circle in Figure 1.1. Furthermore, applying a letter-to-letter substitution (or *coding*) to a purely morphic sequence produces a *morphic sequence*, see [All+17] for a detailed guide. Lastly, by introducing a topology and a *shift* map $S : (x_i)_{i \in \mathbb{N}} \mapsto (x_{i+1})_{i \in \mathbb{N}}$ on right-infinite sequences (or $S : (x_i)_{i \in \mathbb{Z}} \mapsto (x_{i+1})_{i \in \mathbb{Z}}$ on bi-infinite sequences), a *subshift* is a closed shift-invariant set of sequences. Then, an iterable substitution generates a *substitution subshift*. In the setting of D0L-systems, morphic sequences or substitution subshifts, the goal is to connect the properties of the substitution and the properties of the associated system. Beyond single iterable substitutions, the most general setting is *S-adic sequences* or *S-adic subshifts*, which are systems generated from a (possibly infinite) number of substitutions on different alphabets.

As a variation of purely morphic sequences, the *Oldenburger-Kolakoski sequence* (A000002 in the OEIS [Inc26]) is defined as a fixed point of the *run-length encoding*

$$\kappa = \underbrace{22}_2 \underbrace{11}_2 \underbrace{2}_1 \underbrace{1}_1 \underbrace{22}_2 \underbrace{1}_1 \underbrace{22}_2 \underbrace{11}_2 \underbrace{2}_1 \underbrace{11}_2 \underbrace{22}_2 \underbrace{1}_1 \underbrace{2}_1 \underbrace{11}_2 \underbrace{2}_1 \underbrace{1}_1 \underbrace{22}_2 \dots = \kappa$$

The sequence κ is a fixed point of a 2-block substitution, that is, a substitution generalized to non-overlapping blocks of length two (see [DK23] for example). The 2-block substitution in question is $11 \mapsto 21, 12 \mapsto 211, 21 \mapsto 221, 22 \mapsto 2211$, but it is not iterable since the images 211 and 221 cannot be decomposed in non-overlapping blocks of length two. This is seemingly close to a purely morphic sequence, but here we know almost nothing about κ . This reveals that the theory of substitutions is still largely under-developed outside of the classical substitutive setting.

One of the main combinatorial tools to measure the complexity of sequences over a finite alphabet is *factor complexity*. It is the function $p : \mathbb{N} \rightarrow \mathbb{N}$ that counts the number of *factors* of each length, that is, the finite words that occur in the sequence. It was introduced by Morse and Hedlund in [MH38], with the idea that the asymptotical behavior of this function describes the quantity of information contained in a sequence. As a primary illustration of this idea, Morse and Hedlund showed in [MH38] that the eventually periodic sequences, that is, the sequences containing the fewest information, are the sequences with bounded factor complexity. A corollary of this result is that the factor complexity of an aperiodic sequence always satisfies $p(n) \geq n + 1$. On that note, the aperiodic sequences which have minimal factor complexity, i.e., $p(n) = n + 1$, are called *Sturmian* sequences, they have numerous interesting properties and the Fibonacci word x_{Fibo} is one of them, see [MH40] for an overview. On the opposite side of the spectrum, π is conjectured to be a *rich* (or *disjunctive*) number, meaning that its digit expansion has maximal factor complexity $p(n) = 10^n$. The link with numeration systems extends further, for instance in [AB05] the authors described the factor complexity of algebraic numbers. To measure the quantity of information of infinite sequences, the *topological entropy* is the exponential growth rate of the factor complexity $\lim_{n \rightarrow \infty} \log(p(n)) / n$. For example, Sturmian sequences have zero entropy while π is believed to have positive entropy. Note that the factor complexity has also been investigated in higher dimensions: similarly to Morse-Hedlund theorem, Nivat

conjectured in [Niv97] that a configuration of \mathbb{Z}^2 with rectangle pattern complexity $p(m, n) \leq mn$ has to be periodic. This question has seen great progress with the recent work of Kari, Szabados and Moutot in [Sza18; KM19]. Back to the one-dimensional case of sequences, this is where the study of factor complexity is the most developed. The best available tool is *special* and *bispecial factors*, they were largely popularized by Cassaigne in [Cas97] and allow to compute factor complexity very efficiently. Lastly, note that several variations of the factor complexity have been introduced, like the *maximal pattern complexity* in [KZ02], the *abelian complexity* in [RSZ09], the *binomial complexity* in [RS15], the *additive complexity* in [PSS24] or the *reflection complexity* in [All+25].

Important results have been shown for the factor complexity of D0L-systems and morphic sequences. This started with the seminal work of Ehrenfeucht, Lee and Rozenberg for D0L-systems in [ELR75] where the authors obtained an upper bound of the factor complexity of the associated language for three classes of substitutions. In particular, they show that the general upper bound is $\mathcal{O}(n^2)$ and that it is attained, which yields zero entropy for all substitutive systems. As a purely morphic sequence has the same language as the associated substitution, these results about D0L-systems also apply to purely morphic sequences. With that said, the main result for purely morphic sequences is due to Pansiot [Pan84]: the asymptotic factor complexity of purely morphic sequences is either $\Theta(1)$, $\Theta(n)$, $\Theta(n \log \log n)$, $\Theta(n \log n)$ or $\Theta(n^2)$, depending on the properties of the substitution. Now, for morphic sequences, the coding heavily impacts the factor complexity and produces the complexity classes $\Theta(n^{1+1/k})$ for every $k \geq 1$. No other complexity class is known, and Devyatov [Dev18] showed that there is no other complexity class above $\Theta(n \log n)$. For this result however, the characterization of each complexity class is much more complex than in Pansiot's theorem. The theorems of Pansiot and Devyatov provide a necessary condition for an infinite sequence to be morphic. For instance, Dekking's conjecture [Dek81] that the factor complexity of the Oldenburger-Kolakoski sequence grows like $\Theta(n^{\log(3)/\log(3/2)})$ would imply that κ is not morphic.

Although factor complexity describes how many factors occur in a sequence, it does not tell anything about how each factor occurs: does it occur infinitely often? how far apart are two occurrences? To express this, a sequence is *recurrent* if all its factors occur infinitely, and it is *uniformly recurrent* if it is recurrent and each of its factors occurs with bounded gaps. Even stronger, x is *linearly recurrent* (or *linearly repetitive*) if it is uniformly recurrent and the gaps between two occurrences of a factor are bounded linearly by the length of the factor. In particular, in [DHS99] the authors showed that a linearly recurrent sequence has at most linear factor complexity. Uniform recurrence can also be defined for subshifts, and in that case we say that the subshift is *minimal*. Minimality of subshifts has two topological interpretations, it is equivalent to having no sub-system and to the fact that the orbit of every point by the shift S is dense.

It is well-known that *primitive* substitutions generate minimal subshifts, but primitivity is not a necessary condition for minimality: the Chacon substitution $\sigma : 0 \mapsto 0010, 1 \mapsto 1$ is a famous non-primitive example that produces a minimal subshift. Recently,

Maloney and Rust [MR18] showed that a property called *tameness* is necessary for minimality. Shimomura [Shi19] then introduced a weaker version of primitivity, called *ℓ -primitivity*, which led to a characterization of minimal substitution subshifts: a substitution subshift is minimal if and only if it is generated by an ℓ -primitive and tame substitution (possibly different from the original one). A corollary of Shimomura's result is that every minimal substitution subshift is linearly recurrent. In the S-adic setting, Durand [Dur00] also found a criterion for minimality called *weak primitivity*, and a criterion for linear recurrence called *strong primitivity*. With these results, primitivity of substitutions is a key property to characterize minimality. However, outside of the substitutive setting, the recurrence of the Oldenburger-Kolakoski sequence is still one of the main open questions.

Main contributions

In this thesis we investigate factor complexity and minimality in two frameworks: the well-known substitutive systems, and the generalization of the Oldenburger-Kolakoski sequence called *smooth sequences*. This leads us to adopt different approaches for each notion: we study factor complexity algorithmically in the first framework and with bispecial factors in the second; we study minimality topologically in the first framework and combinatorially in the second.

We first focus on the theorems of Pansiot and Devyatov about the factor complexity of morphic sequences. We are interested in the following problem: given a morphic sequence, can we compute its complexity class? To do so, we proceed in three steps: we detail the proofs of the theorems that characterize the complexity classes, we transpose them into a formal algorithm, and we solve the computational problems raised by the algorithm. Based on our work in [Hen24], we begin with purely morphic sequences where Pansiot's classification is complete and straightforward. Then, Devyatov's result for morphic sequences is more layered and relies on complex combinatorial structures so we mainly focus on giving a condensed version of the proof.

Then we turn our attention to substitution subshifts. Now that minimality has been characterized by Shimomura, we investigate the opposite case, that is, what happens when a substitution subshift is not minimal? How non-minimal can it be? These questions are part of a recent interest for non-minimal substitution subshift in [MR18; Shi19; BPR24]. In the last paper, the authors showed that any substitution subshift is *quasi-minimal*, meaning that it contains a finite number of subshifts. In particular, the minimal sub-systems are called *minimal components*, and their number can be seen as a measure of the complexity of the subshift. For instance, in [BKM09] the authors showed that any growing substitution subshift on d letters has at most d minimal components. Our contribution ([Hen25]) is to characterize the minimal components of any substitution subshift: we show that they are either smaller substitution subshifts or periodic orbits composed of bounded letters. Our characterization is explicit so we obtain upper bounds for the number of minimal components: over d letters, we find back that growing substitutions have at most d minimal components, and for non-growing substitutions we find at most $2d - 4$ minimal components since

the periodic orbits can be twice as numerous. We also provide a Python code that computes minimal components of a given substitution subshift.

Next we leave the well-known framework of substitutions to focus on a generalization of the Oldenburger-Kolakoski sequence κ called *smooth sequences*. The first level of generalization is that the run-length encoding has fixed points over other 2-integer alphabets, like $\{1, 3\}$ with

$$\kappa_{3,1} = \underbrace{333}_3 \underbrace{111}_3 \underbrace{333}_3 \underbrace{1}_1 \underbrace{3}_1 \underbrace{1}_1 \underbrace{333}_3 \underbrace{111}_3 \underbrace{333}_3 \underbrace{1}_1 \underbrace{333}_3 \underbrace{1}_1 \underbrace{333}_3 \underbrace{111}_3 \underbrace{333}_3 \underbrace{1}_1 \dots = \kappa_{3,1}$$

Similarly to κ , $\kappa_{3,1}$ (A064353 in the OEIS [Inc26]) is the fixed point of a 2-block substitution. These fixed points were considered soon after the Oldenburger-Kolakoski sequence was rediscovered by Kolakoski in [Kol65], and a systematic study was made by Sing in [BS04] and [Sin10]. Now, smooth sequences are not only the fixed points of run-length encoding, they are the infinite sequences over a 2-integer alphabet $\mathcal{A} = \{a, b\}$ such that all their successive run-length encodings remain over the same alphabet \mathcal{A} . General smooth sequences had been mentioned in a few papers like [Dek01] but their systematic study began in [BJP08] and continued very recently with [Jam+26]. The motivation to study this generalization is to understand what is difficult about κ , to see if we can prove more conclusive results about other smooth sequences and if they translate to κ . With this in mind, we study smooth sequences with two approaches.

We first consider the language of *smooth words*, which is crucial in the study of smooth sequences. For instance, over $\{1, 2\}$ it is conjectured to be exactly the set of factors of κ and to have complexity $\Theta(n^{\log(3)/\log(3/2)})$, which is why κ is conjectured to have the same behavior. The definitions yield that every factor of a smooth sequence is a smooth word, and conversely we conjecture that the factors of each smooth sequence over an alphabet $\{a, b\}$ where $a + b$ is odd are exactly the smooth words. In this direction, we begin by proving that, over any alphabet, every smooth word occurs in at least one smooth sequence. Regarding the complexity of the language of smooth words, over an alphabet $\{a, b\}$ it is conjectured to grow like $\Theta(n^{\log(a+b)/\log((a+b)/2)})$. Partial results have been proven towards this conjecture, first over $\{1, 2\}$ where Dekking showed polynomial upper and lower bounds in [Dek81] and Weakley studied the bispecial smooth words in [Wea89], then over any alphabet where Sing generalized the polynomial bounds in [Sin02]. In a joint work with Julien Cassaigne ([CH26]), we build on the work of Weakley: we prove the conjectured lower bound over any alphabet, the conjectured upper bound over even alphabets (i.e., a and b are even) and we improve the known upper bound over odd alphabets (i.e., a and b are odd).

Finally we consider specifically smooth sequences over even and odd alphabets. Contrary to the other alphabets where smooth sequences like κ are very difficult to study, it appears that smooth sequences over these alphabets have simpler behaviors. Therefore more results have been shown in this setting, for example in [BJP08] the authors showed that all smooth sequences are recurrent using palindroms. In fact, smooth sequences have an S-adic structure over these alphabets, like the sequence

$\kappa_{3,1}$ which is morphic and has been extensively studied in [BS04]. Very recently, in [Jam+26] the authors also exploited this S-adic structure to study letter frequencies in smooth sequences over $\{1, 3\}$. In another joint work in preparation with Julien Cassaigne, we continue to study smooth sequences over even and odd alphabets: we first prove uniform recurrence for every smooth sequence, then we obtain linear factor complexity over every even alphabet and every odd alphabet of the form $\{a, a + 2\}$. Finally, as a corollary we obtain unique ergodicity for all smooth sequences over specific alphabets.

Organization of the manuscript

This manuscript is organized in two independent parts, the first gathers our contributions to the theory of substitutions and the second gathers our contributions about smooth sequences. But first, in Chapter 2 we introduce the basic notations and definitions about words, factor complexity and minimality.

Chapter 3 is preliminary to our contributions to the theory of substitutions. After recalling the basic definitions of substitutions and growth rate of letters, we highlight an important property of substitutions, *tameness*, that is crucial in the next two chapters.

In Chapter 4 we analyze the results of Pansiot and Devyatov on the factor complexity of morphic sequences. The first result is primarily based on the growth rate of letters, and the second result relies on a more complex combinatorial structure called *k*-blocks.

In Chapter 5 we characterize the minimal components of substitution subshifts (Theorem 5.1.24). We treat separately the minimal components that contain growing letters, which are smaller substitution subshifts generated by a restriction to a power of the substitution, and the minimal components that contain only bounded letters, which are periodic orbits. Then we count the minimal components (Corollary 5.1.30).

Chapter 6 is preliminary to our contributions about smooth sequences. Inspired by the original Oldenburger-Kolakoski sequence, we introduce smooth sequences and explain how they can be computed. We also emphasize on the fixed points of the run-length encoding that are morphic, and we explain the dichotomy within smooth sequences that led to the two approaches we take in the next chapters.

In Chapter 7 we study smooth words. We first introduce another language of *r*-smooth words, then we show that every smooth word occurs in an *r*-smooth word and that every *r*-smooth word is the prefix of a smooth sequence (Theorem 7.1.18). Then, for the complexity of smooth words, we determine the bispecial words as Weakley did over $\{1, 2\}$ and we use technical arguments to obtain lower and upper bounds from the average and the minimal length of bispecial words (Theorems 7.1.19 and 7.1.20).

In Chapter 8 we deal with three different properties of smooth sequences over even and odd alphabets. We prove uniform recurrence of smooth sequences (Theorem 8.1.5) and linear factor complexity over even alphabets and some odd alphabets (Theorem 8.1.6). Lastly, we determine the alphabets over which the factor complexity of smooth sequences is sufficiently low to apply a criterion of Boshernitzan [Bos84] for unique ergodicity (Theorem 8.1.7).

2. Notation and basic concepts

C'est bien beau de jouer l'poète libre
Alors qu'aucun mot ne vient sans un chiffre

Vald

Content

| | |
|------------------------------------|----|
| 2.1. Words | 20 |
| 2.1.1. Finite words | 20 |
| 2.1.2. Infinite sequences | 21 |
| 2.1.3. Subshifts | 21 |
| 2.2. Factor complexity | 22 |
| 2.2.1. Asymptotic behavior | 22 |
| 2.2.2. Low factor complexity | 23 |
| 2.3. Uniform recurrence/minimality | 23 |
| 2.3.1. Recurrence | 23 |
| 2.3.2. Minimality | 23 |

2.1. Words

A *finite alphabet* \mathcal{A} is a non-empty finite set, its elements are called *letters*. We usually use letters $0, 1, 2, \dots$ in examples and letters a, b, c, \dots in proofs.

2.1.1. Finite words

Over a finite alphabet \mathcal{A} , a *word* u is a concatenation $u_1 u_2 \dots u_{n-1} u_n$ of letters of \mathcal{A} , where $n = 0$ corresponds to the empty word denoted by ε . We define the length of the word $u = u_1 u_2 \dots u_{n-1} u_n$ by $|u| := n$, and if $a \in \mathcal{A}$ we define $|u|_a$ as the number of occurrences of the letter a in u . For $n \geq 0$, we denote the set of words of length n over \mathcal{A} by \mathcal{A}^n . We also write \mathcal{A}^* for the set of all finite words and $\mathcal{A}^+ := \mathcal{A}^* \setminus \{\varepsilon\}$.

If $u, v \in \mathcal{A}^*$ are two finite words, we write uv their natural concatenation. Also, if $w, w' \in \mathcal{A}^*$ are such that $u = wvw'$, we say that v is a *factor* of u and we write $v \sqsubset u$. In particular, we say that v is a *prefix* (resp. *suffix*) of u if $w = \varepsilon$ (resp. $w' = \varepsilon$).

Over a finite alphabet \mathcal{A} , a *language* is a set of finite words $\mathcal{L} \subseteq \mathcal{A}^*$, and it is *factorial* if $u \sqsubset v \in \mathcal{L}$ implies $u \in \mathcal{L}$. Then the set of factors of a finite word $u \in \mathcal{A}^*$ is

the factorial language

$$\mathcal{L}(u) := \{v \in \mathcal{A}^* \mid v \sqsubset u\}.$$

2.1.2. Infinite sequences

Over a finite alphabet \mathcal{A} , a *right-infinite sequence* (resp. a *bi-infinite sequence*) x is a concatenation $x_0x_1x_2\dots$ (resp. $\dots x_{-2}x_{-1}x_0x_1x_2\dots$) of letters of \mathcal{A} . We write $\mathcal{A}^{\mathbb{N}}$ (resp. $\mathcal{A}^{\mathbb{Z}}$) the set of right-infinite sequences (resp. bi-infinite sequences) over \mathcal{A} , where \mathbb{N} denotes non-negative integers including 0.

If $u \in \mathcal{A}^*$ and $v \in \mathcal{A}^+$, we write the right-infinite sequence $v^\omega := vvv\dots$ and we can do the concatenation uv^ω . A sequence of the form v^ω (resp. uv^ω) is *periodic* (resp. *eventually periodic*), otherwise it is *aperiodic*. Similarly, if $u \in \mathcal{A}^*$ and $v, w \in \mathcal{A}^+$, we write the bi-infinite sequences ${}^\omega v^\omega := \dots vvv.vvv\dots$ and ${}^\omega vuw^\omega := \dots vvv.uw^\omega$ where the middle dot indicates the separation $\dots x_{-1}.x_0\dots$. A sequence of the form ${}^\omega v^\omega$ is *periodic*, otherwise it is *aperiodic*.

If $x \in \mathcal{A}^{\mathbb{N}}$ (resp. $x \in \mathcal{A}^{\mathbb{Z}}$), a *factor* of x is a finite word $x_{[i,j]} := x_i x_{i+1} \dots x_{j-1}$ where $i, j \in \mathbb{N}$ (resp. $i, j \in \mathbb{Z}$) and $i \leq j$, with the convention that $x_{[i,i]} := \varepsilon$. We write $u \sqsubset x$ if $u \in \mathcal{A}^*$ is a factor of x , and the set of factors of x is the factorial language

$$\mathcal{L}(x) := \{u \in \mathcal{A}^* \mid u \sqsubset x\}.$$

In particular, if $x \in \mathcal{A}^{\mathbb{N}}$, a factor $u = x_{[0,j]}$ is a *prefix* of x and we write $u \sqsubset_p x$.

2.1.3. Subshifts

We define the following distance on $\mathcal{A}^{\mathbb{N}}$: for $x, y \in \mathcal{A}^{\mathbb{N}}$, if $x \neq y$ then we set $d(x, y) := 2^{-n}$ where $n \geq 0$ is the first integer for which $x_n \neq y_n$, otherwise we set $d(x, y) = 0$. We define a similar distance on $\mathcal{A}^{\mathbb{Z}}$: for $x, y \in \mathcal{A}^{\mathbb{Z}}$, if $x \neq y$ then we set $d(x, y) := 2^{-n}$ where $n \geq 0$ is the first integer for which $x_n \neq y_n$ or $x_{-n} \neq y_{-n}$, otherwise we set $d(x, y) = 0$. These distances induce the *prodiscrete topology* on $\mathcal{A}^{\mathbb{N}}$ and on $\mathcal{A}^{\mathbb{Z}}$.

Also, for right-infinite and bi-infinite sequences, we define the left shift

$$\begin{array}{ll} S: & \mathcal{A}^{\mathbb{N}} \longrightarrow \mathcal{A}^{\mathbb{N}} \\ & (x_i)_{i \geq 0} \longmapsto (x_{i+1})_{i \geq 0} \end{array} \qquad \begin{array}{ll} S: & \mathcal{A}^{\mathbb{Z}} \longrightarrow \mathcal{A}^{\mathbb{Z}} \\ & (x_i)_{i \in \mathbb{Z}} \longmapsto (x_{i+1})_{i \in \mathbb{Z}} \end{array}$$

Then, a *subshift* on $\mathcal{A}^{\mathbb{N}}$ (resp. on $\mathcal{A}^{\mathbb{Z}}$) is a non-empty set $X \subseteq \mathcal{A}^{\mathbb{N}}$ (resp. $X \subseteq \mathcal{A}^{\mathbb{Z}}$) that is closed for the prodiscrete topology and satisfies $S(X) = X$.

Lemma 2.1.1. (i) If $x \in \mathcal{A}^{\mathbb{N}}$, the smallest subshift containing x is the set

$$X(x) := \overline{\{S^k(x) \mid k \in \mathbb{N}\}}.$$

(ii) If $x \in \mathcal{A}^{\mathbb{Z}}$, the smallest subshift containing x is the set

$$X(x) := \overline{\{S^k(x) \mid k \in \mathbb{Z}\}}.$$

2. Notation and basic concepts – 2.2. Factor complexity

Generally, if $X \subseteq \mathcal{A}^{\mathbb{N}}$ or $X \subseteq \mathcal{A}^{\mathbb{Z}}$, e.g., X is a subshift, we define the factorial language

$$\mathcal{L}(X) := \bigcup_{x \in X} \mathcal{L}(x).$$

2.2. Factor complexity

In order to count the number of factors in an infinite factorial language, combinatorics on words provides the *factor complexity*.

Definition 2.2.1. *Over a finite alphabet \mathcal{A} , we define the following complexity functions $p : \mathbb{N} \rightarrow \mathbb{N}$.*

- *The complexity of an infinite factorial language $\mathcal{L} \subseteq \mathcal{A}^*$ is $p_{\mathcal{L}}(n) := |\mathcal{L} \cap \mathcal{A}^n|$,*
- *The factor complexity of a sequence $x \in \mathcal{A}^{\mathbb{N}} \cup \mathcal{A}^{\mathbb{Z}}$ is $p_x(n) := p_{\mathcal{L}(x)}(n)$,*
- *The factor complexity of a set $X \subseteq \mathcal{A}^{\mathbb{N}}$ or $X \subseteq \mathcal{A}^{\mathbb{Z}}$ is $p_X(n) := p_{\mathcal{L}(X)}(n)$.*

Remark 2.2.2. *A complexity function is always non-decreasing.*

2.2.1. Asymptotic behavior

In order to describe the asymptotics of sequences or functions $\mathbb{N} \rightarrow \mathbb{N}$, e.g., complexity functions, we will frequently use the following notation.

Definition 2.2.3. *Given two functions $f, g : \mathbb{N} \rightarrow \mathbb{N}$, we write*

- *$f(n) = \mathcal{O}(g(n))$ if there exist $C > 0$ and $N \geq 0$ such that $f(n) \leq Cg(n)$ for all $n \geq N$,*
- *$f(n) = \Omega(g(n))$ if there exist $C > 0$ and $N \geq 0$ such that $f(n) \geq Cg(n)$ for all $n \geq N$,*
- *$f(n) = \Theta(g(n))$ if $f(n) = \mathcal{O}(g(n))$ and $f(n) = \Omega(g(n))$.*

Because the complexity of a factorial language $\mathcal{L} \subseteq \mathcal{A}^*$ is *sub-multiplicative*, i.e., $p_{\mathcal{L}}(m+n) \leq p_{\mathcal{L}}(m)p_{\mathcal{L}}(n)$ for all $m, n \geq 0$, the quantity

$$h(\mathcal{L}) := \lim_{n \rightarrow \infty} \log(p_{\mathcal{L}}(n)) / n$$

always exists and defines a notion of entropy.

Definition 2.2.4. *Given an infinite factorial language $\mathcal{L} \subseteq \mathcal{A}^*$, its topological entropy is the quantity $h(\mathcal{L})$.*

2.2.2. Low factor complexity

A seminal result on factor complexity is the characterization of eventually periodic sequences.

Theorem 2.2.5 ([MH38]). *Let $x \in \mathcal{A}^{\mathbb{N}}$. Then the following properties are equivalent.*

1. x is eventually periodic.
2. The function $p_x : \mathbb{N} \rightarrow \mathbb{N}$ is bounded.
3. There exists $n \geq 0$ such that $p_x(n+1) = p_x(n)$.

With the third property in Theorem 2.2.5, an aperiodic sequence x always satisfies $p_x(n+1) \geq p_x(n) + 1$. As $p_x(0)$ is always equal to 1, we deduce that $p_x(n) \geq n + 1$ for all $n \geq 0$. Aperiodic sequences with minimal factor complexity $p_x(n) = n + 1$ do exist, and are in fact the well-studied class of *Sturmian* sequences.

2.3. Uniform recurrence/minimality

Uniform recurrence and minimality define the same notion in different setting. The first one is a combinatorial property of sequences, the second one is a dynamical property of subshifts.

2.3.1. Recurrence

On the combinatorial side, there are different levels of recurrence for a sequence.

Definition 2.3.1. *Let $x \in \mathcal{A}^{\mathbb{N}} \cup \mathcal{A}^{\mathbb{Z}}$ be a sequence.*

- x is recurrent if all its factors occur infinitely often in it.
- A sequence $u \in \mathcal{A}^*$ occurs with bounded gaps in x if there exists $L \geq 1$ such that, for all $v \in \mathcal{L}(x)$, if $|v| \geq L$ then $u \sqsubset v$. Then, stronger than being recurrent, x is uniformly recurrent if all its factors occur with bounded gaps in it.
- Even stronger than being uniformly recurrent, x is linearly recurrent if there exists a constant $L \geq 1$ such that, for all $u, v \in \mathcal{L}(x)$, if $|v| \geq L|u|$ then $u \sqsubset v$.

2.3.2. Minimality

On the dynamical side, there are several equivalent definitions of minimality. But first, we say that a word $u \in \mathcal{A}^*$ occurs with bounded gaps in a subshift X if there exists $L \geq 1$ such that, for all $v \in \mathcal{L}(X)$, if $|v| \geq L$ then $u \sqsubset v$.

2. Notation and basic concepts – 2.3. Uniform recurrence/minimality

Definition/Proposition 2.3.2. A subshift $X \subseteq \mathcal{A}^{\mathbb{Z}}$ is minimal if one of the three equivalent conditions holds:

1. X contains no subshift other than itself.
2. For every $x \in X$, $X(x) = X$.
3. Every word of $\mathcal{L}(X)$ occurs with bounded gaps in X .

The first definition is dynamical, it means that the subshift has no sub-system. The second definition is topological, it means that the orbit of every point is dense. The third definition is combinatorial, it is the closest to uniform recurrence. A proof of the equivalence between the three definitions can be found in [Fog+02].

We also say that a subshift X is *linearly recurrent* if there exists a constant $L \geq 1$ such that, for all $u, v \sqsubset \mathcal{L}(X)$, if $|v| \geq L|u|$ then $u \sqsubset v$.

Remark 2.3.3. Every sequence in a minimal subshift is uniformly recurrent, but the converse is not true: the union of two different minimal subshifts is not minimal but all the sequences are still uniformly recurrent. However, a sequence x is uniformly recurrent if and only if the subshift $X(x)$ is minimal.

The same holds for linear recurrence.

Part I.
Substitutions

3. Substitutions properties

Courir permet l'évasion, c'est le verbe des bandits et des héros.

Amélie Nothomb

Content

| | |
|---|----|
| 3.1. Iterable morphisms and substitutions | 26 |
| 3.2. Growth rate of letters | 27 |
| 3.3. Tame morphisms | 28 |

This chapter is preliminary to the contributions of this thesis in the theory of substitutions, it introduces important notions that are common to Chapters 4 and 5. First, Section 3.1 recalls the basic definitions of morphisms and substitutions, as well as usual properties like primitive or uniform substitutions.

Because we study general morphisms, even the non-growing or erasing ones, we have to distinguish letters based on their growth rate (defined in Section 3.2). In particular we separate growing letters and bounded letters, which are essential to study the factor complexity of substitutive systems. Let us cite Remark 4.3 of [DLR13] on this topic: "To build an S-adic sequence with a very high complexity, either you need bounded letters, or [the set of substitutions] S to be infinite and the alphabets must grow exponentially in the minimal length of the images".

Because bounded letters are so important, it is equally important to understand how they are produced. The adequate property is *tameness* (or its opposite *wildness*), it plays a central role for factor complexity and minimality. In Section 3.3 we define wild morphisms with two equivalent properties \mathfrak{P}_1 and \mathfrak{P}_2 , each of them being convenient for different purposes. In Chapter 4 \mathfrak{P}_1 is strongly related to factor complexity, and in Chapter 5 \mathfrak{P}_2 is one of the two properties that characterizes minimality.

3.1. Iterable morphisms and substitutions

In this thesis we consider exclusively *iterable* morphisms, that is, maps $\sigma : \mathcal{A}^* \rightarrow \mathcal{A}^*$ over a finite alphabet \mathcal{A} such that $\sigma(uv) = \sigma(u)\sigma(v)$ for all $u, v \in \mathcal{A}^*$. Such a map is determined by its image on every letter and we write $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$. Then the language

3. Substitutions properties – 3.2. Growth rate of letters

of the morphism σ is

$$\mathcal{L}(\sigma) := \{u \in \mathcal{A}^* \mid \exists a \in \mathcal{A}, \exists n \geq 0, u \sqsubset \sigma^n(a)\},$$

and its *incidence matrix* is $M_\sigma \in |\mathcal{A}| \times |\mathcal{A}|$ with coefficients $m_{a,b} = |\sigma(b)|_a$ for $(a, b) \in \mathcal{A} \times \mathcal{A}$. Also, the morphism naturally extends to sequences with the partial maps

$$\begin{array}{ll} \sigma : \mathcal{A}^{\mathbb{N}} \longrightarrow \mathcal{A}^{\mathbb{N}} & \sigma : \mathcal{A}^{\mathbb{Z}} \longrightarrow \mathcal{A}^{\mathbb{Z}} \\ x \longmapsto \sigma(x_0)\sigma(x_1)\sigma(x_2)\dots & x \longmapsto \dots\sigma(x_{-2})\sigma(x_{-1}).\sigma(x_0)\sigma(x_1)\sigma(x_2)\dots \end{array}$$

These maps are undefined when the sequence $x \in \mathcal{A}^{\mathbb{N}} \cup \mathcal{A}^{\mathbb{Z}}$ contains a finite number of letters that are not erased by σ (a letter $a \in \mathcal{A}$ is *erased* by σ if $\sigma(a) = \varepsilon$). A *substitution* is a morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ that is *non-erasing*, and we write $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$.

Example 3.1.1. A famous example is the Fibonacci substitution $\sigma : 0 \mapsto 01, 1 \mapsto 0$, which has the incidence matrix

$$M_\sigma = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}.$$

Here are three common types of substitutions.

Definition 3.1.2. Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution.

- We say that σ is *primitive* if its incidence matrix M_σ is primitive, i.e., there exists $n \geq 1$ such that M^n has positive entries.
- We say that σ is ℓ -uniform for some $\ell \geq 1$ if $|\sigma(a)| = \ell$ for all $a \in \mathcal{A}$.
- We say that σ is a *coding* if it is 1-uniform, and we write $\sigma : \mathcal{A} \rightarrow \mathcal{A}$.

Remark 3.1.3. An equivalent definition of a primitive substitution is that there exists $n \geq 1$ such that $a \sqsubset \sigma^n(b)$ for every $a, b \in \mathcal{A}$.

3.2. Growth rate of letters

Given a morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$, the *growth rate* of a letter $a \in \mathcal{A}$ (with regard to σ) is the asymptotic behavior of the sequence $(|\sigma^n(a)|)_{n \geq 0}$. Linear algebra provides a precise description of this behavior.

Theorem 3.2.1 ([SS78]). For each $a \in \mathcal{A}$ there exists $(\alpha_a, \beta_a) \in (\mathbb{N} \times \mathbb{R}_{\geq 1}) \cup \{(0, 0)\}$ such that $|\sigma^n(a)| = \Theta(n^{\alpha_a} \beta_a^n)$.

The case $(\alpha_a, \beta_a) = (0, 0)$ corresponds to erased letters. When studying morphisms, an important distinction has to be made between two types of letters.

Definition 3.2.2. Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ be a morphism and let $a \in \mathcal{A}$.

- If $(|\sigma^n(a)|)_{n \geq 0}$ is bounded, we say that the letter a is bounded.
- If $|\sigma^n(a)| \xrightarrow{n \rightarrow \infty} \infty$, we say that the letter a is growing.

Furthermore, \mathcal{B} denotes the set of bounded letters and \mathcal{C} denotes the set of growing letters. In particular, we say that σ is growing if $\mathcal{B} = \emptyset$.

With Theorem 3.2.1, a letter $a \in \mathcal{A}$ is bounded if and only if $(\alpha_a, \beta_a) \in \{(0, 0), (0, 1)\}$. Bounded letters are pathological in the sense that they only generate finite information, whereas growing letters are the letters that generate infinite information and allow to define sequences.

3.3. Tame morphisms

The property that describes how bounded letters are produced is *tameness*, we define it in two ways.

Definition 3.3.1. Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ be a morphism and let \mathcal{E} be the set of letters erased by σ .

- If the set of words $\mathcal{L}(\sigma) \cap \mathcal{B}^*$ is infinite, we say that σ has property \mathfrak{P}_1 .
- If there exists $c \in \mathcal{C}$, $n \geq 1$, $d \in \mathcal{B} \setminus \mathcal{E}$, $u \in \mathcal{B}^+$ and $v \in \mathcal{A}^*$ such that $d \sqsubset u$, and $\sigma^n(c) = ucv$ or $\sigma^n(c) = vcu$, we say that σ has property \mathfrak{P}_2 .

Note that a growing substitution does not have property \mathfrak{P}_1 nor \mathfrak{P}_2 .

Remark 3.3.2. The formulation of \mathfrak{P}_2 for substitutions is simpler: there exists $c \in \mathcal{C}$, $n \geq 1$, $u \in \mathcal{B}^+$ and $v \in \mathcal{A}^*$ such that $\sigma^n(c) = ucv$ or $\sigma^n(c) = vcu$.

Note that the two properties exist in the framework of D0L-systems under different names: property \mathfrak{P}_1 is called *pushyness* and property \mathfrak{P}_2 is called the *edge condition*. In this framework, Ehrenfeucht and Rozenberg showed in [ER83c, Lemma 2.1] that the two properties are equivalent. This equivalence was also proven for substitution subshifts in [MR18, Theorem 2.9] and in [Shi19, Proposition 3.17], and it easily extends to morphisms.

Proposition 3.3.3. For any morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$, the properties \mathfrak{P}_1 and \mathfrak{P}_2 are equivalent.

In Chapter 5 we provide another proof of this equivalence with Remark 5.3.35 (ii), but let us quickly show here that \mathfrak{P}_2 implies \mathfrak{P}_1 .

3. Substitutions properties – 3.3. Tame morphisms

Proof that \mathfrak{P}_2 implies \mathfrak{P}_1 . Suppose that σ has property \mathfrak{P}_2 , that is, there exist $c \in \mathcal{C}$, $n \geq 1$, $d \in \mathcal{B} \setminus \mathcal{E}$, $u \in \mathcal{B}^+$ and $v \in \mathcal{A}^*$ such that $d \sqsubset u$, and $\sigma^n(c) = uc v$ or $\sigma^n(c) = vcu$. Without loss of generality, suppose that $\sigma^n(c) = vcu$. Then a quick induction provides $v_\ell \in \mathcal{A}^*$ such that $\sigma^{n\ell}(c) = v_\ell c u \sigma(u) \dots \sigma^{\ell-1}(u)$ for all $\ell \geq 1$. Since u contains only bounded letters, among which is the letter d that is not erased by σ , each word $\sigma^\ell(u)$ contains only bounded letters and is non-empty so the words $u \sigma(u) \dots \sigma^{\ell-1}(u) \in \mathcal{L}(\sigma) \cap \mathcal{B}^*$ have arbitrarily large length when ℓ grows. Therefore σ has property \mathfrak{P}_1 . \square

We now define *tameness* with the two equivalent properties.

Definition 3.3.4. *A morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ is wild if it has property \mathfrak{P}_1 or equivalently \mathfrak{P}_2 , otherwise it is tame.*

Note that a different notion called *tameness* exists for general dynamical systems related to semigroups, but it is not related to our topic.

Authors have defined tameness with \mathfrak{P}_1 or \mathfrak{P}_2 depending on what was more convenient for their purpose. On the one hand, \mathfrak{P}_1 expresses how bounded letters accumulate. For that reason, Pansiot used it in [Pan84] to study the factor complexity of purely morphic sequences, and Maloney and Rust were the first to call *tame* and *wild* substitutions in [MR18]. On the other hand, \mathfrak{P}_2 is easy to check. For instance, Shimomura used it to characterize minimal substitution subshifts. In Chapter 5 we will also use \mathfrak{P}_2 because it allows to analyze the periodic structure of words in $\mathcal{L}(\sigma) \cap \mathcal{B}^*$.

4. Factor complexity of morphic sequences

Si ça se trouve c'est vrai, si c'est vrai ça se trouve.

Content

| | |
|---|----|
| 4.1. Preliminaries | 32 |
| 4.1.1. D0L-systems | 32 |
| 4.1.2. Morphic sequences | 32 |
| 4.1.3. Complexity classes of purely morphic sequences | 33 |
| 4.1.4. Complexity classes of morphic sequences | 34 |
| 4.2. Factor complexity of purely morphic sequences | 35 |
| 4.2.1. Classification of morphisms | 35 |
| 4.2.1.1. Growing morphisms | 35 |
| 4.2.1.2. Tame morphisms | 36 |
| 4.2.1.3. The complete classification | 37 |
| 4.2.2. Pansiot's theorem | 38 |
| 4.2.2.1. The statement | 38 |
| 4.2.2.2. Computability | 39 |
| 4.3. Factor complexity of morphic sequences | 40 |
| 4.3.1. Classification of k -blocks | 41 |
| 4.3.1.1. Order of letters | 41 |
| 4.3.1.2. k -blocks | 42 |
| 4.3.1.3. Evolutions of k -blocks | 42 |
| 4.3.1.4. Continuously periodic evolutions | 43 |
| 4.3.2. Devyatov's theorem | 44 |
| 4.3.2.1. The statement | 44 |
| 4.3.2.2. Examples | 45 |
| 4.4. Open questions | 46 |
| 4.4.1. Improved complexity classes | 46 |
| 4.4.2. New complexity classes | 46 |

Theorem 4.1.15 is not well-known, so this section also serves as a condensed review of it. We first detail the classification of morphic sequences based on the notion of k -blocks, where tameness remains a key property. Then we give the five propositions that assemble into Devyatov’s theorem. Finally we illustrate how the theorem applies to simple examples.

We conclude this chapter with two open questions: we make a conjecture for an asymptotic equivalent instead of a Θ equivalent for some complexity classes, and we discuss how the classification of morphic sequences could be completed.

4.1. Preliminaries

In this section we define D0L-systems and morphic sequences, then we explain what led to Pansiot’s theorem for purely morphic sequences and to Devyatov’s theorem for morphic sequences. Recall that, given a morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$, the set of bounded letters is denoted by \mathcal{B} and the set of growing letters is denoted by \mathcal{C} (cf. Definition 3.2.2).

4.1.1. D0L-systems

L-systems are a large class of systems originally introduced by Lindenmayer to study mathematically the development of simple filamentous organisms. For a general overview of L-systems we refer to [RS76]. The purpose of these systems is to generate a language from simple rules, among them the *D0L-systems* are *deterministic* (hence the letter D) and *context-free* (hence the digit 0).

Definition 4.1.1. A D0L-system is a triplet $G := (\mathcal{A}, \sigma, w)$ where \mathcal{A} is a finite alphabet, $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ is a morphism and w is a word in \mathcal{A}^+ . Its associated language is

$$\mathcal{L}(G) := \{u \in \mathcal{A}^* \mid \exists n \geq 0, u \sqsubset \sigma^n(w)\}.$$

The definition of the language $\mathcal{L}(G)$ is close to the language of the substitution $\mathcal{L}(\sigma) := \{u \in \mathcal{A}^* \mid \exists a \in \mathcal{A}, \exists n \geq 0, u \sqsubset \sigma^n(a)\}$, but the difference comes from the *axiom* w of G .

4.1.2. Morphic sequences

Purely morphic sequences are the primary way to construct right-infinite sequences from a single substitution.

Definition 4.1.2. If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ is a morphism and there exist $a \in \mathcal{A}$ and $u \in \mathcal{A}^+$ such that $\sigma(a) = au$ and $\sigma^n(u)$ is never the empty sequence ε , we say that the letter a is *prolongeable* (by σ). In that case the letter a is *growing* and we have $\sigma^n(a) = au\sigma(u)\sigma^2(u)\dots\sigma^{n-1}(u)$ for all $n \geq 1$, which defines the *purely morphic* sequence

$$\sigma^\infty(a) := au\sigma(u)\sigma^2(u)\sigma^3(u)\dots \in \mathcal{A}^{\mathbb{N}}.$$

4. Factor complexity of morphic sequences – 4.1. Preliminaries

Furthermore, they showed that each upper bound is attained.

Theorem 4.1.9 ([ELR75, Theorems 3, 5, 7]). *There exists a DOL-system G such that $p_G(n) = \Theta(n^2)$ (resp. $p_G(n) = \Theta(n \log n)$) (resp. $p_G(n) = \Theta(n)$).*

With Remark 4.1.5, the upper bounds of Theorem 4.1.8 also hold for purely morphic sequences. Pansiot completed these results in [Pan84]: he found the new complexity class $\Theta(n \log \log n)$ and completed the classification of the factor complexity of purely morphic sequences.

Theorem 4.1.10 ([Pan84]). *The factor complexity of a purely morphic sequence is either $\Theta(1)$, $\Theta(n)$, $\Theta(n \log \log n)$, $\Theta(n \log n)$ or $\Theta(n^2)$.*

Note that, with Theorem 2.2.5, the complexity class $\Theta(1)$ corresponds to eventually periodic sequences. More interestingly, the complexity class $\Theta(n)$ contains the Sturmian purely morphic sequences like the Fibonacci word (see σ_1 in Example 4.2.3), and the complexity class $\Theta(n^2)$ corresponds to wild morphisms (cf. Section 3.3).

4.1.4. Complexity classes of morphic sequences

With morphic sequences, we first observe that the coding decreases the factor complexity of the associated purely morphic sequence.

Lemma 4.1.11. *Let $x \in \mathcal{A}^{\mathbb{N}}$ be a sequence and let $\tau : \mathcal{A} \rightarrow \mathcal{A}$ be a coding. Then we have $p_{\tau(x)}(n) \leq p_x(n)$ for all $n \geq 0$.*

Proof. We simply observe that the map $\tau : \mathcal{L}_n(x) \rightarrow \tau(\mathcal{L}_n(x)) = \mathcal{L}_n(y)$ is surjective for every $n \geq 0$. □

Of course, the identity coding preserves the five complexity classes of Pansiot, but, interestingly, codings also create an infinite number of new complexity classes.

Proposition 4.1.12 ([Pan85, Lemma 2.2] [CN10, Proposition 4.7.2]). *For all $k \geq 1$, there exists a binary morphic sequence x such that $p_x(n) = \Theta(n^{1+1/k})$.*

Remark 4.1.13. *Once again we are faced with wild morphisms. Indeed, if a morphic sequence $y = \tau(x)$ satisfies $p_y(n) = \Theta(n^{1+1/k})$ for some $k \geq 1$, then Lemma 4.1.11 yields $p_x(n) = \Omega(p_y(n))$ and with Theorem 4.1.10 we necessarily have $p_x(n) = \Theta(n^2)$, which corresponds to wild morphisms.*

Example 4.1.14. *Let $k \geq 1$. Over the alphabet $\mathcal{A} = \{a, b_1, b_2, \dots, b_{k+1}\}$, consider the substitution defined as $\sigma(a) = ab_k$, $\sigma(b_1) = b_1$ and $\sigma(b_i) = b_i b_{i-1}$ for all $i \in \llbracket 2, k+1 \rrbracket$ and the coding $\tau : \mathcal{A}^* \rightarrow \{0, 1\}^*$ defined as $\tau(a) = 0$, $\tau(b_i) = 0$ for all $i \in \llbracket 1, k \rrbracket$ and $\tau(b_{k+1}) = 1$. Then the morphic sequence $y = \tau(\sigma^\infty(a))$ satisfies $p_y(n) = \Theta(n^{1+1/k})$.*

In fact, Devyatov showed in [Dev18] that there is no other complexity class above $\Theta(n \log n)$.

4. Factor complexity of morphic sequences – 4.2. Factor complexity of purely morphic sequences

Theorem 4.1.15 ([Dev18, Theorem 1.1]). *The factor complexity of morphic sequences is either $\Theta(n^{1+1/k})$ for some $k \geq 1$ or $\mathcal{O}(n \log n)$.*

However, the classification of morphic sequences remains incomplete because we do not know if there exist complexity classes below $n \log n$ other than the purely morphic classes $\Theta(1)$, $\Theta(n)$, $\Theta(n \log \log n)$ and $\Theta(n \log n)$ given by Theorem 4.1.10.

4.2. Factor complexity of purely morphic sequences

In order to analyze Theorem 4.1.10, we use the more precise statement Theorem 4.2.11 that works in two steps:

1. Develop a classification of morphisms based on the growth rate of letters.
2. Compute the factor complexity for each class of morphisms.

For step 1 we introduce the growth rate of letters, we summarize the classification in Figure 4.1 and we provide various examples. For step 2 we directly state Theorem 4.2.11, then we obtain a formal algorithm displayed in Figure 4.2 that computes the complexity class of a given purely morphic sequence, and we discuss its feasibility by detailing the related computability problems.

From now on, x will always denote the purely morphic sequence $\sigma^\infty(x_0) \in \mathcal{A}^\mathbb{N}$ generated by the morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ and the prolongeable letter $x_0 \in \mathcal{A}$.

Assumption 4.2.1. *Without loss of generality, we assume that every letter of \mathcal{A} occurs in x (otherwise we can remove the non-occurring letters from \mathcal{A}). In particular we have $\mathcal{L}(\sigma) = \mathcal{L}(x)$, which means that Theorem 4.1.10 also holds for the language of a morphism.*

4.2.1. Classification of morphisms

4.2.1.1. Growing morphisms

The classification of growing morphisms relies on the coefficients (α_a, β_a) associated to the growth rate of letters in Theorem 3.2.1.

Definition 4.2.2. *If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ is growing, then it belongs to one of the three categories:*

- quasi-uniform (q.u.) *if there exists $\beta \in \mathbb{R}_{>1}$ such that, for all $a \in \mathcal{A}$, $(\alpha_a, \beta_a) = (0, \beta)$,*
- polynomially divergent (p.d.) *if there exists $\beta \in \mathbb{R}_{\geq 1}$ such that, for all $a \in \mathcal{A}$, $\beta_a = \beta$, and there exists $a \in \mathcal{A}$ such that $\alpha_a \geq 1$,*
- exponentially divergent (e.d.) *if there exist two different letters $a, b \in \mathcal{A}$ such that $\beta_a > \beta_b$.*

4. Factor complexity of morphic sequences – 4.2. Factor complexity of purely morphic sequences

Note that every uniform morphism and every primitive morphism is quasi-uniform.

Example 4.2.3. We give an example for each class of growing morphisms.

- The Fibonacci substitution $\sigma_1 : 0 \mapsto 01, 1 \mapsto 0$ is primitive.
- $\sigma_2 : 0 \mapsto 1, 1 \mapsto 12, 2 \mapsto 1$ is quasi-uniform but not uniform nor primitive.
- $\sigma_3 : 0 \mapsto 010, 1 \mapsto 11$ is polynomially divergent.
- $\sigma_4 : 0 \mapsto 0010, 1 \mapsto 11$ is exponentially divergent.

It is clear that σ_1 is primitive and that σ_2 is quasi-uniform but not primitive. We treat σ_3 and σ_4 in detail in Appendix A.1.

4.2.1.2. Tame morphisms

In some cases, a purely morphic sequence generated by a non-growing morphism can be conjugated to a different purely morphic sequence generated by a growing morphism. This relies on *tameness*, which we define here with the property \mathfrak{P}_1 of Definition 3.3.1.

Definition 4.2.4. We say that a substitution σ is tame if the set $\mathcal{L}(\sigma) \cap \mathcal{B}^*$ is finite, otherwise we say that σ is wild.

Definition 4.2.5. If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ is non-growing and tame, then $\mathcal{L}(x) \cap \mathcal{B}^*$ is finite so the alphabet

$$\hat{\mathcal{A}} := \{\langle aub \rangle \mid a, b \in \mathcal{C}, u \in \mathcal{B}^*, aub \in \mathcal{L}(\sigma)\}$$

is finite. We then define the morphism

$$\begin{aligned} \hat{\sigma} : \quad \hat{\mathcal{A}} &\longrightarrow \hat{\mathcal{A}}^* \\ \langle aub \rangle &\longmapsto \langle c_1 v_1 c_2 \rangle \langle c_2 v_2 c_3 \rangle \dots \langle c_n v_n c_{n+1} \rangle \end{aligned}$$

where $c_i \in \mathcal{C}$ and $v_i, v'_i \in \mathcal{B}^*$ are such that $\sigma(au) = v_0 c_1 v_1 c_2 v_2 \dots v_{n-1} c_n v'_n$, $\sigma(b) = v''_0 c_{n+1} \dots$ and $v_n = v'_n v''_0$. In particular, $\hat{\sigma}$ is growing because every letter $\langle aub \rangle \in \hat{\mathcal{A}}$ has the same growth rate as $a \in \mathcal{C}$. Finally, we define the morphism

$$\begin{aligned} \varphi : \quad \hat{\mathcal{A}} &\longrightarrow \mathcal{A}^* \\ \langle aub \rangle &\longmapsto au \end{aligned}$$

Proposition 4.2.6. If $x := \sigma^\infty(x_0)$ is aperiodic and σ is non-growing and tame, then there exists a letter $\hat{x}_0 \in \hat{\mathcal{A}}$ that is prolongeable by $\hat{\sigma}$ such that $x = \varphi(\hat{\sigma}^\infty(\hat{x}_0))$. In particular, we have $p_x(n) = \Theta(p_{\hat{\sigma}^\infty(\hat{x}_0)}(n))$.

Proof. Let $u \in \mathcal{A}^+$ be the sequence such that $\sigma(x_0) = x_0 u$. Then u must contain a growing letter, otherwise $x = x_0 u \sigma(u) \sigma^2(u) \sigma^3(u) \dots$ would be eventually periodic. Because $\sigma^k(u)$ contains at least one growing letter for all $k \geq 0$, we can write $x = a_1 u_1 a_2 u_2 a_3 u_3 \dots$ where $a_i \in \mathcal{C}$ and $u_i \in \mathcal{B}^*$ for all $i \geq 1$. Then the letter $\hat{x}_0 := \langle a_1 u_1 a_2 \rangle \in \hat{\mathcal{A}}$ is prolongeable by $\hat{\sigma}$ and we get $x = \tau(\hat{\sigma}^\infty(\hat{x}_0))$. \square

4. Factor complexity of morphic sequences – 4.2. Factor complexity of purely morphic sequences

Example 4.2.7. Consider the Chacon substitution $\sigma_5 : 0 \mapsto 0010, 1 \mapsto 1$. We have $\mathcal{B} = \{1\}$ so it is non-growing, and we have $\mathcal{L}(\sigma) \cap \mathcal{B}^* = \{\varepsilon, 1\}$ so it is tame. We then have $\hat{\mathcal{A}} = \{\langle 00 \rangle, \langle 010 \rangle\} = \{a, b\}$, $\hat{\sigma}_5 : a \mapsto aba, b \mapsto abb$ and $\varphi : a \mapsto 0, b \mapsto 01$. In particular, $\hat{\sigma}_5$ is primitive, and Proposition 4.2.6 yields

$$\sigma_5^\infty(0) = \varphi(\hat{\sigma}_5^\infty(a)) = 0010001010010001000101001010010001010010001000101001\dots$$

Definition 4.2.8. If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ is tame, then it belongs to one of the three categories:

- virtually quasi-uniform (v.q.u.) if σ is quasi-uniform or if σ is non-growing and $\hat{\sigma}$ is quasi-uniform.
- virtually polynomially divergent (v.p.d.) if σ is polynomially divergent or if σ is non-growing and $\hat{\sigma}$ is polynomially divergent.
- virtually exponentially divergent (v.e.d.) if σ is exponentially divergent or if σ is non-growing and $\hat{\sigma}$ is exponentially divergent.

Example 4.2.9. We give a non-growing example for each class of tame morphisms, and an example of a wild morphism.

- The Chacon substitution $\sigma_5 : 0 \mapsto 0010, 1 \mapsto 1$ is virtually quasi-uniform.
- $\sigma_6 : 0 \mapsto 01010, 1 \mapsto 121, 2 \mapsto 2$ is polynomially divergent.
- $\sigma_7 : 0 \mapsto 0101010, 1 \mapsto 121, 2 \mapsto 2$ is exponentially divergent.
- $\sigma_8 : a \mapsto ab, b \mapsto bc, c \mapsto c$ is wild.

The Chacon substitution σ_5 was treated in Example 4.2.7: σ_5 is non-growing and tame and $\hat{\sigma}_5$ is primitive so σ_5 is virtually quasi-uniform. We treat σ_6, σ_7 in Appendix A.2 and σ_8 in Appendix A.3.

4.2.1.3. The complete classification

We now have the classification of morphisms that we need for step 1 of Theorem 4.2.11, we illustrate it in Figure 4.1.

Note that, up to this point, we only gave examples of morphisms that generate a purely morphic sequence that is aperiodic, but we can also find some periodic morphisms.

Example 4.2.10. The following morphisms generate a purely morphic sequence that is eventually periodic.

- $\sigma_9 : 0 \mapsto 01, 1 \mapsto 01$ is primitive.
- $\sigma_{10} : 0 \mapsto 010, 1 \mapsto 1$ is non-growing, tame and virtually quasi-uniform.
- $\sigma_{11} : 0 \mapsto 01, 1 \mapsto 1$ is wild.

We treat them in Appendix A.4.

4. Factor complexity of morphic sequences – 4.2. Factor complexity of purely morphic sequences

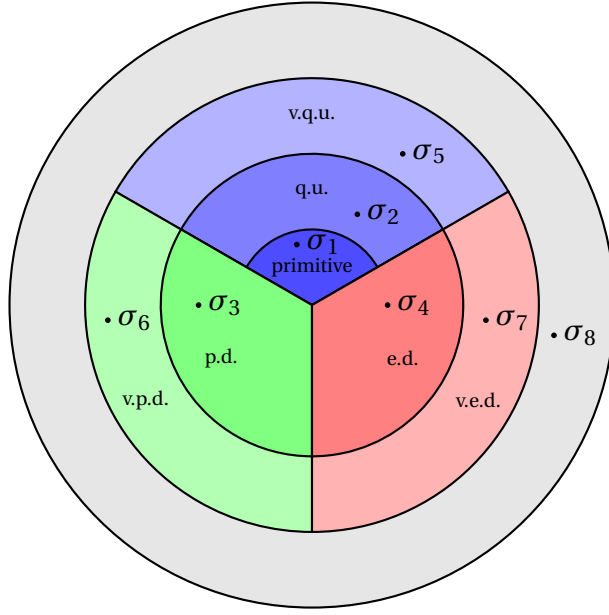


Figure 4.1.: Classification of morphisms

The outer ring represents all morphisms, the middle ring represents tame morphisms and the inner ring represents growing morphisms. The points σ_i are given in Examples 4.2.3 and 4.2.9.

4.2.2. Pansiot's theorem

4.2.2.1. The statement

With the classification presented in Figure 4.1, the precise formulation of Theorem 4.1.10 is the following.

Theorem 4.2.11 ([Pan84]). *Let $x := \sigma^\infty(x_0)$ be a purely morphic sequence that satisfies Assumption 4.2.1.*

- (i) *If x is eventually periodic, then $p_x(n) = \Theta(1)$.*
- (ii) *If x is aperiodic and σ is virtually quasi-uniform, then $p_x(n) = \Theta(n)$.*
- (iii) *If x is aperiodic and σ is virtually polynomially divergent, then $p_x(n) = \Theta(n \log \log n)$.*
- (iv) *If x is aperiodic and σ is virtually exponentially divergent, then $p_x(n) = \Theta(n \log n)$.*
- (v) *If x is aperiodic and σ is wild, then $p_x(n) = \Theta(n^2)$.*

Remark 4.2.12. (i) Assumption 4.2.1 is mandatory for Theorem 4.2.11, otherwise some letters that do not occur in x could have growth rates that modify the classification of σ .

(ii) Theorem 4.2.11 (i) is due to Theorem 2.2.5.

(iii) If σ is non-growing and tame, Proposition 4.2.6 states that $p_x(n) = \Theta(p_{\hat{\sigma}^\infty(\hat{x}_0)}(n))$ where $\hat{\sigma}$ is growing so it suffices to show Theorem 4.2.11 (ii), (iii) and (iv) for growing morphisms.

4.2.2.2. Computability

Theorem 4.2.11 provides a method to solve the following algorithmic problem:

PMClass: **Input:** A morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ and a prolongeable letter $x_0 \in \mathcal{A}$
 Output: The complexity class of the purely morphic sequence $\sigma^\infty(x_0)$

We give a formal algorithm in Figure 4.2.

```

PMClass( $\sigma, x_0$ ) :
  if  $\sigma^\infty(x_0)$  is eventually periodic:
    return " $\Theta(1)$ "
  else if  $\sigma$  is tame:
    if  $\sigma$  is non-growing:
       $\sigma = \hat{\sigma}$ 
    if  $\sigma$  is quasi uniform:
      return " $\Theta(n)$ "
    else if  $\sigma$  is polynomially divergent:
      return " $\Theta(n \log \log n)$ "
    else if  $\sigma$  is exponentially divergent:
      return " $\Theta(n \log n)$ "
  else if  $\sigma$  is wild:
    return " $\Theta(n^2)$ "
  
```

Figure 4.2.: A formal algorithm for **PMClass**

This algorithm raises three computability problems.

1. Compute the growth rate coefficients (α_a, β_a) of any letter a .
2. Decide if a purely morphic sequence is eventually periodic or not.
3. Decide if a morphism is tame or wild.

For Problem 1, we refer to the proof of Theorem 3.2.1 given in [CN10, Theorem 4.7.15] which uses the incidence matrix of morphisms. For each $a \in \mathcal{A}$, consider the incidence matrix M_a of the morphism σ restricted to the letters that appear in a $\sigma^n(a)$ for some $n \geq 0$. Then, with the Jordan normal form of M_a , β_a is the spectral radius and α_a is the degree of a polynomial.

Problem 2 was solved by Pansiot in [Pan86] by reducing the problem to irreducible morphisms and solving it in this case.

4. Factor complexity of morphic sequences – 4.3. Factor complexity of morphic sequences

For Problem 3, recall that we defined wildness with property \mathfrak{P}_1 (cf. Definition 3.3.1). Thanks to Proposition 3.3.3, property \mathfrak{P}_1 is equivalent to property \mathfrak{P}_2 , which is easily checkable.

4.3. Factor complexity of morphic sequences

Theorem 4.1.15 is obtained from Propositions 4.3.25 to 4.3.29, in order to state them we proceed in two steps:

1. Develop a combinatorial structure for the factors of the morphic sequence.
2. Compute the factor complexity from the combinatorial structure.

Step 1 begins with a refinement of the distinction between bounded and growing letters called the *order* of letters, then we introduce *k-blocks* and their *evolutions*, and we explain the crucial notion of *continuously periodic* evolutions. For step 3 we directly state the results that link the continuously periodic evolutions to the factor complexity.

With the previous section, we can already classify the morphisms that produce the complexity classes $\Theta(n^{1+1/k})$.

Lemma 4.3.1. *Let $y := \tau(\sigma^\infty(x_0))$ be a morphic sequence. If σ is tame, then we have $p_y(n) = \mathcal{O}(n \log n)$. A fortiori, if $p_y(n) = \Theta(n^{1+1/k})$ for some $k \geq 1$, then σ is non-growing and wild.*

Proof. Let $x := \sigma^\infty(x_0)$ be the purely morphic sequence such that $y = \tau(x)$. Lemma 4.1.11 states that $p_y(n) = \mathcal{O}(p_x(n))$ and σ is tame so with Theorem 4.2.11 we necessarily have $p_x(n) = \mathcal{O}(n \log n)$. \square

From now on, y will always denote the morphic sequence $\tau(\sigma^\infty(x_0)) \in \mathcal{A}^{\mathbb{N}}$ generated by the coding $\tau : \mathcal{A} \rightarrow \mathcal{A}$, the morphism $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ and the prolongeable letter $x_0 \in \mathcal{A}$, and x will always denote the purely morphic sequence $\sigma^\infty(x_0)$.

Assumption 4.3.2. *Without loss of generality, we assume that*

1. *Every letter of \mathcal{A} occurs in x (cf. Assumption 4.2.1),*
2. *σ is a substitution (i.e., it is non-erasing) thanks to [AS03, Theorem 7.7.1],*
3. *σ is non-growing and wild thanks to Lemma 4.3.1.*

4.3.1. Classification of k -blocks

4.3.1.1. Order of letters

Definition 4.3.3. Let $a \in \mathcal{A}$ have growth rate coefficients $(\alpha_a, \beta_a) \in (\mathbb{N}, \mathbb{R}_{\geq 1})$ (cf. Theorem 3.2.1).

- If $\beta_a = 1$, we say that a has order $\alpha_a + 1 \geq 1$. This means that $(|\sigma^n(a)|)_{n \geq 0}$ grows polynomially.
- If $\beta_a > 1$, we say that a has order ∞ . This means that $(|\sigma^n(a)|)_{n \geq 0}$ grows exponentially.

Remark 4.3.4. The letters of order 1 are the bounded letters.

We can then define the maximal finite order of letters.

Definition 4.3.5. With Assumption 4.3.2 3., σ has letters of order 1 and we define $\mathcal{O}(\sigma)$ as the maximal finite order of letters in \mathcal{A} .

The following lemma describes the behavior of letters depending on their order.

Lemma 4.3.6 ([Dev18, Lemma 3.1]). Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution and let $a \in \mathcal{A}$.

- (i) If a has order k , then $\sigma(a)$ contains at least one letter of order k .
- (ii) If a has order $k < \infty$ and there exists $n \geq 1$ such that $a \sqsubset \sigma^n(a)$, then for all $n \geq 0$ a unique letter c_n of order k occurs in $\sigma^n(a)$ and the sequence $(c_n)_{n \geq 0}$ is periodic.
- (iii) If a has order $k < \infty$ and $a \not\sqsubset \sigma^n(a)$ for all $n \geq 1$, then there exists n such that every letter of order k occurring in $\sigma^n(a)$ is periodic.
- (iv) If a has order ∞ , then there exists $n \geq 1$ such that $\sigma^n(a)$ contains two letters of order ∞ .
- (v) If a has order $k > 1$ and occurs in $\sigma^n(a)$ for some $n \geq 1$, then at least one letter of order $k - 1$ occurs in $\sigma^n(a)$.

Remark 4.3.7. A consequence of Lemma 4.3.6 (v) is that there exist letters of finite order k for each $k \in \llbracket 1, \mathcal{O}(\sigma) \rrbracket$.

Definition 4.3.8. Let $k \geq 1$. If $u \in \mathcal{A}^*$ contains a letter of order $> k$, we denote the first letter of order $> k$ in u by $LL_k(u)$ and the prefix before $LL_k(u)$ by $P_k(u)$ which contains only letters of order $\leq k$. Symmetrically, we denote the last letter of order $> k$ in u by $RL_k(u)$ and the suffix after $RL_k(u)$ by $S_k(u)$ which contains only letters of order $\leq k$.

Example 4.3.9. Consider the substitution $\sigma : 0 \mapsto 010, 1 \mapsto 12, 2 \mapsto 23, 3 \mapsto 3$. The letter 3 has order 1, the letter 2 has order 2, the letter 1 has order 3 and the letter 0 has order ∞ . We can then decompose the word 20123 in several ways:

$$\begin{array}{c} \boxed{\varepsilon} \boxed{2} \boxed{01} \boxed{2} \boxed{3} \\ \underbrace{\hspace{1em}}_{P_1} \underbrace{\hspace{1em}}_{LL_1} \underbrace{\hspace{1em}}_{RL_1} \underbrace{\hspace{1em}}_{S_1} \end{array} \quad \text{or} \quad \begin{array}{c} \boxed{2} \boxed{0} \boxed{1} \boxed{23} \\ \underbrace{\hspace{1em}}_{P_2} \underbrace{\hspace{1em}}_{LL_2} \underbrace{\hspace{1em}}_{RL_2} \underbrace{\hspace{1em}}_{S_2} \end{array} \quad \text{or} \quad \begin{array}{c} \boxed{2} \boxed{0} \boxed{123} \\ \underbrace{\hspace{1em}}_{P_3} \underbrace{\hspace{1em}}_{LL_3 = RL_3} \underbrace{\hspace{1em}}_{S_3} \end{array} .$$

4.3.1.2. k -blocks

In [Dev18], Devyatov introduced k -blocks as occurrences in the purely morphic sequence x . We give here an equivalent definition that uses words without considering their positions.

Definition 4.3.10. For $k \geq 1$, a k -block of $x := \sigma^\infty(x_0)$ is a triplet $(a, u, b) \in \mathcal{A} \times \mathcal{A}^* \times \mathcal{A}$ such that u contains only letters of order $\leq k$, a and b are letters of order $> k$ and $aub \sqsubset x$. In particular, if $u := (a, u, b)$ is a k -block of x , then the letter a (resp. b) is its left (resp. right) border and $\text{core}(u) := u$ is its core.

With this definition, the purely morphic sequence x can naturally be split into cores of k -blocks.

Lemma 4.3.11 ([Dev18, Lemma 4.3]). (i) If x_0 has order $K < \infty$, then x_0 is the only letter of order K in $x := \sigma^\infty(x_0)$, it only occurs once, and x has no k -block for $k \geq K - 1$. Also, for each $k \in \llbracket 1, K - 2 \rrbracket$, x splits into a concatenation of (possibly empty) cores of k -blocks and letters of order $> k$.

(ii) If x_0 has order ∞ , then x has k -blocks for all $k \geq 1$, but they do not depend on k as soon as $k \geq \mathcal{O}(\sigma)$. Also, for each $k \in \llbracket 1, \mathcal{O}(\sigma) \rrbracket$, x splits into a concatenation of (possibly empty) cores of k -blocks and letters of order $> k$.

The notion of 1-block had already appeared in the litterature because it only requires to distinguish bounded letters from growing letters. We recall that \mathcal{B} denotes the set of bounded letters and $\mathcal{C} = \mathcal{A} \setminus \mathcal{B}$ denotes the set of growing letters (with regard to σ), so that the 1-blocks of $x := \sigma^\infty(x_0)$ are the triplets $(a, u, b) \in \mathcal{C} \times \mathcal{B}^* \times \mathcal{C}$ such that $aub \sqsubset x$. Then we get that 1-blocks are very special k -blocks because they always have a strong periodic structure.

Proposition 4.3.12 ([CN10, Proposition 4.7.62]). If $x := \sigma^\infty(x_0)$ is a purely morphic sequence, then there exists a finite subset $Q \subseteq \mathcal{C} \times (\mathcal{B}^*)^5 \times \mathcal{C}$ such that $\mathcal{L}(x) \cap (\mathcal{C}\mathcal{B}^*\mathcal{C})$ equals the set of all words of the form

$$au_1v_1^k wv_2^k u_2b$$

with $(a, u_1, v_1, w, v_2, u_2, b) \in Q$ and $k \geq 1$.

4.3.1.3. Evolutions of k -blocks

The borders of k -blocks are also valuable because they contribute to building chains of k -blocks.

Definition 4.3.13. Let $k \geq 1$ and let $u = (a, u, b)$ be a k -block of $x := \sigma^\infty(x_0)$. As $aub \sqsubset x$, we have $\sigma(a)\sigma(u)\sigma(b) \sqsubset \sigma(x) = x$, and by Lemma 4.3.6 (i), $\sigma(a)$ and $\sigma(b)$ both contain a letter of order $> k$ so

$$\mathfrak{v} = (RL_k(\sigma(a)), S_k(\sigma(a))\sigma(u)P_k(\sigma(b)), LL_k(\sigma(b)))$$

is a k -block of x . We call \mathfrak{v} the descendant of u and we write $\mathfrak{v} = \mathfrak{D}(u)$.

4. Factor complexity of morphic sequences – 4.3. Factor complexity of morphic sequences

Definition 4.3.14. Let $k \geq 1$ and let $u = (a, u, b)$ be k -block of x . If there is no k -block v of x that occurs before u in x such that $u = \mathfrak{D}^\ell(v)$ for some $\ell \geq 1$, we say that u is an origin. In particular, this implies that $aub \sqsubset \sigma(c)$ for some $c \in \mathcal{A}$.

This leads to the following definition.

Definition 4.3.15. For $k \geq 1$, an evolution of k -blocks of $x := \sigma^\infty(x_0)$ is a sequence $\mathcal{E} = (\mathcal{E}_\ell)_{\ell \geq 0}$ where \mathcal{E}_0 is an origin of order k in x and $\mathcal{E}_{\ell+1} = \mathfrak{D}(\mathcal{E}_\ell)$ for all $\ell \geq 0$.

Remark 4.3.16. For all $k \geq 1$, every k -block belongs to an evolution, each evolution of k -blocks is determined by its origin and x has a finite number of evolutions of k -blocks.

Example 4.3.17. Following Example 4.3.9, the substitution σ produces

$$x := \sigma^\infty(0) = 0101201012230101201012232330101201012230101201012232332333\dots$$

The three evolutions of 1-blocks of x are

$$\begin{aligned} (0, \varepsilon, 1) &\triangleleft \\ (1, \varepsilon, 0) &\mapsto (2, \varepsilon, 0) \mapsto (2, 3, 0) \mapsto (2, 33, 0) \mapsto (2, 333, 0) \mapsto \dots \\ (1, \varepsilon, 2) &\mapsto (2, \varepsilon, 2) \mapsto (2, 3, 2) \mapsto (2, 33, 2) \mapsto (2, 333, 2) \mapsto \dots \end{aligned}$$

The two evolutions of 2-blocks of x are

$$\begin{aligned} (0, \varepsilon, 1) &\triangleleft \\ (1, \varepsilon, 0) &\mapsto (1, 2, 0) \mapsto (1, 223, 0) \mapsto (1, 223233, 0) \mapsto (1, 2232332333, 0) \mapsto \dots \end{aligned}$$

Note that $(0, \varepsilon, 1)$ and $(1, \varepsilon, 0)$ are origins of order 1 and 2 at the same time. Finally, the unique evolution of 3-blocks of x is

$$(0, 1, 0) \mapsto (0, 12, 0) \mapsto (0, 1223, 0) \mapsto (0, 1223233, 0) \mapsto (0, 12232332333, 0) \mapsto \dots$$

Lastly, let us highlight the length of k -blocks.

Lemma 4.3.18 ([Dev18, Lemma 6.13]). If $k \geq 1$ and \mathcal{E} is an evolution of k -blocks of x , then $|\text{core}(\mathcal{E}_\ell)| = \mathcal{O}(\ell^k)$.

Remark 4.3.19. This last lemma is the reason why the order of a letter was defined by $\alpha_a + 1$ instead of α_a . In Example 4.3.17 we can observe that the length of an evolution of k -blocks does not necessarily grow like $\Theta(\ell^k)$, in fact it will always grow like $\Theta(\ell^{k'})$ for some $k' \in \llbracket 0, k \rrbracket$.

4.3.1.4. Continuously periodic evolutions

To simplify the proof of Theorem 4.1.15, Devyatov replaced the substitution σ by a sufficiently large power σ^n . Then he proceeded to describe very precisely the structure of k -blocks and their evolutions, this is the part where the coding τ comes into play.

4. Factor complexity of morphic sequences – 4.3. Factor complexity of morphic sequences

This relies entirely on a property of evolutions of k -blocks called *continuous periodicity*, that has to do with the images $\tau(\text{core}(\mathcal{E}_\ell))$ of any evolution \mathcal{E} . Instead of giving the extended definition, we choose to give an approximate definition and to illustrate it progressively.

Definition 4.3.20 ([Dev18]). *For $k \geq 1$, an evolution of k -blocks \mathcal{E} of x is continuously periodic in y if the images $\tau(\text{core}(\mathcal{E}_\ell))$ have a strong periodic structure when ℓ is large.*

Let us give several cases of continuously periodic evolutions.

Remark 4.3.21. *If y is eventually periodic, then every evolution of x is continuously periodic.*

Remark 4.3.22. (i) *Due to Proposition 4.3.12, the 1-blocks already have a strong periodic structure so their image by τ has it too. Therefore, as Devyatov emphasized in [Dev18, Remark 7.17], every evolution of 1-blocks of x is continuously periodic.*

(ii) *On the contrary, for $k \geq 2$, no k -block has a strong periodic structure unless it contains only letters of order 1 or y is eventually periodic, so only the coding τ can create continuously periodic evolutions of k -blocks.*

Lemma 4.3.23. *Suppose that there exist $K \geq 1$ and $a \in \mathcal{A}$ such that $\tau(b) = a$ for all $b \in \mathcal{A}$ of order $\leq K$. Then, for all $k \in \llbracket 1, K \rrbracket$, every evolution of k -blocks of x is continuously periodic.*

Proof. Let $k \in \llbracket 1, K \rrbracket$ and let \mathcal{E} be an evolution of k -blocks of x . Then, by assumption, we have $\tau(\text{core}(\mathcal{E}_\ell)) \in \{a\}^*$ for all $\ell \geq 0$. □

We provide a less trivial example of continuously periodic evolution of 2-blocks.

Example 4.3.24. *Consider the substitution $\sigma : 0 \mapsto 010, 1 \mapsto 32123, 2 \mapsto 2, 3 \mapsto 3$ and the coding $\tau : 0 \mapsto 0, 1 \mapsto 3, 2 \mapsto 2, 3 \mapsto 3$. Then the unique evolution of 2-blocks of $\sigma^\infty(0)$ is*

$$\mathcal{E} : (0, 1, 0) \mapsto (0, 32123, 0) \mapsto (0, (32)^2 1 (23)^2, 0) \mapsto (0, (32)^3 1 (23)^3, 0) \mapsto \dots$$

We deduce that $\tau(\text{core}(\mathcal{E}_\ell)) = (32)^{2^\ell} 3$ for all $\ell \geq 0$. Therefore, \mathcal{E} is continuously periodic.

4.3.2. Devyatov's theorem

4.3.2.1. The statement

We now have all the tools to state the propositions that yield Theorem 4.1.15 (Propositions 1.2 to 1.6 in [Dev18]). For that, recall that $x := \sigma^\infty(x_0)$ is the purely morphic sequence and that $y := \tau(x)$ is the morphic sequence.

4. Factor complexity of morphic sequences – 4.3. Factor complexity of morphic sequences

Proposition 4.3.25. *If x_0 has order 2, then $p_y(n) = \Theta(1)$.*

Proof. With Lemma 4.3.6 (ii), x_0 is the only letter of order ≥ 2 in x and it occurs only once. This means that $\sigma(x_0) = x_0 u$ where u contains only bounded letters. We then have $x = x_0 u \sigma(u) \sigma^2(u) \sigma^3(u) \dots$, and the sequence $(\sigma^n(u))_{n \geq 0}$ is eventually periodic so x is eventually periodic. Finally, y is also eventually periodic and Theorem 2.2.5 yields $p_y(n) = \Theta(1)$. \square

Proposition 4.3.26. *If x_0 has finite order $K \geq 3$ and there exists $k \in \llbracket 2, K-2 \rrbracket$ such that x has an evolution of k -blocks that is not continuously periodic, let $k \in \llbracket 2, K-2 \rrbracket$ be the first of these integers. Then we have $p_y(n) = \Theta(n^{1+1/(k-1)})$.*

Proposition 4.3.27. *Suppose that x_0 has finite order $K \geq 3$ and that, for all $k \in \llbracket 2, K-2 \rrbracket$, every evolution of k -blocks of x is continuously periodic.*

(i) *If y is eventually periodic, then $p_y(n) = \Theta(1)$.*

(ii) *If y is not eventually periodic, then $p_y(n) = \Theta(n^{1+1/(K-2)})$.*

Proposition 4.3.28. *If x_0 has order ∞ and there exists $k \geq 2$ such that x has an evolution of k -blocks that is not continuously periodic, let $k \geq 2$ be the first of these integers. Then we have $p_y(n) = \Theta(n^{1+1/(k-1)})$.*

Proposition 4.3.29. *If x_0 has order ∞ and there exists $k > \mathcal{O}(\sigma)$ such that all k -blocks is continuously periodic, then we have $p_y(n) = \mathcal{O}(n \log(n))$.*

Remark 4.3.30. *Thanks to Remark 4.3.22 (i), the smallest order k for which we need to check if the evolutions are continuously periodic in Propositions 4.3.26, 4.3.27 and 4.3.29 is 2.*

4.3.2.2. Examples

Let us see how Devyatov's theorem applies to some examples of increasing difficulty. We start with examples for which the factor complexity is already known.

Example 4.3.31. *If $y = \tau(\sigma^\infty(x_0))$ is an eventually periodic morphic sequence, Theorem 2.2.5 directly states that $p_y(n) = \Theta(1)$.*

Now, in Remark 4.3.21 we saw that all its evolutions of k -blocks are continuously periodic, so if x_0 has finite order then Proposition 4.3.27 states that $p_y(n) = \Theta(1)$, and if x_0 has order ∞ then Proposition 4.3.29 states that $p_y(n) = \mathcal{O}(n \log(n))$.

Example 4.3.32. *If τ is the identity and y is aperiodic, y is simply a purely morphic sequence, and, since we assumed that σ is wild, Theorem 4.2.11 (v) states that $p_y(n) = \Theta(n^2)$.*

Now, thanks to Remark 4.3.22 (ii), for all $k \in \llbracket 2, \mathcal{O}(\sigma) \rrbracket$, no evolution of k -blocks is continuously periodic. Then, Proposition 4.3.27 (ii) states that $p_y(n) = \Theta(n^2)$ if x_0 has order 3, Proposition 4.3.26 states that $p_y(n) = \Theta(n^2)$ if x_0 has finite order $K \geq 4$ and Proposition 4.3.28 states that $p_y(n) = \Theta(n^2)$ if x_0 has order ∞ .

4. Factor complexity of morphic sequences – 4.4. Open questions

Example 4.3.33. Let $k \geq 1$, let $x := \sigma^\infty(a)$ and $y := \tau(x)$ be the sequences defined in Example 4.1.14. It was already proven by Pansiot that $p_y(n) = \Theta(n^{1+1/k})$.

Now, the letter b_i has order i for all $i \in \llbracket 1, k+1 \rrbracket$ and the letter a has order $k+2$. Then, because $\tau(b_i) = 0$ for all $i \in \llbracket 1, k \rrbracket$, Lemma 4.3.23 ensures that, for every $i \in \llbracket 1, k \rrbracket$, every evolution of i -blocks of x is continuously periodic. Also, y is aperiodic so Proposition 4.3.27 (ii) states that $p_y(n) = \Theta(n^{1+1/k})$.

Let us also study a new example.

Example 4.3.34. Following Example 4.3.24, we get that every evolution of 2-blocks of x is continuously periodic. However, the evolution of 3-blocks of x is not continuously periodic so Proposition 4.3.28 states that $p_y(n) = \Theta(n^{3/2})$.

4.4. Open questions

4.4.1. Improved complexity classes

Regarding the factor complexity of purely morphic sequences, we report a conjecture suggested by Julien Cassaigne.

Conjecture 4.4.1. *If a purely morphic sequence x satisfies $p_x(n) = \Theta(f(n))$ where $f(n) \geq n \log \log n$, then there exists a constant $C > 0$ such that $p_x(n) \sim Cf(n)$.*

This is supported by all the examples we could find in the litterature. However, this does not hold for the $\Theta(n)$ complexity: with the factor complexity of the Thue-Morse sequence (cf. Example 4.1.3), we have $\liminf_{n \rightarrow \infty} p_x(n)/n = 2$ and $\limsup_{n \rightarrow \infty} p_x(n)/n = 4$.

4.4.2. New complexity classes

After the partial classification of morphic sequences in Theorem 4.1.15, we naturally ask wether there are morphic sequences with new factor complexity below $n \log n$. So far, no such sequence has been found, which suggests the following conjecture.

Conjecture 4.4.2. *The factor complexity of morphic sequences is either $\Theta(1)$, $\Theta(n)$, $\Theta(n \log \log n)$ or $\Omega(n \log n)$.*

Another related question would be to understand what can be the complexity of a purely morphic sequence x given the factor complexity of the morphic sequence $y := \tau(x)$. We already have the following remarks:

1. Lemma 4.1.11 yields $p_x(n) = \Omega(p_y(n))$,
2. We can always have $p_y(n) = p_x(n)$ with the identity coding,
3. If $p_y(n) = \Theta(1)$, then x can have any factor complexity since we can take a coding τ that maps every letter to a single letter,

4. Factor complexity of morphic sequences – 4.4. Open questions

We summarize the possibilities in the following table.

| $p_x(n) \backslash p_y(n)$ | $\Theta(1)$ | $\Theta(n)$ | $\Theta(n \log \log n)$ | $\Theta(n \log n)$ | $\Theta(n^{1+1/k})$ |
|----------------------------|-------------|-------------|-------------------------|--------------------|---------------------|
| $\Theta(1)$ | ✓ | ✗ | ✗ | ✗ | ✗ |
| $\Theta(n)$ | ✓ | ✓ | ✗ | ✗ | ✗ |
| $\Theta(n \log \log n)$ | ✓ | ? | ✓ | ✗ | ✗ |
| $\Theta(n \log n)$ | ✓ | ? | ? | ✓ | ✗ |
| $\Theta(n^2)$ | ✓ | ? | ? | ? | ✓ |

5. Minimal components of substitution subshifts

- Maman ?
 - Qu'est-ce qu'il y a ?
 - Rien. Je voulais seulement m'assurer qu'il y a encore un joli mot qui sert à quelque chose.

Quino

Content

| | |
|---|----|
| 5.1. Preliminaries | 51 |
| 5.1.1. Substitution subshifts | 51 |
| 5.1.2. Minimal substitution subshifts | 51 |
| 5.1.3. Minimal components | 53 |
| 5.1.3.1. Minimal components for growing substitutions | 53 |
| 5.1.3.2. Minimal components for non-growing substitutions | 54 |
| 5.1.4. New results | 54 |
| 5.2. Tame minimal components | 57 |
| 5.2.1. Dynamics of alphabets | 57 |
| 5.2.1.1. Minimal alphabets | 57 |
| 5.2.1.2. ℓ -primitivity on subalphabets | 59 |
| 5.2.1.3. Dynamics of subshifts | 59 |
| 5.2.2. Proof of Theorem 5.1.24 (i) | 60 |
| 5.3. Wild minimal components | 61 |
| 5.3.1. Maximal bounded factors | 61 |
| 5.3.1.1. 1-blocks | 61 |
| 5.3.1.2. Decomposition of 1-blocks | 63 |
| 5.3.1.3. Decomposition of $LB(\sigma^k(c))$ | 64 |
| 5.3.1.4. Decomposition of $RB(\sigma^k(c))$ | 68 |
| 5.3.1.5. Decomposition of maximal bounded factors | 69 |
| 5.3.2. Proof of Theorem 5.1.24 (ii) | 70 |
| 5.4. Dynamics of minimal components | 71 |
| 5.4.1. Dynamics of tame minimal components | 71 |
| 5.4.2. Dynamics of wild minimal components | 72 |

5. Minimal components of substitution subshifts –

| | |
|--|----|
| 5.5. Counting minimal components | 73 |
| 5.5.1. Computing bounded letters | 73 |
| 5.5.2. Computing minimal components | 74 |
| 5.5.3. Upper bounds for minimal components | 74 |
| 5.6. Minimal components on two letters | 76 |
| 5.6.1. Growing substitutions | 76 |
| 5.6.1.1. $\mathcal{D} = \{0\}, \mathcal{D}' = \{1\}$ | 76 |
| 5.6.1.2. $\mathcal{D} = \{1\}, \mathcal{D}' = \{1\}$ | 77 |
| 5.6.1.3. $\mathcal{D} = \{1\}, \mathcal{D}' = \{0\}$ | 77 |
| 5.6.1.4. $\mathcal{D} = \{0, 1\}, \mathcal{D}' = \{1\}$ | 77 |
| 5.6.1.5. $\mathcal{D} = \{0, 1\}, \mathcal{D}' = \{0\}$ | 78 |
| 5.6.1.6. $\mathcal{D} = \{0, 1\}, \mathcal{D}' = \{0, 1\}$ | 78 |
| 5.6.2. Non-growing substitutions | 78 |
| 5.6.2.1. Tame substitutions | 78 |
| 5.6.2.2. Wild substitutions | 78 |
| 5.7. Open questions | 79 |

In symbolic dynamics, a very natural property of subshifts is *minimality*, meaning that the orbit of every point is dense. In this chapter we consider substitution subshifts on $\mathcal{A}^{\mathbb{Z}}$, that are generated by a single substitution. Many results were proven in this framework, for instance, Damanik and Lenz [DL06] proved that a substitution subshift is minimal if and only if it is linearly recurrent. As a simple way to construct minimal subshifts, it is well-known that *primitive* substitutions generate minimal subshifts (see for example [Que87]), like the Fibonacci substitution $\sigma_F : 0 \mapsto 01, 1 \mapsto 0$. Some non-primitive substitutions also generate minimal subshifts, like the Chacon substitution $\sigma_C : 0 \mapsto 0010, 1 \mapsto 1$, which is even *non-growing* because the letter 1 is *bounded*. Acknowledging that it is crucial to distinguish growing and bounded letters, Shimomura [Shi19] introduced a weaker version of primitivity, called ℓ -*primitivity*, that means being primitive when removing the bounded letters. Moreover, in a recent interest towards non-minimal substitution subshifts, Maloney and Rust [MR18] noticed that *tameness* (cf. Section 3.3) is essential for minimality. The idea is that a *wild* (i.e., not tame) substitution will generate an infinite periodic sequence which might prevent minimality. In his same article, Shimomura showed that these two properties characterize the minimal substitution subshifts: a substitution subshift X_σ is minimal if and only if there exists a tame and ℓ -primitive substitution σ' such that $X_\sigma = X_{\sigma'}$.

Now that minimal substitution subshifts are well-understood, the next step is to describe the structure of non-minimal substitution subshifts. In this direction, Béal, Perrin and Restivo [BPR24] showed that any substitution subshift is *quasi-minimal*, meaning that it has a finite number of subshifts; and Bezuglyi, Kwiatowski and Medynets [BKM09] showed that any growing substitution subshift on d letters has at most d minimal components. The number of subshifts - or the number of minimal

components - can then be seen as a measure of the complexity of the subshift, and this chapter is dedicated to the study of the minimal components of substitution subshifts.

For growing substitutions, we have encouraging results. The upper bound of their number in [BKM09] is already important, but it does not describe the minimal components themselves. On that note, one known family of minimal components is given by what Durand called *main sub-substitutions*: they are primitive substitutions obtained by restricting a suitable power of the original substitution to a suitable subalphabet. Each of them generates a minimal component, and they are thought to generate all the minimal components. To support this idea, the analog of main sub-substitutions was introduced by Cortez and Solomyak [CS10] in the framework of tiling substitutions on \mathbb{R}^d , and they show that the minimal components are exactly the subshifts generated by these sub-substitutions. In particular, it is remarkable that every minimal component is itself a substitution system.

For non-growing substitutions, another family of minimal components arises from wildness. The idea, mentioned earlier, originates from the closely related framework of DOL-systems, where Ehrenfeucht and Rozenberg [ER83c] begin the investigation of infinite repetitions in their language and Klouda and Starosta [KS15] provide effective results. More precisely, if a substitution is wild, then there exists a growing letter that generates a periodic sequence over bounded letters to its left or to its right. Notably, this was already used by Pansiot [Pan84] to characterize purely morphic sequences with factor complexity $\Theta(n^2)$. With the work of Maloney and Rust [MR18] and Shimomura [Shi19], this transposes to the framework of substitution subshifts, where, in particular, each such periodic sequence provides a minimal component.

Based on our paper [Hen25], this chapter presents advances in the study of the minimal components of substitution subshifts.

- In Theorem 5.1.24 we characterize the minimal components of any substitution subshift. We distinguish two types of minimal components that we study separately: the first type are smaller substitution subshifts generated by a generalization of main-substitutions that are ℓ -primitive and tame, the second type are periodic orbits over bounded letters produced by a wild substitution.
- In Propositions 5.4.1, 5.4.4 and 5.4.5 we show that the substitution induces a permutation on the minimal components that can be read from directed graphs.
- In Corollary 5.1.30, we bound the number of minimal components as a function of the alphabet size, generalizing the previously known bound to the non-growing case.

In our effort to make the characterization as effective as possible, we obtain that the number of minimal components of a given substitution subshift is computable. This is paired with a Python implementation of the computation of minimal components at <https://codeberg.org/RaphaelHENRY/MinimalComponents.git>. As a way to apply our results to various examples, we compute the minimal components for all substitutions on two letters. We conclude this chapter by opening the study to what

we call *irreducible components* and briefly discussing potential generalizations of our results to other types of subshifts.

5.1. Preliminaries

5.1.1. Substitution subshifts

Recall that, for a given substitution $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$, the set of bounded letters is denoted by \mathcal{B} and the set of bounded letters is denoted by \mathcal{C} (cf. Definition 3.2.2).

Definition 5.1.1. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. If $u \in \mathcal{A}^*$ (resp. if $X \subseteq \mathcal{A}^{\mathbb{Z}}$), we define $\text{alph}_{\mathcal{C}}(u) := \mathcal{L}(u) \cap \mathcal{C}$ (resp. $\text{alph}_{\mathcal{C}}(X) := \mathcal{L}(X) \cap \mathcal{C}$) as the set of growing letters that occurs in u (resp. in X).*

Definition 5.1.2. *If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ is a substitution and there exists a growing letter $a \in \mathcal{A}$, then the associated substitution subshift is the set*

$$X_{\sigma} := \{x \in \mathcal{A}^{\mathbb{Z}} \mid \mathcal{L}(x) \subseteq \mathcal{L}(\sigma)\}.$$

In particular, we always have $\mathcal{L}(X_{\sigma}) \subseteq \mathcal{L}(\sigma)$. If $\mathcal{A} \subset \mathcal{L}(X_{\sigma})$, we say that σ is *admissible*.

Example 5.1.3. *The substitution $\sigma : 0 \mapsto 12, 1 \mapsto 22, 2 \mapsto 11$ is not admissible because $X_{\sigma} = X^{(\omega 12^{\omega})} \cup X^{(\omega 21^{\omega})}$ and $0 \notin \mathcal{L}(X_{\sigma})$.*

5.1.2. Minimal substitution subshifts

Primitive substitutions are known to generate minimal systems.

Theorem 5.1.4 (Folklore). *If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ is a primitive substitution, then the subshift X_{σ} is minimal and linearly recurrent.*

In fact, primitivity is more than a sufficient condition for minimality.

Theorem 5.1.5 ([MR18, Theorem 2.1]). *Let $X_{\sigma} \subseteq \mathcal{A}^{\mathbb{Z}}$ be a minimal substitution subshift. Then there exists another finite alphabet \mathcal{D} and a primitive substitution $\tau : \mathcal{D} \rightarrow \mathcal{D}^+$ such that X_{τ} is topologically conjugate to X_{σ} .*

Non-primitive substitutions can also generate minimal subshifts, like the non-growing Chacon substitution $\sigma_{\mathcal{C}} : 0 \mapsto 0010, 1 \mapsto 1$. Acknowledging that it is crucial to distinguish growing and bounded letters in order to characterize minimality, Shimomura introduced in [Shi19] a weaker version of primitivity.

Definition 5.1.6. *We say that a substitution σ is ℓ -primitive if there exists $n \geq 1$ such that $a \sqsubset \sigma^n(b)$ for all $a, b \in \mathcal{C}$ (recall that \mathcal{C} (resp. \mathcal{B}) denotes the set of growing (resp. bounded) letters).*

5. Minimal components of substitution subshifts – 5.1. Preliminaries

Note that, for growing substitutions, ℓ -primitivity is the same as primitivity. Now, the ℓ -primitivity of a substitution depends only on its incidence matrix, but it is not sufficient to ensure minimality because an ℓ -primitive substitution generating a minimal subshift can have the same incidence matrix as an ℓ -primitive substitution generating a non-minimal subshift.

Example 5.1.7. Consider the two substitutions $\sigma : 0 \mapsto 0010, 1 \mapsto 1$ and $\sigma' : 0 \mapsto 0001, 1 \mapsto 1$. For both we have $\mathcal{C} = \{0\}$ and $\mathcal{B} = \{1\}$, they have the same incidence matrix and they are both ℓ -primitive. However, X_σ is known to be minimal but $\{\omega 1^\omega\} \not\subseteq X_{\sigma'}$ so $X_{\sigma'}$ is not minimal.

The missing property to solve this ambiguity is *tameness*, we define it with property $\mathfrak{P}2$ in Definition 3.3.4. First, if $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ is a substitution, we say that a letter $c \in \mathcal{C}$ is *left-isolated* (resp. *right-isolated*) if there exist $n \geq 1, u \in \mathcal{B}^+$ and $v \in \mathcal{A}^*$ such that $\sigma^n(c) = ucv$ (resp. $\sigma^n(c) = vcu$). We define \mathcal{C}_{liso} (resp. \mathcal{C}_{riso}) the set of left-isolated (resp. right-isolated) letters, and \mathcal{C}_{niso} the set of growing letters that are neither left- nor right-isolated.

Definition 5.1.8. We say that a substitution σ is tame if $\mathcal{C}_{liso} = \mathcal{C}_{riso} = \emptyset$, otherwise we say that σ is wild.

Now Proposition 3.3.3 states that a substitution σ is wild if and only if the set $\mathcal{L}(\sigma) \cap \mathcal{B}^*$ is infinite. In that case, by compactity the subshift X_σ will contain a sequence in $\mathcal{B}^{\mathbb{Z}}$, which can be an obstacle for minimality.

Example 5.1.9. Following Example 5.1.7, σ does not have property $\mathfrak{P}2$ so it is tame, but σ' has property $\mathfrak{P}2$ so it is wild and the resulting sequence $\omega 1^\omega \in \mathcal{B}^{\mathbb{Z}}$ is the obstacle to minimality.

Shimomura then showed that these two properties characterize the minimal substitution subshifts.

Theorem 5.1.10 ([Shi19, Theorem A]). Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution.

(i) Suppose that σ is tame and ℓ -primitive. Then, X_σ is minimal.

(ii) Suppose that X_σ is minimal and is not a single periodic orbit. Then σ is tame and there exists a unique subalphabet $\mathcal{D} \subseteq \mathcal{A}$ and a tame and ℓ -primitive restriction $\sigma|_{\mathcal{D}}$ such that $X_\sigma = X_{\sigma|_{\mathcal{D}}}$.

As single periodic orbits are minimal and can be expressed as a substitution subshift, this result gives an equivalent class for minimal substitution subshifts.

Corollary 5.1.11 ([Shi19, Corollary B]). Let \mathcal{M} be the class of all minimal substitution subshifts. Let \mathcal{M}' be the class of all X_σ such that σ is a tame and ℓ -primitive substitution. Then, it follows that $\mathcal{M} = \mathcal{M}'$.

This also allowed to generalize [DL06, Theorem 1].

Corollary 5.1.12 ([Shi19, Corollary 3.33]). A substitution subshift is minimal if and only if it is linearly recurrent.

5.1.3. Minimal components

If X is a subshift, a minimal subshift $Y \subseteq X$ is called a *minimal component* of X . If X has a unique minimal component, we say it is *essentially minimal*, which is not equivalent to being minimal as illustrated with Example 5.1.20. More generally, if X contains a finite number of subshifts, we say it is *quasi-minimal*. In particular, every substitution subshift is quasi-minimal.

Proposition 5.1.13 ([BPR24, Proposition 10.8]). *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^*$ be a morphism. Then X_σ is quasi-minimal.*

Note that the same result had already been shown with various assumptions on the substitution, see [MR18, Lemma 5.13] or [Ber+19, Proposition 5.14].

5.1.3.1. Minimal components for growing substitutions

When $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ is a growing substitution, Durand [Dur02] defines *main sub-substitutions*. As our goal is to generalize them, we will describe them only briefly.

By raising the incidence matrix of σ to a suitable power p , Durand introduces *principal primitive components* of \mathcal{A} , that are the disjoint alphabets $\mathcal{A}_i \subseteq \mathcal{A}$ such that $\sigma^p(\mathcal{A}_i) \subseteq \mathcal{A}_i^+$ and the restriction $\sigma^p|_{\mathcal{A}_i}$ is primitive. The $\sigma^p|_{\mathcal{A}_i}$ are called *main sub-substitutions* and they generate the minimal components $X_{\sigma^p|_{\mathcal{A}_i}}$.

Example 5.1.14. *Consider the growing substitution $\sigma : 0 \mapsto 11, 1 \mapsto 00$. Looking at σ^2 , its main sub-substitutions are $\sigma^2|_{\{0\}}$ and $\sigma^2|_{\{1\}}$, so $X_{\sigma^2|_{\{0\}}} = \{\omega 0^\omega\}$ and $X_{\sigma^2|_{\{1\}}} = \{\omega 1^\omega\}$ are minimal components of X_σ . Note that, in this example, these are the only minimal components.*

More generally, we have an upper bound of the number of minimal components that does not require to describe them.

Proposition 5.1.15 ([BKM09, Remark 5.7]). *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a growing substitution. Then X_σ has at most $|\mathcal{A}|$ minimal components.*

Let us draw a parallel with a result from Cortez and Solomyak [CS10] in the framework of admissible substitution tiling spaces on \mathbb{R}^d .

Lemma 5.1.16 ([CS10, Lemma 2.9]). *Let ω be an admissible tile substitution. Then, for all $k \geq 2$ we have $X_{\omega^k} = X_\omega$.*

This allows to replace ω by a suitable power ω^p such that its incidence matrix provides alphabets \mathcal{A}_i and primitive substitutions $\omega^p|_{\mathcal{A}_i}$ like Durand. What is remarkable in this framework is that this describes all the minimal components.

Proposition 5.1.17 ([CS10, Lemma 2.10 (i)]). *Let ω be an admissible tile substitution that provides primitive substitutions $\omega|_{\mathcal{A}_i}$. Then the minimal components of X_ω are the $X_{\omega|_{\mathcal{A}_i}}$.*

As the \mathcal{A}_i are pairwise disjoint, we get that any admissible tiling substitution system on a set of tiles \mathcal{A} has at most $|\mathcal{A}|$ minimal components, as in Proposition 5.1.15.

5.1.3.2. Minimal components for non-growing substitutions

When a substitution is not growing, the production of bounded letters is closely related to tameness because, thanks to Proposition 3.3.3, a substitution σ is wild if and only if the set $\mathcal{L}(\sigma) \cap \mathcal{B}^*$ is infinite. With property \mathfrak{P}_1 however, we can be more precise about the structure of sequences composed of bounded letters.

Lemma 5.1.18 ([MR18, Lemma 2.8],[Shi19, Lemma 3.8]). *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. If σ is wild, then X_σ contains a periodic sequence in $\mathcal{B}^{\mathbb{Z}}$.*

Proof. Let $c \in \mathcal{C}_{liso}$ (resp. $c \in \mathcal{C}_{riso}$), so that there exist $n \geq 1$, $u \in \mathcal{B}^+$ and $v \in \mathcal{A}^*$ such that $\sigma^n(c) = uc v$ or $\sigma^n(c) = vcu$. Without loss of generality, suppose that $\sigma^n(c) = vcu$. Then a quick induction provides $v_\ell \in \mathcal{A}^*$ such that $\sigma^{n\ell}(c) = v_\ell c u \sigma(u) \dots \sigma^{\ell-1}(u)$ for all $\ell \geq 1$. Since $u \in \mathcal{B}^+$ is not erased by σ , the sequence $(\sigma^\ell(u))_{\ell \geq 0}$ is eventually periodic and non-empty so, as ℓ grows, the words $\sigma^{n\ell}(c)$ produce an infinite periodic sequence in $X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$. \square

Remark 5.1.19. *The proof of Lemma 5.1.18 is in fact more precise than its statement: it shows that every letter of \mathcal{C}_{liso} (resp. of \mathcal{C}_{riso}) generates a periodic sequence $x \in X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$, therefore it generates the minimal component $X(x) \subseteq X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$.*

Example 5.1.20. *Consider the substitution $\sigma : 0 \mapsto 101, 1 \mapsto 1$, for which $\mathcal{C} = \{0\}$ and $\mathcal{B} = \{1\}$. We have $0 \in \mathcal{C}_{liso} \cap \mathcal{C}_{riso}$ and ${}^\omega 1^\omega \in X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$ so $\{{}^\omega 1^\omega\} \subseteq X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$ is a minimal component. Note that $X_\sigma = X({}^\omega 101^\omega)$ so $\{{}^\omega 1^\omega\}$ is its unique minimal component, thus X_σ is essentially minimal but not minimal.*

More generally, every sequence in $X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$ has a periodic structure.

Proposition 5.1.21 ([BPR24, Proposition 4.3]). *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. If $x \in X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$, then there exists $u, v, w \in \mathcal{B}^+$ and $k \in \mathbb{Z}$ such that $x = S^k({}^\omega u w v^\omega)$, where the lengths of u, v, w are bounded by a computable integer depending only on σ .*

We deduce a description of the minimal components of X_σ in $\mathcal{B}^{\mathbb{Z}}$.

Corollary 5.1.22. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution and let $X \subseteq \mathcal{B}^{\mathbb{Z}}$ be a minimal component of X_σ , then there exists a word $u \in \mathcal{B}^+$ such that $X = X({}^\omega u^\omega)$. In other words, every minimal component of $X_\sigma \cap \mathcal{B}^{\mathbb{Z}}$ is a single periodic orbit.*

Proof. Let $x \in X$. Then, by Proposition 5.1.21, there exist $u, v, w \in \mathcal{B}^+$ and $k \in \mathbb{Z}$ such that $x = S^k({}^\omega u w v^\omega)$. By closeness of subshifts, we have ${}^\omega u^\omega \in X$, but X is minimal so we have $X = X({}^\omega u^\omega)$. \square

5.1.4. New results

In order to state our results, we distinguish two types of minimal components.

Definition 5.1.23. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution and let X be a minimal component of X_σ . If $X \subseteq \mathcal{B}^{\mathbb{Z}}$, we say that X is wild, otherwise we say that X is tame.*

5. Minimal components of substitution subshifts – 5.1. Preliminaries

This definition is inspired by the two types of minimal components previously described. On the one hand, main sub-substitutions provide tame minimal components in the growing case, so our goal is to generalize them to the general case and to show that they are precisely the minimal components. On the other hand, Corollary 5.1.22 states that all the wild minimal components are single periodic orbits, so our goal is to give an explicit characterization that is suited for describing the dynamics, computing, and counting the minimal components. This is our main result:

Theorem 5.1.24. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution.*

(i) *The tame minimal components of X_σ are the X_τ where τ is a main sub-substitution of σ , i.e., a computable substitution of the form $\sigma^k|_{\mathcal{D} \cup \mathcal{B}}$ where \mathcal{D} is a suitable subalphabet of \mathcal{C} and k is an integer that characterizes \mathcal{D} .*

(ii) *The wild minimal components of X_σ are the $X^{(\omega LP(c)^\omega)}$ where $c \in \mathcal{C}_{liso}$ and $LP(c)$ is a computable word in \mathcal{B}^+ , and the $X^{(\omega RP(c)^\omega)}$ where $c \in \mathcal{C}_{riso}$ and $RP(c)$ is a computable word in \mathcal{B}^+ .*

Remark 5.1.25. *In the growing case, Durand’s definition of main sub-substitutions and Proposition 5.1.17 rely on raising the substitution to a suitable power. This has the benefit of simplifying the cyclic behaviour of subalphabets, but we do not do this here for several reasons:*

(i) *Lemma 5.1.16 also holds for admissible substitutions on $\mathcal{A}^{\mathbb{Z}}$ but not for non-admissible ones: in Example 5.1.3, we have $X_\sigma = X^{(\omega 12^\omega)} \cup X^{(\omega 21^\omega)} \neq X_{\sigma^2} = X^{(\omega 21^\omega)}$.*

(ii) *It would hide the dynamical aspects we focus on in Sections 5.2.1.3 and 5.4.*

(iii) *It would make computation significantly longer when we want our result to be as efficient as possible.*

In Section 5.2 we prove Theorem 5.1.24 (i). We first introduce minimal alphabets (Definition 5.2.5), not with matrices but with an oriented graph, and we show that the restrictions to these alphabets provide tame minimal components which generalize main sub-substitutions (Definition 5.2.16). To prove that all tame components have this form, we use the fact that the substitution σ induces a permutation $\tilde{\sigma}$ on the subshifts of X_σ .

In Section 5.3 we prove Theorem 5.1.24 (ii) by constructing the computable words $LP(c)$ and $RP(c)$ (Equations (2) and (3)) that depend only on σ and the letter c . By construction, ${}^\omega LP(c)^\omega$ and ${}^\omega RP(c)^\omega$ are the periodic sequences exhibited in Remark 5.1.19, and we take inspiration from a result in D0L-systems to show that they are precisely the periodic sequences in Corollary 5.1.22.

In Section 5.4 we show that $\tilde{\sigma}$ induces a permutation on the tame (resp. wild) minimal components of X_σ , and that its action is described by the directed graphs we built to prove our theorem.

Example 5.1.26. *The different constructions and results throughout Sections 5.2 to 5.4 will be illustrated with the substitution*

$$\sigma : 0 \mapsto 141, 1 \mapsto 00, 2 \mapsto 242, 3 \mapsto 5435, 4 \mapsto 5, 5 \mapsto 6, 6 \mapsto 5,$$

for which $\mathcal{C} = \{0, 1, 2, 3\}$, $\mathcal{B} = \{4, 5, 6\}$ and $\mathcal{C}_{liso} = \mathcal{C}_{riso} = \{3\}$.

5. Minimal components of substitution subshifts – 5.1. Preliminaries

Remark 5.1.27. *In practice, we display minimal components in their reduced form: for every tame component $X = X_{\sigma^k|_{\mathcal{D} \cup \mathcal{B}}}$, there is a unique alphabet \mathcal{E} such that $\mathcal{D} \subseteq \mathcal{E} \subseteq \mathcal{D} \cup \mathcal{B}$, $\sigma^k|_{\mathcal{E}}$ is defined and admissible and $X = X_{\sigma^k|_{\mathcal{E}}}$; for every wild component $X = X^{(\omega} u^\omega)$, there is a primitive word v such that $X = X^{(\omega} v^\omega)$.*

In the last two sections we emphasize the effectiveness of our characterization. In Section 5.5 we begin by showing a known computability result for which we could not find a proper proof:

Proposition 5.1.28. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. Then \mathcal{B} and \mathcal{C} are computable.*

As an application of our theorem, we are able to compute and count the minimal components of a given substitution subshift. If $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ is a substitution, the number of minimal components is denoted by $MC(\sigma)$.

Corollary 5.1.29. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. Then $MC(\sigma)$ is computable.*

In particular, we can decide if X_σ is essentially minimal. A Python implementation of the computation of \mathcal{B} and \mathcal{C} , of the minimal components of a given substitution subshift as well as their number can be found at <https://codeberg.org/RaphaelHENRY/MinimalComponents.git>. According to Remark 5.1.27, we output the tame components of the form $X_{\sigma^k|_{\mathcal{E}}}$ as the couple (\mathcal{E}, k) and the wild components as the primitive word v .

We also bound $MC(\sigma)$ by the size of the alphabet, as in Proposition 5.1.15.

Corollary 5.1.30. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution.*

- (i) *If $|\mathcal{B}| = 0$, then $MC(\sigma) \leq |\mathcal{C}| = |\mathcal{A}|$.*
- (ii) *If $|\mathcal{B}| = 1$, then $MC(\sigma) \leq |\mathcal{C}| = |\mathcal{A}| - 1$.*
- (iii) *If $|\mathcal{B}| \geq 2$, then $MC(\sigma) \leq 2|\mathcal{C}| \leq 2|\mathcal{A}| - 4$.*

We provide examples to show that these upper bounds - in fact every number between 1 and the upper bound - can be reached.

Finally, in Section 5.6 we detail the computation of the minimal components for all substitutions on two letters, which provides numerous examples to illustrate the tools developed.

Remark 5.1.31. *The non-erasing assumption on the substitution can be removed from all our results, the only difference being that computing bounded letters is more complicated.*

In particular, as D0L-systems are defined with morphisms instead of substitutions, the way we characterize, compute and count the wild components of substitution subshifts can be directly used to characterize, compute and count the infinite repetitions over bounded letters in a D0L-system.

To conclude, we discuss in Section 5.7 some open questions and generalizations of our results.

5.2. Tame minimal components

5.2.1. Dynamics of alphabets

5.2.1.1. Minimal alphabets

In this section we identify the subalphabets $\mathcal{D} \subseteq \mathcal{A}$ for which there exists $k \geq 1$ such that $\sigma^k(\mathcal{D}) \subseteq \mathcal{D}^+$ in order to study the restrictions $\sigma^k|_{\mathcal{D}} : \mathcal{D} \rightarrow \mathcal{D}^+$. Our goal is to determine when such a restriction is ℓ -primitive and tame (so that it generates a minimal substitution subshift) so we study the action of σ on the subalphabets of \mathcal{C} rather than on the subalphabets of \mathcal{A} , and we later add the bounded letters.

Definition 5.2.1. *In order to represent how σ acts on the subalphabets of \mathcal{C} , we define the directed graph $G := (V, E)$ by*

- $V := \mathcal{P}(\mathcal{C}) \setminus \{\emptyset\}$,
- $E := \left\{ \left(\mathcal{D}, \bigcup_{a \in \mathcal{D}} \text{alph}_{\mathcal{C}}(\sigma(a)) \right) \mid \mathcal{D} \in V \right\} \subseteq V^2$.

If $(\mathcal{D}, \mathcal{D}') \in E$, we write $\mathcal{D} \rightarrow \mathcal{D}'$. If $k \geq 1$ and $\mathcal{D} \xrightarrow[k \text{ times}]{\dots} \mathcal{D}'$, we write $\mathcal{D} \xrightarrow[k]{\dots} \mathcal{D}'$, and in that

case $\mathcal{D}' = \bigcup_{a \in \mathcal{D}} \text{alph}_{\mathcal{C}}(\sigma^k(a))$.

Example 5.2.2. *Following Example 5.1.26, the graph G has 15 vertices so let us display the orbit of the singletons only:*



The following lemma shows that G behaves well with inclusion and union.

Lemma 5.2.3. *Let $k \geq 1$ and $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \mathcal{D}_4 \in V$.*

(i) *If $\mathcal{D}_1 \subseteq \mathcal{D}_2$, $\mathcal{D}_1 \xrightarrow[k]{\dots} \mathcal{D}_3$ and $\mathcal{D}_2 \xrightarrow[k]{\dots} \mathcal{D}_4$, then $\mathcal{D}_3 \subseteq \mathcal{D}_4$.*

(ii) *If $\mathcal{D}_1 \xrightarrow[k]{\dots} \mathcal{D}_3$ and $\mathcal{D}_2 \xrightarrow[k]{\dots} \mathcal{D}_4$, then $\mathcal{D}_1 \cup \mathcal{D}_2 \xrightarrow[k]{\dots} \mathcal{D}_3 \cup \mathcal{D}_4$.*

Proof. The proof is left to the reader. □

Remark 5.2.4. *Lemma 5.2.3 (ii) implies that the graph G is entirely determined by the alphabets \mathcal{D}_a such that $\{a\} \rightarrow \mathcal{D}_a$ for each $a \in \mathcal{C}$. We call these alphabets the generators of G , they are particularly relevant when we do computations.*

New let us identify the cyclic behaviors in G .

Definition 5.2.5. *We say that $\mathcal{D} \in V$ is a k -periodic alphabet if k is the smallest positive integer such that $\mathcal{D} \xrightarrow[k]{\dots} \mathcal{D}$. We say that $\mathcal{D} \in V$ is a minimal alphabet if there exists $k \geq 1$ such that \mathcal{D} is k -periodic and has no proper periodic subalphabet.*

5. Minimal components of substitution subshifts – 5.2. Tame minimal components

Note that every periodic alphabet contains at least one minimal alphabet.

Lemma 5.2.6. *Let \mathcal{D} be a minimal alphabet of period k and let \mathcal{E} such that $\mathcal{D} \rightarrow \mathcal{E}$. Then \mathcal{E} is a minimal alphabet of period k .*

Proof. First, \mathcal{E} is a k -periodic alphabet. Let $\mathcal{E}' \subseteq \mathcal{E}$ be a k' -periodic subalphabet. Also let \mathcal{D}' be the k' -periodic alphabet such that $\mathcal{E}' \xrightarrow[k-1]{k-1} \mathcal{D}'$. We have $\mathcal{E} \xrightarrow[k-1]{k-1} \mathcal{D}$ so, by Lemma 5.2.3 (i), $\mathcal{D}' \subseteq \mathcal{D}$. Then, as \mathcal{D} is minimal, we have $\mathcal{D}' = \mathcal{D}$ and $k' = k$. Finally, we have $\mathcal{E}' \xrightarrow[k]{k} \mathcal{E}'$ and $\mathcal{E}' \xrightarrow[k-1]{k-1} \mathcal{D} \rightarrow \mathcal{E}$ so $\mathcal{E}' = \mathcal{E}$. \square

We now show an equivalent property to the minimality of alphabets, which will be easier to handle in the proofs.

Lemma 5.2.7. *Let \mathcal{D} be a k -periodic alphabet. Then \mathcal{D} is minimal if and only if for all $a \in \mathcal{D}$, there exists $\ell_a \geq 1$ such that $\{a\} \xrightarrow[k\ell_a]{k\ell_a} \mathcal{D}$.*

Proof. Suppose that \mathcal{D} is minimal. Let $a \in \mathcal{D}$, and for all $\ell \geq 1$, let \mathcal{D}_ℓ be the subalphabet such that $\{a\} \xrightarrow[k\ell]{k\ell} \mathcal{D}_\ell$. Then, as $\{a\} \subseteq \mathcal{D}$ and $\mathcal{D} \xrightarrow[k\ell]{k\ell} \mathcal{D}$, by Lemma 5.2.3 (i), $\mathcal{D}_\ell \subseteq \mathcal{D}$. As V is finite, the sequence $(\mathcal{D}_\ell)_{\ell \geq 1}$ is eventually periodic, which means that there exists $\ell_a \geq 1$ such that \mathcal{D}_{ℓ_a} is a periodic alphabet. Then, by minimality of \mathcal{D} , $\mathcal{D}_{\ell_a} = \mathcal{D}$.

Suppose that for all $a \in \mathcal{D}$, there exists $\ell_a \geq 1$ such that $\{a\} \xrightarrow[k\ell_a]{k\ell_a} \mathcal{D}$. Let $\mathcal{D}' \subseteq \mathcal{D}$ be a minimal k' -periodic alphabet and let $a \in \mathcal{D}'$. By supposition, there exists ℓ_a such that $\{a\} \xrightarrow[k\ell_a]{k\ell_a} \mathcal{D}$. Moreover, by the previous implication, there exists $\ell'_a \geq 1$ such that $\{a\} \xrightarrow[k'\ell'_a]{k'\ell'_a} \mathcal{D}'$. By setting $\ell = \max(\ell_a, \ell'_a)$, we get $\{a\} \xrightarrow[kk'\ell]{kk'\ell} \mathcal{D}$ and $\{a\} \xrightarrow[kk'\ell]{kk'\ell} \mathcal{D}'$, so $\mathcal{D} = \mathcal{D}'$. Hence \mathcal{D} is minimal. \square

Remark 5.2.8. *Lemma 5.2.7 implies that the minimal alphabets are in the orbit of the singletons $\{a\}$ for $a \in \mathcal{C}$. Therefore, when searching for minimal alphabets we only need to compute these orbits instead of the entire graph G .*

Example 5.2.9. *Following Example 5.2.2, the minimal alphabets are the 2-periodic alphabets $\{0\}$ and $\{1\}$ and the 1-periodic alphabets $\{2\}$ and $\{3\}$.*

We also prove a natural fact that will be useful when we count minimal components in Section 5.5.

Proposition 5.2.10. *The minimal alphabets are pairwise disjoint.*

Proof. Let \mathcal{D} and \mathcal{D}' be two minimal alphabets of respective period k and k' such that $\mathcal{D} \cap \mathcal{D}' \neq \emptyset$. Let $a \in \mathcal{D} \cap \mathcal{D}'$. By Lemma 5.2.7, there exists $\ell_a \geq 1$ such that $\{a\} \xrightarrow[k\ell_a]{k\ell_a} \mathcal{D}$ and $\ell'_a \geq 1$ such that $\{a\} \xrightarrow[k'\ell'_a]{k'\ell'_a} \mathcal{D}'$. By setting $\ell = \max(\ell_a, \ell'_a)$, we get $\{a\} \xrightarrow[kk'\ell]{kk'\ell} \mathcal{D}$ and $\{a\} \xrightarrow[kk'\ell]{kk'\ell} \mathcal{D}'$, hence $\mathcal{D} = \mathcal{D}'$. \square

5. Minimal components of substitution subshifts – 5.2. Tame minimal components

5.2.1.2. ℓ -primitivity on subalphabets

If \mathcal{D} is a k -periodic alphabet, we have the restriction $\sigma^k|_{\mathcal{D} \cup \mathcal{B}} : \mathcal{D} \cup \mathcal{B} \rightarrow (\mathcal{D} \cup \mathcal{B})^+$. We show here that the restrictions to minimal alphabets are precisely the ℓ -primitive sub-substitutions of σ .

Proposition 5.2.11. *Let \mathcal{D} be a k -periodic alphabet. Then, \mathcal{D} is minimal if and only if $\sigma^k|_{\mathcal{D} \cup \mathcal{B}}$ is ℓ -primitive.*

Proof. Thanks to Lemma 5.2.7, we are going to show that $\sigma^k|_{\mathcal{D} \cup \mathcal{B}}$ is ℓ -primitive if and only if for all $a \in \mathcal{D}$, there exists $l_a \geq 1$ such that $\{a\} \xrightarrow[k\ell_a]{} \mathcal{D}$.

Suppose that $\sigma^k|_{\mathcal{D} \cup \mathcal{B}}$ is ℓ -primitive, which provides $n \geq 1$ such that $a \sqsubset \sigma^{kn}(b)$ for all $a, b \in \mathcal{D}$. Then, for all $a \in \mathcal{D}$, we have $\text{alph}_{\mathcal{C}}(\sigma^{kn}(a)) = \mathcal{D}$, which yields $\{a\} \xrightarrow[kn]{} \mathcal{D}$.

Suppose that, for all $a \in \mathcal{D}$, there exists $\ell_a \geq 1$ such that $\{a\} \xrightarrow[k\ell_a]{} \mathcal{D}$. By setting $n = \max_{a \in \mathcal{D}} \ell_a$, we get $\{a\} \xrightarrow[kn]{} \mathcal{D}$ for all $a \in \mathcal{D}$, i.e., $\text{alph}_{\mathcal{C}}(\sigma^{kn}(a)) = \mathcal{D}$, which means that $\sigma^k|_{\mathcal{D} \cup \mathcal{B}}$ is ℓ -primitive. \square

5.2.1.3. Dynamics of subshifts

A substitution $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ induces a map $\tilde{\sigma}$ on the subshifts of X_σ : if $X \subseteq X_\sigma$ is a subshift, we define the subshift

$$\tilde{\sigma}(X) := \left\{ S^k(\sigma(x)) \mid x \in X, k \in \mathbb{Z} \right\} \subseteq X_\sigma.$$

We recall that every sequence in a substitution subshift can be desubstituted.

Lemma 5.2.12. *For all $x \in X_\sigma$, there exist $y \in X_\sigma$ and $k \in \mathbb{Z}$ such that $x = S^k(\sigma(y))$.*

A proof of this lemma can be found in [BPR23, Theorem 5.1] for example. This means that the reciprocal of $\tilde{\sigma}$ is defined, it is the subshift

$$\tilde{\sigma}^{-1}(X) := \{x \in X_\sigma \mid \sigma(x) \in X\} \subseteq X_\sigma,$$

such that, for all subshift $X \subseteq X_\sigma$, $\tilde{\sigma}(\tilde{\sigma}^{-1}(X)) = X$. More generally, for all subshift $X \subseteq X_\sigma$ and all $n \geq 1$, $\tilde{\sigma}^n(\tilde{\sigma}^{-n}(X)) = X$.

Remark 5.2.13. *If $X \subseteq X_\sigma$ is a subshift such that $\text{alph}_{\mathcal{C}}(X) \neq \emptyset$, then $\text{alph}_{\mathcal{C}}(\tilde{\sigma}(X)) \neq \emptyset$ and $\text{alph}_{\mathcal{C}}(X) \rightarrow \text{alph}_{\mathcal{C}}(\tilde{\sigma}(X))$. More generally, for $n \geq 1$, $\text{alph}_{\mathcal{C}}(X) \xrightarrow[n]{} \text{alph}_{\mathcal{C}}(\tilde{\sigma}^n(X))$.*

We can now prove that the subshifts of X_σ have a strong cyclic behavior.

Proposition 5.2.14. *Let $X \subseteq X_\sigma$ be a subshift. Then there exists $k \geq 1$ such that $\tilde{\sigma}^k(X) = X$, and if in addition $\text{alph}_{\mathcal{C}}(X) \neq \emptyset$, then $\text{alph}_{\mathcal{C}}(X)$ is a k -periodic alphabet and we have $X_{\sigma^k|_{\text{alph}_{\mathcal{C}}(X) \cup \mathcal{B}}} \subseteq X$.*

5. Minimal components of substitution subshifts – 5.2. Tame minimal components

Proof. Consider the sequence of subshifts $(\tilde{\sigma}^{-n}(X))_{n \geq 0}$. By Proposition 5.1.13, X_σ has a finite number of subshifts so this sequence is ultimately periodic. By setting $\ell \geq 0$ and $k \geq 1$ the smallest integers such that $\tilde{\sigma}^{-\ell}(X) = \tilde{\sigma}^{-(\ell+k)}(X)$, we obtain

$$\tilde{\sigma}^k(X) = \tilde{\sigma}^{\ell+k}(\tilde{\sigma}^{-\ell}(X)) = \tilde{\sigma}^{\ell+k}(\tilde{\sigma}^{-(\ell+k)}(X)) = X.$$

If $\text{alph}_C(X) \neq \emptyset$, with Remark 5.2.13 we have $\text{alph}_C(X) \xrightarrow[k]{\tilde{\sigma}^k} \text{alph}_C(\tilde{\sigma}^k(X)) = \text{alph}_C(X)$, which means that $\text{alph}_C(X)$ is k -periodic. Let $u \in \mathcal{L}\left(X_{\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}}\right)$, then there exist $c \in \text{alph}_C(X)$ and $\ell \geq 1$ such that $u \sqsubset \sigma^{k\ell}(c)$. As $c \in \text{alph}_C(X)$, there exists $x \in X$ such that $c \sqsubset x$, then $u \sqsubset \sigma^{k\ell}(c) \sqsubset \sigma^{k\ell}(x) \in \tilde{\sigma}^{k\ell}(X) = X$ so $u \in \mathcal{L}(X)$. Therefore $\mathcal{L}\left(X_{\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}}\right) \subseteq \mathcal{L}(X)$, which yields $X_{\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}} \subseteq X$. \square

Remark 5.2.15. This means that $\tilde{\sigma}$ is a permutation on the subshifts of X_σ .

5.2.2. Proof of Theorem 5.1.24 (i)

We are now able to generalize main sub-substitutions to the general case.

Definition 5.2.16. Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. If $\mathcal{D} \subseteq \mathcal{C}_{\text{niiso}}$ is a minimal alphabet of period k , we say that the substitution $\sigma^k|_{\mathcal{D} \cup \mathcal{B}}$ is a main sub-substitution of σ . In particular, main sub-substitutions are tame (because $\mathcal{D} \subseteq \mathcal{C}_{\text{niiso}}$) and ℓ -primitive (by Proposition 5.2.11).

Example 5.2.17. Following Example 5.2.9, we recall that $\mathcal{C}_{\text{niiso}} = \{0, 1, 2\}$ so the minimal alphabets included in $\mathcal{C}_{\text{niiso}}$ are $\{0\}$ of period 2, $\{1\}$ of period 2 and $\{2\}$ of period 1. We then add the bounded letters that appear in the associated subshift, so the main sub-substitutions of σ are $\sigma^2|_{\{0,5\}}$, $\sigma^2|_{\{1,4,6\}}$ and $\sigma|_{\{2,4,5,6\}}$.

Theorem 5.1.10 (i) directly provides the following result.

Proposition 5.2.18. Let τ be a main-substitution of σ . Then X_τ is a minimal component of X_σ .

The converse relies on Proposition 5.2.14.

Proposition 5.2.19. Let X be a tame minimal component of X_σ . Then there exists a main sub-substitution τ of σ such that $X = X_\tau$.

Proof. First, Proposition 5.2.14 provides $k \geq 1$ such that $\text{alph}_C(X)$ is k -periodic and $X_{\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}} \subseteq X$. Then, by minimality of X , we have equality. It remains to show that $\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}$ is a main sub-substitution of σ .

As $\text{alph}_C(X) \neq \emptyset$ and X is minimal, growing letters occur with bounded gaps in X . In other words, the words in $\mathcal{L}\left(X_{\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}}\right) \cap \mathcal{B}^*$ have bounded length, so, by Proposition 3.3.3, $\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}$ is tame, i.e., $\text{alph}_C(X) \subseteq \mathcal{C}_{\text{niiso}}$.

Proposition 5.2.14 also states that $\tilde{\sigma}^k(X) = X$, so we have $\sigma^{k\ell}(d) \in \mathcal{L}(X)$ for all $d \in \text{alph}_C(X)$ and all $\ell \geq 0$. As X is minimal, for ℓ large enough, $\sigma^{k\ell}(d)$ contains

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

every growing letter of $\text{alph}_C(X)$. This means that $\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}$ is ℓ -primitive, and by Proposition 5.2.11 $\text{alph}_C(X)$ is minimal. Therefore, $\sigma^k|_{\text{alph}_C(X) \cup \mathcal{B}}$ is a main substitution. \square

Propositions 5.2.18 and 5.2.19 complete the proof of Theorem 5.1.24 (i).

Example 5.2.20. *Following Example 5.2.17, the tame minimal components of X_σ are $X_{\sigma^2|_{\{0,5\}}}$, $X_{\sigma^2|_{\{1,4,6\}}}$ and $X_{\sigma|_{\{2,4,5,6\}}}$.*

5.3. Wild minimal components

5.3.1. Maximal bounded factors

The purpose of this section is to prove Proposition 5.3.34, which is an analog of a result proved for D0L-systems:

Proposition 5.3.1 ([KS15, Theorem 12]). *Let $G = (A, \sigma, w)$ be a pushy D0L-system (i.e., σ is wild). Then there exists $L \geq 1$ and a finite subset $U \subseteq \mathcal{B}^+$ such that any factor from $\mathcal{L}(G) \cap \mathcal{B}^*$ has one of the following forms:*

- w_1
 - $w_1 u_1^{k_1} w_2$
 - $w_1 u_1^{k_1} w_2 u_2^{k_2} w_3$
- where $u_1, u_2 \in U$, $|w_j| < L$ for all $j \in \{1, 2, 3\}$, and $k_1, k_2 \geq 1$.

Note that Proposition 4.3.12 is a similar result for purely morphic sequences. We are going to be more precise by explicitly defining the words u_1 and u_2 as the computable words $LP(c)$ and $RP(c)$ for specific growing letters c . To achieve that, we define the following.

Definition 5.3.2. *If $c \in \mathcal{C}$ and $k \geq 1$, we say that v is a maximal bounded factor of $\sigma^k(c)$ if $v \in \mathcal{L}(\sigma^k(c)) \cap \mathcal{B}^*$ and it is not a factor of any other word of $\mathcal{L}(\sigma^k(c)) \cap \mathcal{B}^*$. We also say that a word is a maximal bounded factor of σ if it is a maximal bounded factor of a $\sigma^k(c)$ for some $c \in \mathcal{C}$ and $k \geq 0$.*

The purpose of maximal bounded factors is that they contain the structure of every word of $\mathcal{L}(X_\sigma) \cap \mathcal{B}^*$, as highlighted by the following remark.

Remark 5.3.3. *As only growing letters generate images of arbitrarily large length, for all $u \in \mathcal{L}(X_\sigma) \cap \mathcal{B}^*$, there exists $c \in \mathcal{C}$ and $k \geq 0$ such that $u \sqsubset \sigma^k(c)$. In particular, u is a factor of a maximal bounded factor of $\sigma^k(c)$.*

5.3.1.1. 1-blocks

We adapt here the notion of 1-blocks, introduced by Devyatov for morphic sequences (cf. Section 4.3.1.2), to substitution subshifts. For that we introduce different notations.

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

Definition 5.3.4. If $u \in \mathcal{A}^*$ contains a growing letter, we denote the first growing letter in u by $LC(u)$ and the prefix before $LC(u)$ by $LB(u) \in \mathcal{B}^*$. Symmetrically, we denote the last growing letter in u by $RC(u)$ and the suffix after $RC(u)$ by $RB(u) \in \mathcal{B}^*$.

Example 5.3.5. If $\mathcal{C} = \{0, 1\}$ and $\mathcal{B} = \{2\}$, the words 22012001 and 202 have the following decomposition:

$$\begin{array}{|c|c|c|c|c|} \hline 22 & 0 & 1200 & 1 & \varepsilon \\ \hline \end{array} \quad \text{and} \quad \begin{array}{|c|c|c|} \hline 2 & 0 & 2 \\ \hline \end{array}$$

$\underbrace{\hspace{1.5cm}}_{LB} \quad \underbrace{\hspace{1.5cm}}_{LC} \quad \underbrace{\hspace{1.5cm}}_{RC} \quad \underbrace{\hspace{1.5cm}}_{RB}$ $\underbrace{\hspace{1.5cm}}_{LB} \quad \underbrace{\hspace{1.5cm}}_{LC=RC} \quad \underbrace{\hspace{1.5cm}}_{RB}$

Definition 5.3.6. If $c \in \mathcal{C}$ and $k \geq 1$, a 1-block of $\sigma^k(c)$ is a triplet $(a, u, b) \in \mathcal{C} \times \mathcal{B}^* \times \mathcal{C}$ such that $aub \sqsubset \sigma^k(c)$. In general, we say that (a, u, b) is a 1-block of σ if it is a 1-block of $\sigma^k(c)$ for some $c \in \mathcal{C}$ and $k \geq 1$.

We now note an important fact.

Remark 5.3.7. Let $c \in \mathcal{C}$ and $k \geq 1$. The maximal bounded factors of $\sigma^k(c)$ have one of the following forms:

- $LB(\sigma^k(c))$,
- u where (a, u, b) is a 1-block of $\sigma^k(c)$,
- $RB(\sigma^k(c))$.

Note that a 1-block of σ can be a 1-block of several $\sigma^k(c)$, but we need to consider a special case.

Definition 5.3.8. A 1-block of $\sigma(c)$ for some $c \in \mathcal{C}$ is called an origin of σ . In particular, σ has a finite number of origins.

We now show that every 1-block of σ can be recovered from an origin of σ .

Definition 5.3.9. Let $u = (a, u, b)$ be a 1-block of $\sigma^k(c)$ for some $c \in \mathcal{C}$ and $k \geq 1$. As $aub \sqsubset \sigma^k(c)$, we have $\sigma(a) \sigma(u) \sigma(b) \sqsubset \sigma^{k+1}(c)$, and in particular

$$v = (RC(\sigma(a)), RB(\sigma(a)) \sigma(u) LB(\sigma(b)), LC(\sigma(b)))$$

is a 1-block of $\sigma^{k+1}(c)$. We call v the descendant of u and we write $v = \mathfrak{D}(u)$. For $\ell \geq 0$, we also write \mathfrak{D}^ℓ the ℓ -th iteration of \mathfrak{D} , in particular $\mathfrak{D}^0(u) = u$.

Lemma 5.3.10. Let $c \in \mathcal{C}$. For all $k \geq 1$ and all 1-block u of $\sigma^k(c)$, there exist an origin v of σ and $0 \leq \ell < k$ such that $u = \mathfrak{D}^\ell(v)$.

Proof. Let us prove it by induction on k . If $k = 1$, u is itself an origin of σ and $u = \mathfrak{D}^0(u)$. Let $k \geq 1$ be such that, for every $c \in \mathcal{C}$ and every 1-block u of $\sigma^k(c)$, there exist an origin v of σ and $0 \leq \ell < k$ such that $u = \mathfrak{D}^\ell(v)$. Let $c \in \mathcal{C}$ and let $u = (a, u, b)$ be a 1-block of $\sigma^{k+1}(c)$. When desubstituting once, the growing letters a and b come from growing letters in $\sigma^k(c)$, and we have two possibilities:

- Either a and b come from the same growing letter $a' \sqsubset \sigma^k(c)$, which means that $aub \sqsubset \sigma(a') \sqsubset \sigma^{k+1}(c)$. This is illustrated by Figure 5.1:

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

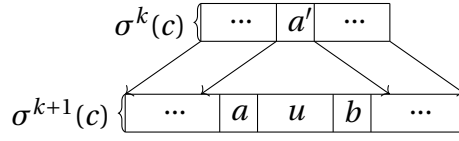


Figure 5.1.: a and b come from the same growing letter.

Then $aub \sqsubset \sigma(a')$ means that u is itself an origin.

• Or a and b come from two different growing letters $a', b' \sqsubset \sigma^k(c)$, which means that there exists $u' \in \mathcal{A}^*$ such that $a'u'b' \sqsubset \sigma^k(c)$ and $aub \sqsubset \sigma^{k+1}(c)$. This is illustrated by Figure 5.2:

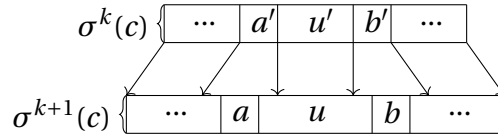


Figure 5.2.: a and b come from two different growing letters.

As $\sigma(u') \sqsubset u \in \mathcal{B}^*$, we have $u' \in \mathcal{B}^*$. In particular, $u' = (a', u', b')$ is a 1-block of $\sigma^k(c)$ so, by hypothesis, there exists an origin v and $0 \leq \ell < k$ such that $u' = \mathcal{D}^\ell(v)$. We have $a = RC(\sigma(a'))$ and $b = LC(\sigma(b'))$ so $u = RB(\sigma(a')) \sigma(u') LB(\sigma(b'))$, therefore $u = \mathcal{D}(u') = \mathcal{D}^{\ell+1}(v)$. \square

This leads to the following definition.

Definition 5.3.11. An evolution of 1-blocks of σ is a sequence $\mathcal{E} = (\mathcal{E}_\ell)_{\ell \geq 0}$ where \mathcal{E}_0 is an origin of σ and, for all $\ell \geq 1$, $\mathcal{E}_\ell = \mathcal{D}^\ell(\mathcal{E}_0)$. In particular, each evolution is determined by its origin, σ has a finite number of evolutions of 1-blocks and Lemma 5.3.10 means that every 1-block of σ belongs to an evolution.

5.3.1.2. Decomposition of 1-blocks

In order to study the structure of 1-blocks, let us start with a useful lemma.

Lemma 5.3.12. Let u be a word containing a growing letter. Then

$$LB(\sigma(u)) = \sigma(LB(u)) LB(\sigma(LC(u))),$$

$$RB(\sigma(u)) = RB(\sigma(RC(u))) \sigma(RB(u)).$$

Proof. We have $LB(\sigma(u)) = LB(\sigma(LB(u)) \sigma(LC(u)) \dots) = \sigma(LB(u)) LB(\sigma(LC(u)))$. The right case is symmetric. \square

We deduce a decomposition of 1-blocks.

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

Lemma 5.3.13. *Let \mathcal{E} be an evolution of 1-blocks of σ and set $\mathcal{E}_0 = (a, u, b)$. For all $\ell \geq 0$, we have*

$$\mathcal{E}_\ell = \left(RC(\sigma^\ell(a)), RB(\sigma^\ell(a)) \sigma^\ell(u) LB(\sigma^\ell(b)), LC(\sigma^\ell(b)) \right).$$

Proof. Let us prove it by induction on ℓ . If $\ell = 0$, this is directly true. Let $\ell \geq 0$ such that $\mathcal{E}_\ell = (RC(\sigma^\ell(a)), RB(\sigma^\ell(a)) \sigma^\ell(u) LB(\sigma^\ell(b)), LC(\sigma^\ell(b)))$. If $\mathcal{E}_{\ell+1} = (a', u', b')$, we have $\mathcal{E}_{\ell+1} = \mathfrak{D}(\mathcal{E}_\ell)$ so $a' = RC(\sigma(RC(\sigma^\ell(a)))) = RC(\sigma^{\ell+1}(a))$,

$$\begin{aligned} u' &= RB(\sigma(RC(\sigma^\ell(a)))) \sigma(RB(\sigma^\ell(a))) \sigma^{\ell+1}(u) \sigma(LB(\sigma^\ell(b))) LB(\sigma(LC(\sigma^\ell(b)))) \\ &= RB(\sigma^{\ell+1}(a)) \sigma^{\ell+1}(u) LB(\sigma^{\ell+1}(b)) \text{ with Lemma 5.3.12,} \end{aligned}$$

and $b' = LC(\sigma(LC(\sigma^\ell(b)))) = LC(\sigma^{\ell+1}(b))$. □

This lemma allows us to refine Remark 5.3.7:

Remark 5.3.14. *Let $c \in \mathcal{C}$ and $k \geq 1$. The maximal bounded factors of $\sigma^k(c)$ have one of the following forms:*

- (i) $LB(\sigma^k(c))$,
- (ii) $RB(\sigma^\ell(a)) \sigma^\ell(u) LB(\sigma^\ell(b))$ where $0 \leq \ell < k$ and (a, u, b) is an origin of σ ,
- (iii) $RB(\sigma^k(c))$.

Now let us decompose the words $LB(\sigma^k(c))$ and $RB(\sigma^k(c))$ for $c \in \mathcal{C}$ and $k \geq 0$.

5.3.1.3. Decomposition of $LB(\sigma^k(c))$

Decomposing the words $LB(\sigma^k(c))$ is strongly related to the prefix automaton of the substitution, but we take here a different approach. To begin with, we will heavily use the following notation: if $(u_i)_{0 \leq i \leq n-1} \in (\mathcal{A}^*)^n$, we write the concatenation from left to right as

$$\bigsqcup_{i=0}^{n-1} u_i := u_0 u_1 \dots u_{n-2} u_{n-1} \quad \text{with the convention that } \bigsqcup_{i=0}^{-1} u_i := \varepsilon. \quad (1)$$

The following lemma decomposes of $LB(\sigma^k(c))$ into k parts,

Lemma 5.3.15. *Let $c \in \mathcal{C}$. Then, for all $k \geq 1$,*

$$LB(\sigma^k(c)) = \bigsqcup_{j=0}^{k-1} \sigma^{k-1-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right).$$

Proof. Let us prove it by induction on k . If $k = 1$, this is directly true. Now let $k \geq 1$ be such that $LB(\sigma^k(c)) = \bigsqcup_{j=0}^{k-1} \sigma^{k-1-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right)$. Then

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

$$\begin{aligned}
 LB(\sigma^{k+1}(c)) &= \sigma(LB(\sigma^k(c))) LB(\sigma(LC(\sigma^k(c)))) \text{ with Lemma 5.3.12} \\
 &= \bigsqcup_{j=0}^{k-1} \sigma^{k-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right) \cdot LB(\sigma(LC(\sigma^k(c)))) \\
 &= \bigsqcup_{j=0}^k \sigma^{k-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right).
 \end{aligned}$$

□

Now we are going to describe the periodic structure of this decomposition. First, let us introduce the directed graph G_L , which corresponds to the directed graph UL_G in [KS15, Section 3.3].

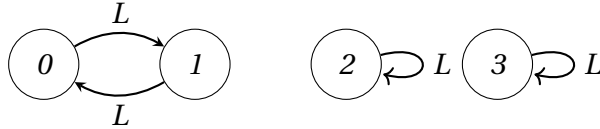
Definition 5.3.16. In order to represent how σ acts on the leftmost letter of $\sigma^k(c)$ for $c \in \mathcal{C}$, we define the directed graph $G_L := (V_L, E_L)$ by

- $V_L = \mathcal{C}$,
- $E_L := \{(c, LC(\sigma(c))) \mid c \in \mathcal{C}\}$.

If $(a, b) \in E_L$, we write $a \xrightarrow{L} b$. Note that every vertex of G_L has out degree 1.

Definition 5.3.17. We say that a collection $\mathcal{C} = (c_i)_{0 \leq i \leq p-1} \in \mathcal{C}^p$ is a p -cycle of G_L if the c_i are all distinct and, for all $i \in \llbracket 0, p-1 \rrbracket$, $c_i \xrightarrow{L} c_{i+1[p]}$ where $i+1[p]$ denotes $i+1 \bmod p$. If $c \in \mathcal{C}$ belongs to a p -cycle of G_L , we say that c is left- p -periodic.

Example 5.3.18. Following Example 5.1.26, the graph G_L is the following:



The letters 0 and 1 are left-2-periodic, and the letters 2 and 3 are left-1-periodic.

Definition 5.3.19. If $\mathcal{C} = (c_i)_{0 \leq i \leq p-1}$ is a p -cycle of G_L , for all $i \in \llbracket 0, p-1 \rrbracket$, we define the word

$$L(c_i) := \bigsqcup_{j=0}^{p-1} \sigma^{p-1-j} \left(LB(\sigma(c_{i+j[p]})) \right).$$

Remark 5.3.20. Let \mathcal{C} be a p -cycle of G_L . For all $c \in \mathcal{C}$, Lemma 5.3.15 directly provides $L(c) = LB(\sigma^p(c))$. As p is the first integer such that $LC(\sigma^p(c)) = c$, this means that $L(c)$ is the smallest bounded word generated to the left of c : there exists $v \in \mathcal{A}^*$ such that $\sigma^p(c) = L(c)cv$. In particular, $c \in \mathcal{C}_{\text{iso}}$ if and only if $L(c) \neq \varepsilon$.

Also, by construction, $L(c) \neq \varepsilon$ if and only if there exists $c' \in \mathcal{C}$ such that $LB(\sigma(c')) \neq \varepsilon$. This means that, in a cycle of G_L , either every letter is left-isolated or no letter is left-isolated.

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

Definition 5.3.21. Let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be a p -cycle of G_L . For all $i \in \llbracket 0, p-1 \rrbracket$, $L(c_i) \in \mathcal{B}^*$ so the sequence $(\sigma^{pj}(L(c_i)))_{j \geq 0}$ is eventually periodic. Set $q_{\mathcal{C}} \geq 0$ and $p_{\mathcal{C}} \geq 1$ the first integers such that, for all $i \in \llbracket 0, p-1 \rrbracket$, $\sigma^{p(q_{\mathcal{C}}+p_{\mathcal{C}})}(L(c_i)) = \sigma^{p q_{\mathcal{C}}}(L(c_i))$. We then define the part before the left period

$$LQ(c) := \bigsqcup_{j=0}^{q_{\mathcal{C}}-1} \sigma^{p(q_{\mathcal{C}}-1-j)}(L(c)),$$

and the left period

$$LP(c) := \bigsqcup_{j=0}^{p_{\mathcal{C}}-1} \sigma^{p(q_{\mathcal{C}}+p_{\mathcal{C}}-1-j)}(L(c)). \quad (2)$$

In particular, $c \in \mathcal{C}_{\text{li so}}$ if and only if $LP(c) \neq \varepsilon$.

Example 5.3.22. Following Example 5.3.18, we compute LP of the left-periodic letters.

- In the 2-cycle of $G_L \{0, 1\}$, $\sigma(LB(\sigma(0))) = \sigma(LB(\sigma(1))) = \varepsilon$ so, by Remark 5.3.20, we have $L(0) = L(1) = \varepsilon$. This means that 0 and 1 are not left-isolated, and $LP(0) = LP(1) = \varepsilon$.
- In the 1-cycle of $G_L \{2\}$, $\sigma(LB(\sigma(2))) = \varepsilon$ so, similarly, $2 \notin \mathcal{C}_{\text{li so}}$ and $LP(2) = \varepsilon$.
- In the 1-cycle of $G_L \mathcal{C} = \{3\}$, $\sigma(LB(\sigma(3))) = 54$. We have $L(3) = \sigma(LB(\sigma(3))) = 54$, and $q_{\mathcal{C}} = 1$ and $p_{\mathcal{C}} = 2$ because $\sigma^{1+2}(54) = 65 = \sigma^1(54)$. Then we have $3 \in \mathcal{C}_{\text{li so}}$ and $LP(3) = \sigma^2(54) \sigma(54) = 5665$.

We must also take into account the fact that not every growing letter is left-periodic.

Definition 5.3.23. If $c \in \mathcal{C}$, we write $r_c \geq 0$ the first integer such that $LC(\sigma^{r_c}(c))$ is a left-periodic letter. Note that $r_c = 0$ if c is itself left-periodic.

Let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be the p -cycle of G_L such that $c_0 = LC(\sigma^{r_c}(c))$. For $k \geq r_c + pq_{\mathcal{C}}$, we can uniquely write $k = r_c + i + p(q_{\mathcal{C}} + \ell p_{\mathcal{C}} + \ell')$ with $i \in \llbracket 0, p \rrbracket$, $\ell \geq 0$ and $\ell' \in \llbracket 0, p_{\mathcal{C}} \rrbracket$. We then define the word

$$LE_k(c) := \bigsqcup_{j=0}^{r_c+i+p\ell'-1} \sigma^{k-1-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right).$$

Note that the words $LE_k(c)$ have bounded length because $r_c + i + p\ell' - 1$ is bounded and $LB(\sigma(LC(\sigma^j(c)))) \in \mathcal{B}^*$ so the words $\sigma^{k-1-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right)$ have bounded length.

We defined the words LE , LP and LQ in order to obtain the desired decomposition.

Proposition 5.3.24. Let $c \in \mathcal{C}$ and let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be the p -cycle of G_L such that $c_0 = LC(\sigma^{r_c}(c))$. For all $k = r_c + i + p(q_{\mathcal{C}} + \ell p_{\mathcal{C}} + \ell') \geq r_c + pq_{\mathcal{C}}$, we have

$$LB(\sigma^k(c)) = LE_k(c) LP(c_i)^\ell LQ(c_i).$$

Proof. We first observe that, for all $j \geq 0$, $LC(\sigma^{r_c+j}(c)) = c_{j \lfloor p \rfloor}$. Then, with Lemma 5.3.15,

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

we have

$$\begin{aligned}
LB\left(\sigma^k(c)\right) &= \bigsqcup_{j=0}^{r_c+i+p\ell'-1} \sigma^{k-1-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right) \cdot \bigsqcup_{j=r_c+i+p\ell'}^{k-1} \sigma^{k-1-j} \left(LB(\sigma(LC(\sigma^j(c)))) \right) \\
&= LE_k(c) \bigsqcup_{j=0}^{p(q_\ell+\ell p_\ell)-1} \sigma^{p(q_\ell+\ell p_\ell)-1-j} \left(LB(\sigma(LC(\sigma^{j+r_c+i+p\ell'}(c)))) \right) \\
&= LE_k(c) \bigsqcup_{j=0}^{p(q_\ell+\ell p_\ell)-1} \sigma^{p(q_\ell+\ell p_\ell)-1-j} \left(LB(\sigma(c_{j+i|p})) \right) \\
&= LE_k(c) \bigsqcup_{j=0}^{q_\ell+\ell p_\ell-1} \bigsqcup_{j'=0}^{p-1} \sigma^{p(q_\ell+\ell p_\ell)-1-pj-j'} \left(LB(\sigma(c_{j'+i|p})) \right) \\
&= LE_k(c) \bigsqcup_{j=0}^{q_\ell+\ell p_\ell-1} \sigma^{p(q_\ell+\ell p_\ell-1-j)} \left(\bigsqcup_{j'=0}^{p-1} \sigma^{p-1-j'} \left(LB(\sigma(c_{j'+i|p})) \right) \right) \\
&= LE_k(c) \bigsqcup_{j=0}^{q_\ell+\ell p_\ell-1} \sigma^{p(q_\ell+\ell p_\ell-1-j)} (L(c_i)) \\
&= LE_k(c) \bigsqcup_{j=0}^{\ell p_\ell-1} \sigma^{p(q_\ell+\ell p_\ell-1-j)} (L(c_i)) \cdot \bigsqcup_{j=\ell p_\ell}^{q_\ell+\ell p_\ell-1} \sigma^{p(q_\ell+\ell p_\ell-1-j)} (L(c_i)) \\
&= LE_k(c) \bigsqcup_{j=0}^{\ell-1} \bigsqcup_{j'=0}^{p_\ell-1} \sigma^{p(q_\ell+\ell p_\ell-1-p_\ell j-j')} (L(c_i)) \cdot \bigsqcup_{j=0}^{q_\ell} \sigma^{p(q_\ell-1-j)} (L(c_i)) \\
&= LE_k(c) \bigsqcup_{j=0}^{\ell-1} \bigsqcup_{j'=0}^{p_\ell-1} \sigma^{p(q_\ell+p_\ell-1-j')} (L(c_i)) \cdot LQ(c_i) \\
&= LE_k(c) \bigsqcup_{j=0}^{\ell-1} LP(c_i) \cdot LQ(c_i) \\
&= LE_k(c) LP(c_i)^\ell LQ(c_i).
\end{aligned}$$

□

Corollary 5.3.25. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. There exists a finite set $Q_L \subseteq \mathcal{B}^* \times \mathcal{B}^*$ for which, for every $c \in \mathcal{C}$ and every $k \geq 0$, there exist a left-periodic letter $a \in \mathcal{C}$, $\ell \geq 0$ and $(u_1, u_2) \in Q_L$ such that*

$$LB\left(\sigma^k(c)\right) = u_1 LP(a)^\ell u_2.$$

Proof. Let $c \in \mathcal{C}$ and let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be the p -cycle of G_L such that $c_0 = LC(\sigma^{r_c}(c))$. If $k \geq r_c + pq_\ell$, with Proposition 5.3.24 we can write $LB(\sigma^k(c)) = u_1 LP(c_i)^\ell u_2$ where $u_1 = LE_k(c)$ and $u_2 = LQ(c_i)$ for some ℓ and some i , and in that case there is a finite number of such u_1 and u_2 . If $k < r_c + pq_\ell$, we can write $LB(\sigma^k(c)) = u_2$ and there is a finite number of such u_2 . □

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

5.3.1.4. Decomposition of $RB(\sigma^k(c))$

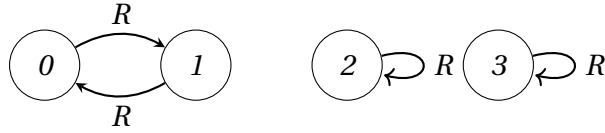
This is the exact symmetric of the decomposition of $LB(\sigma^k(c))$, so we only give the key definitions and results.

Definition 5.3.26. In order to represent how σ acts on the rightmost letter of $\sigma^k(c)$ for $c \in \mathcal{C}$, we define the directed graph $G_R := (V_R, E_R)$ by

- $V_R = \mathcal{C}$,
- $E_R := \{(c, RC(\sigma(c))) \mid c \in \mathcal{C}\}$.

We have the same notions of p -cycle of G_R and right- p -periodic letters.

Example 5.3.27. Following Example 5.1.26, the graph G_R is the following:



The letters 0 and 1 are right-2-periodic, and the letters 2 and 3 are right-1-periodic.

Definition 5.3.28. If $\mathcal{C} = (c_i)_{0 \leq i \leq p-1}$ is a p -cycle of G_R , for all $i \in \llbracket 0, p-1 \rrbracket$, we define the word

$$R(c_i) := \bigsqcup_{j=0}^{p-1} \sigma^j (RB(\sigma(c_{i+p-1-j|p}))).$$

Definition 5.3.29. Let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be a p -cycle of G_R . For all $i \in \llbracket 0, p-1 \rrbracket$, $R(c_i) \in \mathcal{B}^*$ so the sequence $(\sigma^{pj}(R(c_i)))_{j \geq 0}$ is eventually periodic. Set $q_{\mathcal{C}} \geq 0$ and $p_{\mathcal{C}} \geq 1$ the first integers such that, for all $i \in \llbracket 0, p-1 \rrbracket$, $\sigma^{p(q_{\mathcal{C}}+p_{\mathcal{C}})}(R(c_i)) = \sigma^{p q_{\mathcal{C}}}(R(c_i))$. We then define the part before the right period

$$RQ(c) := \bigsqcup_{j=0}^{q_{\mathcal{C}}-1} \sigma^{pj}(R(c)),$$

and the right period

$$RP(c) := \bigsqcup_{j=0}^{p_{\mathcal{C}}-1} \sigma^{p(q_{\mathcal{C}}+j)}(R(c)). \quad (3)$$

In particular, $c \in \mathcal{C}_{\text{riso}}$ if and only if $RP(c) \neq \varepsilon$.

Example 5.3.30. Following Example 5.3.27, we compute RP of the right-periodic letters.

- Similarly to the left side, $RP(0) = RP(1) = RP(2) = \varepsilon$.
- In the 1-cycle of G_R $\mathcal{C} = \{3\}$, $\sigma(RB(\sigma(3))) = 5$. We have $R(3) = \sigma(LB(\sigma(3))) = 5$, $q_{\mathcal{C}} = 0$ and $p_{\mathcal{C}} = 2$. Then $3 \in \mathcal{C}_{\text{riso}}$ and $RP(3) = 5 \sigma(5) = 56$.

We must also take into account the fact that not every growing letter is right-periodic.

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

Definition 5.3.31. If $c \in \mathcal{C}$, we write $r_c \geq 0$ the first integer such that $RC(\sigma^{r_c}(c))$ is a right-periodic letter. Note that $r_c = 0$ if c is itself right-periodic.

Let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be the p -cycle of G_R such that $c_0 = RC(\sigma^{r_c}(c))$. For $k \geq r_c + pq_{\mathcal{C}}$, we can uniquely write $k = r_c + i + p(q_{\mathcal{C}} + \ell p_{\mathcal{C}} + \ell')$ with $i \in \llbracket 0, p \rrbracket$, $\ell \geq 0$ and $\ell' \in \llbracket 0, p_{\mathcal{C}} \rrbracket$. We then define the word

$$RE_k(c) := \bigsqcup_{j=q_{\mathcal{C}}+\ell p_{\mathcal{C}}}^{k-1} \sigma^j \left(RB(\sigma(RC(\sigma^{k-1-j}(c)))) \right).$$

Similarly to $LE_k(c)$, the words $RE_k(c)$ have bounded length.

We finally obtain the desired decomposition.

Proposition 5.3.32. Let $c \in \mathcal{C}$ and let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be the p -cycle of G_R such that $c_0 = RC(\sigma^{r_c}(c))$. For all $k = r_c + i + p(q_{\mathcal{C}} + \ell p_{\mathcal{C}} + \ell') \geq r_c + pq_{\mathcal{C}}$, we have

$$RB(\sigma^k(c)) = RQ(c_i) RP(c_i)^\ell RE_k(c).$$

Corollary 5.3.33. Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. There exists a finite set $Q_R \subseteq \mathcal{B}^* \times \mathcal{B}^*$ for which, for every $c \in \mathcal{C}$ and every $k \geq 0$, there exist a right-periodic letter $a \in \mathcal{C}$, $\ell \geq 0$ and $(u_1, u_2) \in Q_R$ such that

$$RB(\sigma^k(c)) = u_1 RP(a)^\ell u_2.$$

5.3.1.5. Decomposition of maximal bounded factors

To recapitulate, Remark 5.3.14 states that every maximal bounded factor of σ can be decomposed with some $LB(\sigma^k(c))$ for $c \in \mathcal{C}$ and $k \geq 1$, some $\sigma^k(u)$ where u is part of an origin of σ and $k \geq 1$, and some $RB(\sigma^k(c))$ for $c \in \mathcal{C}$ and $k \geq 1$. The words $\sigma^k(u)$ are easy to understand, and with Corollaries 5.3.25 and 5.3.33 we decomposed $LB(\sigma^k(c))$ and $RB(\sigma^k(c))$ for all $c \in \mathcal{C}$. We now deduce a precise decomposition of the maximal bounded factors.

Proposition 5.3.34. Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. There exists a finite subset $Q \subseteq \mathcal{B}^* \times \mathcal{B}^* \times \mathcal{B}^*$ such that every maximal bounded factor of σ has the form

$$u_1 RP(a)^p u_2 LP(b)^q u_3$$

where $(u_1, u_2, u_3) \in Q$, $a \in \mathcal{C}$ is right-periodic, $b \in \mathcal{C}$ is left-periodic and $p, q \geq 0$.

Proof. If u is a maximal bounded factor of $\sigma^k(c)$ for $c \in \mathcal{C}$ and $k \geq 1$, Remark 5.3.14 gives three cases to consider:

(i) $u = LB(\sigma^k(c))$. By Corollary 5.3.25, there exists $(u_2, u_3) \in Q_L$, $b \in \mathcal{C}$ left-periodic and $q \geq 0$ such that $u = u_2 LP(b)^q u_3$, and there is a finite number of such u_2 and u_3 .

(ii) $u = RB(\sigma^\ell(a)) \sigma^\ell(v) LB(\sigma^\ell(b))$ where $0 \leq \ell < k$ and (a, v, b) is an origin of σ . By Corollary 5.3.33, there exists $(u_1, v_1) \in Q_R$, $a \in \mathcal{C}$ right-periodic and $p \geq 0$ such

5. Minimal components of substitution subshifts – 5.3. Wild minimal components

that $RB(\sigma^\ell(a)) = u_1 RP(a)^p v_1$. By Corollary 5.3.25, there exists $(v_2, u_3) \in Q_L$, $b \in \mathcal{C}$ left-periodic and $q \geq 0$ such that $LB(\sigma^\ell(b)) = v_2 LP(b)^q u_3$. By setting $u_2 = v_1 \sigma^\ell(v) v_2$, we can write $u = u_1 RP(a)^p u_2 LP(b)^q u_3$, and there is a finite number of such u_1 , u_2 and u_3 because $v \in \mathcal{B}^*$.

(iii) $u = RB(\sigma^k(c))$. Similarly to the first case, we can write $u = u_1 RP(a)^p u_2$ with $a \in \mathcal{C}$ right-periodic, and there is a finite number of such u_1 and u_2 . \square

Remark 5.3.35. *This proposition decomposes the maximal bounded factor of σ , which will naturally provide a decomposition of every factor of $\mathcal{L}(\sigma) \cap \mathcal{B}^*$. It makes two improvements from Proposition 5.3.1:*

(i) *It explicitly defines the words that occur periodically in the decomposition.*

(ii) *It holds not only when σ is wild, but in the general case. As a consequence, one can recover Proposition 3.3.3: if σ has property $\mathfrak{A}2$, then one $LP(c)$ (resp. $RP(c)$) is non-empty so, by Proposition 5.3.24 (resp. Proposition 5.3.32), the words of $\mathcal{L}(\sigma) \cap \mathcal{B}^*$ have unbounded length, i.e., σ has property $\mathfrak{A}1$; conversely, if σ does not have property $\mathfrak{A}2$, then every $LP(c)$ and $RP(c)$ is empty so Proposition 5.3.34 ensures that the maximal bounded factors have bounded length and that σ does not have property $\mathfrak{A}1$.*

5.3.2. Proof of Theorem 5.1.24 (ii)

We first show that the repetitions arising in Propositions 5.3.24 and 5.3.32 provide the wild minimal components exhibited in Remark 5.1.19.

Proposition 5.3.36. *Let $c \in \mathcal{C}_{liso}$ (resp. $c \in \mathcal{C}_{riso}$). Then the subshift $X^{(\omega LP(c)^\omega)}$ (resp. $X^{(\omega RP(c)^\omega)}$) is a wild minimal component of X_σ .*

Proof. Let $c \in \mathcal{C}_{liso}$ be a left- p -periodic letter. We have $r_c = 0$ so Proposition 5.3.24 states that $LP(c)^\ell \sqsubset LB(\sigma^{p(q_\ell + \ell p_\ell)}(c))$ for all $\ell \geq 0$. Therefore $X^{(\omega LP(c)^\omega)} \subseteq X_\sigma$. Symmetrically, if $c \in \mathcal{C}_{riso}$ we obtain $X^{(\omega RP(c)^\omega)} \subseteq X_\sigma$. \square

The converse relies on Corollaries 5.3.25 and 5.3.33, this is a more precise version of Corollary 5.1.22.

Proposition 5.3.37. *Let X be a wild minimal component of X_σ . Then X satisfies at least one of the following properties:*

- *There exists $c \in \mathcal{C}_{liso}$ such that $X = X^{(\omega LP(c)^\omega)}$.*
- *There exists $c \in \mathcal{C}_{riso}$ such that $X = X^{(\omega RP(c)^\omega)}$.*

Proof. Let $x \in X$. Suppose that, for every $c \in \mathcal{C}_{liso}$ (resp. $c \in \mathcal{C}_{riso}$), there exists $k_c \geq 1$ such that $LP(c)^{k_c} \not\sqsubset x$ (resp. $RP(c)^{k_c} \not\sqsubset x$). Set K to be the maximum of all k_c for $c \in \mathcal{C}_{liso}$ and $c \in \mathcal{C}_{riso}$, and set $\ell_p := \max_{c \text{ left-periodic}} |LP(c)^{K+2}| + \max_{c \text{ right-periodic}} |RP(c)^{K+2}|$. Also set $\ell_Q := \max_{(u_1, u_2, u_3) \in Q} |u_1| + |u_2| + |u_3|$ where Q is the finite set from Proposition 5.3.34.

Now, for any $u \sqsubset x$, we have $u \in \mathcal{L}(X_\sigma) \cap \mathcal{B}^*$ so there exists a maximal bounded factor v of σ such that $u \sqsubset v$, and, by Proposition 5.3.34, there exist $(v_1, v_2, v_3) \in Q$, $a \in \mathcal{C}$ right-periodic, $b \in \mathcal{C}$ left-periodic and $p, q \geq 0$ such that $v = v_1 RP(a)^p v_2 LP(b)^q v_3$.

We can then write $u = u_1 u_2 u_3 u_4 u_5$ where $u_1 \sqsubset v_1$, $u_2 \sqsubset RP(a)^p$, $u_3 \sqsubset v_2$, $u_4 \sqsubset LP(b)^q$ and $u_5 \sqsubset v_3$. We have $|u_1| + |u_3| + |u_5| \leq |v_1| + |v_2| + |v_3| \leq \ell_Q$. If $a \in \mathcal{C}_{riso}$, by definition of K we have $RP(a)^K \not\sqsubset u_2$ so $|u_2| < |RP(a)^{K+2}|$, otherwise $|v_2| = 0$. Similarly, if $b \in \mathcal{C}_{liso}$, we have $|v_4| < |LP(b)^{K+2}|$, otherwise $|v_4| = 0$. In any case, we have $|u_2| + |u_4| \leq \ell_P$, so $|u| \leq \ell_Q + \ell_P$. We just proved that every factor of x has bounded length, which is a contradiction.

Therefore, either there exists $c \in \mathcal{C}_{liso}$ such that for all $k \geq 1$, $LP(c)^k \sqsubset x$, or there exists $c \in \mathcal{C}_{riso}$ such that for all $k \geq 1$, $RP(c)^k \sqsubset x$. This means that $X^{(\omega LP(c)^\omega)} \subseteq X$ or $X^{(\omega RP(c)^\omega)} \subseteq X$, and as X is minimal we get the equality. \square

Propositions 5.3.36 and 5.3.37 complete the proof of Theorem 5.1.24 (ii).

Example 5.3.38. Following Examples 5.3.22 and 5.3.30, the wild minimal components of X_σ are $X^{(\omega(5665)^\omega)}$ and $X^{(\omega(56)^\omega)}$.

5.4. Dynamics of minimal components

We explained in Section 5.2.1.3 that a substitution σ induces a permutation $\tilde{\sigma}$ on its subshifts. In fact, one can prove that $\tilde{\sigma}$ preserves the minimal components, but we will go further by describing how $\tilde{\sigma}$ acts respectively on the tame and wild minimal components.

5.4.1. Dynamics of tame minimal components

Proposition 5.4.1. Let $\mathcal{D} \subseteq \mathcal{C}_{niso}$ be a minimal alphabet of period k and let \mathcal{E} be such that $\mathcal{D} \rightarrow \mathcal{E}$. Then $\mathcal{E} \subseteq \mathcal{C}_{niso}$ is a minimal alphabet of period k and $\tilde{\sigma} \left(X_{\sigma^k|_{\mathcal{D} \cup \mathcal{B}}} \right) = X_{\sigma^k|_{\mathcal{E} \cup \mathcal{B}}}$.

Proof. By Lemma 5.2.6, \mathcal{E} is a minimal alphabet of period k . In particular, we have $\mathcal{E} \xrightarrow[k-1]{} \mathcal{D}$. If $e \in \mathcal{E}$ is a left-periodic letter, let \mathcal{C} be the cycle of G_L such that $e \in \mathcal{C}$ and let $d \in \mathcal{C}$ be such that $e \xrightarrow[k-1]{L} d$. We have $d = LC(\sigma^{k-1}(e)) \sqsubset \sigma^{k-1}(e)$ so $d \in \mathcal{D}$. In particular, $d \notin \mathcal{C}_{liso}$ so, by Remark 5.3.20, $e \notin \mathcal{C}_{liso}$. We just proved that $\mathcal{E} \cap \mathcal{C}_{liso} = \emptyset$, and symmetrically we get $\mathcal{E} \cap \mathcal{C}_{riso} = \emptyset$, therefore $\mathcal{E} \subseteq \mathcal{C}_{niso}$.

Let $x \in \tilde{\sigma} \left(X_{\sigma^k|_{\mathcal{D} \cup \mathcal{B}}} \right)$. For all $u \in \mathcal{L}(x)$, it follows from the definitions that there exist $d \in \mathcal{D}$ and $\ell \geq 1$ such that $u \sqsubset \sigma^{k\ell+1}(d)$. We also have $\mathcal{E} \xrightarrow[k-1]{} \mathcal{D}$ so there exists $e \in \mathcal{E}$ such that $d \sqsubset \sigma^{k-1}(e)$. We get $u \sqsubset \sigma^{k(\ell+1)}(e)$ so $u \in \mathcal{L} \left(X_{\sigma^k|_{\mathcal{E} \cup \mathcal{B}}} \right)$. We just proved that $\mathcal{L}(x) \subseteq \mathcal{L} \left(X_{\sigma^k|_{\mathcal{E} \cup \mathcal{B}}} \right)$ for all $x \in \tilde{\sigma} \left(X_{\sigma^k|_{\mathcal{D} \cup \mathcal{B}}} \right)$, which means that $\tilde{\sigma} \left(X_{\sigma^k|_{\mathcal{D} \cup \mathcal{B}}} \right) \subseteq X_{\sigma^k|_{\mathcal{E} \cup \mathcal{B}}}$. Finally, by Proposition 5.2.18, $X_{\sigma^k|_{\mathcal{E} \cup \mathcal{B}}}$ is minimal so we have the equality. \square

This allows us to define the following graph.

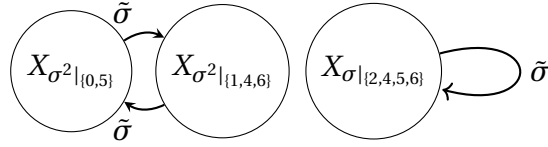
5. Minimal components of substitution subshifts – 5.4. Dynamics of minimal components

Definition 5.4.2. We define the directed graph $G_t = (V_t, E_t)$ by

- V_t the set of tame minimal components of X_σ ,
- $E_t := \{(X, Y) \mid X, Y \in V_t, \tilde{\sigma}(X) = Y\}$.

Then Proposition 5.4.1 means that G_t is in correspondence with the subgraph of G restricted to the minimal alphabets included in $\mathcal{C}_{ni\,so}$. In particular, $\tilde{\sigma}$ induces a permutation on the tame minimal components.

Example 5.4.3. Following Examples 5.2.2 and 5.2.20, the graph G_t is the following:



5.4.2. Dynamics of wild minimal components

Proposition 5.4.4. Let $a, b \in \mathcal{C}_{li\,so}$ be such that $a \xrightarrow{L} b$. Then $\tilde{\sigma}(X^{(\omega LP(a)^\omega)}) = X^{(\omega LP(b)^\omega)}$.

Proof. Let $\mathcal{C} = \{c_i\}_{0 \leq i \leq p-1}$ be the p -cycle of G_L such that $a = c_0$ and $b = c_{1|p}$.

On the first hand, for all $\ell \geq 0$, Proposition 5.3.24 provides

$$LB\left(\sigma^{p(q_{\mathcal{C}} + \ell p_{\mathcal{C}}) + 1}(c_0)\right) = LE_{p(q_{\mathcal{C}} + \ell p_{\mathcal{C}}) + 1}(c_0) LP(c_{1|p})^\ell LQ(c_{1|p}).$$

On the other hand, for all $\ell \geq 0$, we have

$$\begin{aligned} LB\left(\sigma^{p(q_{\mathcal{C}} + \ell p_{\mathcal{C}}) + 1}(c_0)\right) &= LB\left(\sigma\left(\sigma^{p(q_{\mathcal{C}} + \ell p_{\mathcal{C}})}(c_0)\right)\right) \\ &= \sigma\left(LB\left(\sigma^{p(q_{\mathcal{C}} + \ell p_{\mathcal{C}})}(c_0)\right)\right) LB\left(\sigma(LC(\sigma^{p(q_{\mathcal{C}} + \ell p_{\mathcal{C}})}(c_0)))\right) \text{ by Lemma 5.3.12} \\ &= \sigma(LP(c_0))^\ell \sigma(LQ(c_0)) LB(\sigma(c_0)) \text{ by Proposition 5.3.24.} \end{aligned}$$

Both expressions are equal, and only $LP(b)^\ell$ on the one side and $\sigma(LP(a))^\ell$ on the other side have unbounded length, which means that there exists $n \in \mathbb{Z}$ such that $T^n(\sigma^{(\omega LP(a)^\omega)}) = \omega LP(b)^\omega$. \square

With the same arguments, we have the symmetric result.

Proposition 5.4.5. Let $a, b \in \mathcal{C}_{ri\,so}$ such that $a \xrightarrow{R} b$. Then $\tilde{\sigma}(X^{(\omega RP(a)^\omega)}) = X^{(\omega RP(b)^\omega)}$.

Remark 5.4.6. In fact, a more precise fact holds: if $a \xrightarrow{L} b$ (resp. $a \xrightarrow{R} b$), then $LP(b)$ (resp. $RP(b)$) is a cyclic shift of $LP(a)$ (resp. $RP(a)$).

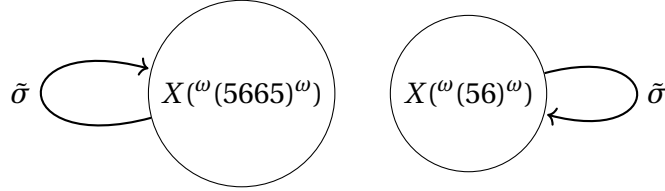
This allows us to define the following graph.

Definition 5.4.7. We define the directed graph $G_w := (V_w, E_w)$ by

- V_w the set of wild minimal components of X_σ ,
- $E_w := \{(X, Y) \mid X, Y \in V_w, \tilde{\sigma}(X) = Y\}$.

Then Propositions 5.4.4 and 5.4.5 mean that G_w is determined by G_L and G_R . In particular, $\tilde{\sigma}$ induces a permutation on the wild minimal components.

Example 5.4.8. Following Examples 5.3.18, 5.3.27 and 5.3.38, the graph G_w is the following:



5.5. Counting minimal components

5.5.1. Computing bounded letters

We take inspiration in [Dev18, Lemmas 3.1 and 3.2] to define the set of *periodic* letters $\mathcal{P} \subseteq \mathcal{B}$ which is essential to characterize \mathcal{B} .

Definition 5.5.1. If $a \in \mathcal{B}$ occurs in $\sigma^n(a)$ for some $n \geq 1$ and we say that a is periodic, otherwise we say that a is pre-periodic. We write \mathcal{P} the set of periodic letters and \mathcal{P}' the set of pre-periodic letters, such that $\mathcal{B} = \mathcal{P} \cup \mathcal{P}'$.

Lemma 5.5.2. Let $a \in \mathcal{A}$. Then the following holds.

- (i) $a \in \mathcal{P}$ if and only if there exists $n \geq 1$ such that $\sigma^n(a) = a$.
- (ii) If $a \in \mathcal{P}$, then, for all $m \geq 1$, $\sigma^m(a) \in \mathcal{P}$.
- (iii) $a \in \mathcal{B}$ if and only if $\sigma^{|\mathcal{A}|}(a) \in \mathcal{P}^+$.

Proof. (i) If $a \in \mathcal{P}$, there exist $n \geq 1$ and $u, v \in \mathcal{A}^*$ such that $\sigma^n(a) = uav$. If $(u, v) \neq (\varepsilon, \varepsilon)$, then, iterating the substitution σ^n on a would yield $a \in \mathcal{C}$, therefore $\sigma^n(a) = a$.

If there exists $n \geq 1$ such that $\sigma^n(a) = a$, then, for all $m \geq 1$, $\sigma^{mn}(a) = a$ so we get $1 = |\sigma^{mn}(a)| \geq |\sigma^m(a)| \geq 1$ and $|\sigma^m(a)| = 1$. This means that $a \in \mathcal{B}$, and it occurs in $\sigma^n(a)$ so $a \in \mathcal{P}$.

(ii) If $a \in \mathcal{P}$, we already proved that, for all $m \geq 1$, $|\sigma^m(a)| = 1$. (i) provides $n \geq 1$ such that $\sigma^n(a) = a$, so $\sigma^n(\sigma^m(a)) = \sigma^m(\sigma^n(a)) = \sigma^m(a)$ and, by (i), $\sigma^m(a) \in \mathcal{P}$.

(iii) If $a \in \mathcal{B}$, suppose that $\sigma^n(a) \notin \mathcal{P}^+$ for some $n \geq 0$. In particular, there exists a letter $a_n \in \mathcal{P}'$ such that $a_n \sqsubset \sigma^n(a)$. If $n \geq 1$, this provides $a_{n-1} \in \mathcal{A}$ such that $a_{n-1} \sqsubset \sigma^{n-1}(a)$ and $a_n \sqsubset \sigma(a_{n-1})$, and, by (ii), $a_{n-1} \in \mathcal{P}'$. By iterating this construction, we obtain a finite sequence of pre-periodic letters $(a_i)_{0 \leq i \leq n}$ such that $a_{i+1} \sqsubset \sigma(a_i)$. Now, if $\sigma^{|\mathcal{A}|}(a) \notin \mathcal{P}^+$, we obtain the family of pre-periodic letters $(a_i)_{0 \leq i \leq |\mathcal{A}|}$ and there exists $i < j$ such that $a_i = a_j$. However, we have $a_i \sqsubset \sigma^{j-i}(a_i)$ so $a_i \notin \mathcal{P}'$, contradiction.

If $\sigma^{|\mathcal{A}|}(a) \in \mathcal{P}^+$, in particular $\sigma^{|\mathcal{A}|}(a) \in \mathcal{B}^+$ so $a \in \mathcal{B}$. □

We now have a method to compute \mathcal{B} , as stated in Proposition 5.1.28.

Proof of Proposition 5.1.28. We first compute the alphabet \mathcal{P} : for $a \in \mathcal{A}$, we compute the $\sigma^n(a)$ and either we reach n such that $|\sigma^n(a)| \geq 2$, and by Lemma 5.5.2 (ii) $a \notin \mathcal{P}$, or we reach n for which there exists $m < n$ such that $\sigma^n(a) = \sigma^m(a) \in \mathcal{A}$, and if $m = 0$ then, by Lemma 5.5.2 (ii), $a \in \mathcal{P}$, or if $m > 0$ then $a \notin \mathcal{P}$.

We can now compute \mathcal{B} : for $a \in \mathcal{A}$, we compute $\sigma^{|\mathcal{A}|}(a)$ and we check if it belongs to \mathcal{P}^+ to use Lemma 5.5.2 (iii). The remaining letters are in \mathcal{C} . \square

5.5.2. Computing minimal components

Essentially, computing the minimal components is the same as counting them. We recall that $MC(\sigma)$ denotes the number of minimal components of X_σ , and we also define $TMC(\sigma)$ the number of tame minimal components and $WMC(\sigma)$ the number of wild minimal components.

Proposition 5.5.3. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. Then $TMC(\sigma)$ is computable.*

Proof. Once Proposition 5.1.28 provides \mathcal{C} , we can compute G , the minimal alphabets and their period. By checking if the minimal alphabets contain a left or right-isolated letter, we obtain $TMC(\sigma)$. \square

Proposition 5.5.4. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. Then $WMC(\sigma)$ is computable.*

Proof. Once Proposition 5.1.28 provides \mathcal{B} , we can compute G_L , G_R , $LP(c)$ and $RP(c)$ for every left or right-periodic letter. In order to differentiate the $X^{(\omega}LP(c)^\omega)$ and $X^{(\omega}RP(c)^\omega)$, we compute the primitive root of the words $LP(c)$ and $RP(c)$ and we check if one is a cyclic shift of another to remove duplicates. \square

Propositions 5.5.3 and 5.5.4 complete the proof of Corollary 5.1.29. For details about the implementation of these algorithms, the reader can take a look at the attached Python code.

5.5.3. Upper bounds for minimal components

Let us first bound $TMC(\sigma)$ and $WMC(\sigma)$ separately.

Lemma 5.5.5. *Let $\sigma : \mathcal{A} \rightarrow \mathcal{A}^+$ be a substitution. Then*

- (i) $TMC(\sigma) \leq |\mathcal{C}_{niso}|$,
- (ii) $WMC(\sigma) \leq |\mathcal{C}_{liso}| + |\mathcal{C}_{riso}|$.

Proof. (i) Set n to be the number of minimal alphabets included in \mathcal{C}_{niso} . Then Theorem 5.1.24 states that $TMC(\sigma) = n$ and with Proposition 5.2.10 yields $n \leq |\mathcal{C}_{niso}|$.

(ii) $WMC(\sigma) \leq |\mathcal{C}_{liso}| + |\mathcal{C}_{riso}|$ comes directly from Theorem 5.1.24 (ii). \square

We can then bound $MC(\sigma)$ using the size of the alphabet, as stated in Corollary 5.1.30.

5. Minimal components of substitution subshifts – 5.5. Counting minimal components

Proof of Corollary 5.1.30. By Lemma 5.5.5, we have

$$MC(\sigma) = TMC(\sigma) + WMC(\sigma) \leq |\mathcal{C}_{niso}| + |\mathcal{C}_{liso}| + |\mathcal{C}_{riso}|.$$

(i) If $\mathcal{B} = \emptyset$, then $\mathcal{C}_{niso} = \mathcal{C} = \mathcal{A}$ and $\mathcal{C}_{liso} = \mathcal{C}_{riso} = \emptyset$ so $MC(\sigma) \leq |\mathcal{C}_{niso}| = |\mathcal{C}| = |\mathcal{A}|$.

(ii) If $\mathcal{B} = \{b\}$, then $\{\omega b^\omega\}$ is the only possible wild minimal component and we have $WMC(\sigma) \leq 1$. If $\mathcal{C}_{liso} = \mathcal{C}_{riso} = \emptyset$, then $MC(\sigma) \leq |\mathcal{C}_{niso}| \leq |\mathcal{C}| = |\mathcal{A}| - 1$. If $\mathcal{C}_{liso} \neq \emptyset$ or $\mathcal{C}_{riso} \neq \emptyset$, then we have $MC(\sigma) \leq |\mathcal{C}_{niso}| + 1 \leq |\mathcal{C}| = |\mathcal{A}| - 1$.

(iii) If $|\mathcal{B}| \geq 2$, then we have $|\mathcal{C}_{liso}| \leq |\mathcal{C}| - |\mathcal{C}_{niso}|$ and $|\mathcal{C}_{riso}| \leq |\mathcal{C}| - |\mathcal{C}_{niso}|$, which yields $MC(\sigma) \leq 2|\mathcal{C}| - |\mathcal{C}_{niso}| \leq 2|\mathcal{C}| \leq 2|\mathcal{A}| - 4$. \square

The following examples show that these bounds are optimal.

(i) The bound $MC(\sigma) = |\mathcal{C}| = |\mathcal{A}|$ can only be reached when every singleton in G is a minimal alphabet.

Example 5.5.6. Let $k \geq 1$, $\mathcal{A} = \{a_1, \dots, a_k\}$ and $\sigma : a_i \mapsto a_i a_i$ for all $i \in \llbracket 1, k \rrbracket$. We have $\mathcal{B} = \emptyset$, the minimal alphabets are the 1-periodic alphabets $\{a_i\}$ and they generate the tame minimal components $\{\omega a_i^\omega\}$, therefore $MC(\sigma) = |\mathcal{A}|$.

(ii) The bound $MC(\sigma) = |\mathcal{C}| = |\mathcal{A}| - 1$ can be reached in two ways:

- $TMC(\sigma) = |\mathcal{A}| - 1$ and $WMC(\sigma) = 0$

Example 5.5.7. Let $k \geq 2$, $\mathcal{A} = \{b, a_1, \dots, a_{k-1}\}$, $\sigma : b \mapsto b, a_1 \mapsto a_1 b a_1$. The minimal alphabets included in \mathcal{C}_{niso} are the $\{a_i\}$ for $i \in \llbracket 1, k-1 \rrbracket$ and they generate the tame minimal components $X_{\sigma|_{\{a_i, b\}}}$. In particular, for all $n \geq 0$, $\sigma^n(a_i) = (a_i b)^{2^n - 1} a_i$ so $X_{\sigma|_{\{a_i, b\}}} = \{\omega (a_i b)^\omega\}$. Therefore, $TMC(\sigma) = k-1 = |\mathcal{A}| - 1$. Also, σ is tame so $WMC(\sigma) = 0$.

- $TMC(\sigma) = |\mathcal{A}| - 2$ and $WMC(\sigma) = 1$

Example 5.5.8. Let $k \geq 2$, $\mathcal{A} = \{b, a_1, \dots, a_{k-1}\}$ and $\sigma : a_i \mapsto a_i a_i$ for all $i \in \llbracket 1, k-2 \rrbracket$, $a_{k-1} \mapsto a_{k-1} b, b \mapsto b$. We have $\mathcal{B} = \{b\}$ and $a_{k-1} \in \mathcal{C}_{riso}$ so the minimal alphabets included in \mathcal{C}_{niso} are the 1-periodic alphabets $\{a_i\}$ for $i \in \llbracket 1, k-2 \rrbracket$ and they generate the tame minimal components $X_{\sigma|_{\{a_i\}}} = \{\omega a_i^\omega\}$ for $i \in \llbracket 1, k-2 \rrbracket$. Moreover, $\{\omega b^\omega\}$ is the unique wild component, therefore $MC(\sigma) = |\mathcal{A}| - 1$.

(iii) The bound $MC(\sigma) = 2|\mathcal{C}| = 2|\mathcal{A}| - 4$ can only be reached when $\mathcal{C}_{liso} = \mathcal{C}_{riso} = \mathcal{C}$, $|\mathcal{B}| = 2$ and all $X(\omega LP(c)^\omega)$ and $X(\omega RP(c)^\omega)$ are distinct.

Example 5.5.9. Let $k \geq 3$, $\mathcal{A} = \{a, b, c_1, \dots, c_{k-2}\}$ and $\sigma : a \mapsto a, b \mapsto b, c_i \mapsto a^{2i-2} b c_i a^{2i-1} b$. We have $\mathcal{B} = \{a, b\}$, and every c_i is growing, left-isolated with period 1 and right-isolated with period 1. It immediately implies that X_σ has no tame minimal components. Also, for all $i \in \llbracket k-2 \rrbracket$, $LP(c_i) = L(c_i) = a^{2i-2} b$ and $RP(c_i) = R(c_i) = a^{2i-1} b$ so the wild minimal components of X_σ are the $X(\omega (a^i b)^\omega)$ for all $0 \leq i \leq 2k-5$. Therefore $MC(\sigma) = 2k-4 = 2|\mathcal{C}| = 2|\mathcal{A}| - 4$.

Remark 5.5.10. We can modify Example 5.5.9 step by step by making two wild minimal components equal, which decreases the number of minimal components by 1 (for example, if we change the image of c_1 to be $b c_1 b$, then $MC(\sigma) = 2k-5$). That way, $MC(\sigma)$ reaches every number between 1 and $2|\mathcal{A}| - 4$.

5.6. Minimal components on two letters

Substitutions on the binary alphabet $\{0, 1\}$ already generate interesting subshifts and provide a reasonable number of cases to analyse. Note that the number of cases grows faster than exponentially with the size of the alphabet, so let us compute and display the minimal components of X_σ for all cases on the alphabet $\{0, 1\}$. We recall that, in order to generate a subshift, substitutions must have a growing letter so, without loss of generality, we assume that the letter 0 is always growing. Then the first distinction is whether 1 is growing or bounded.

5.6.1. Growing substitutions

Suppose that $\mathcal{C} = \{0, 1\}$. Then Theorem 5.1.24 tells us that X_σ only has tame minimal components, and Corollary 5.1.30 (i) tells us that it has at most two. They are characterized by the minimal alphabets, and we recall from Remark 5.2.4 that G is uniquely determined by its generators \mathcal{D} and \mathcal{D}' such that $\{0\} \rightarrow \mathcal{D}$ and $\{1\} \rightarrow \mathcal{D}'$. The following table displays the minimal components in each case, white cells mean that X_σ is minimal, light gray cells mean that X_σ is sometimes minimal and sometimes not, and gray cells mean that X_σ is not minimal.

| $\mathcal{D} \backslash \mathcal{D}'$ | $\{1\}$ | $\{0\}$ | $\{0, 1\}$ |
|---------------------------------------|--|--|-----------------------|
| $\{0\}$ | $\{\omega 0^\omega, \omega 1^\omega\}$ | $\{\omega 0^\omega\}$ | $\{\omega 0^\omega\}$ |
| $\{1\}$ | $\{\omega 1^\omega\}$ | $\{\omega 0^\omega, \omega 1^\omega\}$ | X_σ |
| $\{0, 1\}$ | $\{\omega 1^\omega\}$ | X_σ | X_σ |

Let us detail the computations. Note that switching 0 and 1 brings the analysis down to six cases.

5.6.1.1. $\mathcal{D} = \{0\}, \mathcal{D}' = \{1\}$

This is one of the two cases where X_σ has two minimal components. The orbits in G are



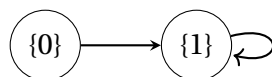
The minimal alphabets are the 1-periodic alphabets $\{0\}$ and $\{1\}$ so the main sub-substitutions of σ are $\sigma|_{\{0\}}$ and $\sigma|_{\{1\}}$ and the minimal components of X_σ are $X_{\sigma|_{\{0\}}} = \{\omega 0^\omega\}$ and $X_{\sigma|_{\{1\}}} = \{\omega 1^\omega\}$.

Example 5.6.1. Consider the substitution $\sigma : 0 \mapsto 00, 1 \mapsto 11$.

5. Minimal components of substitution subshifts – 5.6. Minimal components on two letters

5.6.1.2. $\mathcal{D} = \{1\}, \mathcal{D}' = \{1\}$

The orbits in G are



The unique minimal alphabet is the 1-periodic alphabet $\{1\}$ so the unique main sub-substitution of σ is $\sigma|_{\{1\}}$ and the unique minimal component of X_σ is $X_{\sigma|_{\{1\}}} = \{\omega 1^\omega\}$. In fact this case is quite trivial since $\sigma(\{0, 1\}) \in \{1\}^+$, so $X_\sigma = \{\omega 1^\omega\}$ and it is minimal.

Example 5.6.2. Consider the substitution $\sigma : 0 \mapsto 1, 1 \mapsto 11$.

5.6.1.3. $\mathcal{D} = \{1\}, \mathcal{D}' = \{0\}$

This is the second case when X_σ has two minimal components. The orbits in G are

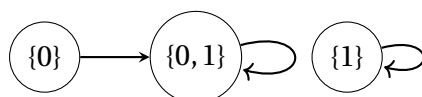


The minimal alphabets are the 2-periodic alphabets $\{0\}$ and $\{1\}$ so the main sub-substitutions are $\sigma^2|_{\{0\}}$ and $\sigma^2|_{\{1\}}$ and the minimal components of X_σ are $X_{\sigma^2|_{\{0\}}} = \{\omega 0^\omega\}$ and $X_{\sigma^2|_{\{1\}}} = \{\omega 1^\omega\}$.

Example 5.6.3. Consider the substitution $\sigma : 0 \mapsto 11, 1 \mapsto 00$.

5.6.1.4. $\mathcal{D} = \{0, 1\}, \mathcal{D}' = \{1\}$

The orbits in G are



The unique minimal alphabet is the 1-periodic alphabet $\{1\}$ so the unique main sub-substitution of σ is $\sigma|_{\{1\}}$ and the unique minimal component of X_σ is $X_{\sigma|_{\{1\}}} = \{\omega 1^\omega\}$. In this case, there are examples where X_σ is not minimal and others where it is minimal.

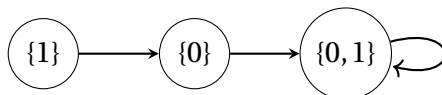
Example 5.6.4. Consider the substitution $\sigma : 0 \mapsto 101, 1 \mapsto 11$. Then $X_\sigma = \{\omega 101^\omega\}$ is not minimal since $\{\omega 1^\omega\} \not\subseteq X$.

Example 5.6.5. Consider the substitution $\sigma : 0 \mapsto 01, 1 \mapsto 11$. We have $X_\sigma = \{\omega 1^\omega\}$ so it is minimal.

5. Minimal components of substitution subshifts – 5.6. Minimal components on two letters

5.6.1.5. $\mathcal{D} = \{0, 1\}$, $\mathcal{D}' = \{0\}$

The orbits in G are

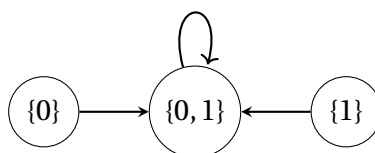


The unique minimal alphabet is the 1-periodic alphabet $\{0, 1\}$ so the unique main sub-substitution is σ itself and X_σ is minimal.

Example 5.6.6. Consider the Fibonacci substitution $\sigma : 0 \mapsto 01, 1 \mapsto 0$.

5.6.1.6. $\mathcal{D} = \{0, 1\}$, $\mathcal{D}' = \{0, 1\}$

The orbits in G are



Similarly to the previous case, X_σ is minimal.

Example 5.6.7. Consider the Thue-Morse substitution $\sigma : 0 \mapsto 01, 1 \mapsto 10$.

5.6.2. Non-growing substitutions

Suppose that $\mathcal{C} = \{0\}$, $\mathcal{B} = \{1\}$. Then σ is ℓ -primitive and the unique minimal alphabet is $\{0\}$. Also, Corollary 5.1.30 (ii) tells us that X_σ has a unique minimal component, which will only depend on whether σ is tame or wild.

5.6.2.1. Tame substitutions

The alphabet $\{0\} \notin \mathcal{C}_{niso}$ is minimal of period 1 so, by Proposition 5.2.11, σ is ℓ -primitive. Then, by Theorem 5.1.24 (i), X_σ is minimal.

Example 5.6.8. Consider the Chacon substitution $\sigma : 0 \mapsto 0010, 1 \mapsto 1$.

5.6.2.2. Wild substitutions

We necessarily have $LP(0) \in \{1\}^+$ or $RP(0) \in \{1\}^+$, so, by Theorem 5.1.24 (ii), the unique minimal component of X_σ is $\{\omega 1^\omega\}$. There are examples where X_σ is not minimal and others where it is minimal.

Example 5.6.9. In Example 5.1.20, $X_\sigma = X(\omega 101^\omega)$ and its unique minimal component is $\{\omega 1^\omega\}$, so X_σ is not minimal.

Example 5.6.10. Consider Example 5.5.8 where $k = 2$, that is $\sigma : 0 \mapsto 01, 1 \mapsto 1$. We have $X_\sigma = \{\omega 1^\omega\}$ so it is minimal.

5.7. Open questions

As mentioned in the introduction of this chapter, the number of total subshifts would be an interesting data to evaluate how far a subshift is from being minimal. This was already discussed by Maloney and Rust in Section 5 of [MR18], but we think that our approach based on giving a combinatorial characterization of the subshifts could also be useful here. As the union of subshifts is also a subshift, it is natural to consider components that are not a disjoint union of subshifts, which we might call *irreducible components*. With this definition, computing the lattice of the irreducible components ordered by inclusion would be a way to describe the larger structure of the subshift.

The question of characterizing and counting the minimal or irreducible components can also be asked for broader types of subshifts, like morphic or linearly recurrent subshifts. Notably, in the morphic case, the number of minimal components might be bigger than in the purely morphic case.

Part II.

Smooth sequences

6. More Kolakoski sequences

It's never good to solve a problem completely, because nobody cites you.

Srečko Brlek

Content

| | |
|--|----|
| 6.1. Smooth sequences | 82 |
| 6.1.1. Conjugacy to the full-shift | 83 |
| 6.1.2. Fixed points | 84 |
| 6.1.3. Dichotomy | 86 |
| 6.2. Factor complexity and bispecial words | 87 |

This chapter introduces the framework of smooth sequences and prepares to our contributions in Chapters 7 and 8. The topic of smooth sequences originates from the famous Oldenburger-Kolakoski sequence

$$\kappa = \underbrace{22}_2 \underbrace{11}_2 \underbrace{2}_1 \underbrace{1}_1 \underbrace{22}_2 \underbrace{1}_1 \underbrace{22}_2 \underbrace{11}_2 \underbrace{2}_1 \underbrace{11}_2 \underbrace{22}_2 \underbrace{1}_1 \underbrace{2}_1 \underbrace{11}_2 \underbrace{2}_1 \underbrace{1}_1 \underbrace{22}_2 \dots = \kappa$$

defined as a fixed point of the *run-length encoding*. Originally introduced by Oldenburger in [Old39] and rediscovered by Kolakoski in [Kol65], it is now registered as sequence A000002 in the OEIS [Inc26]. Despite its simple definition, κ raises some surprisingly difficult questions: it was quickly shown in [Old39] and in [KÜ66] that κ is aperiodic, but almost nothing else is known about it.

Conjecture 6.0.1. *The three major conjectures for κ are*

1. κ is (uniformly) recurrent,
2. $p_\kappa(n) = \Theta(n^{\log(3)/\log(3/2)})$,
3. κ has uniform frequencies, in particular it has equal letter frequencies.

By *uniform frequencies* we mean that the subshift generated by κ is uniquely ergodic. These conjectures first appeared in the seminal work of Dekking in [Dek79; Dek81], but the third conjecture is often attributed to Keane in [Kea91].

Facing the difficulties raised by κ , researchers started to investigate similar sequences in the hope of finding better answers. In Section 6.1 we introduce the generalization called *smooth sequences*: over an alphabet of integers $\{a, b\}$, smooth sequences are sequences that are stable by the run-length encoding. After defining these sequences, we explain how they can be computed thanks to a conjugacy with the full-shift over $\{a, b\}$. Then we focus on remarkable examples of morphic sequences obtained as fixed points like κ , to which we can directly apply classical results about substitutions. Finally we conjecture a key dichotomy within smooth sequences, which translates into the two complementary approaches for the study smooth sequences that we will take in Chapters 7 and 8 respectively.

Finally, Section 6.2 recalls how to compute the complexity over a binary alphabet from the bispecial factors, as it will be a central tool in Chapters 7 and 8.

6.1. Smooth sequences

Definition 6.1.1. A binary alphabet is a set $\mathcal{A} = \{a, b\}$ where a and b are integers such that $1 \leq a < b$. If a and b are both even (resp. both odd), we say that \mathcal{A} is even (resp. odd). Otherwise, i.e., if $a + b$ is odd, we say that \mathcal{A} is mixed.

Definition 6.1.2. Over a binary alphabet $\mathcal{A} = \{a, b\}$, the complement of a word $u \in \mathcal{A}^*$ (resp. $x \in \mathcal{A}^{\mathbb{N}}$) is the word \bar{u} (resp. \bar{x}) where every letter a is replaced with the letter b and vice versa.

Over a binary alphabet $\mathcal{A} = \{a, b\}$, any sequence $x \in \mathcal{A}^{\mathbb{N}}$ where the letters a and b occur infinitely often is canonically factorized as $x_0^{p_0} \bar{x}_0^{p_1} x_0^{p_2} \bar{x}_0^{p_3} \dots$ where x_0 is the first letter of x and $p_i \geq 1$ for all $i \geq 0$. We can now define the run-length encoding with the vocabulary of derivable functions as it is the norm in the literature.

Definition 6.1.3. Over a binary alphabet \mathcal{A} , we say that a sequence $x \in \mathcal{A}^{\mathbb{N}}$ is derivable if its canonical factorization is $x_0^{p_0} \bar{x}_0^{p_1} x_0^{p_2} \bar{x}_0^{p_3} \dots$ with $p_i \in \mathcal{A}$ for all $i \geq 0$. We write \mathcal{C}^1 the set of derivable sequences and we define the derivative as the map

$$\begin{aligned} \mathcal{D} : \quad \mathcal{C}^1 &\longrightarrow \mathcal{A}^{\mathbb{N}} \\ x_0^{p_0} \bar{x}_0^{p_1} x_0^{p_2} \dots &\longmapsto p_0 p_1 p_2 \dots \end{aligned}$$

Example 6.1.4. Over $\{1, 2\}$ we have $(221)^\omega \in \mathcal{C}^1$ and $\mathcal{D}((221)^\omega) = (21)^\omega \in \mathcal{C}^1$ but $\mathcal{D}((21)^\omega) = 1^\omega \notin \mathcal{C}^1$.

Definition 6.1.5. For $n \geq 2$, we define by induction the set of sequences that are derivable n -times

$$\mathcal{C}^n := \{x \in \mathcal{C}^1 \mid \mathcal{D}(x) \in \mathcal{C}^{n-1}\}.$$

We then define the set of smooth sequences, that is the set of sequences that are infinitely derivable

$$\mathcal{C}^\infty := \bigcap_{n \geq 1} \mathcal{C}^n.$$

With this definition, the Oldenburger-Kolakoski sequence κ satisfies $\mathcal{D}(\kappa) = \kappa$ so it is trivially smooth, and the proof that κ is aperiodic is easily adaptable to any smooth sequence.

Proposition 6.1.6. *No smooth sequence is eventually periodic.*

Proof. Suppose that a sequence $x \in \mathcal{A}^{\mathbb{N}}$ is smooth, eventually periodic, and that its period $u \in \mathcal{A}^+$ is the smallest period among all smooth sequences. Then $\mathcal{D}(x)$ is also smooth and eventually periodic, and its period is strictly shorter than u unless $\mathcal{A} = \{1, b\}$ and $u \in \{1b, b1\}$. In the first case it contradicts the minimality of u , and in the second case $\mathcal{D}(x)$ is not smooth. \square

Lastly, we must warn the reader that the set \mathcal{C}^{∞} is not a subshift.

Remark 6.1.7. *The set \mathcal{C}^{∞} is closed but it is not shift-invariant: if $\mathcal{A} \neq \{1, 2\}$, we can shift any smooth sequence until it is not derivable. Over $\{1, 2\}$, we have for example $\mathcal{D}^4(S(\kappa_{2,1})) = \mathcal{D}^4(2112122212211211\dots) = \mathcal{D}^3(12112122212\dots) = \mathcal{D}^2(1121121\dots) = \mathcal{D}(2121\dots) = 111\dots \notin \mathcal{C}^1$, so $S(\kappa_{2,1}) \notin \mathcal{C}^{\infty}$.*

6.1.1. Conjugacy to the full-shift

In [Dek01], Dekking characterized smooth sequences over $\{1, 2\}$ by conjugating them to $\{1, 2\}^{\mathbb{N}}$. This was later generalized in the series of articles [BBC05; Br1+06; BJP08; Jam+26] with the map called Φ , we define it in a slightly different way here and we call it the *trace*.

Definition 6.1.8. *Over a binary alphabet $\mathcal{A} = \{a, b\}$, the trace is the map*

$$\begin{aligned} \mathcal{T} : \mathcal{C}^{\infty} &\longrightarrow \mathcal{A}^{\mathbb{N}} \\ x &\longmapsto (\mathcal{D}^n(x)_0)_{n \geq 0}. \end{aligned}$$

In Section 4 of [Dek01], Dekking noticed that the trace is a bijection over $\{1, 2\}$. This easily extends to any binary alphabet, and we build here the reciprocal of \mathcal{T} as a computable sequence of prefixes.

Definition 6.1.9. *Over a binary alphabet $\mathcal{A} = \{a, b\}$, the r-primitives are the maps*

$$\begin{aligned} \mathcal{P}_{r,a} : \mathcal{A}^* &\longrightarrow \mathcal{A}^+ & \mathcal{P}_{r,b} : \mathcal{A}^* &\longrightarrow \mathcal{A}^+ \\ u_1 \dots u_n &\longmapsto \begin{cases} a^{u_1} b^{u_2} \dots b^{u_n} a^a & \text{if } n \text{ is even,} \\ a^{u_1} b^{u_2} \dots a^{u_n} b^a & \text{otherwise.} \end{cases} & u &\longmapsto \overline{\mathcal{P}_{r,a}(u)} \end{aligned}$$

If $u \in \mathcal{A}^n$, we also define the r- u -primitive as the composition

$$\mathcal{P}_{r,u} := \mathcal{P}_{r,u_1} \circ \mathcal{P}_{r,u_2} \circ \dots \circ \mathcal{P}_{r,u_n}.$$

tion 6.1.12 provides a characterization of these smooth sequences.

Lemma 6.1.14. *Over a binary alphabet $\mathcal{A} = \{a, b\}$, the two fixed points of \mathcal{D} are the smooth sequences $\mathcal{E}(a^\omega)$ and $\mathcal{E}(b^\omega)$. In particular, if $a = 1$, we observe that $\mathcal{E}(1^\omega) = 1\mathcal{E}(b^\omega)$.*

Proof. Proposition 6.1.12 directly yields $\mathcal{D}(\mathcal{E}(a^\omega)) = \mathcal{E}(S(a^\omega)) = \mathcal{E}(a^\omega)$ and $\mathcal{D}(\mathcal{E}(b^\omega)) = \mathcal{E}(S(b^\omega)) = \mathcal{E}(b^\omega)$. Conversely, if $x \in \mathcal{C}^\infty$ satisfies $\mathcal{D}(x) = x$, Proposition 6.1.12 yields $S(\mathcal{T}(x)) = \mathcal{T}(\mathcal{D}(x)) = \mathcal{T}(x)$ so $\mathcal{T}(x) \in \{a^\omega, b^\omega\}$. \square

Definition 6.1.15. *Over a binary alphabet $\mathcal{A} = \{a, b\}$, we write $\kappa_{a,b} := \mathcal{E}(a^\omega)$ and $\kappa_{b,a} := \mathcal{E}(b^\omega)$.*

Example 6.1.16. *The Oldenburger-Kolakoski sequence is now denoted by $\kappa_{2,1}$. Here are other examples of fixed points:*

$$\begin{aligned}\kappa_{3,1} &= 333111333131333111333133313331113331313331113331333111333133\dots \\ \kappa_{2,4} &= 224422224444224422442222444422224444224422224444224422224444\dots \\ \kappa_{2,5} &= 2255222255552255225522555522225555222255552255222255\dots\end{aligned}$$

Dekking noticed in [Dek79] the remarkable fact that $\kappa_{3,1}$ is a morphic sequence (cf. Section 4.1.2). He later gave the details in the introduction of [Dek01], the idea is to mark the parity of the letters 1 and 3 in $\kappa_{3,1}$ with four letters. Then classical results about substitutions directly apply and provide strong results.

Proposition 6.1.17. *(i) We have $\kappa_{3,1} = \tau(\sigma^\infty(3))$ with the substitution $\sigma : 1 \mapsto 3, 1' \mapsto 1', 3 \mapsto 33'3, 3' \mapsto 1'11'$ and the coding $\tau : 1 \mapsto 1, 1' \mapsto 1, 3 \mapsto 3, 3' \mapsto 3$,
(ii) $\kappa_{3,1}$ is linearly recurrent,
(iii) $p_{\kappa_{3,1}}(n) = \Theta(n)$,
(iv) $\kappa_{3,1}$ has uniform frequencies, in particular its frequency of 3's is an algebraic number approximately equal to 0.6028.*

Proof. (i) We refer to Section 4.1.2.

(ii) We observe that σ is ℓ -primitive and tame (see the definitions in Section 5.1.2). Then Theorem 5.1.10 states that the subshift X_σ is minimal and Corollary 5.1.12 states that X_σ is linearly recurrent, so a fortiori the sequence $\sigma^\infty(3)$ is linearly recurrent. Finally, $\kappa_{3,1}$ remains linearly recurrent when applying the coding τ .

(iii) Since $\kappa_{3,1}$ is linearly recurrent, [DHS99, Theorem 24 i)] states that it has at most linear factor complexity, but Proposition 6.1.6 ensures that $\kappa_{3,1}$ is aperiodic so Theorem 2.2.5 states that it has at least linear factor complexity.

(iv) Since $\kappa_{3,1}$ is linearly recurrent, [Dur00, Theorem 15] states that it has uniform frequencies. Also, the letter frequencies of the purely morphic sequence $\sigma^\infty(3)$ are given by the Perron-Frobenius eigenvector of its incidence matrix, and the frequencies of the letters 3 and 3' in $\sigma^\infty(3)$ sum to the frequency of the letter 3 in $\kappa_{3,1}$. \square

We notice that $\kappa_{3,1}$ differs from Conjecture 6.0.1 about the Oldenburger-Kolakoski sequence on properties (iii) and (iv). In [BS04] the authors extensively studied the properties of $\kappa_{3,1}$, but they used a different substitutive representation relying on a primitive substitution over three letters instead of the ℓ -primitive and tame substitution over four letters given in Proposition 6.1.17. In fact, Dekking’s construction and Sing’s construction can both be generalized to every fixed point over an even and odd alphabet, which provides linearly recurrence, linear factor complexity and computable frequencies for all. Sing detailed his construction in the overview [Sin10], let us illustrate it on the simple case of the even alphabet $\{2, 4\}$.

Proposition 6.1.18. (i) We have $\kappa_{2,4} = \tau(\sigma^\infty(A))$ whith the substitutions $\sigma : A \mapsto AB, B \mapsto AAB B$ and $\tau : A \mapsto 22, B \mapsto 44,$

(ii) $\kappa_{2,4}$ is linearly recurrent,

(iii) $p_{\kappa_{2,4}}(n) = \Theta(n),$

(iv) $\kappa_{2,4}$ has uniform frequencies, in particular it has equal letter frequencies.

Proof. The proof is identical to Proposition 6.1.17. □

Once again, we notice that $\kappa_{2,4}$ differs from Conjecture 6.0.1 about the Oldenburger-Kolakoski sequence on property (iii). However, we easily observe that any smooth sequence over an even alphabet has equal letter frequencies.

6.1.3. Dichotomy

By studying the sequences $\kappa_{a,b}$ over various alphabets with Conjecture 6.0.1 or Propositions 6.1.17 and 6.1.18, we observe a huge gap of difficulty between the Oldenburger-Kolakoski $\kappa_{2,1}$ and the morphic smooth sequences like $\kappa_{3,1}$ or $\kappa_{2,4}$. This is the symptom of a profound dichotomy within smooth sequences that we state for the first time in its most general form.

Conjecture 6.1.19. *Smooth sequences are split between two groups that have the same properties.*

1. *Over a mixed alphabet $\mathcal{A} = \{a, b\}$, every smooth sequence has the same language and its factor complexity grows like $\Theta(n^\rho)$ where $\rho = \frac{\log(a+b)}{\log\left(\frac{a+b}{2}\right)}$.*
2. *Over even and odd alphabets, smooth sequences have different languages but they have a substitutive structure.*

Remark 6.1.20. *If the conjecture for mixed alphabets holds then there is no morphic smooth sequence over a mixed alphabet, otherwise with Theorem 4.1.15 its factor complexity would grow like $\Theta(n^{1+1/k})$ for some $k \geq 1$ and we would get $a + b = 2^{k+1}$.*

The next two chapters will explore one side of this dichotomy each. In Chapter 7 we will study the complexity of the language behind the conjecture for mixed alphabets, and in Chapter 8 we will take advantage of the substitutive structure of smooth sequences over even and odd alphabets to obtain uniform recurrence, to describe their factor complexity and even obtain unique ergodicity in some cases.

6.2. Factor complexity and bispecial words

Let us recall how to compute the complexity of a language over a binary alphabet from the bispecial factors (we refer to Section 3.3 of [Cas97]).

First, recall that a language \mathcal{L} is *factorial* if $u \sqsubset v \in \mathcal{L}$ implies $u \in \mathcal{L}$. Also, \mathcal{L} is *left-extendable* (resp. *right-extendable*) if, for all $u \in \mathcal{L}$, there exists $a \in \mathcal{A}$ such that $au \in \mathcal{L}$ (resp. $ua \in \mathcal{L}$).

Definition 6.2.1. *Let $\mathcal{A} = \{a, b\}$ be a binary alphabet and let $\mathcal{L} \subseteq \mathcal{A}^*$ be a factorial, left- and right-extendable language. A word $u \in \mathcal{L}$ is bispecial if $\{au, bu, ua, ub\} \subseteq \mathcal{L}$. In that case, the set $\{aua, aub, bua, bub\} \cap \mathcal{L}$ has 2, 3 or 4 elements and its cardinal determines the type of u : we say that u is weak if it has 2 elements, neutral if it has 3 and strong if it has 4.*

Let $BS(n)$ be the set of bispecial factors of \mathcal{L} of length n . Let $s(n)$ be the first finite difference of the factor complexity, i.e., $s(n) = p_{\mathcal{L}}(n+1) - p_{\mathcal{L}}(n)$, and let $b(n)$ be the second finite difference, i.e., $b(n) = s(n+1) - s(n)$. The seminal result that allows to compute the complexity of the language \mathcal{L} is the following.

Theorem 6.2.2 ([Cas97, Proposition 3.2]). *Let \mathcal{L} be a factorial, left- and right-extendable language, and let $bs(n)$ (resp. $bw(n)$) denote the number of strong (resp. weak) bispecial words of \mathcal{L} . Then for all $n \geq 0$ we have*

$$b(n) = bs(n) - bw(n).$$

With this theorem, studying the strong and weak bispecial words of a language, and specifically their length, allows to recover precisely the factor complexity (instead of describing all words of each length).

7. Complexity of smooth words

Oh ain't that funny
All these words are made for me

Caravan Palace

Content

| | |
|---|-----|
| 7.1. Preliminaries | 90 |
| 7.1.1. Smooth words | 90 |
| 7.1.1.1. smooth words | 90 |
| 7.1.1.2. The language of smooth sequences | 91 |
| 7.1.1.3. The complexity of smooth words | 91 |
| 7.1.1.4. A mistake in a paper | 93 |
| 7.1.2. New results | 94 |
| 7.2. Factors of smooth sequences | 94 |
| 7.2.1. r -smooth words | 95 |
| 7.2.2. Prefixes and factors of smooth sequences | 95 |
| 7.3. Bispecial smooth words | 96 |
| 7.3.1. Finite primitives | 96 |
| 7.3.2. Reduction of bispecial smooth words | 97 |
| 7.3.3. Families of bispecial smooth words | 98 |
| 7.3.4. Computation of the complexity | 99 |
| 7.4. Bounds of the complexity | 100 |
| 7.4.1. The lower bound over any alphabet | 101 |
| 7.4.2. An upper bound from ℓ_i | 103 |
| 7.4.3. The upper bound over even alphabets | 103 |
| 7.4.4. An upper bound over odd alphabets | 104 |

In the framework of smooth sequences, we conjectured in Section 6.1.3 that, over a mixed alphabet $\mathcal{A} = \{a, b\}$, every smooth sequence has the same language and its factor complexity grows like $\Theta(n^\rho)$ where $\rho = \log(a + b)/\log((a + b)/2)$. The language in question was first described over $\{1, 2\}$ by Dekking in [Dek81] and was called C^∞ -words. The definition naturally extends to any binary alphabet, but we change the name to fit with our general framework of smooth sequences: *smooth words* are

the finite words that are infinitely derivable with the derivative \mathcal{D}_f on finite words that simulates locally the derivative \mathcal{D} on infinite words. If \mathcal{C}_f^∞ denotes the set of smooth words, it follows from the definition that, over any alphabet, every smooth sequence x satisfies $\mathcal{L}(x) \subseteq \mathcal{C}_f^\infty$. In [Dek81], Dekking conjectured that $\mathcal{L}(\kappa) = \mathcal{C}_f^\infty$ for the Oldenburger-Kolakoski sequence κ over $\{1, 2\}$, and we generalize this conjecture to every smooth sequence over mixed alphabets.

Conjecture 7.0.1. *Over a mixed alphabet, every smooth sequence x satisfies $\mathcal{L}(x) = \mathcal{C}_f^\infty$, and in particular $p_x(n) = p_{\mathcal{C}^\infty}(n) = p_{\mathcal{C}_f^\infty}(n)$.*

Nothing is known about this conjecture, but the conjecture about the Oldenburger-Kolakoski sequence κ has led to a thorough investigation of the properties of smooth words. For instance, heavy computation in [Chv93; Rao12; Nil14] suggest that the asymptotical frequencies of letters in smooth words over $\{1, 2\}$ is $1/2$, supporting Keane’s conjecture that κ has equal letter frequencies. Researchers also studied repetitions in smooth words: Carpi showed in [Car94] that smooth words are cube-free, and Huang generalized it in [Hua11b] by showing that smooth words are $(b+1)$ -th-power-free over any alphabet $\{a, b\}$ where $a < b$, except over $\{1, 3\}$ where they are 5-th-power-free.

In this chapter we focus on the complexity of the language \mathcal{C}_f^∞ . Over $\{1, 2\}$, Dekking conjectured in [Dek81] that $p_{\mathcal{C}_f^\infty}(n) = \Theta(n^{\log(3)/\log(3/2)})$, and it was generalized by Sing in [Sin10].

Conjecture 7.0.2 ([Sin10]). *Over a binary alphabet $\{a, b\}$, we have $p_{\mathcal{C}_f^\infty}(n) = \Theta(n^\rho)$ where $\rho = \frac{\log(a+b)}{\log\left(\frac{a+b}{2}\right)}$.*

Towards this conjecture, Dekking showed in [Dek81] that $p_{\mathcal{C}_f^\infty}(n)$ has polynomial lower and upper bounds over $\{1, 2\}$ and Sing [Sin02] generalized these bounds to any alphabet. Later Weakley studied the bispecial smooth words, in [Wea89] he described how they can be used to prove Conjecture 7.0.2 over $\{1, 2\}$, and in [HW10] he and Huang obtained other polynomial bounds that depend on letter frequencies. Huang later claimed in [Hua11a] to generalize this last result to any alphabet, but we found a mistake that we explain in Section 7.1.1.4. Now we arrive to our contributions to the study of smooth words, based on a joint work with Julien Cassaigne (preprint [CH26]).

- In Theorem 7.1.18 we show that, over any binary alphabet, any smooth words occurs in at least one smooth sequences. This yields the equality of complexities $p_{\mathcal{C}^\infty}(n) = p_{\mathcal{C}_f^\infty}(n)$ and further motivates the study of \mathcal{C}_f^∞ .
- In Theorem 7.1.19 we prove the conjectured lower bound of $p_{\mathcal{C}_f^\infty}(n)$ over any alphabet and the conjectured upper bound over *even* alphabets (i.e., a and b are even).
- In Theorem 7.1.20 we give a new upper bound of $p_{\mathcal{C}_f^\infty}(n)$ over *odd* alphabets (i.e., a and b are odd).

7.1. Preliminaries

Recall that, over a binary alphabet $\mathcal{A} = \{a, b\}$, the *complement* of a word $u \in \mathcal{A}^*$ (resp. $x \in \mathcal{A}^{\mathbb{N}}$) is the word \bar{u} (resp. \bar{x}) where every letter a is replaced with the letter b and vice versa.

7.1.1. Smooth words

7.1.1.1. smooth words

Any binary word $u \in \mathcal{A}^*$ is canonically factorized as $a_1^{p_1} \dots a_n^{p_n}$ where $a_i \in \mathcal{A}$, $a_{i+1} = \bar{a}_i$ and $p_i \geq 1$, with the convention that $n = 0$ if $u = \varepsilon$.

Definition 7.1.1. Over a binary alphabet \mathcal{A} , we say that a word $u \in \mathcal{A}^*$ is *derivable* if its canonical factorization is $a_1^{p_1} \dots a_n^{p_n}$ with $p_i \in \mathcal{A}$ for all $i \in \llbracket 2, n-1 \rrbracket$ and $p_1, p_n \in \llbracket 1, b \rrbracket$. We write \mathcal{C}_f^1 the set of finite derivable words and we define the finite derivative as the map

$$\begin{aligned} \mathcal{D}_f : \quad \mathcal{C}_f^1 &\longrightarrow \mathcal{A}^* \\ a_1^{p_1} \dots a_n^{p_n} &\longmapsto \text{cut}(p_1) p_2 \dots p_{n-1} \text{cut}(p_n) \end{aligned}$$

where $\text{cut}(p) = \begin{cases} \varepsilon & \text{if } 1 \leq p \leq a \\ b & \text{if } a < p \leq b \end{cases}$, $\mathcal{D}_f(\varepsilon) = \varepsilon$ and $\mathcal{D}_f(a_1^{p_1}) = \text{cut}(p_1)$.

Example 7.1.2. Over $\{1, 2\}$, \mathcal{C}_f^1 is the set of words that do not contain the factors 111 or 222, and we have $\mathcal{D}_f(\varepsilon) = \mathcal{D}_f(1) = \mathcal{D}_f(21) = \varepsilon$ and $\mathcal{D}_f(2211) = \mathcal{D}_f(122112) = 22$. Over $\{1, 3\}$, we have $\mathcal{D}_f(331113) = 33$.

Definition 7.1.3. For $n \geq 2$, we define by induction the set of finite words that are derivable n times

$$\mathcal{C}_f^n := \left\{ u \in \mathcal{C}_f^1 \mid \mathcal{D}_f(u) \in \mathcal{C}_f^{n-1} \right\}.$$

We then define the set of smooth words, that is the set of finite words that are infinitely derivable

$$\mathcal{C}_f^\infty := \bigcap_{n \geq 1} \mathcal{C}_f^n.$$

We notice that the map \mathcal{D}_f is *contracting*, i.e., $|\mathcal{D}_f(u)| < |u|$ if $u \in \mathcal{C}_f^1 \setminus \{\varepsilon\}$. This means that a finite word is smooth if and only if it can be derived down to ε .

Definition 7.1.4. The *height* of a smooth word u is the first integer $n \geq 0$ such that $\mathcal{D}_f^n(u) = \varepsilon$.

Example 7.1.5. Over $\{1, 2\}$, we have $\mathcal{D}_f^4(221121221) = \mathcal{D}_f^3(22112) = \mathcal{D}_f^2(22) = \mathcal{D}_f(2) = \varepsilon$ so $221121221 \in \mathcal{C}_f^\infty$ with height 4, but $\mathcal{D}_f(12121) = 111 \notin \mathcal{C}_f^1$ so $12121 \notin \mathcal{C}_f^\infty$.

One can easily show that the set \mathcal{C}_f^∞ is factorial, left- and right-extendable, we refer to [Wea89, Proposition 2] for the proof over $\{1, 2\}$.

7. Complexity of smooth words – 7.1. Preliminaries

While this does not prove Conjecture 7.0.2, it still provides the first polynomial bounds for any binary alphabet. This implies zero entropy for all smooth sequences (recall that the entropy of a word $x \in \mathcal{A}^{\mathbb{N}}$ is the finite quantity $\lim_{n \rightarrow \infty} \log(p_x(n)/n)$).

Corollary 7.1.9. *Every smooth sequence has zero entropy.*

In [Wea89], Weakley took another approach by studying the bispecial smooth words over $\{1, 2\}$.

Theorem 7.1.10 ([Wea89, Corollary 9]). *Over $\{1, 2\}$, let $\rho = \log(3)/\log(3/2)$. Then there exists two constants $0 < C_1 < C_2$ such that*

$$C_1 n^\rho \leq p_{\mathcal{C}_f^\infty}(n) \leq C_2 n^\rho$$

for each n satisfying $L_{i-1} + 1 \leq n \leq \ell_i + 1$ for some $i \geq 0$, where ℓ_i (resp. L_i) is the minimum (resp. maximum) length of bispecial smooth words of height i .

This result led Weakley to ask when $L_{i-1} \leq \ell_i$ holds, computations suggest that it holds for all $i \geq 0$ but it is still an open question. In fact this directly relates to the letter frequencies in smooth words, as shown in the next result of Weakley and Huang.

Theorem 7.1.11 ([HW10, Theorem 3]). *Over $\{1, 2\}$, if $\phi < 1/2$ is a positive real number and N is a positive integer such that $\frac{|u|_2}{|u|} > \frac{1}{2} - \phi$ for all $u \in \mathcal{C}_f^\infty$ such that $|u| > N$, then there are positive constants $C_1, C_2 > 0$ such that, for all $n \geq 1$, we have*

$$C_1 n^\delta < p_{\mathcal{C}_f^\infty}(n) < C_2 n^\gamma$$

where $\delta = \frac{\log(3)}{\log(3/2 + \phi + 2/N)}$ and $\gamma = \frac{\log(3)}{\log(3/2 - \phi)}$.

Remark 7.1.12. *Keane famously conjectured in [Kea91] that $\kappa_{2,1}$ has equal letter frequencies. More generally, over any mixed alphabet, we think that the letter frequencies in smooth words are asymptotically $1/2$. In particular this supports Conjecture 7.0.2 when combined with Theorem 7.1.11. However, the conjecture about letter frequencies fails over odd alphabets: for example the smooth sequence $\kappa_{3,1}$ studied in Proposition 6.1.17 has frequencies of 1 and 3 that are not equal.*

Theorem 7.1.11 can be combined with the computations of [Rao12] and [Nil14] that provide $\limsup_{u \in \mathcal{C}_f^\infty} \frac{|u|_2}{|u|} > \frac{1}{2} - 0.00008$ over $\{1, 2\}$.

Corollary 7.1.13. *Over $\{1, 2\}$, let $\rho = \log(3)/\log(3/2)$. Then there exists a constant $C > 0$ such that, for all $n \geq 0$,*

$$p_{\mathcal{C}_f^\infty}(n) < C n^{\rho + 0.00036}.$$

7.1.1.4. A mistake in a paper

In [Hua11a], Huang claimed to generalize Theorem 7.1.11 to any binary alphabet.

Claim 7.1.14 ([Hua11a, Theorem 12]). *Let $\mathcal{A} = \{a, b\}$ be a binary alphabet. If ϕ is a positive real number and N is a positive integer such that $\frac{|u|_2}{|u|} > \phi$ for all $u \in \mathcal{C}_f^\infty$ such that $|u| > N$, then there are positive constants $C_1, C_2 > 0$ such that, for all $n \geq 0$, we have*

$$C_1 n^\delta < p_{\mathcal{C}_f^\infty}(n) < C_2 n^\gamma$$

where $\delta = \frac{\log(2b-1)}{\log(1+(a+b-2)(1-\phi))}$ and $\gamma = \frac{\log(2b-1)}{\log(1+(a+b-2)\phi)}$.

If we expect ϕ to be arbitrarily close to 1/2, as it is conjectured for even and mixed alphabets, Claim 7.1.14 would suggest that $p_{\mathcal{C}_f^\infty}(n) = \Theta(n^{\rho'})$ where $\rho' = \frac{\log(2b-1)}{\log\left(\frac{a+b}{2}\right)}$. In the same paper, Huang also claimed to give the asymptotic behavior of $p_{\mathcal{C}_f^\infty}(n)$ over even alphabets.

Claim 7.1.15 ([Hua11a, Theorem 16]). *Let $\mathcal{A} = \{a, b\}$ be an even alphabet. Then*

$$p_{\mathcal{C}_f^\infty}(n) = \Theta(n^{\rho'}) \quad \text{where } \rho' = \frac{\log(2b-1)}{\log\left(\frac{a+b}{2}\right)}.$$

We notice that Claims 7.1.14 and 7.1.15 conflict with Conjecture 7.0.2, and in fact they are false. The mistake comes from the fact that the proofs use an incorrect definition of finite derivative: the correct operation is \mathcal{D}_f as introduced in Definition 7.1.1, note that it is also defined in [Hua11a] and denoted by ρ ; but the author uses the different operation D defined as follows.

Definition 7.1.16. *Over a binary alphabet \mathcal{A} , we define the map*

$$D : \quad \mathcal{C}_f^1 \longrightarrow \mathcal{A}^* \\ a_1^{p_1} \dots a_n^{p_n} \longmapsto \text{cut}'(p_1) p_2 \dots p_{n-1} \text{cut}'(p_n)$$

where $\text{cut}'(p) = \begin{cases} \varepsilon & \text{if } 1 \leq p < b \\ b & \text{if } p = b \end{cases}$, with the convention that $D(\varepsilon) = \varepsilon$.

The map D happens to be identical to \mathcal{D}_f when $a = b - 1$, but not anymore when $a < b - 1$. The consequence is that the set of "smooth words" defined by D is bigger than \mathcal{C}_f^∞ (hence $\rho' > \rho$) and contains finite words that cannot occur in smooth sequences, as illustrated in the following example.

Example 7.1.17. *Over $\{1, 4\}$, the word $u := 4^4 1^4 4^4 1^4 4^3$ is "smooth" with regard to D since $D^3(u) = D^2(4^4) = D(4) = \varepsilon$, but every occurrence of u in an infinite word has to be followed by a 4 and the word $u4$ cannot occur in a smooth sequence. However, u is not smooth with regard to \mathcal{D}_f since $\mathcal{D}_f(u) = 4^5 \notin \mathcal{C}_f^1$.*

7.1.2. New results

In this chapter we contribute to Conjectures 7.0.1 and 7.0.2. We begin in Section 7.2 by showing that the smooth words are exactly the factors of smooth sequences.

Theorem 7.1.18. *Over any binary alphabet, we have $\mathcal{L}(\mathcal{C}^\infty) = \mathcal{C}_f^\infty$, and in particular $p_{\mathcal{C}^\infty}(n) = p_{\mathcal{C}_f^\infty}(n)$.*

This does not prove Conjecture 7.0.1 but it motivates even more the study of the complexity $p_{\mathcal{C}_f^\infty}(n)$. Next, in Section 7.3 we extend the description of the bispecial smooth words of [Wea89] to any binary alphabet, and in Section 7.4 we solve Conjecture 7.0.2 over even alphabets and we obtain the lower bound over the other alphabets.

Theorem 7.1.19. *Let $\mathcal{A} = \{a, b\}$ and let $\rho = \frac{\log(a+b)}{\log\left(\frac{a+b}{2}\right)}$.*

- (i) *We have $p_{\mathcal{C}_f^\infty}(n) = \Omega(n^\rho)$.*
- (ii) *If \mathcal{A} is even, then $p_{\mathcal{C}_f^\infty}(n) = \Theta(n^\rho)$.*

With the same techniques we obtain a new upper bound over odd alphabets.

Theorem 7.1.20. *Let $\mathcal{A} = \{a, b\}$ be an odd alphabet and let $\zeta := \frac{\log(2\lambda)}{\log(\lambda)}$ where*

$$\lambda := \begin{cases} \frac{1+\sqrt{2b-1}}{2} & \text{if } a = 1, \\ \text{the dominant root of } X^3 - \frac{a+b}{2}X^2 + \frac{(b-a)^2}{4} & \text{otherwise.} \end{cases}$$

Then we have $p_{\mathcal{C}_f^\infty}(n) = \mathcal{O}(n^\zeta)$.

Remark 7.1.21. *We do not provide an explicit formula for ζ but we claim that our upper bound improves Theorem 7.1.8. In the following table we compute and compare the exponents ρ provided by Conjecture 7.0.2, ζ provided by Theorem 7.1.20 and β provided by Theorem 7.1.8 for various odd alphabets.*

| \mathcal{A} | {1,3} | {1,5} | {3,5} | {1,7} | {3,7} | {5,7} | {1,9} | {3,9} | {5,9} | {7,9} |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| ρ | 2 | 1.63 | 1.5 | 1.5 | 1.431 | 1.387 | 1.431 | 1.387 | 1.356 | 1.333 |
| ζ | 2.44 | 2 | 1.51 | 1.831 | 1.44 | 1.388 | 1.74 | 1.397 | 1.358 | 1.334 |
| β | 7.129 | 7.658 | 2.96 | 8.193 | 3.195 | 2.6 | 8.565 | 3.383 | 2.734 | 2.465 |

7.2. Factors of smooth sequences

In this section we prove Theorem 7.1.18. To do so, we use a right version of smooth words called *r-smooth words*, originally introduced in [BMP07], which will play the role of prefixes in smooth sequences.

7.2.1. r-smooth words

Definition 7.2.1. Over a binary alphabet \mathcal{A} , we say that a word $u \in \mathcal{A}^*$ is right-derivable if it is factorized as $a_1^{p_1} \dots a_n^{p_n}$ where $p_i \in \mathcal{A}$ for all $i \in \llbracket 1, n-1 \rrbracket$ and $p_n \in \llbracket 1, b \rrbracket$. We write \mathcal{C}_r^1 the set of right-derivable words and we define the right-derivative as the map

$$\begin{aligned} \mathcal{D}_r : \quad \mathcal{C}_r^1 &\longrightarrow \mathcal{A}^* \\ a_1^{p_1} \dots a_n^{p_n} &\longmapsto p_1 \dots p_{n-1} \text{ cut}(p_n) \end{aligned}$$

where the function cut is the same as in \mathcal{D}_f and $\mathcal{D}_r(a_1^{p_1}) = \text{cut}(p_1)$.

Example 7.2.2. Over $\{1, 2\}$, we have $\mathcal{D}_r(\varepsilon) = \mathcal{D}_r(2) = \varepsilon$, $\mathcal{D}_r(21) = 1$ and $\mathcal{D}_r(211) = 12$.

Definition 7.2.3. For $n \geq 2$, we define by induction the set of words that are right-derivable n times

$$\mathcal{C}_r^n := \{u \in \mathcal{C}_r^1 \mid \mathcal{D}_r(x) \in \mathcal{C}_r^{n-1}\}.$$

We then define the set of r-smooth words, that is the set of words that are infinitely right-derivable

$$\mathcal{C}_r^\infty := \bigcap_{n \geq 1} \mathcal{C}_r^n.$$

Similarly to \mathcal{C}_f^∞ , one can easily prove by induction that $u \sqsubset_p v \in \mathcal{C}_r^\infty$ implies $u \in \mathcal{C}_r^\infty$, and that the language \mathcal{C}_r^∞ is right-extendable.

7.2.2. Prefixes and factors of smooth sequences

Let us prove Theorem 7.1.18 using the fact that \mathcal{C}_r^∞ is exactly the set of prefixes of smooth sequences.

Proof of Theorem 7.1.18. Let $\mathcal{A} = \{a, b\}$ be a binary alphabet. We already observed that $\mathcal{L}(\mathcal{C}^\infty) \subseteq \mathcal{C}_f^\infty$, so it remains to prove that $\mathcal{C}_f^\infty \subseteq \mathcal{L}(\mathcal{C}^\infty)$.

The first step is to show that every smooth word is factor of an r-smooth word. For that we are going to show that, for all $u \in \mathcal{C}_f^\infty$, there exists $v \in \mathcal{A}^*$ such that $|v| \geq a + b$ and $vu \in \mathcal{C}_r^\infty$. We proceed by induction on the height of u : first, if u has height 0, i.e., $u = \varepsilon$, then the word $v = a^b b^a$ suffices. Now let $h \geq 0$ be such that, for all $u \in \mathcal{C}_f^\infty$ of height h , there exists $v \in \mathcal{A}^*$ such that $|v| \geq a + b$ and $vu \in \mathcal{C}_r^\infty$. Let $u \in \mathcal{C}_f^\infty$ be of height $h + 1$ and consider its canonical factorization $u = a_1^{p_1} \dots a_n^{p_n}$ with $n \geq 1$. The word $\mathcal{D}_f(u)$ is smooth and has height h so, by hypothesis, there exists $v \in \mathcal{A}^*$ such that $|v| \geq a + b$ and $v\mathcal{D}_f(u) \in \mathcal{C}_r^\infty$, and we can write $v = v_1 \dots v_\ell$ with $\ell \geq a + b$. In particular, $v \in \mathcal{C}_r^\infty$ and it is long enough to contain at least a occurrences of the letter b . We deduce that

$$\sum_{i=1}^{\ell} v_i = a|v|_{a+b}|v|_b = a(\ell - |v|_b) + b|v|_b = a\ell + (b-a)|v|_b \geq a(a+b) + (b-a)a \geq (a+b) + a.$$

If $p_1 \leq a$, we have $\mathcal{D}_f(u) = p_2 \dots p_{n-1} \text{cut}(p_n)$ and we set $w := c_1^{v_1} \dots c_\ell^{v_\ell} c_{\ell+1}^{p_2} \dots c_{\ell+n-1}^{p_n}$ where $c_{i+1} = \overline{c}_i$ and $c_{\ell+1} = a_2$. We first observe that $w = v'u$ where $v' := c_1^{v_1} \dots c_{\ell-1}^{v_{\ell-1}} c_\ell^{v_\ell - p_1}$

7. Complexity of smooth words – 7.3. Bispecial smooth words

and that $|v'| = \left(\sum_{i=1}^{\ell} v_i \right) - p_1 \geq (a + b + a) - a = a + b$. We also observe that $w \in \mathcal{C}_r^1$ and that $\mathcal{D}_r(w) = v\mathcal{D}_f(u)$, which implies that $w = v'u \in \mathcal{C}_r^\infty$.

If $p_1 > a$, we have $\mathcal{D}_f(u) = bp_2 \dots p_{n-1} \text{cut}(p_n)$ and we set $w := c_1^{v_1} \dots c_\ell^{v_\ell} c_{\ell+1}^b c_{\ell+2}^{p_2} \dots c_{\ell+n}^{p_n}$ where $c_{i+1} = \bar{c}_i$ and $c_{\ell+1} = a_1$. We first observe that $w = v'u$ where $v' := c_1^{v_1} \dots c_\ell^{v_\ell} c_{\ell+1}^{b-p_1}$ and that $|v'| \geq \sum_{i=1}^{\ell} v_i \geq a + b + a$. We also observe that $w \in \mathcal{C}_r^1$ and that $\mathcal{D}_r(w) = v\mathcal{D}_f(u)$, which implies that $w = v'u \in \mathcal{C}_r^\infty$.

The second step is to show that a sequence $x \in \mathcal{A}^{\mathbb{N}}$ such that all its prefixes are r -smooth is smooth. We show by induction on n that, for all $n \geq 1$, a sequence $x \in \mathcal{A}^{\mathbb{N}}$ such that all its prefixes are r -smooth satisfies $x \in \mathcal{C}^n$. If $n = 1$, we easily notice that a sequence such that all its prefixes are r -smooth is derivable. Now let $n \geq 1$ be such that a sequence $x \in \mathcal{A}^{\mathbb{N}}$ such that all its prefixes are r -smooth satisfies $x \in \mathcal{C}^n$, and let $x \in \mathcal{A}^{\mathbb{N}}$ be such that all its prefixes are r -smooth. In particular, the words $\mathcal{D}_r(x_{[0, bn)})$ are r -smooth for all $n \geq 0$ and are prefixes of strictly increasing length of the sequence $\mathcal{D}(x)$. We deduce that all prefixes of $\mathcal{D}(x)$ are r -smooth, and by hypothesis we have $\mathcal{D}(x) \in \mathcal{C}^n$ so $x \in \mathcal{C}^{n+1}$.

The third step is to show that every r -smooth word is prefix of a smooth sequence. If v is r -smooth, the fact that the language \mathcal{C}_r^∞ is right-extendable provides a sequence of r -smooth words of strictly increasing length, starting with v , which are all prefixes of a sequence $x \in \mathcal{A}^{\mathbb{N}}$. By construction we have $v \sqsubset_p x$, and all prefixes of x are r -smooth so x is smooth.

Finally, if $u \in \mathcal{C}_f^\infty$, we proved that there exists $v \in \mathcal{C}_r^\infty$ such that $u \sqsubset v$ and that there exists $x \in \mathcal{C}^\infty$ such that $v \sqsubset_p x$, so we have $u \sqsubset x$. Therefore, we have $\mathcal{C}_f^\infty \subseteq \mathcal{L}(\mathcal{C}^\infty)$. \square

7.3. Bispecial smooth words

In this section we detail how to compute $p_{\mathcal{C}_f^\infty}(n)$ from the bispecial smooth words, extending the work of Weakley [Wea89] to any binary alphabet. First we split the strong and weak bispecial smooth words into five families, each forming an infinite binary tree introduced in Definition 7.3.7. Then we show in Proposition 7.3.10 that it suffices to study a single family of strong bispecial smooth words to bound $p_{\mathcal{C}_f^\infty}(n)$.

7.3.1. Finite primitives

In order to determine the bispecial smooth words, we introduce the notion of *finite primitives*. This starts with the observation that the two longest words with derivative $u \in \mathcal{A}^n$ are canonically factorized as $a_0^a a_1^{u_1} a_2^{u_2} \dots a_n^{u_n} a_{n+1}^a$.

7. Complexity of smooth words – 7.3. Bispecial smooth words

Definition 7.3.1. Over a binary alphabet $\mathcal{A} = \{a, b\}$, the finite primitives are the maps

$$\begin{aligned} \mathcal{P}_{f,a} : \quad \mathcal{A}^* &\longrightarrow \mathcal{A}^+ & \mathcal{P}_{f,b} : \mathcal{A}^* &\longrightarrow \mathcal{A}^+ \\ u_1 \dots u_n &\longmapsto \begin{cases} a^a b^{u_1} a^{u_2} \dots a^{u_n} b^a & \text{if } n \text{ is even,} \\ a^a b^{u_1} a^{u_2} \dots b^{u_n} a^a & \text{otherwise.} \end{cases} & u &\longmapsto \overline{\mathcal{P}_{f,a}(u)} \end{aligned}$$

Example 7.3.2. Over $\{1, 2\}$, we have $\mathcal{P}_{f,1}(2) = 1221$ and $\mathcal{P}_{f,2}(2) = 2112$.

Over $\{2, 4\}$, we have $\mathcal{P}_{f,2}(42) = 2244442244$.

We make two important remarks.

Remark 7.3.3. If $u \in \mathcal{A}^*$ and $c \in \mathcal{A}$, then $\mathcal{D}_f(\mathcal{P}_{f,c}(u)) = u$. We deduce that, if u is smooth then $\mathcal{P}_{f,c}(u)$ is also smooth. If $u \sqsubset_p v \in \mathcal{A}^*$ and $c \in \mathcal{A}$, then $\mathcal{P}_{f,c}(u) \sqsubset_p \mathcal{P}_{f,c}(v)$.

7.3.2. Reduction of bispecial smooth words

Now we show that bispecial smooth words can be expressed with finite primitives and reduced to short bispecial words with the same type.

Definition 7.3.4. If $u \in \mathcal{A}^*$ is canonically factorized as $a_1^{p_1} \dots a_n^{p_n}$, we define its factorized length as $\|u\| := n$. Then we say that u is short if $\|u\| \leq 1$, otherwise we say that u is long.

Lemma 7.3.5. Let u be a bispecial smooth word over $\mathcal{A} = \{a, b\}$.

(i) If u is long, then $u = \mathcal{P}_{f,u_1}(\mathcal{D}_f(u))$ and $\mathcal{D}_f(u)$ is also bispecial, of the same type.

(ii) The words $\mathcal{P}_{f,a}(u)$ and $\mathcal{P}_{f,b}(u)$ are also bispecial, of the same type.

Proof. (i) Let $a_1^{p_1} \dots a_n^{p_n}$ be the canonical factorization of u , where $n = \|u\| \geq 2$. We have $a_1^{p_1+1} \dots a_n^{p_n} = a_1 u \in \mathcal{C}_f^\infty$ so $p_1 + 1 \leq b$, and $\overline{a_1} a_1^{p_1} \dots a_n^{p_n} = \overline{a_1} u \in \mathcal{C}_f^\infty$ so $p_1 \in \{a, b\}$, therefore $p_1 = a$; and symmetrically we obtain $p_n = a$. We deduce that $u = a_1^a a_2^{p_2} \dots a_{n-1}^{p_{n-1}} a_n^a$ and $\mathcal{D}_f(u) = p_2 \dots p_{n-1}$, which yields $u = \mathcal{P}_{f,u_1}(\mathcal{D}_f(u))$ with the fact that $a_1 = u_1$.

Next, the fact that $\mathcal{D}_f(u)$ is also a bispecial smooth word with the same type as u relies on the following equivalences:

$$a_1 u a_n \in \mathcal{C}_f^\infty \iff b \mathcal{D}_f(u) b \in \mathcal{C}_f^\infty \quad (4)$$

$$a_1 u \overline{a_n} \in \mathcal{C}_f^\infty \iff b \mathcal{D}_f(u) a \in \mathcal{C}_f^\infty \quad (5)$$

$$\overline{a_1} u a_n \in \mathcal{C}_f^\infty \iff a \mathcal{D}_f(u) b \in \mathcal{C}_f^\infty \quad (6)$$

$$\overline{a_1} u \overline{a_n} \in \mathcal{C}_f^\infty \iff a \mathcal{D}_f(u) a \in \mathcal{C}_f^\infty \quad (7)$$

These equivalences rely on the following trivial equivalence: if $v \in \mathcal{A}^*$, then $v \in \mathcal{C}_f^\infty$ if and only if $v \in \mathcal{C}_f^1$ and $\mathcal{D}_f(v) \in \mathcal{C}_f^\infty$. Then $\mathcal{D}_f(a_1 u a_n) = b \mathcal{D}_f(u) b$ yields (4), $\mathcal{D}_f(a_1 u \overline{a_n}) = b \mathcal{D}_f(u) a$ yields (5), $\mathcal{D}_f(\overline{a_1} u a_n) = a \mathcal{D}_f(u) b$ yields (6) and $\mathcal{D}_f(\overline{a_1} u \overline{a_n}) = a \mathcal{D}_f(u) a$ yields (7).

Now we have $\overline{a_1} u \in \mathcal{C}_f^\infty$ because u is bispecial, so either $\overline{a_1} u a_n$ or $\overline{a_1} u \overline{a_n} \in \mathcal{C}_f^\infty$, and then (4) and (5) yield $b \mathcal{D}_f(u) \in \mathcal{C}_f^\infty$. Similarly we show that $a \mathcal{D}_f(u)$, $\mathcal{D}_f(u) a$ and

7. Complexity of smooth words – 7.3. Bispecial smooth words

$\mathcal{D}_f(u)b \in \mathcal{C}_f^\infty$, which means that $\mathcal{D}_f(u)$ is bispecial. Finally, (4), (5), (6) and (7) ensure that $\mathcal{D}_f(u)$ has the same type as u .

(ii) Let $v = \mathcal{P}_{f,c}(u)$ with $c \in \mathcal{A}$. In particular, we have $\|v\| \geq 2$ so, if $v = a_1^{p_1} \dots a_n^{p_n}$, then (4), (5), (6) and (7) hold for v and $\mathcal{D}_f(v) = u$. Now we have $au \in \mathcal{C}_f^\infty$ because u is bispecial, so either aua or $aub \in \mathcal{C}_f^\infty$, and then (6) and (7) yield $\overline{a_1}v \in \mathcal{C}_f^\infty$. Similarly we show that a_1v , va_1 and $v\overline{a_1} \in \mathcal{C}_f^\infty$, which means that v is bispecial. Finally, (4), (5), (6) and (7) ensure that v has the same type as u . \square

This means that every bispecial smooth word can be derived down to a short bispecial smooth word that we call its *root* and that has the same type.

7.3.3. Families of bispecial smooth words

Now let us determine the strong and weak roots of bispecial smooth words.

Lemma 7.3.6. *Let $\mathcal{A} = \{a, b\}$ be a binary alphabet.*

(i) *If $a = b - 1$, then ε is the unique short strong bispecial smooth word and there is no short weak bispecial smooth word.*

(ii) *If $a < b - 1$, then the short strong bispecial smooth words are ε , a^a and b^a ; and the short weak bispecial smooth words are a^{b-1} and b^{b-1} .*

Proof. The short smooth words are all the words of the form c^n with $c \in \mathcal{A}$ and $n \in \llbracket 0, b \rrbracket$, so we directly check their bi-extensions in \mathcal{C}_f^∞ .

| | $cc^n c$ | $cc^n \overline{c}$ | $\overline{c}c^n c$ | $\overline{c}c^n \overline{c}$ | type of c^n |
|--|-------------------------------|-------------------------------|-------------------------------|--------------------------------|---------------|
| $n = 0$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | strong |
| $n \in \llbracket 1, a-1 \rrbracket \cup \llbracket a+1, b-2 \rrbracket$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\notin \mathcal{C}_f^\infty$ | neutral |
| $n = a = b - 1$ | $\notin \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | neutral |
| $n = a < b - 1$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | strong |
| $n = b - 1 > a$ | $\notin \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | $\notin \mathcal{C}_f^\infty$ | weak |
| $n = b$ | $\notin \mathcal{C}_f^\infty$ | $\notin \mathcal{C}_f^\infty$ | $\notin \mathcal{C}_f^\infty$ | $\in \mathcal{C}_f^\infty$ | not bispecial |

\square

Definition 7.3.7. *Over a binary alphabet $\mathcal{A} = \{a, b\}$, we define the infinite binary tree T as follows: its root is ε and the two children of a vertex $u \in \mathcal{A}^*$ are $\mathcal{P}_{f,a}(u)$ and $\mathcal{P}_{f,b}(u)$. By Lemma 7.3.5, the vertices of T are all the strong bispecial smooth words of root ε .*

If $a < b - 1$, we define in the same way the following infinite binary trees:

- $T^{(1)}$ is the tree of strong bispecial smooth words of root a^a ,
- $T^{(2)}$ is the tree of strong bispecial smooth words of root b^a ,
- $T^{(3)}$ is the tree of weak bispecial smooth words of root a^{b-1} ,
- $T^{(4)}$ is the tree of weak bispecial smooth words of root b^{b-1} .

Example 7.3.8. Over alphabets $\mathcal{A} = \{a, a+1\}$, the strong bispecial smooth words form the tree T . We display it over $\{1, 2\}$ in Figure 7.1.

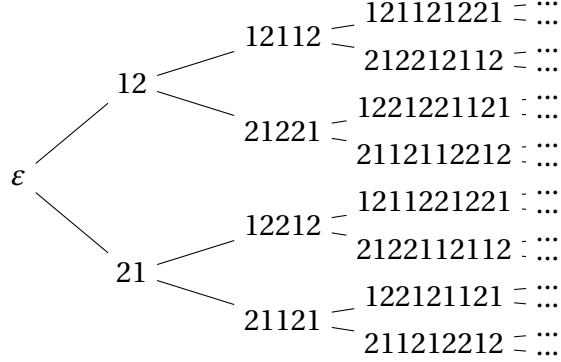


Figure 7.1.: The tree T over $\{1, 2\}$.

7.3.4. Computation of the complexity

In this subsection we apply the theory of bispecial factors (cf. Section 6.2) to compute $p_{\mathcal{C}_f^\infty}(n)$. We show that it suffices to study T in order to bound $p_{\mathcal{C}_f^\infty}(n)$, even when $a < b - 1$ and the strong and weak bispecial smooth words are spread among T , $T^{(1)}$, $T^{(2)}$, $T^{(3)}$ and $T^{(4)}$.

Definition 7.3.9. For all $i \geq 0$, we define T_i as the i -th generation of T , i.e., the set of strong bispecial smooth words whose i -th ancestor in T is ε . In particular, $|T_i| = 2^i$.

In order to count the contribution of the bispecial smooth words of T to the complexity of \mathcal{C}_f^∞ , we define the following numbers for all $n, i \geq 0$:

$$b_i(n) := |T_i \cap \mathcal{A}^n|, \quad s_i(n) := \sum_{m=0}^{n-1} b_i(m),$$

$$p_i(n) := \sum_{m=0}^{n-1} s_i(m), \quad p(n) := \sum_{i=0}^{\infty} p_i(n).$$

Note that, for a fixed n , the definition of $p(n)$ is a finite sum since $b_i(n)$ is eventually zero when i grows.

Proposition 7.3.10. Over a binary alphabet $\mathcal{A} = \{a, b\}$, for all $n \geq 0$ we have

$$1 + n + p(n) \leq p_{\mathcal{C}_f^\infty}(n) \leq 1 + n + 3p(n).$$

Proof. Let $s(n)$ be the first finite difference and let $b(n)$ be the second finite difference of $p_{\mathcal{C}_f^\infty}(n)$, as defined in Section 6.2.

If $a = b - 1$, there is no weak bispecial smooth word and every strong bispecial smooth word belongs to T , so, with Theorem 6.2.2 and by summation, for all $n \geq 0$ we

7. Complexity of smooth words – 7.4. Bounds of the complexity

have

$$\begin{aligned}
 b(n) &= |T \cap \mathcal{A}^n| = \sum_{i=0}^{\infty} b_i(n), \\
 s(n) &= s(0) + \sum_{m=0}^{n-1} b(m) = 1 + \sum_{i=0}^{\infty} \sum_{m=0}^{n-1} b_i(m) = 1 + \sum_{i=0}^{\infty} s_i(n), \\
 p_{C_f}^{\infty}(n) &= 1 + \sum_{m=0}^{n-1} s(m) = 1 + n + \sum_{i=0}^{\infty} \sum_{m=0}^{n-1} s_i(m) = 1 + n + \sum_{i=0}^{\infty} p_i(n) = 1 + n + p(n).
 \end{aligned}$$

If $a < b-1$, for $t \in \llbracket 1, 4 \rrbracket$ we define the i -th generation of $T^{(t)}$ as $T_i^{(t)}$, and the numbers $b_i^{(t)}(n)$, $s_i^{(t)}(n)$, $p_i^{(t)}(n)$ and $p^{(t)}(n)$ exactly as for T . Now the strong bispecial smooth words belong to T , $T^{(1)}$ or $T^{(2)}$ and the weak ones belong to $T^{(3)}$ or $T^{(4)}$, so with Theorem 6.2.2 and by summation, we obtain

$$p_{C_f}^{\infty}(n) = 1 + n + p(n) + p^{(1)}(n) + p^{(2)}(n) - p^{(3)}(n) - p^{(4)}(n). \quad (8)$$

Now, we observe that $\varepsilon \sqsubset_p a^a$ so a quick induction on i with Remark 7.3.3 (ii) shows that every word $u \in T_i$ is the prefix of a word $f(u) \in T_i^{(1)}$ for all $i \geq 0$. By construction, the map $f : T \rightarrow T^{(1)}$ is an injection, and we have $|T_i| = |T_i^{(1)}| = 2^i$ for all $i \geq 0$ so f is a bijection and we have $T_i^{(1)} = f(T_i)$ for all $i \geq 0$. Then, by summation, for all $i \geq 0$ and all $n \geq 0$ we have

$$\begin{aligned}
 p_i^{(1)}(n) &= \sum_{u \in T_i^{(1)}} \max(0, n - |u| - 1) = \sum_{u \in T_i} \max(0, n - |f(u)| - 1) \\
 &\leq \sum_{u \in T_i} \max(0, n - |u| - 1) = p_i(n),
 \end{aligned}$$

which yields $p_i^{(1)}(n) \leq p_i(n)$ for all $n \geq 0$. We also have $\varepsilon \sqsubset_p b^a$, $a^a \sqsubset_p a^{b-1}$ and $b^a \sqsubset_p b^{b-1}$ so the same argument yields $p^{(2)}(n) \leq p(n)$, $p^{(3)}(n) \leq p^{(1)}(n)$ and $p^{(4)}(n) \leq p^{(2)}(n)$ for all $n \geq 0$. Finally, combining these inequalities with Equation (8) yields the result. \square

7.4. Bounds of the complexity

In this section we prove Theorems 7.1.19 and 7.1.20. Thanks to Proposition 7.3.10, it remains to study the asymptotics of $p(n)$ from the tree T . To do so, we define the minimal and maximal length of the words in each generation of T .

Definition 7.4.1. *If $\mathcal{A} = \{a, b\}$ is a binary alphabet, define $\ell_i := \min_{u \in T_i} |u|$ and $L_i := \max_{u \in T_i} |u|$.*

Recall that ℓ_i and L_i already appeared in Theorem 7.1.10, and that Weakley aimed to show that $L_{i-1} \leq \ell_i$ for all $i \geq 1$, which would help proving Conjecture 7.0.2. We get around this difficult task by considering the average length in each T_i and we obtain

the conjectured lower bound. And for the upper bounds we look for a lower bound of ℓ_i .

7.4.1. The lower bound over any alphabet

Let us prove the conjectured lower bound over any binary alphabet, this is Theorem 7.1.19 (i).

Proof of Theorem 7.1.19 (i). The first step is to compute the average length of each generation of T . For $i \geq 0$, set $f(i) := \sum_{m=\ell_i}^{L_i} mb_i(m) = \sum_{u \in T_i} |u|$ where ℓ_i and L_i are defined in Definition 7.4.1. We define the map

$$\begin{aligned} g : T \setminus \{\varepsilon\} &\longrightarrow T \setminus \{\varepsilon\} \\ u &\longmapsto \mathcal{P}_{f, \bar{u}_1}(\overline{\mathcal{D}_f(u)}) \end{aligned}$$

Also, for $i \geq 1$ we define the sets $T_{i,a} := \{u \in T_i \mid u_1 = a\}$ and $T_{i,b} := \{u \in T_i \mid u_1 = b\}$, and we are going to show that $T_{i,b} = g(T_{i,a})$. First, if $u \in T_{i,a}$, then $\mathcal{D}_f(u) \in T_{i-1}$ and $\overline{\mathcal{D}_f(u)} \in T_{i-1}$ so $g(u) = \mathcal{P}_{f,b}(\overline{\mathcal{D}_f(u)}) \in T_{i,b}$, therefore $g(T_{i,a}) \subseteq T_{i,b}$. Symmetrically, if $u \in T_{i,b}$ we have $g(u) \in T_{i,a}$, and

$$g(g(u)) = \mathcal{P}_{f,b}(\overline{\mathcal{D}_f(\mathcal{P}_{f,a}(\overline{\mathcal{D}_f(u)})})) = \mathcal{P}_{f,b}(\overline{\overline{\mathcal{D}_f(u)}}) = u,$$

therefore $T_{i,b} \subseteq g(T_{i,a})$. We deduce that $f(i) = \sum_{u \in T_{i,a}} [|u| + |g(u)|]$ for all $i \geq 1$. Now, for all $u \in T_{i,a}$, we have

$$\begin{aligned} |u| &= a |\mathcal{D}_f(u)|_a + b |\mathcal{D}_f(u)|_b + 2a, \\ |g(u)| &= a |\mathcal{D}_f(g(u))|_a + b |\mathcal{D}_f(g(u))|_b + 2a = a |\mathcal{D}_f(u)|_b + b |\mathcal{D}_f(u)|_a + 2a. \end{aligned}$$

Moreover, we have $\mathcal{D}_f(T_{i,a}) = T_{i-1}$ and $|T_{i,a}| = 2^{i-1}$ for all $i \geq 1$. Then, for all $i \geq 0$, we have

$$\begin{aligned} f(i+1) &= \sum_{u \in T_{i+1,a}} [|u| + |g(u)|] = \sum_{u \in T_{i+1,a}} [(a+b) |\mathcal{D}_f(u)| + 4a] \\ &= (a+b) \sum_{u \in T_{i+1,a}} |\mathcal{D}_f(u)| + \sum_{u \in T_{i+1,a}} 4a = (a+b) \sum_{u \in T_i} |u| + 4a2^i = (a+b)f(i) + 4a2^i. \end{aligned}$$

Then, with $f(0) = 0$ and by setting $c := \frac{4a}{a+b-2}$, a quick induction yields

$$f(i) = c(a+b)^i - c2^i \tag{9}$$

for all $i \geq 0$.

7. Complexity of smooth words – 7.4. Bounds of the complexity

The second step is to compute $p_i(n)$ for large enough n . For $i \geq 0$, if $n < \ell_i$ then $b_i(n) = s_i(n) = p_i(n) = 0$, and $s_i(\ell_i) = 0$. Also, if $n > L_i$ then $b_i(n) = 0$ and $s_i(n) = |T_i| = 2^i$. Then, for all $n > L_i$, we get

$$\begin{aligned} p_i(n) &= \sum_{m=0}^{\ell_i} s_i(m) + \sum_{m=\ell_i+1}^{L_i} s_i(m) + \sum_{m=L_i+1}^{n-1} s_i(m) = 0 + \sum_{m=\ell_i+1}^{L_i} \sum_{m'=\ell_i}^{m-1} b_i(m') + \sum_{m=L_i+1}^{n-1} 2^i \\ &= \sum_{m=\ell_i}^{L_i} (L_i - m)b_i(m) + (n - L_i - 1)2^i = L_i \sum_{m=\ell_i}^{L_i} b_i(m) - f(i) + (n - L_i - 1)2^i \\ &= L_i 2^i - c(a+b)^i + c2^i + (n - L_i - 1)2^i \text{ thanks to Equation (9),} \end{aligned}$$

which yields

$$p_i(n) = (n + c - 1)2^i - c(a+b)^i \quad (10)$$

for all $n > L_i$.

The last step is to deduce the lower bound of $p_{C_f^\infty}(n)$. If we fix $i \geq 0$, we notice that the function $p_i : \mathbb{N} \rightarrow \mathbb{N}$ is convex since its second finite difference is $b_i : \mathbb{N} \rightarrow \mathbb{N}$, and Equation (10) states that it is eventually affine, so we deduce that

$$p_i(n) \geq \max\left(0, (n + c - 1)2^i - c(a+b)^i\right)$$

for all $i \geq 0$ and all $n \geq 0$. Then, for $n \geq 1$, the set $I_n := \{i \geq 0 \mid (n + c - 1)2^i \geq c(a+b)^i\}$ satisfies $I_n = \llbracket 0, j \rrbracket$ where $j := \left\lfloor \frac{\log(m)}{\log\left(\frac{a+b}{2}\right)} \right\rfloor$ and $m := \frac{n+c-1}{c} \geq 1$, and we get

$$\begin{aligned} p(n) &= \sum_{i=0}^{\infty} p_i(n) \geq \sum_{i \in I_n} \left[(n + c - 1)2^i - c(a+b)^i \right] = \sum_{i=0}^j \left[cm2^i - c(a+b)^i \right] \\ &\geq c \left[m \left(2^{j+1} - 1 \right) - \frac{(a+b)^{j+1} - 1}{a+b-1} \right] = c \left[m2^{j+1} - \frac{(a+b)^{j+1}}{a+b-1} \right] - cm + \frac{c}{a+b-1} \\ &\geq ch(j) - n + 1 - \frac{4a}{a+b-1} \geq ch(j) - n - 1 \text{ because } a+b-1 \geq 2a \end{aligned}$$

where $h(x) = m2^{x+1} - \frac{(a+b)^{x+1}}{a+b-1}$. The function $h : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ is derivable and $h'(x) = \frac{m2^{x+1}}{\log(2)} - \frac{(a+b)^{x+1}}{(a+b-1)\log(a+b)}$. We deduce that h is increasing then decreasing, which implies that $h(j) \geq \min\left(h\left(\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)} - 1\right), h\left(\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)}\right)\right)$. Now we compute

$$h\left(\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)} - 1\right) = m2^{\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)}} - \frac{(a+b)^{\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)}}}{a+b-1} = \left[1 - \frac{1}{a+b-1}\right] m^\rho$$

7. Complexity of smooth words – 7.4. Bounds of the complexity

and

$$\begin{aligned} h\left(\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)}\right) &= m2^{\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)}+1} - \frac{(a+b)^{\frac{\log(m)}{\log\left(\frac{a+b}{2}\right)}+1}}{a+b-1} = 2m^{\frac{\log(2)}{\log\left(\frac{a+b}{2}\right)}+1} - \frac{(a+b)}{a+b-1}m^\rho \\ &= \left[2 - \frac{a+b}{a+b-1}\right]m^\rho = \left[1 - \frac{1}{a+b-1}\right]m^\rho. \end{aligned}$$

We deduce that $p(n) \geq c\left[1 - \frac{1}{a+b-1}\right]m^\rho - n - 1$ for all $n \geq 1$. If $c \geq 1$, then $m \geq \frac{n}{c}$ so Proposition 7.3.10 yields $p_{C_f^\infty}(n) \geq c^{1-\rho}\left[1 - \frac{1}{a+b-1}\right]n^\rho$ for all $n \geq 1$. If $c < 1$, then $m \geq n$ so Proposition 7.3.10 yields $p_{C_f^\infty}(n) \geq c\left[1 - \frac{1}{a+b-1}\right]n^\rho$ for all $n \geq 1$. \square

7.4.2. An upper bound from ℓ_i

We show here that a lower bound of ℓ_i provides an upper bound of $p_{C_f^\infty}(n)$.

Proposition 7.4.2. *Let $\mathcal{A} = \{a, b\}$ be a binary alphabet. If there exists $\lambda > 1$, $C > 0$ and $D \geq 0$ such that $\ell_i \geq C\lambda^i - D - 1$ for all $i \geq 0$, then we have*

$$p_{C_f^\infty}(n) = \mathcal{O}(n^\zeta) \quad \text{where } \zeta = \frac{\log(2\lambda)}{\log(\lambda)}.$$

Proof. With the notation introduced in Definition 7.3.9, we have $s_i(n) = 0$ if $n \leq \ell_i$ and $s_i(n) \leq |T_i| = 2^i$ otherwise, therefore $p_i(n) \leq \max(0, 2^i(n - \ell_i - 1))$ for all $i \geq 0$. Now, for $n \geq \max(0, C - D)$, the set $I_n := \{i \geq 0 \mid n - \ell_i - 1 \geq 0\}$ satisfies $I_n \subseteq \llbracket 0, j \rrbracket$ where $j := \left\lfloor \frac{\log(m)}{\log(\lambda)} \right\rfloor$ and $m := \frac{n+D}{C} \geq 1$, and we get

$$p(n) = \sum_{i=0}^{\infty} p_i(n) \leq \sum_{i \in I_n} (n - \ell_i - 1)2^i \leq \sum_{i=0}^j n2^i = n2^{j+1} - n.$$

Then, for all $n \geq D$, we have $j \leq \frac{\log(m)}{\log(\lambda)}$ and $m \leq \frac{2n}{C}$ so

$$p(n) \leq 2n2^{\frac{\log(m)}{\log(\lambda)}} - n = 2nm^{\frac{\log(2)}{\log(\lambda)}} - n \leq 2\left(\frac{2}{C}\right)^{\frac{\log(2)}{\log(\lambda)}}n^\zeta - n.$$

Finally, Proposition 7.3.10 yields $p_{C_f^\infty}(n) \leq 6\left(\frac{2}{C}\right)^{\frac{\log(2)}{\log(\lambda)}}n^\zeta - 2n + 1 \leq 6\left(\frac{2}{C}\right)^{\frac{\log(2)}{\log(\lambda)}}n^\zeta$ for all $n \geq \max(1, C - D, D)$. \square

Now it remains to find the best lower bound of ℓ_i .

7.4.3. The upper bound over even alphabets

Over even alphabets, ℓ_i is easy to compute and we deduce the conjectured upper bound of $p_{C_f^\infty}(n)$.

7. Complexity of smooth words – 7.4. Bounds of the complexity

Lemma 7.4.3. *Let $\mathcal{A} = \{a, b\}$ be an even alphabet and let $c := \frac{4a}{a+b-2}$. Then we have $\ell_i = c \left(\frac{a+b}{2}\right)^i - c$ for all $i \geq 0$.*

Proof. Because a and b are even, a quick induction shows that $|u|_a = |u|_b = \frac{|u|}{2}$ for all $u \in T$. We deduce that every word in T has even length, and we showed with Equation (9) that the average length of words in T_i is $c(a+b)^i - c2^i$ so we deduce that $\ell_i = L_i = c \left(\frac{a+b}{2}\right)^i - c$ for all $i \geq 0$. \square

Finally, combining Theorem 7.1.19 (i), Proposition 7.4.2 and Lemma 7.4.3 yields Theorem 7.1.19 (ii).

7.4.4. An upper bound over odd alphabets

Over odd alphabets, we compute the best lower bound of ℓ_i and deduce a new upper bound of $p_{C_f^\infty}(n)$.

Lemma 7.4.4. *Let $\mathcal{A} = \{a, b\}$ be an odd alphabet and let*

$$\lambda := \begin{cases} \frac{1+\sqrt{2b-1}}{2} & \text{if } a = 1, \\ \text{the dominant root of } X^3 - \frac{a+b}{2}X^2 + \frac{(b-a)^2}{4} & \text{otherwise.} \end{cases}$$

Then there exists $C > 0$ and $D \geq 0$ such that $\ell_i \geq C\lambda^i - D - 1$ for all $i \geq 0$.

Proof. The first step is to show that every word in T has even length. We proceed by induction on the generation of T : for $T_0 = \{\varepsilon\}$ this is trivial, now let $i \geq 0$ be such that every word of T_i has even length. If $u \in T_{i+1}$, by construction of T there exists $v \in T_i$ such that $u = \mathcal{P}_{u_1}(v)$, which yields $|u| = a|v|_a + b|v|_b + 2a$. By hypothesis, v has even length, so $|v|_a$ and $|v|_b$ are either both even or both odd, and in the two cases $|u|$ is even.

The second step is to show that $\ell_i = \left| \mathcal{P}_{f,a}^i(\varepsilon) \right|$ for all $i \geq 0$. To do so, we introduce the following notation: if $c \in \mathcal{A}$ and $u = u_1 \dots u_n \in \mathcal{A}^n$, then $|u|_{c,1}$ (resp. $|u|_{c,0}$) denotes the number of occurrences of the letter c at odd (resp. even) indexes in u . For $u \in \mathcal{A}^*$, we define

$$V(u) := \begin{pmatrix} |u|_{a,0} \\ |u|_{a,1} \\ |u|_{b,0} \\ |u|_{b,1} \end{pmatrix}, M := \begin{pmatrix} \frac{a-1}{2} & 0 & \frac{b-1}{2} & 0 \\ \frac{a+1}{2} & 0 & \frac{b+1}{2} & 0 \\ 0 & \frac{a+1}{2} & 0 & \frac{b+1}{2} \\ 0 & \frac{a-1}{2} & 0 & \frac{b-1}{2} \end{pmatrix}, N := \begin{pmatrix} \frac{a-1}{2} \\ \frac{a+1}{2} \\ \frac{a+1}{2} \\ \frac{a-1}{2} \end{pmatrix} \text{ and } P := \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix},$$

and we observe that, for all $u \in \mathcal{A}^*$ of even length,

$$V(\mathcal{P}_{f,a}(u)) = M V(u) + N, \tag{11}$$

$$V(\mathcal{P}_{f,b}(u)) = P V(\mathcal{P}_{f,a}(u)). \tag{12}$$

We also define $\ell'_i := \min_{u \in T_i} |u|_b$ and $\ell''_i := \min_{u \in T_i} |u|_{b,1}$ for all $i \geq 0$. Note that, since every word of T has even length and each T_i is mirror-invariant, we also have $\ell''_i = \min_{u \in T_i} |u|_{b,0}$.

7. Complexity of smooth words – 7.4. Bounds of the complexity

Now we show by induction on i that, for all $i \geq 0$, $\ell_i = |\mathcal{P}_{f,a}^i(\varepsilon)|$, $\ell'_i = |\mathcal{P}_{f,a}^i(\varepsilon)|_b$ and $\ell''_i = |\mathcal{P}_{f,a}^i(\varepsilon)|_{b,1}$. For $i = 0$ this is trivial, now let $i \geq 0$ be such that the three equalities hold. Let $u \in T_{i+1}$, and let $c \in \mathcal{A}$ and $v \in T_i$ be such that $u = \mathcal{P}_{f,c}(v)$. If $c = a$, by noticing that $|v|_{a,1} + |v|_{b,1} = \frac{|v|}{2}$, Equation (11) yields

$$\begin{aligned} |u|_{b,1} &= \frac{a-1}{2} \frac{|v|}{2} + \frac{b-a}{2} |v|_{b,1} + \frac{a-1}{2} \geq \frac{a-1}{2} \frac{\ell_i}{2} + \frac{b-a}{2} \ell''_i + \frac{a-1}{2} = |\mathcal{P}_{f,a}^{i+1}(\varepsilon)|_{b,1}, \\ |u|_b &= a \frac{|v|}{2} + (b-a) |v|_{b,1} + a \geq a \frac{\ell_i}{2} + (b-a) \ell''_i + a = |\mathcal{P}_{f,a}^{i+1}(\varepsilon)|_b, \\ |u| &= a|v| + (b-a) |v|_b + 2a \geq a \ell_i + (b-a) \ell'_i + 2a = |\mathcal{P}_{f,a}^{i+1}(\varepsilon)|. \end{aligned}$$

If $c = b$, then Equation (12) yields

$$\begin{aligned} |u|_{b,1} &= \frac{a+1}{2} \frac{|v|}{2} + \frac{b-a}{2} |v|_{b,0} + \frac{a+1}{2} \geq \frac{a-1}{2} \frac{\ell_i}{2} + \frac{b-a}{2} \ell''_i + \frac{a-1}{2} = |\mathcal{P}_{f,a}^{i+1}(\varepsilon)|_{b,1}, \\ |u|_b &= a \frac{|v|}{2} + (b-a) |v|_{b,0} + a \geq a \frac{\ell_i}{2} + (b-a) \ell''_i + a = |\mathcal{P}_{f,a}^{i+1}(\varepsilon)|_b, \\ |u| &= a|v| + (b-a) |v|_b + 2a \geq a \ell_i + (b-a) \ell'_i + 2a = |\mathcal{P}_{f,a}^{i+1}(\varepsilon)|. \end{aligned}$$

The last step is to deduce a lower bound of ℓ_i . First, Equation (11) yields

$$V(\mathcal{P}_{f,a}^{i+1}(u)) = M V(\mathcal{P}_{f,a}^i(u)) + N$$

for all $i \geq 0$, and a quick induction provides $V(\mathcal{P}_{f,a}^i(u)) = \sum_{j=0}^{i-1} M^j N$ for all $i \geq 0$. For all $i \geq 1$, we deduce that

$$\ell_i = |\mathcal{P}_{f,a}^i(\varepsilon)| = \|V(\mathcal{P}_{f,a}^i(u))\|_1 \geq \|M^{i-1} N\|_1 \quad (13)$$

If $a = 1$, we have $M = \left(\begin{array}{c|c} R & S \\ \mathbf{0} & \frac{b-1}{2} \end{array} \right)$ where $R := \begin{pmatrix} 0 & 0 & \frac{b-1}{2} \\ 1 & 0 & \frac{b+1}{2} \\ 0 & 1 & 0 \end{pmatrix}$. Then, for all $i \geq 0$

there exists $S_i \in \mathbb{R}^{1 \times 3}$ such that $M^i = \left(\begin{array}{c|c} R^i & S_i \\ \mathbf{0} & \left(\frac{b-1}{2}\right)^i \end{array} \right)$ and $M^i N = \left\| R^i \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\|_1$.

check that R^5 has positive entries so that R is primitive and Perron-Frobenius theorem states that every entry of R^i grows like $\Theta(\lambda^i)$ where $\lambda \in \mathbb{R}_+$ is the spectral radius of R . Moreover, the characteristic polynomial of R is $(X+1)\left(X^2 - X - \frac{b-1}{2}\right)$ so we deduce that $\lambda = \frac{1+\sqrt{2b-1}}{2}$. With Equation (13), this directly provides $C > 0$ and $D \geq 0$ such that $\ell_i \geq C\lambda^i - D - 1$ for all $i \geq 0$.

7. Complexity of smooth words – 7.4. Bounds of the complexity

If $a > 1$, one can check that M^2 has positive entries so that M is primitive and Perron-Frobenius theorem states that every entry of M^i grows like $\Theta(\lambda^i)$ where $\lambda \in \mathbb{R}_+$ is the spectral radius of M . Moreover, the characteristic polynomial of M is $(X + 1) \left(X^3 - \frac{a+b}{2} X^2 + \frac{(b-a)^2}{4} \right)$ so λ is the dominant root of $X^3 - \frac{a+b}{2} X^2 + \frac{(b-a)^2}{4}$. With Equation (13), this directly provides $C > 0$ and $D \geq 0$ such that $\ell_i \geq C\lambda^i - D - 1$ for all $i \geq 0$. \square

Finally, combining Proposition 7.4.2 and Lemma 7.4.4 yields Theorem 7.1.20.

Remark 7.4.5. *Studying ℓ_i over even alphabets led us to prove the conjectured upper bound of $p_{C_f^\infty}(n)$ thanks to Proposition 7.4.2, but this method appears to be insufficient over odd alphabets. Let us explain.*

Firstly, over even alphabets ℓ_i grows like $\left(\frac{a+b}{2}\right)^i$, but over odd alphabets it grows only like λ^i so Proposition 7.4.2 cannot provide a better upper bound than n^ζ .

Secondly, over even alphabets we have $\ell_i = L_i$ so we immediatly get $L_i \leq \ell_{i+1}$ for all $i \geq 0$ as suggested by Weakley in [Wea89]. However, this inequality fails over odd alphabets since L_i has a greater growth rate than ℓ_i : in the same way we proved that ℓ_i grows like λ^i , one can prove that L_i grows like r^i where r is the spectral radius of MPM where M and P are defined in the last proof. For example, over $\{1, 3\}$ we have $L_i > \ell_{i+1}$ for all $i \geq 5$, starting with $L_5 = 86 > \ell_6 = 64$.

8. Smooth sequences over even and odd alphabets

Plus tu y penses, plus t'es inconfortable
Plus t'es inconfortable, plus tu réfléchis.

Adib Alkhalidey

Content

| | |
|---|-----|
| 8.1. Preliminaries | 108 |
| 8.1.1. S-adic smooth sequences | 108 |
| 8.1.2. New results | 110 |
| 8.2. Occurrences in smooth sequences | 111 |
| 8.2.1. Occurrences in derivatives | 111 |
| 8.2.2. Almost-short words | 112 |
| 8.2.3. Complementation of factors | 113 |
| 8.2.4. Uniform recurrence | 114 |
| 8.3. Factor complexity over even alphabets | 114 |
| 8.3.1. Forbidden and mandatory factors | 114 |
| 8.3.2. Bispecial factors | 115 |
| 8.3.2.1. Short bispecial factors | 115 |
| 8.3.2.2. Long bispecial factors | 116 |
| 8.3.2.3. First-long bispecial factors | 117 |
| 8.3.3. Computation of the factor complexity | 119 |
| 8.3.4. Unique ergodicity | 120 |
| 8.4. Factor complexity over odd alphabets | 121 |
| 8.4.1. Forbidden and mandatory factors | 121 |
| 8.4.2. Bispecial factors | 123 |
| 8.4.2.1. Short bispecial factors | 123 |
| 8.4.2.2. Long bispecial factors | 123 |
| 8.4.2.3. First-long bispecial factors | 123 |
| 8.4.3. Computation of factor complexity | 124 |
| 8.4.4. Unique ergodicity | 125 |

In the framework of smooth sequences, we explained in Section 6.1.2 how the fixed points of the run-length encoding over any even or odd alphabet is morphic with a primitive substitution. We extrapolate this in Conjecture 6.1.19 where we state that, over an even or odd alphabet, smooth sequences have a substitutive structure. This was already used [BJP08] where the authors showed that all smooth sequences over even or odd alphabets are recurrent, characterized the lexicographic minimal and maximal smooth sequences and computed their letter frequencies. More recently, in [Jam+26] the authors use the S-adic behavior of smooth sequences over $\{1, 3\}$ to prove the existence of frequencies and characterize their letter frequencies.

In this chapter we continue to take advantage of the S-adic structure of smooth sequences over even and odd alphabets.

- In Theorem 8.1.5 we show that, over any even or odd alphabet, every smooth sequence is uniformly recurrent.
- In Theorem 8.1.6 we show that, over any even alphabet and any odd alphabet of the form $\{a, a + 2\}$, every smooth sequence has linear factor complexity.
- In Theorem 8.1.7 we show that, over some even alphabet and any odd alphabet of the form $\{a, a + 2\}$, every smooth sequence has uniform frequencies.

8.1. Preliminaries

8.1.1. S-adic smooth sequences

A dual way to look at smooth sequences is to consider the two canonical pseudo-reciprocals of the map \mathcal{A} , that inspired the notion of r-primitives in Definition 6.1.9.

Definition 8.1.1. *Over $\mathcal{A} = \{a, b\}$, the primitives are the maps*

$$\begin{aligned} \mathcal{P}_a : \mathcal{A}^{\mathbb{N}} &\longrightarrow \mathcal{C}^1 & \mathcal{P}_b : \mathcal{A}^{\mathbb{N}} &\longrightarrow \mathcal{C}^1 \\ x &\longmapsto a^{x_0} b^{x_1} a^{x_2} \dots & x &\longmapsto b^{x_0} a^{x_1} b^{x_2} \dots \end{aligned}$$

We immediatly observe that $x = \mathcal{P}_{x_0}(\mathcal{D}(x))$ for all $x \in \mathcal{C}^1$, and that $x = \mathcal{D}(\mathcal{P}_a(x)) = \mathcal{D}(\mathcal{P}_b(x))$ for $x \in \mathcal{A}^{\mathbb{N}}$, which yields $\mathcal{C}^\infty = \mathcal{P}_a(\mathcal{C}^\infty) \cup \mathcal{P}_b(\mathcal{C}^\infty)$.

Inspired by the substitutive construction of Dekking illustrated in Proposition 6.1.17, we are going to show that, over even or odd alphabets, the set \mathcal{C}^∞ is conjugated to an S-adic system with two substitutions. Given a binary alphabet $\mathcal{A} = \{a, b\}$, we consider the alphabet $\hat{\mathcal{A}} = \{a, a', b, b'\}$ and the space $X := \{x \in \hat{\mathcal{A}}^{\mathbb{N}} \mid x_{2i} \in \{a, b\}, x_{2i+1} \in \{a', b'\}\}$. Then the coding

$$\tau : a \mapsto a, a' \mapsto a, b \mapsto b, b' \mapsto b$$

induces a map $\tau : X \rightarrow \mathcal{A}^{\mathbb{N}}$ that is bijective with reciprocal

$$\begin{aligned} \tau^{-1} : \mathcal{A}^{\mathbb{N}} &\longrightarrow X \\ x &\longmapsto x_0(x_1)'x_2(x_3)' \dots \end{aligned}$$

8. Smooth sequences over even and odd alphabets – 8.1. Preliminaries

Now the action of \mathcal{P}_a and \mathcal{P}_b can be replicated by one substitution each on the alphabet $\hat{\mathcal{A}}$.

Proposition 8.1.2. (i) Given an even alphabet $\mathcal{A} = \{a, b\}$, we define the substitutions

$$\begin{aligned}\sigma_a : a &\mapsto (aa')^{\frac{a}{2}}, a' \mapsto (bb')^{\frac{a}{2}}, b \mapsto (aa')^{\frac{b}{2}}, b' \mapsto (bb')^{\frac{b}{2}}, \\ \sigma_b : a &\mapsto (bb')^{\frac{a}{2}}, a' \mapsto (aa')^{\frac{a}{2}}, b \mapsto (bb')^{\frac{b}{2}}, b' \mapsto (aa')^{\frac{b}{2}}.\end{aligned}$$

(ii) Given an odd alphabet $\mathcal{A} = \{a, b\}$, we define the substitutions

$$\begin{aligned}\sigma_a : a &\mapsto (aa')^{\frac{a-1}{2}} a, a' \mapsto (b'b)^{\frac{a-1}{2}} b', b \mapsto (aa')^{\frac{b-1}{2}} a, b' \mapsto (b'b)^{\frac{b-1}{2}} b', \\ \sigma_b : a &\mapsto (bb')^{\frac{a-1}{2}} b, a' \mapsto (a'a)^{\frac{a-1}{2}} a', b \mapsto (bb')^{\frac{b-1}{2}} b, b' \mapsto (a'a)^{\frac{b-1}{2}} a' .\end{aligned}$$

Then, in both cases we have $\mathcal{P}_c(x) = \tau(\sigma_c(\tau^{-1}(x)))$ for all $x \in \mathcal{A}^{\mathbb{N}}$ and all $c \in \mathcal{A}$.

Proof. (i) If $\mathcal{A} = \{a, b\}$ is even, we observe that $\sigma_a(c) = (aa')^{\frac{c}{2}}$ and $\sigma_a(c') = (bb')^{\frac{c}{2}}$ for all $c \in \mathcal{A}$. Now, if $x \in \mathcal{A}^{\mathbb{N}}$, we have $\tau^{-1}(x) = x_0(x_1)'x_2(x_3)'\dots$ so

$$\sigma_a(\tau^{-1}(x)) = (aa')^{\frac{x_0}{2}} (bb')^{\frac{x_1}{2}} (aa')^{\frac{x_2}{2}} (bb')^{\frac{x_3}{2}} \dots$$

and

$$\tau(\sigma_a(\tau^{-1}(x))) = a^{x_0} b^{x_1} a^{x_2} b^{x_3} \dots = \mathcal{P}_a(x).$$

In the same way we obtain $\tau(\sigma_a(\tau^{-1}(x))) = \mathcal{P}_b(x)$.

(ii) If $\mathcal{A} = \{a, b\}$ is odd, we observe that $\sigma_a(c) = (aa')^{\frac{c-1}{2}} a$ and $\sigma_a(c') = (b'b)^{\frac{c-1}{2}} b'$ for all $c \in \mathcal{A}$. Now, if $x \in \mathcal{A}^{\mathbb{N}}$, we have $\tau^{-1}(x) = x_0(x_1)'x_2(x_3)'\dots$ so

$$\sigma_a(\tau^{-1}(x)) = (aa')^{\frac{x_0-1}{2}} a (b'b)^{\frac{x_1-1}{2}} b' (aa')^{\frac{x_2-1}{2}} a (b'b)^{\frac{x_3-1}{2}} b' \dots$$

and

$$\tau(\sigma_a(\tau^{-1}(x))) = a^{x_0} b^{x_1} a^{x_2} b^{x_3} \dots = \mathcal{P}_a(x).$$

In the same way we obtain $\tau(\sigma_b(\tau^{-1}(x))) = \mathcal{P}_b(x)$. □

We easily deduce that the words with a periodic trace are morphic.

Proposition 8.1.3. Let $\mathcal{A} = \{a, b\}$ be even or odd and let $u \in \mathcal{A}^n$ where $n \geq 1$.

(i) If $a \neq 1$ or $b \sqsubset u$, then the smooth sequence $\mathcal{E}(u^\omega)$ is the morphic word $\tau(\sigma^\infty(u_1))$ where $\sigma = \sigma_{u_1} \circ \sigma_{u_2} \circ \dots \circ \sigma_{u_n}$.

(ii) We have $\mathcal{E}(1^\omega) = 1\tau(\sigma_b^\infty(b))$.

Proof. Let $x = \mathcal{E}(u^\omega)$. With Proposition 6.1.12 we have $\mathcal{D}^n(x) = x$, and with primitives it becomes $x = \mathcal{P}_{u_1} \circ \mathcal{P}_{u_2} \circ \dots \circ \mathcal{P}_{u_n}(x)$. Then Proposition 8.1.2 yields $x = \tau(\sigma(\tau^{-1}(x)))$ where $\sigma := \sigma_{u_1} \circ \sigma_{u_2} \circ \dots \circ \sigma_{u_n}$. We deduce that $\tau^{-1}(x)$ is a fixed point of the substitution σ and that its first term is $u_1 \in \hat{\mathcal{A}}$.

(i) If $a \neq 1$ or $b \sqsubset u$, we observe that the letter u_1 is prolongeable by σ , so $\tau^{-1}(x)$ is the purely morphic word $\sigma^\infty(u_1)$.

8. Smooth sequences over even and odd alphabets – 8.1. Preliminaries

(ii) If $a = 1$ and $u = 1^n$, the letter u_1 is not prolongeable by σ but we have $x = \mathcal{E}(1^\omega) = 1\mathcal{E}(b^\omega)$ and (i) provides $\mathcal{E}(b^\omega) = \tau(\sigma_b^\infty(b))$. \square

We then derive important properties of these morphic words, which generalizes Propositions 6.1.17 and 6.1.18.

Corollary 8.1.4. *Over an even or odd alphabet \mathcal{A} , if $x \in \mathcal{C}^\infty$ and $\mathcal{T}(x)$ is eventually periodic, then x is linearly recurrent and has linear factor complexity.*

Proof. Let us first show it when $\mathcal{T}(x)$ is periodic. If $x \neq \mathcal{E}(1^\omega)$, Proposition 8.1.3 (i) states that x is a morphic word $\tau(\sigma^\infty(c))$, and we easily observe that σ is ℓ -primitive and tame (cf. Section 5.1.2). Then Theorem 5.1.10 states that the subshift X_σ is minimal and Corollary 5.1.12 states that X_σ is linearly recurrent, so a fortiori the word $\sigma^\infty(c)$ is linearly recurrent. Finally, x remains linearly recurrent by applying the coding τ .

Now, if $\mathcal{T}(x)$ is eventually periodic, this provides $n \geq 0$, $u \in \mathcal{A}^n$ and $v \in \mathcal{A}^+$ such that $\mathcal{T}(x) = uv^\omega$, so Proposition 8.1.2 yields $\tau^{-1}(x) = \sigma_{u_1} \circ \dots \circ \sigma_{u_n}(\tau^{-1}(\mathcal{E}(v^\omega)))$. Finally, we already showed that $\tau^{-1}(\mathcal{E}(v^\omega))$ is linearly recurrent and the substitutions preserve the linear recurrence of $\tau^{-1}(x)$.

In particular, [DHS99, Theorem 24 i)] states x has at most linear factor complexity, but Proposition 6.1.6 ensures that x is aperiodic so Theorem 2.2.5 states that it has at least linear factor complexity. \square

This contrasts with Remark 6.1.20 where we explained that, over mixed alphabets, no smooth sequence is expected to be morphic.

8.1.2. New results

In the preparatory Section 8.2, we prove uniform recurrence for all smooth sequences over even and odd alphabets.

Theorem 8.1.5. *Over even and odd alphabets, every smooth sequence is uniformly recurrent.*

We then show that, over all even alphabets and over odd alphabets of the form $\{a, a + 2\}$, every smooth sequence has linear factor complexity.

Theorem 8.1.6. (i) *Over an even alphabet, every smooth sequence x has the same factor complexity and satisfies $p_x(n) = \Theta(n)$.*

(ii) *Over an odd alphabet of the form $\{a, a + 2\}$, every smooth sequence x satisfies $p_x(n) = \Theta(n)$.*

In the light of Theorem 7.1.18, where we proved that $p_{\mathcal{C}_f^\infty}(n)$ grows at least like n^ρ where $\rho > 1$, we deduce that these smooth sequences are far from containing all smooth words. We also obtain unique ergodicity by applying a criterion of [Bos84].

8. Smooth sequences over even and odd alphabets – 8.2. Occurrences in smooth sequences

Theorem 8.1.7. (i) Over an even alphabet $\mathcal{A} = \{a, b\}$ such that $\frac{a+b}{2}a(a+2) > b(b-2)$, every smooth sequence x has uniform frequencies.

(ii) Over an odd alphabet $\mathcal{A} = \{a, a+2\}$, every smooth sequence x has uniform frequencies.

Example 8.1.8. Theorem 8.1.7 (i) applies to all even alphabets of the form $\{a, a+2\}$. It also applies to $\{2, 4\}$, $\{2, 6\}$, $\{4, 6\}$, $\{4, 8\}$ and $\{6, 8\}$, but not to $\{2, 8\}$.

8.2. Occurrences in smooth sequences

This section provides mostly technical lemmas that are crucial in understanding the combinatorial properties of the factors of smooth sequences, which lead in particular to proving uniform recurrence. We first show how occurrences in a smooth sequence are connected to occurrences in its derivative or in its primitives, then we introduce the useful *almost-short* words, we study how the factors of smooth sequences behave under complementation and finally we prove Theorem 8.1.5.

8.2.1. Occurrences in derivatives

We first introduce key notations.

Definition 8.2.1. If $x \in \mathcal{A}^{\mathbb{N}}$ and $u \in \mathcal{A}^n$, recall that u occurs with bounded gaps in x if it occurs infinitely often and the gap between two consecutive occurrences is bounded, and in that case we write $u < x$.

Also, a word $u \in \mathcal{A}^n$ occurs at an even (resp. odd) position in x if there exists an even (resp. odd) index $i \geq 0$ such that $x_{[i, i+n]} = u$, and we say that u occurs at both parities in x if it occurs at both at an even and an odd position in x .

Moreover, we say that u occurs with bounded gaps at both parities in x if it has infinitely many occurrences at even and odd positions in x and the gap between two consecutive occurrences at the same parity is bounded. In that case we write $u \ll x$.

Also, recall that the complementation \bar{u} of a word $u \in \mathcal{A}^*$ swaps the two letters. The next lemma describes how occurrences in a smooth sequence are connected to occurrences in its derivative.

Lemma 8.2.2. Let $\mathcal{A} = \{a, b\}$ be a binary alphabet, let $x \in \mathcal{C}^\infty$ and let $u \in \mathcal{C}_f^\infty$.

- (i) If $u \sqsubset x$ and $\bar{u} \sqsubset x$, then $\mathcal{D}_f(u)$ occurs at both parities in $\mathcal{D}(x)$.
- (ii) If $\mathcal{D}_f(u)$ occurs in $\mathcal{D}(x)$ not as a prefix, then either u or \bar{u} occurs in x .
- (iii) If $\mathcal{D}_f(u)$ occurs at both parities in $\mathcal{D}(x)$ not as a prefix, then $u \sqsubset x$ and $\bar{u} \sqsubset x$.
- (iv) If $\mathcal{D}_f(u) < \mathcal{D}(x)$, then $u < x$ or $\bar{u} < x$.
- (v) If $\mathcal{D}_f(u) \ll \mathcal{D}(x)$, then $u < x$ and $\bar{u} < x$.

Proof. (i) If $u = \varepsilon$ this is trivial, otherwise let $u = a_1^{p_1} \dots a_n^{p_n}$ with $n \geq 1$. If $p_1 \leq a$, set v (resp. v') as the element of $\{u, \bar{u}\}$ that starts with x_0 (resp. with \bar{x}_0). If $v \sqsubset x$, there

8. Smooth sequences over even and odd alphabets – 8.2. Occurrences in smooth sequences

exists $i \geq 0$ such that $x = x_0^{y_0} \overline{x_0}^{y_1} \dots x_0^{y_{2i}} \overline{x_0}^{p_2} x_0^{p_3} \dots$ so

$$\mathcal{D}(x) = y_0 y_1 \dots y_{2i} p_2 p_3 \dots = y_0 y_1 \dots y_{2i} \mathcal{D}_f(u) \dots,$$

which means that $\mathcal{D}_f(u)$ occurs at an odd position in $\mathcal{D}(x)$. Similarly, if $v' \sqsubset x$, there exists $i \geq 0$ such that $x = x_0^{y_0} \overline{x_0}^{y_1} \dots \overline{x_0}^{y_{2i+1}} x_0^{p_2} \overline{x_0}^{p_3} \dots$ so

$$\mathcal{D}(x) = y_0 y_1 \dots y_{2i+1} p_2 p_3 \dots = y_0 y_1 \dots y_{2i+1} \mathcal{D}_f(u) \dots,$$

which means that $\mathcal{D}_f(u)$ occurs at an even position in $\mathcal{D}(x)$. If $p_1 > a$, similar arguments yield the same result.

(ii)(iii) If $u = \varepsilon$ this is trivial. Otherwise let $u = a_1^{p_1} \dots a_n^{p_n}$ with $n \geq 1$ and suppose that $p_1 \leq a$ and $p_n \leq a$. With the fact that $x = x_0^{y_0} \overline{x_0}^{y_1} x_0^{y_2} \overline{x_0}^{y_3} \dots$ where $y := \mathcal{D}(x)$, we deduce that each non-prefix occurrence of $\mathcal{D}_f(u) = p_2 \dots p_{n-1}$ in y at an even position gives rise to a different non-prefix occurrence in x of the word $v := b_2^{p_2} \dots b_{n-1}^{p_{n-1}}$ where $b_2 = x_0$. Furthermore, each such occurrence of v is included in a different occurrence of the word $w := b_1^{p_1} b_2^{p_2} \dots b_{n-1}^{p_{n-1}} b_n^{p_n}$ where $b_1 = \overline{x_0}$ and $b_n = \overline{b_{n-1}}$. Similarly, each occurrence of $\mathcal{D}_f(u)$ at an odd position gives rise to a different occurrence of the word \overline{v} in x , that is included in a different occurrence of the word \overline{w} in x . Finally, observing that $\{w, \overline{w}\} = \{u, \overline{u}\}$ yields the result, and we treat the three other cases ($p_1 \leq a$ and $p_n > a$; $p_1 > a$ and $p_n \leq a$; $p_1 > a$ and $p_n > a$) with the same arguments.

(iv)(v) Following the proof of (ii) and (iii), it suffices to observe that the gap between two occurrences of w (resp. \overline{w}) in x is at most b times the gap between the two associated non-prefix occurrences of $\mathcal{D}_f(u)$ in $\mathcal{D}(x)$. \square

8.2.2. Almost-short words

Definition 8.2.3. A word $u \in \mathcal{A}^*$ is almost-short if it is of the form c^n where $c \in \mathcal{A}$ and $n \in \llbracket 0, b-1 \rrbracket$, otherwise it is almost-long. In particular, every almost-short word is short.

We show that almost-short words occur very regularly in smooth sequences.

Lemma 8.2.4. Let $\mathcal{A} = \{a, b\}$ be an even or odd alphabet, and let $x \in \mathcal{C}^\infty$. If $a \neq 1$ or $\mathcal{D}(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, and $u \in \mathcal{A}^*$ is almost-short, then $u \ll x$.

Proof. Let $u = c^n$ with $c \in \mathcal{A}$ and $n \in \llbracket 0, b-1 \rrbracket$.

If $a \neq 1$, we have $\mathcal{D}(x) \in \mathcal{C}^1$ so necessarily $b < \mathcal{D}(x)$, and every occurrence of b is preceded or followed by another occurrence of b because $a \geq 2$, so $b \ll \mathcal{D}(x)$. Then Lemma 8.2.2 (v) ensures that $c^b < x$. In particular, because c^n occurs with both parities in c^b , we get that $c^n \ll x$.

If $a = 1$ and $\mathcal{D}(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, we will use the substitutive formalism introduced in Section 8.1.1. Let us first observe that the condition on $\mathcal{D}(x)$ means that either $11 \sqsubset \mathcal{T}(\mathcal{D}(x))$ or $bb \sqsubset \mathcal{T}(\mathcal{D}(x))$. If $11 \sqsubset \mathcal{T}(\mathcal{D}(x))$, let us write $\mathcal{T}(\mathcal{D}(x)) = u11cy$ with $u \in \mathcal{A}^n$, $n \geq 0$, $c \in \mathcal{A}$ and $y \in \mathcal{A}^{\mathbb{N}}$. In particular, we have

$$\mathcal{D}(x) = \mathcal{P}_{u_1} \circ \dots \circ \mathcal{P}_{u_n} \circ \mathcal{P}_1^2 \circ \mathcal{P}_c(\mathcal{E}(y))$$

8. Smooth sequences over even and odd alphabets – 8.2. Occurrences in smooth sequences

and Proposition 8.1.2 yields

$$\tau^{-1}(\mathcal{D}(x)) = \sigma_{u_1} \circ \dots \circ \sigma_{u_n} \circ \sigma_1^2 \circ \sigma_c(\tau^{-1}(\mathcal{E}(y))).$$

In particular, $\mathcal{E}(y) \in \mathcal{C}^\infty$ so $b < \tau^{-1}(\mathcal{E}(y))$ or $b' < \tau^{-1}(\mathcal{E}(y))$. By observing that $1' \sqsubset \sigma_c(b)$ and $b' \sqsubset \sigma_c(b')$ if $c = 1$, or $b' \sqsubset \sigma_c(b)$ and $1' \sqsubset \sigma_c(b')$ if $c = b$, we deduce that $1' < \sigma_c(\tau^{-1}(\mathcal{E}(y)))$ or $b' < \sigma_c(\tau^{-1}(\mathcal{E}(y)))$. Then, by observing that $b, b' \sqsubset \sigma_1^2(1')$ and $b, b' \sqsubset \sigma_1^2(b')$, we deduce that $b < \sigma_1^2 \circ \sigma_c(\tau^{-1}(\mathcal{E}(y)))$ and $b' < \sigma_1^2 \circ \sigma_c(\tau^{-1}(\mathcal{E}(y)))$. By observing that $b, b' \sqsubset \sigma_1(b')$ and $b, b' \sqsubset \sigma_b(b)$, we deduce that $b < \tau^{-1}(\mathcal{D}(x))$ and $b' < \tau^{-1}(\mathcal{D}(x))$, which implies that $a^b < x$ and $b^b < x$. Finally, because c^n occurs with both parities in c^b , we deduce that $c^n \ll x$. If $bb \sqsubset \mathcal{T}(\mathcal{D}(x))$, similar arguments yield the result. \square

Then, combining Lemma 8.2.4 and Lemma 8.2.2 (v) directly yields the following result.

Corollary 8.2.5. *Let $\mathcal{A} = \{a, b\}$ be an even or odd alphabet and let $x \in \mathcal{C}^\infty$. If $a \neq 1$ or $\mathcal{D}^2(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$ and $u \in \mathcal{C}_f^\infty$ is such that $\mathcal{D}_f(u)$ is almost-short, then $u < x$.*

Remark 8.2.6. *The four pathological words that satisfy $\mathcal{D}(x) \in \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$ are $\mathcal{E}((1b)^\omega)$, $\mathcal{E}((b1)^\omega)$, $\mathcal{E}(1(1b)^\omega)$ and $\mathcal{E}(b(b1)^\omega)$. They must be treated separately because the first two do not contain the word bb (cf. Example 7.1.7) and the other two do not contain the word 11 .*

8.2.3. Complementation of factors

This is where we start to see why smooth sequences over even and odd alphabets have much less factors than \mathcal{C}_f^∞ .

Lemma 8.2.7. *Let $x \in \mathcal{C}^\infty$ over an alphabet $\mathcal{A} = \{a, b\}$.*

(i) *If \mathcal{A} is even, then the words $x_0\bar{x}_0$ and \bar{x}_0x_0 occur only at odd positions in x .*

(ii) *If \mathcal{A} is odd, then the word $x_0\bar{x}_0$ occurs only at even positions and the word \bar{x}_0x_0 occurs only at odd positions in x .*

Proof. It suffices to consider the canonical factorization $x = x_0^{p_0}\bar{x}_0^{p_1}x_0^{p_2}\bar{x}_0^{p_3}\dots$ and to observe that $p_i + p_{i+1}$ is always even. \square

We deduce that the factors of smooth sequences over even and odd alphabets behave badly under complementation.

Proposition 8.2.8. *Let $x \in \mathcal{C}^\infty$ over an even or odd alphabet $\mathcal{A} = \{a, b\}$. If $u \in \mathcal{A}^*$ is such that $\mathcal{D}_f(u)$ is almost-long, then $u \not\sqsubset x$ or $\bar{u} \not\sqsubset x$.*

Proof. Let $u \in \mathcal{A}^*$ be such that $\mathcal{D}_f(u)$ is almost-long and suppose that $u \sqsubset x$ and $\bar{u} \sqsubset x$. Then Lemma 8.2.2 (i) ensures that $\mathcal{D}_f(u)$ occurs at both parities in $\mathcal{D}(x)$. If $\mathcal{D}_f(u) = a^b$, any occurrence of a^b is followed by the letter b so the word a^bb occurs at both parities in $\mathcal{D}(x)$, which contradicts Lemma 8.2.7 (i). If $\mathcal{D}_f(u) = b^b$, the same argument leads to the same contradiction. If $\|\mathcal{D}_f(u)\| \geq 2$, we have $ab \sqsubset \mathcal{D}_f(u)$ or $ba \sqsubset \mathcal{D}_f(u)$ so ab or ba occurs at both parities in $\mathcal{D}(x)$, which contradicts Lemma 8.2.7. \square

8.2.4. Uniform recurrence

We now have all the tools to prove Theorem 8.1.5.

Proof of Theorem 8.1.5. Let $\mathcal{A} = \{a, b\}$ be an even or odd alphabet. We are going to show that, for every $n \geq 0$ and all $x \in \mathcal{C}^\infty$, every word $u \in \mathcal{L}(x) \cap \mathcal{A}^n$ satisfies $u < x$. We proceed by induction on n : if $n = 0$ this is trivial. Now let $n \geq 0$ be such that, for all $x \in \mathcal{C}^\infty$ and all $m \leq n$, every word of $\mathcal{L}(x) \cap \mathcal{A}^m$ satisfies $u < x$. Let $x \in \mathcal{C}^\infty$ and $u \in \mathcal{L}(x) \cap \mathcal{A}^{n+1}$.

If $\mathcal{D}_f(u)$ is almost-short, and $a \neq 1$ or $\mathcal{D}^2(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, Corollary 8.2.5 states that $u < x$.

If $\mathcal{D}_f(u)$ is almost-short, $a = 1$ and $\mathcal{D}^2(x) \in \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, Proposition 6.1.12 yields $\mathcal{S}^2(\mathcal{T}(x)) \in \{(1b)^\omega, (b1)^\omega\}$, in particular $\mathcal{T}(x)$ is eventually periodic so Corollary 8.1.4 states that x is linearly recurrent and a fortiori $u < x$.

If $\mathcal{D}_f(u)$ is almost-long, we have $\mathcal{D}_f(u) \in \mathcal{L}(\mathcal{D}(x))$ and $|\mathcal{D}_f(u)| < |u| = n + 1$ so, by hypothesis, $\mathcal{D}_f(u) < \mathcal{D}(x)$. Now Lemma 8.2.2 (iv) ensures that $u < x$ or $\bar{u} < x$, but Proposition 8.2.8 ensures that $\bar{u} \not\sqsubset x$. \square

8.3. Factor complexity over even alphabets

In this section we prove Theorem 8.1.6 (i) and Theorem 8.1.7 (i). We begin by exhibiting some forbidden and mandatory factors of smooth sequences, which greatly helps describing the bispecial factors of smooth sequences. We then deduce that all smooth sequences have the same factor complexity, which yields linear factor complexity. Finally we compute their factor complexity more precisely and obtain unique ergodicity when the complexity is sufficiently low.

8.3.1. Forbidden and mandatory factors

In addition to Corollary 8.2.5, we give another lemma that will greatly help finding if a finite word occurs in a smooth sequence or not.

Lemma 8.3.1. *If $x \in \mathcal{C}^\infty$ over an even alphabet $\mathcal{A} = \{a, b\}$, we define the forbidden words*

$$\begin{aligned} f_1 &:= x_0^{a+1} \overline{x_0^a} x_0, & f_2 &:= \overline{x_0} x_0^a \overline{x_0^{a+1}}, \\ f_3 &:= x_0 \left[\overline{x_0^a} x_0^a \right]^{\frac{b}{2}} \overline{x_0}, & f_4 &:= \overline{x_0^{a+1}} \left[x_0^b \overline{x_0^b} \right]^{\frac{b-2}{2}} x_0^{a+1}. \end{aligned}$$

Now, if $u \in \mathcal{A}^$ and there exists $i \in \{1, 2, 3, 4\}$ such that $f_i \sqsubset u$, then $u \not\sqsubset x$.*

Proof. It suffices to show that $f_i \not\sqsubset x$ for all $i \in \{1, 2, 3, 4\}$.

If $f_1 \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0^{p_1}} \dots \overline{x_0^{p_{2i-1}}} x_0^b \overline{x_0^a} x_0^{p_{2i+2}} \dots$ and $\mathcal{D}(x) = p_0 p_1 \dots p_{2i-1} b a \dots$ which contradicts Lemma 8.2.7 (i).

If $f_2 \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0^{p_1}} \dots \overline{x_0^{p_{2i-1}}} x_0^a \overline{x_0^b} x_0^{p_{2i+2}} \dots$ and $\mathcal{D}(x) = p_0 p_1 \dots p_{2i-1} a b \dots$ which contradicts Lemma 8.2.7 (i).

8. Smooth sequences over even and odd alphabets – 8.3. Factor complexity over even alphabets

If $f_3 \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0}^{p_1} \dots x_0^{p_{2i}} \left[\overline{x_0}^a x_0^a \right]^{\frac{b}{2}} \overline{x_0}^{p_{2i+b+1}} \dots$. Then, if $p_{2i} = b$ this contradicts Lemma 8.2.7 (i), and if $p_{2i} = a$ we have $\mathcal{D}(x) = p_0 p_1 \dots p_{2i-1} a^{b+1} p_{2i+b+1} \dots \notin \mathcal{C}^1$.

If $f_4 \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0}^{p_1} \dots x_0^{p_{2i}} \left[\overline{x_0}^b x_0^b \right]^{\frac{a}{2}} \overline{x_0}^{p_{2i+b+1}} \dots$. Then, if $p_{2i} = a$ this contradicts Lemma 8.2.7 (i), and if $p_{2i} = b$ we have $\mathcal{D}(x) = p_0 p_1 \dots p_{2i-1} b^{a+1} p_{2i+b+1} \dots \notin \mathcal{C}^1$. \square

Lemma 8.3.2. *If $x \in \mathcal{C}^\infty$ over an even alphabet \mathcal{A} , we define the mandatory words*

$$\begin{aligned} m_1 &:= \overline{x_0}^{a+1} \left[x_0^a \overline{x_0}^a \right]^{\frac{b}{2}} x_0^{a+1}, & m_2 &:= \overline{x_0}^{a+1} \left[x_0^a \overline{x_0}^a \right]^{\frac{a}{2}} x_0^{a+1}, \\ m_3 &:= x_0 \overline{x_0}^a \left[x_0^b \overline{x_0}^b \right]^{\frac{b}{2}} x_0^a \overline{x_0}, & m_4 &:= x_0 \overline{x_0}^a \left[x_0^b \overline{x_0}^b \right]^{\frac{a}{2}} x_0^a \overline{x_0}. \end{aligned}$$

Now, if $u \in \mathcal{A}^*$ and there exists $i \in \{1, 2, 3, 4\}$ such that $u \sqsubset m_i$, then $u \sqsubset x$.

Proof. It suffices to show that $m_i \sqsubset x$ for all $i \in \{1, 2, 3, 4\}$.

$\mathcal{D}_f(ba^b b) = b$ is almost-short so Corollary 8.2.5 states that $ba^b b \sqsubset \mathcal{D}(x)$. Moreover, Theorem 8.1.5 implies that $\mathcal{D}_f(m_1) = ba^b b$ occurs not as a prefix in $\mathcal{D}(x)$ so Lemma 8.2.2 (ii) states that $m_1 \sqsubset x$ or $\overline{m_1} \sqsubset x$, but we notice that $f_1 \sqsubset_p \overline{m_1}$ so Lemma 8.3.1 states that $\overline{m_1} \not\sqsubset x$.

Similarly, we have $\mathcal{D}_f(ba^a b) = a$, $\mathcal{D}_f(m_2) = ba^a b$ and $f_1 \sqsubset_p \overline{m_2}$ so $m_2 \sqsubset x$.

Similarly, we have $\mathcal{D}_f(ab^b a) = b$, $\mathcal{D}_f(m_3) = ab^b a$ and $f_2 \sqsubset_p \overline{m_3}$ so $m_3 \sqsubset x$.

Similarly, we have $\mathcal{D}_f(ab^a a) = a$, $\mathcal{D}_f(m_4) = ab^a a$ and $f_2 \sqsubset_p \overline{m_4}$ so $m_4 \sqsubset x$. \square

8.3.2. Bispecial factors

We first describe the short bispecial factors over even alphabets. Next we show that long bispecial factors can be traced back to *first-long* bispecial factors, and we conclude by describing the first-long bispecial factors.

8.3.2.1. Short bispecial factors

Lemma 8.3.3. *Let $x \in \mathcal{C}^\infty$ over an even alphabet $\mathcal{A} = \{a, b\}$. Then the short strong bispecial factors of x are ε , a^a and b^a ; and the short weak bispecial factors of x are a^{b-1} and b^{b-1} .*

Proof. Let u be a short word over \mathcal{A} , i.e., $u = c^n$ with $c \in \mathcal{A}$ and $n \in \llbracket 0, b \rrbracket$. If $u \in \mathcal{A}^*$ has an almost-short finite-derivative, abbreviated a.s.d., then Corollary 8.2.5 ensures that $u \sqsubset x$.

8. Smooth sequences over even and odd alphabets – 8.3. Factor complexity over even alphabets

| | $cc^n c$ | $cc^n \bar{c}$ | $\bar{c}c^n c$ | $\bar{c}c^n \bar{c}$ | type of c^n |
|--|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---------------|
| $n = 0$ | $\sqsubset x$ | $\sqsubset x$ | $\sqsubset x$ | $\sqsubset x$ | strong |
| $n \in \llbracket 1, a-1 \rrbracket$ | a.s.d. | a.s.d. | a.s.d. | $\notin \mathcal{C}_f^\infty$ | neutral |
| $n = a$ | a.s.d. | a.s.d. | a.s.d. | a.s.d. | strong |
| $n \in \llbracket a+1, b-2 \rrbracket$ | a.s.d. | a.s.d. | a.s.d. | $\notin \mathcal{C}_f^\infty$ | neutral |
| $n = b-1$ | $\notin \mathcal{C}_f^\infty$ | a.s.d. | a.s.d. | $\notin \mathcal{C}_f^\infty$ | weak |
| $n = b$ | $\notin \mathcal{C}_f^\infty$ | $\notin \mathcal{C}_f^\infty$ | $\notin \mathcal{C}_f^\infty$ | a.s.d. | not bispecial |

□

8.3.2.2. Long bispecial factors

We begin with a very similar result to Lemma 7.3.5.

Lemma 8.3.4. *Let $\mathcal{A} = \{a, b\}$ be an even alphabet. Let $x \in \mathcal{C}^\infty$ and let u be a bispecial factor of x .*

(i) *If $\mathcal{D}_f(u)$ is long, then $u = \mathcal{P}_{f, \bar{x}_0}(\mathcal{D}_f(u))$ and $\mathcal{D}_f(u)$ is a bispecial factor of $\mathcal{D}(x)$, of the same type.*

(ii) *If u is long and $c \in \mathcal{A}$, then $\mathcal{P}_{f, \bar{c}}(u)$ is a bispecial factor of $\mathcal{P}_c(x)$, of the same type.*

Proof. (i) We first observe that u is also long, so let $a_1^{p_1} \dots a_n^{p_n}$ be the canonical factorization of u , where $n = \|u\| \geq 2$. u is a bispecial factor of x so a fortiori it is a bispecial word of \mathcal{C}_f^∞ and Lemma 7.3.5 (i) states that $u = \mathcal{P}_{f, u_1}(\mathcal{D}_f(u))$.

Now we show that $v := \mathcal{P}_{f, x_0}(\mathcal{D}_f(u))$ is not a bispecial factor of x . If $p_2 = a$, we notice that $f_1 \sqsubset x_0 v$ so Lemma 8.3.1 states that $x_0 v \not\sqsubset x$; but if $p_2 = b$ we notice that $f_2 \sqsubset \bar{x}_0 v$ so Lemma 8.3.1 states that $\bar{x}_0 v \not\sqsubset x$. Therefore $u_1 = \bar{x}_0$.

For the type we show four equivalences:

$$a_1 u a_n \sqsubset x \iff b \mathcal{D}_f(u) b \sqsubset \mathcal{D}(x) \quad (14)$$

$$a_1 u \bar{a}_n \sqsubset x \iff b \mathcal{D}_f(u) a \sqsubset \mathcal{D}(x) \quad (15)$$

$$\bar{a}_1 u a_n \sqsubset x \iff a \mathcal{D}_f(u) b \sqsubset \mathcal{D}(x) \quad (16)$$

$$\bar{a}_1 u \bar{a}_n \sqsubset x \iff a \mathcal{D}_f(u) a \sqsubset \mathcal{D}(x) \quad (17)$$

For (14) we observe that $a_1 u a_n = a_1^{a+1} a_2^{p_2} \dots a_{n-1}^{p_{n-1}} a_n^{a+1} \in \mathcal{C}_f^1$ and $\mathcal{D}_f(a_1 u a_n) = b p_2 \dots p_{n-1} b = b \mathcal{D}_f(u) b$ so $a_1 u a_n \sqsubset x \implies b \mathcal{D}_f(u) b \sqsubset \mathcal{D}(x)$. Conversely, if $b \mathcal{D}_f(u) b \sqsubset \mathcal{D}(x)$, Theorem 8.1.5 ensures that $b \mathcal{D}_f(u) b$ occurs not as a prefix in $\mathcal{D}(x)$ so Lemma 8.2.2 (ii) states that $a_1 u a_n \sqsubset x$ or $\bar{a}_1 u \bar{a}_n \sqsubset x$, but Proposition 8.2.8 ensures that $\bar{u} \not\sqsubset x$ so we have $a_1 u a_n \sqsubset x$. Finally, the proofs of (15), (16) and (17) are similar.

Now we have $\bar{a}_1 u \sqsubset x$ because u is bispecial in x , so either $\bar{a}_1 u a_n \sqsubset x$ or $\bar{a}_1 u \bar{a}_n \sqsubset x$, and then (14) and (15) yield $b \mathcal{D}_f(u) \sqsubset x$. Similarly we show that $a \mathcal{D}_f(u)$, $\mathcal{D}_f(u) a$ and $\mathcal{D}_f(u) b \sqsubset x$, which means that $\mathcal{D}_f(u)$ is bispecial in x . Finally, (14), (15), (16) and (17) ensure that $\mathcal{D}_f(u)$ has the same type as u .

(ii) We have $\mathcal{D}_f(\mathcal{P}_{f, a}(u)) = u \sqsubset x$ so Lemma 8.2.2 (ii) states that $\mathcal{P}_{f, a}(u) \sqsubset \mathcal{P}_c(x)$ or $\mathcal{P}_{f, b}(u) = \overline{\mathcal{P}_{f, a}(u)} \sqsubset \mathcal{P}_c(x)$, which provides $d \in \mathcal{A}$ such that $\mathcal{P}_{f, d}(u) \sqsubset \mathcal{P}_c(x)$. In

8. Smooth sequences over even and odd alphabets – 8.3. Factor complexity over even alphabets

particular, $\mathcal{D}_f(\mathcal{P}_{f,d}(u)) = u$ is long so (14), (15), (16) and (17) hold for $\mathcal{P}_{f,d}(u)$ and the same arguments as in (i) yield that $d = \bar{c}$ and $\mathcal{P}_{f,d}(u)$ is bispecial in $\mathcal{P}_c(x)$, of the same type as u . \square

We deduce that any long bispecial factor u of x can be traced back to a long bispecial factor v of $\mathcal{D}^n(x)$ such that $\mathcal{D}_f(v)$ is short, with the same type in their respective smooth sequence.

Definition 8.3.5. *If a word u is long but $\mathcal{D}_f(u)$ is short, we call u a first-long bispecial word.*

If a word u is long bispecial in a smooth sequence x , let us define the degree of u to be the first integer $n \geq 0$ such that $\mathcal{D}_f^n(u)$ is a first-long bispecial in $\mathcal{D}^n(x)$, and in that case we call $\mathcal{D}_f^n(u)$ the root of u . In particular, Lemma 8.3.4 (i) ensures that u has the same type as its root.

Definition 8.3.6. *If $u \in \mathcal{A}^*$, $y \in \mathcal{A}^{\mathbb{N}}$ and $n \geq 0$, we define the word $\mathcal{P}_{f,y,n}(u) := \mathcal{P}_{f,y_0} \circ \mathcal{P}_{f,y_1} \dots \circ \mathcal{P}_{f,y_{n-1}}(u)$, with the convention that $\mathcal{P}_{f,y,0}(u) := u$.*

With the previous notations, Lemma 8.3.4 yields a characterization of the long bispecial factors over even alphabets.

Lemma 8.3.7. *Let $x \in \mathcal{C}^\infty$ over an even alphabet \mathcal{A} and let $y := \overline{\mathcal{T}(x)}$. Then the long bispecial factors of degree n of x are the words $\mathcal{P}_{f,y,n}(u)$ where u is a first-long bispecial factor of $\mathcal{D}^n(x)$, and they have the same type in x as their root u in $\mathcal{D}^n(x)$.*

Proof. First, Lemma 8.3.4 (ii) ensures that the words of this form are long bispecial of degree n in x . To show the converse, we proceed by induction on n . If $n = 0$, this is trivial. Let $n \geq 0$ be such that, for all $x \in \mathcal{C}^\infty$, for all long bispecial factor u of x of degree n and of root v , then $u = \mathcal{P}_{f,y,n}(v)$ where $y = \overline{\mathcal{T}(x)}$. Now let $x \in \mathcal{C}^\infty$ and let u be a long bispecial factor of degree $n+1$ in x . In particular, $v := \mathcal{D}_f(u)$ is long so Lemma 8.3.4 (i) states that $u = \mathcal{P}_{f,\bar{x}_0}(v)$ and that v is a long bispecial factor of $\mathcal{D}(x)$, of degree n . By hypothesis, $v = \mathcal{P}_{f,y,n}(w)$ where w is the root of v and $y = \overline{\mathcal{T}(\mathcal{D}(x))}$. We now have $u = \mathcal{P}_{\bar{x}_0}(\mathcal{P}_{f,y,n}(w))$, and by noticing that w is also the root of u and that $\overline{\mathcal{T}(x)} = \bar{x}_0 y$, we obtain the desired result. \square

It remains to identify the first-long bispecial factors and their type.

8.3.2.3. First-long bispecial factors

If $x \in \mathcal{C}^\infty$ and $u \in \mathcal{A}^*$ is a first-long bispecial factor of x , a fortiori it is bispecial in \mathcal{C}_f^∞ so Lemma 7.3.5 (i) states that $u = \mathcal{P}_{f,u_1}(\mathcal{D}_f(u))$ and that $\mathcal{D}_f(u)$ is a bispecial \mathcal{C}_f^∞ word. Also, $\mathcal{D}_f(u)$ is short but a^b and b^b are not bispecial in \mathcal{C}_f^∞ so $\mathcal{D}_f(u)$ must be almost-short. This means that the first-long bispecial factors of x are of the form $\mathcal{P}_{f,d}(c^n)$ where $c, d \in \mathcal{A}$ and $n \in \llbracket 0, b-1 \rrbracket$. Now we are ready to describe the first-long bispecial factors and their type.

8. Smooth sequences over even and odd alphabets – 8.3. Factor complexity over even alphabets

Lemma 8.3.8. *Let $x \in \mathcal{C}^\infty$ over an even alphabet $\mathcal{A} = \{a, b\}$.*

(i) *If $a = b - 2$, then $\mathcal{P}_{f, \bar{x}_0}(\varepsilon)$ is the unique first-long strong bispecial factor of x ; $\mathcal{P}_{f, x_0}(\varepsilon)$ is the unique first-long weak bispecial factor of x .*

(ii) *If $a < b - 2$, then $\mathcal{P}_{f, \bar{x}_0}(\varepsilon)$, $\mathcal{P}_{f, \bar{x}_0}(a^a)$ and $\mathcal{P}_{f, \bar{x}_0}(b^a)$ are the first-long strong bispecial factors of x ; $\mathcal{P}_{f, x_0}(\varepsilon)$, $\mathcal{P}_{f, x_0}(a^{b-2})$ and $\mathcal{P}_{f, x_0}(b^{b-2})$ are the first-long weak bispecial factors of x .*

Proof. For $c, d \in \mathcal{A}$ and $n \in \llbracket 0, b - 1 \rrbracket$, we check the bi-extensions of the first-long word $\mathcal{P}_{f, d}(c^n)$ in x .

By setting $u := \mathcal{P}_{f, x_0}(\varepsilon) = x_0^a \bar{x}_0^a$, we have $x_0 u x_0 = f_1$, $x_0 u \bar{x}_0 \sqsubset m_3$, $\bar{x}_0 u x_0 \sqsubset m_1$ and $\bar{x}_0 u \bar{x}_0 = f_2$, so with Lemmas 8.3.1 and 8.3.2 we deduce that $\mathcal{P}_{f, x_0}(\varepsilon)$ is a weak bispecial factor of x .

By setting $u := \mathcal{P}_{f, \bar{x}_0}(\varepsilon) = \bar{x}_0^a x_0^a$, we have $x_0 u x_0 \sqsubset m_1$, $x_0 u \bar{x}_0 \sqsubset m_1$, $\bar{x}_0 u x_0 \sqsubset m_3$ and $\bar{x}_0 u \bar{x}_0 \sqsubset m_1$, so with Lemmas 8.3.1 and 8.3.2 we deduce that $\mathcal{P}_{f, \bar{x}_0}(\varepsilon)$ is a strong bispecial factor of x .

We show that $\mathcal{P}_{f, x_0}(c^n)$ is not a bispecial factor of x if $n \geq 1$. The proof is the same as in Lemma 8.3.4 (i). Also, we show that $\mathcal{P}_{f, \bar{x}_0}(c^n)$ is not a bispecial factor of x if n is odd. If n is odd, we have $u := \mathcal{P}_{f, \bar{x}_0}(c^n) = \bar{x}_0^a x_0^c \bar{x}_0^c \dots x_0^c \bar{x}_0^a$. If $c = a$, we notice that f_2 is a suffix of $\nu \bar{x}_0$ so Lemma 8.3.1 states that $\nu \bar{x}_0 \not\sqsubset x$; but if $c = b$ we notice that f_1 is a suffix of νx_0 so Lemma 8.3.1 states that $\nu x_0 \not\sqsubset x$.

It remains to check the first-long words $\mathcal{P}_{f, \bar{x}_0}(c^n)$ when n is even, $n \in \llbracket 2, b - 2 \rrbracket$ and $c \in \mathcal{A}$.

By setting $u := \mathcal{P}_{f, \bar{x}_0}(a^n) = \bar{x}_0^a [x_0^a \bar{x}_0^a]^{\frac{n}{2}} x_0^a$, with Lemmas 8.3.1 and 8.3.2 we deduce the bi-extensions of u in x and the type of u in x :

| | $x_0 u x_0$ | $x_0 u \bar{x}_0$ | $\bar{x}_0 u x_0$ | $\bar{x}_0 u \bar{x}_0$ | type of u |
|--|-----------------|-------------------|-------------------------------|-------------------------|-------------|
| $n \in \llbracket 2, a - 2 \rrbracket$ | $\sqsubset m_1$ | $\sqsubset m_1$ | $\notin \mathcal{C}_f^\infty$ | $\sqsubset m_1$ | neutral |
| $n = a = b - 2$ | $\sqsubset m_1$ | $= f_3$ | $= m_2$ | $\sqsubset m_1$ | neutral |
| $n = a < b - 2$ | $\sqsubset m_1$ | $\sqsubset m_1$ | $= m_2$ | $\sqsubset m_1$ | strong |
| $n \in \llbracket a + 2, b - 4 \rrbracket$ | $\sqsubset m_1$ | $\sqsubset m_1$ | $\notin \mathcal{C}_f^\infty$ | $\sqsubset m_1$ | neutral |
| $n = b - 2 > a$ | $\sqsubset m_1$ | $= f_3$ | $\notin \mathcal{C}_f^\infty$ | $\sqsubset m_1$ | weak |

Similarly, by setting $v := \mathcal{P}_{f, \bar{x}_0}(b^n) = \bar{x}_0^a [x_0^b \bar{x}_0^b]^{\frac{n}{2}} x_0^a$, with Lemmas 8.3.1 and 8.3.2 we deduce the bi-extensions of v in x and the type of v in x :

| | $x_0 v x_0$ | $x_0 v \bar{x}_0$ | $\bar{x}_0 v x_0$ | $\bar{x}_0 v \bar{x}_0$ | type of v |
|--|-----------------|-------------------------------|-------------------|-------------------------|-------------|
| $n \in \llbracket 2, a - 2 \rrbracket$ | $\sqsubset m_3$ | $\notin \mathcal{C}_f^\infty$ | $\sqsubset m_3$ | $\sqsubset m_3$ | neutral |
| $n = a = b - 2$ | $\sqsubset m_3$ | $= m_4$ | $= f_4$ | $\sqsubset m_3$ | neutral |
| $n = a < b - 2$ | $\sqsubset m_3$ | $= m_4$ | $\sqsubset m_3$ | $\sqsubset m_3$ | strong |
| $n \in \llbracket a + 2, b - 4 \rrbracket$ | $\sqsubset m_3$ | $\notin \mathcal{C}_f^\infty$ | $\sqsubset m_3$ | $\sqsubset m_3$ | neutral |
| $n = b - 2 > a$ | $\sqsubset m_3$ | $\notin \mathcal{C}_f^\infty$ | $= f_4$ | $\sqsubset m_3$ | weak |

□

8.3.3. Computation of the factor complexity

Finally, combining Lemmas 8.3.3, 8.3.7 and 8.3.8 yields the description of the strong and weak bispecial factors of any smooth sequence.

Proposition 8.3.9. *Let $\mathcal{A} = \{a, b\}$ be an even alphabet. Let $x \in \mathcal{C}^\infty$ and set $y := \overline{\mathcal{T}(x)}$.*

(i) *If $a = b - 2$, then the strong bispecial factors of x are ε , a^a , b^a and the words $\mathcal{P}_{f,y,n+1}(\varepsilon)$ for $n \geq 0$; and the weak bispecial factors of x are a^{a+1} , b^{a+1} and the words $\mathcal{P}_{f,y,n}(\mathcal{P}_{f,\overline{y_n}}(\varepsilon))$ for $n \geq 0$.*

(ii) *If $a < b - 2$, then the strong bispecial factors of x are ε , a^a , b^a and the words $\mathcal{P}_{f,y,n+1}(\varepsilon)$, $\mathcal{P}_{f,y,n+1}(a^a)$ and $\mathcal{P}_{f,y,n+1}(b^a)$ for $n \geq 0$; and the weak bispecial factors of x are a^{b-1} , b^{b-1} and the words $\mathcal{P}_{f,y,n}(\mathcal{P}_{f,\overline{y_n}}(\varepsilon))$, $\mathcal{P}_{f,y,n+1}(a^{b-2})$ and $\mathcal{P}_{f,y,n+1}(b^{b-2})$ for $n \geq 0$.*

We then study the length of the primitives.

Lemma 8.3.10. *Let $\mathcal{A} = \{a, b\}$ be an even alphabet. If $u \in \mathcal{A}^*$ has even length and $y \in \mathcal{A}^\mathbb{N}$, then the sequence $(\ell_n)_{n \geq 0} := (|\mathcal{P}_{f,y,n+1}(u)|)_{n \geq 0}$ satisfies*

$$\begin{cases} \ell_0 = a|u|_a + b|u|_b + 2a \\ \ell_{n+1} = \frac{a+b}{2}\ell_n + 2a \text{ for all } n \geq 0. \end{cases}$$

Proof. Let us first observe that, if $v \in \mathcal{A}^*$ and $c \in \mathcal{A}$, then $|\mathcal{P}_{f,c}(v)| = a|v|_a + b|v|_b + 2a$, which already provides the formula for ℓ_0 . Also, if v has even length, then $|\mathcal{P}_{f,c}(v)|_a = |\mathcal{P}_{f,c}(v)|_b$. We then obtain by immediate induction that $|\mathcal{P}_{f,y,n+1}(u)|_a = |\mathcal{P}_{f,y,n+1}(u)|_b$ for all $n \geq 0$. Finally, with the initial observation we have

$$\begin{aligned} \ell_{n+1} &= a|\mathcal{P}_{f,y,n+1}(u)|_a + b|\mathcal{P}_{f,y,n+1}(u)|_b + 2a \\ &= a\frac{|\mathcal{P}_{f,y,n+1}(u)|}{2} + b\frac{|\mathcal{P}_{f,y,n+1}(u)|}{2} + 2a \\ &= \frac{a+b}{2}\ell_n + 2a \end{aligned}$$

for all $n \geq 0$. □

This allows to show that all smooth sequences over an even alphabet have the same factor complexity, which yields linear complexity, this is Theorem 8.1.6 (i).

Proof of Theorem 8.1.6 (i). Let $\mathcal{A} = \{a, b\}$ be an even alphabet and let $x \in \mathcal{C}^\infty$. Then Lemma 8.3.10 implies that the lengths of the bispecial factors of x $\mathcal{P}_{f,y,n+1}(\varepsilon)$, $\mathcal{P}_{f,y,n}(\mathcal{P}_{f,\overline{y_n}}(\varepsilon))$ (and $\mathcal{P}_{f,y,n+1}(a^a)$, $\mathcal{P}_{f,y,n+1}(b^a)$, $\mathcal{P}_{f,y,n+1}(a^{b-2})$ and $\mathcal{P}_{f,y,n+1}(b^{b-2})$ if $a < b - 2$) do not depend on $y := \overline{\mathcal{T}(x)}$, i.e., they do not depend on x . With Theorem 6.2.2 this means that the second finite difference $b(n)$ does not depend on x , which implies that every smooth sequence has the same factor complexity. Finally, Corollary 8.1.4 provides smooth sequences x that satisfy $p_x(n) = \Theta(n)$. □

8.3.4. Unique ergodicity

In fact, Lemma 8.3.10 allows to precisely compute the length of the bispecial factors, and consequently the factor complexity. In particular, under a numerical condition on the alphabet, we are able to show that $\limsup_{n \rightarrow \infty} \frac{p_x(n)}{n} < 3$, which implies unique ergodicity thanks to [Bos84, Theorem 1.5], this is Theorem 8.1.7 (i).

Proof of Theorem 8.1.7 (i). Let $\mathcal{A} = \{a, b\}$ be an even alphabet such that $\frac{a+b}{2}a(a+2) > b(b-2)$ and let $x \in \mathcal{C}^\infty$.

If $a = b - 2$, Lemma 8.3.10 implies that the bispecial factors of x $\mathcal{P}_{f,y,n+1}(\varepsilon)$ and $\mathcal{P}_{f,y,n}(\mathcal{P}_{f,\overline{y_n}}(\varepsilon))$ have the same length for all $n \geq 0$. Because they have opposite type, their contributions to the factor complexity in Theorem 6.2.2 cancel each other, leaving only the contribution of the bispecial factors ε , a^a , b^a , a^{a+1} and b^{a+1} . We deduce that

$$b(n) = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{if } n \in \llbracket 1, a-1 \rrbracket \\ 2 & \text{if } n = a \\ -2 & \text{if } n = a+1 \\ 0 & \text{if } n \geq a+2 = b \end{cases}, \quad s(n) = \begin{cases} 1 & \text{if } n = 0 \\ 2 & \text{if } n \in \llbracket 1, a \rrbracket \\ 4 & \text{if } n = a+1 \\ 2 & \text{if } n \geq a+2 = b \end{cases}$$

and

$$p_x(n) = \begin{cases} 1 & \text{if } n = 0 \\ 2n & \text{if } n \in \llbracket 1, a+1 \rrbracket \\ 2n+2 & \text{if } n \geq a+2 = b \end{cases}.$$

In particular, we have $\limsup_{n \rightarrow \infty} \frac{p_x(n)}{n} = 2 < 3$ so [Bos84, Theorem 1.5] yields the result.

If $a < b - 2$, we compute the values of $s(n)$ by ordering the lengths of the bispecial factors of x provided by Proposition 8.3.9 (ii) with $y := \overline{\mathcal{T}(x)}$. The ordering of the lengths of the short bispecial factors is the following:

$$|\varepsilon| = 0 < |a^a| = |b^a| = a < |a^{b-1}| = |b^{b-1}| = b-1. \quad (18)$$

As when $a = b - 2$, the contributions of the bispecial factors $\mathcal{P}_{f,y,n+1}(\varepsilon)$ and $\mathcal{P}_{f,y,n}(\mathcal{P}_{f,\overline{y_n}}(\varepsilon))$ cancel each other. Let us introduce notations for the lengths of the other long bispecial factors: for $n \geq 0$, we set

$$p_n := |\mathcal{P}_{f,y,n+1}(a^a)|, \quad q_n := |\mathcal{P}_{f,y,n+1}(a^{b-2})|, \quad r_n := |\mathcal{P}_{f,y,n+1}(b^a)|, \quad s_n := |\mathcal{P}_{f,y,n+1}(b^{b-2})|.$$

In particular, Lemma 8.3.10 and the numerical conditions on a and b yield the ordering

$$p_0 = a(a+2) < q_0 = ab < r_0 = a(b+2) < s_0 = b(b-2) + 2a < p_1 = \frac{a+b}{2}a(a+2) + 2a. \quad (19)$$

8. Smooth sequences over even and odd alphabets – 8.4. Factor complexity over odd alphabets

Moreover, Lemma 8.3.10 states that the four sequences $(p_n)_{n \geq 0}$, $(q_n)_{n \geq 0}$, $(r_n)_{n \geq 0}$ and $(s_n)_{n \geq 0}$ satisfy the recurrence relation $\ell_{n+1} = \frac{a+b}{2}\ell_n + 2a$, so an immediate induction yields $p_n < q_n < r_n < s_n < p_{n+1}$ for all $n \geq 0$. Finally, to link Equation (18) and Equation (19), we have $a < b - 2$ so $a \leq b - 4$ and $\frac{a+b}{2} \leq b - 2$, and then the numerical condition on a and b yields $a(a+2) > \frac{2}{a+b}b(b-2) \geq b > b - 1$. We deduce the complete ordering of the length of bispecial factors of x :

$$|\varepsilon| < |a^a| = |b^a| < |a^{b-1}| = |b^{b-1}| < p_0 < q_0 < r_0 < s_0 < p_1 < q_1 < r_1 < s_1 < \dots$$

With the fact that the numbers p_n and r_n (resp. q_n and s_n) are each the length of a single strong (resp. weak) bispecial factor of x , we compute the values of $s(n)$:

$$s(n) = \begin{cases} 1 & \text{if } n = 0 \\ 2 & \text{if } n \in \llbracket 1, a \rrbracket \\ 4 & \text{if } n \in \llbracket a+1, b-1 \rrbracket \\ 2 & \text{if } n \in \llbracket b, p_0 \rrbracket \end{cases} \quad \text{and} \quad s(n) = \begin{cases} 3 & \text{if } n \in \llbracket p_k+1, q_k \rrbracket \\ 2 & \text{if } n \in \llbracket q_k+1, r_k \rrbracket \\ 3 & \text{if } n \in \llbracket r_k+1, s_k \rrbracket \\ 2 & \text{if } n \in \llbracket s_k+1, p_{k+1} \rrbracket \end{cases} \quad \text{for some } k \geq 0.$$

Finally, because the four sequences $(p_n)_{n \geq 0}$, $(q_n)_{n \geq 0}$, $(r_n)_{n \geq 0}$ and $(s_n)_{n \geq 0}$ satisfy the recurrence relation $\ell_{n+1} = \frac{a+b}{2}\ell_n + 2a$, the lengths of the intervals $\llbracket p_k+1, q_k \rrbracket$, $\llbracket q_k+1, r_k \rrbracket$, $\llbracket r_k+1, s_k \rrbracket$, $\llbracket s_k+1, p_{k+1} \rrbracket$ satisfy the recurrence relation $\ell_{k+1} = \frac{a+b}{2}\ell_k$. In particular, $s(n)$ is equal to 2 with non-zero proportion, which implies that $\limsup_{n \rightarrow \infty} \frac{p_x(n)}{n} < 3$ so [Bos84, Theorem 1.5] yields the result. \square

Remark 8.3.11. *On the contrary, $\frac{a+b}{2}a(a+2) < b(b-2)$ (it cannot be equal), similar computations yield $3 < \limsup_{n \rightarrow \infty} \frac{p_x(n)}{n} < 4$ for all $x \in C^\infty$ so Boshernitzan criterion for unique ergodicity cannot be applied.*

8.4. Factor complexity over odd alphabets

In this section we prove Theorem 8.1.6 (ii) and Theorem 8.1.7 (ii). We proceed exactly like Section 8.3.

8.4.1. Forbidden and mandatory factors

In addition to Corollary 8.2.5, we give another lemma that will greatly help finding if a finite word occurs in a smooth sequence or not.

Lemma 8.4.1. *If $x \in C^\infty$ over an odd alphabet $\mathcal{A} = \{a, b\}$, we define the forbidden words*

$$\begin{aligned} f_1 &:= x_0 \overline{x_0}^a x_0^{a+1}, & f_2 &:= x_0^{a+1} \overline{x_0}^a x_0, \\ f_3 &:= x_0 \left[\overline{x_0}^a x_0^a \right]^{\frac{b-1}{2}} \overline{x_0}^a x_0, & f_4 &:= x_0^{a+1} \left[\overline{x_0}^b x_0^b \right]^{\frac{b-3}{2}} \overline{x_0}^b x_0^{a+1}. \end{aligned}$$

8. Smooth sequences over even and odd alphabets – 8.4. Factor complexity over odd alphabets

- (i) If $\mathcal{D}(x)_0 = a$, if $u \in \mathcal{A}^*$ and there exists $i \in \{1, 2, 3, 4\}$ such that $\overline{f_i} \sqsubset u$, then $u \not\sqsubset x$.
(ii) If $\mathcal{D}(x)_0 = b$, if $u \in \mathcal{A}^*$ and there exists $i \in \{1, 2, 3, 4\}$ such that $\overline{f_i} \sqsubset u$, then $u \not\sqsubset x$.

Proof. (i) As $\mathcal{D}(x)_0 = a$, Lemma 8.2.7 (ii) states that ab occurs only at even positions in $\mathcal{D}(x)$ and ba occurs only at odd positions in $\mathcal{D}(x)$. It suffices to show that $\overline{f_i} \not\sqsubset x$ for all $i \in \{1, 2, 3, 4\}$.

If $\overline{f_1} \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0}^{p_1} \dots x_0^{p_{2i}} \overline{x_0}^a x_0^b \dots$ and $\mathcal{D}(x) = p_0 p_1 \dots p_{2i} ab \dots$ which contradicts Lemma 8.2.7 (ii).

If $\overline{f_2} \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0}^{p_1} \dots \overline{x_0}^{p_{2i-1}} x_0^b \overline{x_0}^a x_0^{p_{2i+2}} \dots$ and $\mathcal{D}(x) = p_0 p_1 \dots p_{2i-1} ba \dots$ which contradicts Lemma 8.2.7 (ii).

If $\overline{f_3} \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0}^{p_1} \dots x_0^{p_{2i}} \left[\overline{x_0}^a x_0^a \right]^{\frac{b-1}{2}} \overline{x_0}^a x_0^{p_{2i+b+1}} \dots$ and $\mathcal{D}(x) = p_0 \dots p_{2i} a^b p_{2i+b+1} \dots$. Then, if $p_{2i} = b$ this contradicts Lemma 8.2.7 (i), and if $p_{2i} = a$ we have $\mathcal{D}(x) \notin \mathcal{C}^1$.

If $\overline{f_4} \sqsubset x$, this provides $i \geq 0$ such that $x = x_0^{p_0} \overline{x_0}^{p_1} \dots \overline{x_0}^{p_{2i-1}} \left[x_0^b \overline{x_0}^b \right]^{\frac{b-1}{2}} x_0^b \overline{x_0}^{p_{2i+b}} \dots$ and $\mathcal{D}(x) = p_0 \dots p_{2i-1} b^b p_{2i+b} \dots$. Then, if $p_{2i-1} = a$ this contradicts Lemma 8.2.7 (i), and if $p_{2i} = b$ we have $\mathcal{D}(x) \notin \mathcal{C}^1$.

(ii) As $\mathcal{D}(x)_0 = b$, Lemma 8.2.7 (ii) states that ab occurs only at odd positions in $\mathcal{D}(x)$ and ba occurs only at even positions in $\mathcal{D}(x)$. Then we obtain in a similar way that $\overline{f_i} \not\sqsubset x$ for all $i \in \{1, 2, 3, 4\}$. \square

Lemma 8.4.2. *If $x \in \mathcal{C}^\infty$ over an odd alphabet $\mathcal{A} = \{a, b\}$ is such that $a \neq 1$ or $\mathcal{D}^2(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, we define the mandatory words*

$$\begin{aligned} m_1 &:= \overline{x_0}^{a+1} \left[x_0^a \overline{x_0}^a \right]^{\frac{b-1}{2}} x_0^a \overline{x_0}^{a+1}, & m_2 &:= \overline{x_0}^{a+1} \left[x_0^a \overline{x_0}^a \right]^{\frac{a-1}{2}} x_0^a \overline{x_0}^{a+1}, \\ m_3 &:= \overline{x_0} x_0^a \left[\overline{x_0}^b x_0^b \right]^{\frac{b-1}{2}} \overline{x_0}^b x_0^a \overline{x_0}, & m_4 &:= \overline{x_0} x_0^a \left[\overline{x_0}^b x_0^b \right]^{\frac{a-1}{2}} \overline{x_0}^b x_0^a \overline{x_0}. \end{aligned}$$

- (i) If $\mathcal{D}(x)_0 = a$, if $u \in \mathcal{A}^*$ and there exists $i \in \{1, 2, 3, 4\}$ such that $u \sqsubset m_i$, then $u \sqsubset x$.
(ii) If $\mathcal{D}(x)_0 = b$, if $u \in \mathcal{A}^*$ and there exists $i \in \{1, 2, 3, 4\}$ such that $u \sqsubset \overline{m_i}$, then $u \sqsubset x$.

Proof. (i) It suffices to show that $m_i \sqsubset x$ for all $i \in \{1, 2, 3, 4\}$.

$\mathcal{D}_f(ba^b b) = b$ is almost-short so Corollary 8.2.5 states that $ba^b b \sqsubset \mathcal{D}(x)$. Moreover, Theorem 8.1.5 implies that $\mathcal{D}_f(m_1) = ba^b b$ occurs not as a prefix in $\mathcal{D}(x)$ so Lemma 8.2.2 (ii) states that $m_1 \sqsubset x$ or $\overline{m_1} \sqsubset x$, but we notice that $\overline{f_2} \sqsubset_p \overline{m_1}$ so Lemma 8.3.1 states that $\overline{m_1} \not\sqsubset x$.

Similarly, we have $\mathcal{D}_f(ba^a b) = a$, $\mathcal{D}_f(m_2) = ba^a b$ and $\overline{f_2} \sqsubset_p \overline{m_2}$ so $m_2 \sqsubset x$.

Similarly, we have $\mathcal{D}_f(ab^b a) = b$, $\mathcal{D}_f(m_3) = ab^b a$ and $\overline{f_1} \sqsubset_p \overline{m_3}$ so $m_3 \sqsubset x$.

Similarly, we have $\mathcal{D}_f(ab^a a) = a$, $\mathcal{D}_f(m_4) = ab^a a$ and $\overline{f_1} \sqsubset_p \overline{m_4}$ so $m_4 \sqsubset x$.

(ii) Similarly to (i), we obtain that $\overline{m_i} \sqsubset x$ for all $i \in \{1, 2, 3, 4\}$. \square

8.4.2. Bispecial factors

8.4.2.1. Short bispecial factors

Lemma 8.4.3. *Let $x \in \mathcal{C}^\infty$ over an odd alphabet $\mathcal{A} = \{a, b\}$. If $a \neq 1$ or $\mathcal{D}(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, then the short strong bispecial factors of x are ε , a^a and b^a ; and the short weak bispecial factors of x are a^{b-1} and b^{b-1} .*

Proof. The proof is identical to Lemma 8.3.3. □

8.4.2.2. Long bispecial factors

We begin with a very similar result to Lemma 8.3.4.

Lemma 8.4.4. *Let $\mathcal{A} = \{a, b\}$ be an odd alphabet. Let $x \in \mathcal{C}^\infty$ and let u be a bispecial factor of x . If $\mathcal{D}_f(u)$ is long, then $u = \mathcal{P}_{f, u_1}(\mathcal{D}_f(u))$ and $\mathcal{D}_f(u)$ is a bispecial factor of $\mathcal{D}(x)$, of the same type.*

Proof. The proof is identical to Lemma 8.3.4 (i), except that we do not determine u_1 . □

We deduce that any long bispecial factor u of x can be traced back to a long bispecial factor v of $\mathcal{D}^n(x)$ such that $\mathcal{D}_f(v)$ is short, with the same type in their respective smooth sequence.

8.4.2.3. First-long bispecial factors

Lemma 8.4.5. *Let $x \in \mathcal{C}^\infty$ over an odd alphabet $\{a, b\}$ such that $a = b - 2$. If $a \neq 1$ or $\mathcal{D}^2(x) \notin \{\mathcal{E}((1b)^\omega), \mathcal{E}((b1)^\omega)\}$, then every first-long bispecial factor of x is neutral.*

Proof. Suppose that $\mathcal{D}(x)_0 = a$.

We check the word $\mathcal{P}_{f, x_0}(\varepsilon)$. By setting $u := \mathcal{P}_{f, x_0}(\varepsilon) = x_0^a \overline{x_0}^a$, we have $x_0 u x_0 = f_2$, $x_0 u \overline{x_0} \sqsubset m_3$, $\overline{x_0} u x_0 \sqsubset m_1$ and $\overline{x_0} u \overline{x_0} \sqsubset m_3$ so with Lemma 8.4.1 (i) and Lemma 8.4.2 (i) we deduce that $\mathcal{P}_{f, x_0}(\varepsilon)$ is a neutral bispecial factor of x .

We check the word $\mathcal{P}_{f, \overline{x_0}}(\varepsilon)$. By setting $u := \mathcal{P}_{f, \overline{x_0}}(\varepsilon) = \overline{x_0}^a x_0^a$, we have $x_0 u x_0 = f_1$, $x_0 u \overline{x_0} \sqsubset m_1$, $\overline{x_0} u x_0 \sqsubset m_3$ and $\overline{x_0} u \overline{x_0} \sqsubset m_1$ so with Lemma 8.4.1 (i) and Lemma 8.4.2 (i) we deduce that $\mathcal{P}_{f, \overline{x_0}}(\varepsilon)$ is a neutral bispecial factor of x .

We show that $\mathcal{P}_{f, x_0}(a^n)$ is not a bispecial factor of x if $n \geq 1$. It suffices to observe that $f_2 \sqsubset x_0 \mathcal{P}_{f, x_0}(a^n)$ so Lemma 8.4.1 (i) states that $x_0 \mathcal{P}_{f, x_0}(a^n) \not\sqsubset x$.

We show that $\mathcal{P}_{f, \overline{x_0}}(a^n)$ is not a bispecial factor of x if $n \geq 1$ and n is even. In that case, we have $\mathcal{P}_{f, \overline{x_0}}(a^n) = \overline{x_0}^a [x_0^a \overline{x_0}^a]^{\frac{n}{2}} x_0^a$ so $f_1 \sqsubset u x_0$ and Lemma 8.4.1 (i) states that $\mathcal{P}_{f, \overline{x_0}}(a^n) x_0 \not\sqsubset x$.

We check the words $\mathcal{P}_{f, \overline{x_0}}(a^n)$ where $n \in \llbracket 1, a \rrbracket$ and n is odd. By setting $u := \mathcal{P}_{f, \overline{x_0}}(a^n) = \overline{x_0}^a [x_0^a \overline{x_0}^a]^{\frac{n+1}{2}}$, with Lemma 8.4.1 (i) and Lemma 8.4.2 (i) we deduce the bi-extensions of u in x and the type of u in x :

8. Smooth sequences over even and odd alphabets – 8.4. Factor complexity over odd alphabets

| | $x_0 u x_0$ | $x_0 u \bar{x}_0$ | $\bar{x}_0 u x_0$ | $\bar{x}_0 u \bar{x}_0$ | type of u |
|--------------------------------------|-----------------|-------------------|-------------------|-------------------------|-------------|
| $n \in \llbracket 1, a-2 \rrbracket$ | $\sqsubset m_1$ | $\sqsubset m_1$ | $\sqsubset m_1$ | $\notin C_f^\infty$ | neutral |
| $n = a = b-2$ | $= f_3$ | $\sqsubset m_1$ | $\sqsubset m_1$ | $= m_2$ | neutral |

We show that $\mathcal{P}_{f,x_0}(b^n)$ is not a bispecial factor of x if $n \geq 1$ and n is even. In that case, we have $\mathcal{P}_{f,x_0}(b^n) = x_0^a \left[\bar{x}_0^b x_0^b \right]^{\frac{n}{2}} \bar{x}_0^a$ so $f_2 \sqsubset u x_0$ and Lemma 8.4.1 (i) states that $\mathcal{P}_{f,x_0}(b^n) x_0 \not\sqsubset x$.

We show that $\mathcal{P}_{f,\bar{x}_0}(b^n)$ is not a bispecial factor of x if $n \geq 1$. It suffices to observe that $f_1 \sqsubset x_0 \mathcal{P}_{f,\bar{x}_0}(b^n)$ so Lemma 8.4.1 (i) states that $x_0 \mathcal{P}_{f,\bar{x}_0}(b^n) \not\sqsubset x$.

It remains to check the words $\mathcal{P}_{f,x_0}(b^n)$ where $n \in \llbracket 1, a \rrbracket$ and n is odd. By setting $u := \mathcal{P}_{f,x_0}(b^n) = x_0^a \left[\bar{x}_0^b x_0^b \right]^{\frac{n-1}{2}} \bar{x}_0^b x_0^a$, with Lemma 8.4.1 (i) and Lemma 8.4.2 (i) we deduce the bi-extensions of u in x and the type of u in x :

| | $x_0 u x_0$ | $x_0 u \bar{x}_0$ | $\bar{x}_0 u x_0$ | $\bar{x}_0 u \bar{x}_0$ | type of u |
|--------------------------------------|-----------------|-------------------|-------------------|-------------------------|-------------|
| $n \in \llbracket 1, a-2 \rrbracket$ | $\sqsubset m_3$ | $\sqsubset m_3$ | $\sqsubset m_3$ | $\notin C_f^\infty$ | neutral |
| $n = a = b-2$ | $= f_4$ | $\sqsubset m_3$ | $\sqsubset m_3$ | $= m_4$ | neutral |

Finally, the proof is similar when $\mathcal{D}(x)_0 = b$. □

8.4.3. Computation of factor complexity

Finally, combining Lemmas 8.4.3 to 8.4.5 yields the description of the strong and weak bispecial factors of any smooth sequence.

Proposition 8.4.6. *Let $\mathcal{A} = \{a, b\}$ be an odd alphabet such that $a = b-2$. If $\mathcal{T}(x)$ is not eventually periodic with period $1b$, then the strong bispecial factors of x are ε , a^a and b^a ; and the weak bispecial factors of x are a^{b-1} and b^{b-1} .*

This allows to show that almost all smooth sequences over an odd alphabet of the form $\{a, a+2\}$ have the same factor complexity, which yields linear complexity, this is Theorem 8.1.6 (ii).

Proof of Theorem 8.1.6 (ii). Let $\mathcal{A} = \{a, b\}$ be an odd alphabet such that $a = b-2$ and let $x \in C^\infty$.

If $\mathcal{T}(x)$ is not eventually periodic with period $1b$, Proposition 8.4.6 means that x has the same strong and weak bispecial factors as $\mathcal{E}(b^\omega)$ for example, so Theorem 6.2.2 ensures that x has the same factor complexity as $\mathcal{E}(b^\omega)$. Finally, Corollary 8.1.4 states that $p_{\mathcal{E}(b^\omega)}(n) = \Theta(n)$.

If $\mathcal{T}(x)$ is eventually periodic with period $1b$, Corollary 8.1.4 states that $p_x(n) = \Theta(n)$. □

8.4.4. Unique ergodicity

In Proposition 8.4.6 we described the bispecial factors of the non-pathological smooth sequences, so we can precisely compute their factor complexity and apply [Bos84, Theorem 1.5] to obtain unique ergodicity. Fortunately, the remaining smooth sequences are primitive morphic so we directly obtain unique ergodicity for them. All in all, this proves Theorem 8.1.7 (ii).

Proof of Theorem 8.1.7 (ii). Let $\mathcal{A} = \{a, a + 2\}$ be an odd alphabet and let $x \in \mathcal{C}^\infty$.

If $\mathcal{T}(x)$ is not eventually periodic with period $1b$, Proposition 8.4.6 describes the strong and weak bispecial factors of x . They are the same as over even alphabets of the form $\{a, a + 2\}$, and we already computed their factor complexity in the proof of Theorem 8.1.7 (i), so we obtain the same factor complexity here:

$$p_x(n) = \begin{cases} 1 & \text{if } n = 0 \\ 2n & \text{if } n \in \llbracket 1, a + 1 \rrbracket \\ 2n + 2 & \text{if } n \geq a + 2 \end{cases} .$$

In particular, we have $\limsup_{n \rightarrow \infty} \frac{p_x(n)}{n} = 2 < 3$ so [Bos84, Theorem 1.5] yields the result.

If $\mathcal{T}(x)$ is eventually periodic with period $1b$, then Corollary 8.1.4 states that $X(x)$ is uniquely ergodic. \square

Bibliography

- [AB05] B. Adamczewski and Y. Bugeaud. “On the complexity of algebraic number I. Expansions in integer bases.” In: *Ann. of Math.* 165 (2 2005), pp. 547–565 (cit. on p. 15).
- [All+25] J. P. Allouche et al. “The Reflection Complexity of Sequences Over Finite Alphabets”. In: *Theory of Computing Systems* (2025). DOI: <https://doi.org/10.1007/s00224-025-10234-x> (cit. on p. 16).
- [All+17] J.P. Allouche, J. Cassaigne, J. Shallit, and L. Q. Zamboni. “A Taxonomy of Morphic Sequences”. In: (2017). arXiv: 1711.10807 [cs.FL] (cit. on pp. 15, 31).
- [AS03] J.P. Allouche and J. Shallit. *Automatic Sequences: Theory, Applications, Generalizations*. Cambridge University Press, 2003. DOI: 10.1017/CB09780511546563 (cit. on pp. 14, 40).
- [BS04] M. Baake and B. Sing. “Kolakoski-(3,1) is a (deformed) model set”. In: *Canadian Mathematical Bulletin* 47.2 (2004), pp. 168–190. DOI: 10.4153/CMB-2004-018-6 (cit. on pp. 18, 19, 86).
- [BDP26] M.P. Béal, F. Durand, and D. Perrin. *Symbolic dynamics and substitutions*. Preprint. 2026 (cit. on p. 14).
- [BPR23] M.P. Béal, D. Perrin, and A. Restivo. “Recognizability of morphisms”. In: *Ergodic Theory and Dynamical Systems* 43.11 (2023), pp. 3578–3602. DOI: 10.1017/etds.2022.109 (cit. on p. 59).
- [BPR24] M.P. Béal, D. Perrin, and A. Restivo. “Decidable problems in substitution shifts”. In: *Journal of Computer and System Sciences* 143 (2024), p. 103529. ISSN: 0022-0000. DOI: <https://doi.org/10.1016/j.jcss.2024.103529> (cit. on pp. 17, 49, 53, 54).
- [BCG82] E. R. Berlekamp, J. H. Conway, and R. K. Guy. “What is Life?” In: *Winning Ways For Your Mathematical Plays*. Vol. 2. Winning Ways For Your Mathematical Plays. Academic Press, 1982. Chap. 25, pp. 817–850 (cit. on p. 14).
- [BBC05] V. Berthé, S. Brlek, and P. Choquette. “Smooth words over arbitrary alphabets”. In: *Theoretical Computer Science* 341.1 (2005), pp. 293–310. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2005.04.010> (cit. on p. 83).

- [Ber+19] V. Berthé, W. Steiner, J.M. Thuswaldner, and R. Yassawi. “Recognizability for sequences of morphisms”. In: *Ergodic Theory and Dynamical Systems* 39.11 (2019), pp. 2896–2931. DOI: [10.1017/etds.2017.144](https://doi.org/10.1017/etds.2017.144) (cit. on p. 53).
- [BKM09] S. Bezuglyi, J. Kwiatkowski, and K. Medynets. “Aperiodic substitution systems and their Bratteli diagrams”. In: *Ergodic Theory and Dynamical Systems* 29.1 (2009), pp. 37–72. DOI: [10.1017/S0143385708000230](https://doi.org/10.1017/S0143385708000230) (cit. on pp. 17, 49, 50, 53).
- [Bos84] M. Boshernitzan. “A unique ergodicity of minimal symbolic flows with linear block growth”. In: *Journal d’analyse mathématique* 44.1 (1984), pp. 77–96. DOI: [10.1007/BF02790191](https://doi.org/10.1007/BF02790191) (cit. on pp. 19, 110, 120, 121, 125).
- [Brl+06] S. Brlek, S. Dulucq, A. Ladouceur, and L. Vuillon. “Combinatorial properties of smooth infinite words”. In: *Theoretical Computer Science* 352.1 (2006), pp. 306–317. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2005.12.003> (cit. on p. 83).
- [BJP08] S. Brlek, D. Jamet, and G. Paquin. “Smooth words on 2-letter alphabets having same parity”. In: *Theoretical Computer Science* 393.1 (2008), pp. 166–181. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2007.11.019> (cit. on pp. 18, 83, 91, 108).
- [BMP07] S. Brlek, G. Melançon, and G. Paquin. “Properties of Extremal Infinite Smooth Words”. In: *Discrete Mathematics and Theoretical Computer Science* 9 (Nov. 2007). DOI: [10.46298/dmtcs.412](https://doi.org/10.46298/dmtcs.412) (cit. on p. 94).
- [Car94] A. Carpi. “On repeated factors in C^∞ -words”. In: *Information Processing Letters* 52.6 (1994), pp. 289–294. ISSN: 0020-0190. DOI: [https://doi.org/10.1016/0020-0190\(94\)00162-6](https://doi.org/10.1016/0020-0190(94)00162-6) (cit. on p. 89).
- [Cas97] J. Cassaigne. “Complexité et facteurs spéciaux”. In: *Bulletin of the Belgian Mathematical Society - Simon Stevin* 4 (Jan. 1997). DOI: [10.36045/bbms/1105730624](https://doi.org/10.36045/bbms/1105730624) (cit. on pp. 16, 87).
- [CH26] J. Cassaigne and R. Henry. *The complexity of smooth words over binary alphabets*. 2026. arXiv: [2603.10733](https://arxiv.org/abs/2603.10733) [cs.FL] (cit. on pp. 18, 89).
- [CN10] J. Cassaigne and F. Nicolas. “Factor complexity”. In: *Combinatorics, Automata and Number Theory*. Ed. by V. Berthé. Encyclopedia of Mathematics and its Applications 135. Cambridge University Press, 2010, pp. 163–247. DOI: [10.1017/CB09780511777653.005](https://doi.org/10.1017/CB09780511777653.005) (cit. on pp. 34, 39, 42).
- [Chv93] V. Chvátal. “Notes on the Kolakoski sequence.” In: *Technical Report 93-84* (1993) (cit. on p. 89).
- [CS10] M.I. Cortez and B. Solomyak. “Invariant measures for non-primitive tiling substitutions”. In: *Journal d’Analyse Mathématique* 115 (2010), pp. 293–342. arXiv: [1007.1686](https://arxiv.org/abs/1007.1686) [math.DS] (cit. on pp. 50, 53).

- [DL06] D. Damanik and D. Lenz. “Substitution dynamical systems: characterization of linear repetitivity and applications”. In: *Journal of Mathematical Analysis and Applications* 321.2 (2006), pp. 766–780. ISSN: 0022-247X. DOI: [10.1016/j.jmaa.2005.09.004](https://doi.org/10.1016/j.jmaa.2005.09.004) (cit. on pp. 49, 52).
- [Dek79] F. M. Dekking. “Regularity and irregularity of sequences generated by automata”. eng. In: *Seminaire de Théorie des Nombres de Bordeaux* 9 (1979), pp. 1–10 (cit. on pp. 81, 85).
- [Dek81] F. M. Dekking. “On the structure of self-generating sequences”. In: *Séminaire de théorie des nombres de Bordeaux* (1981), pp. 1–6. ISSN: 09895558 (cit. on pp. 16, 18, 81, 88, 89, 91).
- [Dek01] F. M. Dekking. “What is the long range order in the Kolakoski sequence?” In: (Aug. 2001). DOI: [10.1007/978-94-015-8784-6_5](https://doi.org/10.1007/978-94-015-8784-6_5) (cit. on pp. 18, 83–85).
- [DK23] M. Dekking and M. Keane. “Two-block substitutions and morphic words”. In: *Advances in Applied Mathematics* 148 (2023), p. 102536. ISSN: 0196-8858. DOI: <https://doi.org/10.1016/j.aam.2023.102536> (cit. on p. 15).
- [Dev18] R. Devyatov. “On factor complexity of morphic sequences”. In: *Moscow Mathematical Journal* 18 (Apr. 2018), pp. 211–303. DOI: [10.17323/1609-4514-2018-18-2-211-303](https://doi.org/10.17323/1609-4514-2018-18-2-211-303) (cit. on pp. 16, 31, 34, 35, 41–44, 73).
- [Dur00] F. Durand. “Linearly recurrent subshifts have a finite number of non-periodic subshift factors”. In: *Ergodic Theory and Dynamical Systems* 20.4 (2000), pp. 1061–1078. DOI: [10.1017/S0143385700000584](https://doi.org/10.1017/S0143385700000584) (cit. on pp. 17, 85).
- [Dur02] F. Durand. “A theorem of Cobham for non-primitive substitutions”. In: *Acta Arithmetica* 104.3 (June 2002), pp. 225–241 (cit. on p. 53).
- [DHS99] F. Durand, B. Host, and C. Skau. “Substitutional dynamical systems, Bratteli diagrams and dimension groups”. In: *Ergodic Theory Dynam. Systems* 19.4 (1999), pp. 953–993. ISSN: 0143-3857,1469-4417. DOI: [10.1017/S0143385799133947](https://doi.org/10.1017/S0143385799133947) (cit. on pp. 16, 85, 110).
- [DLR13] F. Durand, J. Leroy, and G. Richomme. “Do the Properties of an S-adic Representation Determine Factor Complexity?” In: *Journal of Integer Sequences* 2 (Mar. 2013), Art 13.2.6 (cit. on p. 26).
- [DP22] F. Durand and D. Perrin. *Dimension Groups and Dynamical Systems: Substitutions, Bratteli Diagrams and Cantor Systems*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2022 (cit. on p. 14).
- [ELR75] A. Ehrenfeucht, K.P. Lee, and G. Rozenberg. “Subword complexities of various classes of deterministic developmental languages without interactions”. In: *Theoretical Computer Science* 1.1 (1975), pp. 59–75. ISSN: 0304-3975. DOI: [https://doi.org/10.1016/0304-3975\(75\)90012-2](https://doi.org/10.1016/0304-3975(75)90012-2) (cit. on pp. 16, 33, 34).

- [ER81a] A. Ehrenfeucht and G. Rozenberg. “On the subword complexity of D0L languages with a constant distribution”. In: *Information Processing Letters* 13.3 (1981), pp. 108–113. ISSN: 0020-0190. DOI: [https://doi.org/10.1016/0020-0190\(81\)90121-6](https://doi.org/10.1016/0020-0190(81)90121-6) (cit. on p. 33).
- [ER81b] A. Ehrenfeucht and G. Rozenberg. “On the subword complexity of square-free D0L languages”. In: *Theoretical Computer Science* 16.1 (1981), pp. 25–32. ISSN: 0304-3975. DOI: [https://doi.org/10.1016/0304-3975\(81\)90028-1](https://doi.org/10.1016/0304-3975(81)90028-1) (cit. on p. 33).
- [ER82] A. Ehrenfeucht and G. Rozenberg. “On Subword Complexities of Homomorphic Images of Languages”. In: *ITA* 16 (Jan. 1982), pp. 303–316. DOI: [10.1051/ita/1982160403031](https://doi.org/10.1051/ita/1982160403031) (cit. on p. 33).
- [ER83a] A. Ehrenfeucht and G. Rozenberg. “On the subword complexity of locally catenative dol languages”. In: *Information Processing Letters* 16.1 (1983), pp. 7–9. ISSN: 0020-0190. DOI: [https://doi.org/10.1016/0020-0190\(83\)90003-0](https://doi.org/10.1016/0020-0190(83)90003-0) (cit. on p. 33).
- [ER83b] A. Ehrenfeucht and G. Rozenberg. “On the subword complexity of m-free D0L languages”. In: *Information Processing Letters* 17.3 (1983), pp. 121–124. ISSN: 0020-0190. DOI: [https://doi.org/10.1016/0020-0190\(83\)90050-9](https://doi.org/10.1016/0020-0190(83)90050-9) (cit. on p. 33).
- [ER83c] A. Ehrenfeucht and G. Rozenberg. “Repetition of subwords in D0L languages”. In: *Information and Control* 59.1 (1983), pp. 13–35. ISSN: 0019-9958. DOI: [https://doi.org/10.1016/S0019-9958\(83\)80028-X](https://doi.org/10.1016/S0019-9958(83)80028-X) (cit. on pp. 28, 50).
- [Fog+02] N. Pythéas Fogg et al. *Substitutions in dynamics, arithmetics and combinatorics*. Lecture Notes in Mathematics 1794. Springer Berlin Heidelberg, 2002. ISBN: 9783540441410 (cit. on p. 24).
- [Hen24] R. Henry. “Morphic Sequences: Complexity and Decidability”. In: *Electronic Proceedings in Theoretical Computer Science* 403 (2024), pp. 113–117. ISSN: 2075-2180. DOI: [10.4204/eptcs.403.24](https://doi.org/10.4204/eptcs.403.24) (cit. on pp. 17, 31).
- [Hen25] R. Henry. “On the minimal components of substitution subshifts”. In: *Theoretical Computer Science* 1056 (2025), p. 115517. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2025.115517> (cit. on pp. 17, 50).
- [Hua11a] Y. B. Huang. “The complexity of smooth words on 2-letter alphabets”. In: *Theoretical Computer Science* 412.45 (2011), pp. 6327–6339. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2011.07.002> (cit. on pp. 89, 93).
- [Hua11b] Y. B. Huang. “The powers of smooth words over arbitrary 2-letter alphabets”. In: (2011). arXiv: [0904.0562](https://arxiv.org/abs/0904.0562) [math.CO] (cit. on p. 89).
- [HW10] Y. B. Huang and W. D. Weakley. “A note on the complexity of C^∞ -words”. In: *Theor. Comput. Sci.* 411.40-42 (2010), pp. 3731–3735. DOI: [10.1016/J.TCS.2010.06.024](https://doi.org/10.1016/J.TCS.2010.06.024) (cit. on pp. 89, 92).

- [Inc26] OEIS Foundation Inc. *The On-Line Encyclopedia of Integer Sequences*. 2026. URL: <https://oeis.org> (cit. on pp. 13–15, 18, 81).
- [Jam+26] D. Jamet, I. Marcovici, T. de la Rue, and L. Poirier. *Frequency of patterns in smooth sequences over the alphabet* $\{1, 3\}$. 2026. arXiv: 2604.11387 [math.DS] (cit. on pp. 18, 19, 83, 108).
- [KZ02] T. Kamae and L. Zamboni. “Maximal pattern complexity for discrete systems”. In: *Ergodic Theory and Dynamical Systems* 22.4 (2002), pp. 1201–1214. DOI: [10.1017/S0143385702000585](https://doi.org/10.1017/S0143385702000585) (cit. on p. 16).
- [KM19] J. Kari and É. Moutot. “Nivat’s conjecture and pattern complexity in algebraic subshifts”. In: *Theoretical Computer Science* 777 (2019). In memory of Maurice Nivat, a founding father of Theoretical Computer Science - Part I, pp. 379–386. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2018.12.029> (cit. on p. 16).
- [Kea91] M. S. Keane. “Ergodic theory and subshifts of finite type”. In: *Ergodic theory, symbolic dynamics, and hyperbolic spaces (Trieste, 1989)*. Oxford Sci. Publ. Oxford Univ. Press, New York, 1991, pp. 35–70. ISBN: 0-19-853390-X; 0-19-859685-5 (cit. on pp. 81, 92).
- [KS15] K. Klouda and Š. Starosta. “An algorithm for enumerating all infinite repetitions in a DOL-system”. In: *Journal of Discrete Algorithms* 33 (July 2015), pp. 130–138. ISSN: 1570-8667. DOI: [10.1016/j.jda.2015.03.006](https://doi.org/10.1016/j.jda.2015.03.006) (cit. on pp. 50, 61, 65).
- [Kol65] W. Kolakoski. “Self generating runs, Problem 5304”. In: *The American Mathematical Monthly* 72 (1965), p. 674 (cit. on pp. 18, 81).
- [KÜ66] W. Kolakoski and N. Üçoluk. “Solution of Advanced Problem 5304”. In: *The American Mathematical Monthly* 73 (1966), pp. 681–682 (cit. on p. 81).
- [Lan86] C. G. Langton. “Studying artificial life with cellular automata”. In: *Physica D: Nonlinear Phenomena* 22.1 (1986). Proceedings of the Fifth Annual International Conference, pp. 120–149. ISSN: 0167-2789. DOI: [https://doi.org/10.1016/0167-2789\(86\)90237-X](https://doi.org/10.1016/0167-2789(86)90237-X) (cit. on p. 14).
- [MR18] G. Maloney and D. Rust. “Beyond primitivity for one-dimensional substitution subshifts and tiling spaces”. In: *Ergodic Theory and Dynamical Systems* 38.3 (2018), pp. 1086–1117. DOI: [10.1017/etds.2016.58](https://doi.org/10.1017/etds.2016.58) (cit. on pp. 17, 28, 29, 49–51, 53, 54, 79).
- [MH38] M. Morse and G. A. Hedlund. “Symbolic dynamics”. In: *Symbolic dynamics, American journal of mathematics* 60.3 (1938), pp. 815–866 (cit. on pp. 15, 23).
- [MH40] M. Morse and G. A. Hedlund. “Symbolic Dynamics II. Sturmian Trajectories”. In: *American Journal of Mathematics* 62 (1940), p. 1 (cit. on p. 15).

- [Nil14] J. Nilsson. “Letter frequencies in the Kolakoski sequence”. In: *Acta Physica Polonica A* 126 (2014), pp. 549–552. DOI: [10.12693/APhysPolA.126.549](https://doi.org/10.12693/APhysPolA.126.549) (cit. on pp. 89, 92).
- [Niv97] M. Nivat. *Keynote address at the 25th anniversary of EATCS, during ICALP 1997*. Bologna, 1997 (cit. on p. 16).
- [Old39] R. Oldenburger. “Exponent Trajectories in Symbolic Dynamics”. In: *Transactions of the American Mathematical Society* 46.3 (1939), pp. 453–466. ISSN: 00029947 (cit. on p. 81).
- [Pan84] J.J. Pansiot. “Complexité des facteurs des mots infinis engendrés par morphismes itérés”. In: *RAIRO Theor. Informatics Appl.* (1984). Ed. by J. Paredaens, pp. 380–389 (cit. on pp. 16, 29, 31, 34, 38, 50).
- [Pan85] J.J. Pansiot. “Subword complexities and iteration”. In: *Bulletin of the EATCS* 26 (Jan. 1985), pp. 55–62 (cit. on pp. 31, 34).
- [Pan86] J.J. Pansiot. “Decidability of Periodicity for Infinite Words”. In: *RAIRO Theor. Informatics Appl.* 20 (1986), pp. 43–46 (cit. on p. 39).
- [PSS24] P. Popoli, J. Shallit, and M. Stipulanti. “Additive word complexity and Walnut”. In: *44th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*. Vol. 323. LIPIcs. Leibniz Int. Proc. Inform. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2024, Art. No. 32, 18. ISBN: 978-3-95977-355-3. DOI: [10.4230/lipics.fsttcs.2024.32](https://doi.org/10.4230/lipics.fsttcs.2024.32) (cit. on p. 16).
- [Que87] M. Queffélec. *Substitution dynamical systems - spectral analysis*. English. Vol. 1294. Lect. Notes Math. Springer, Cham, 1987. ISBN: 3-540-18692-1. DOI: [10.1007/978-3-642-11212-6_4](https://doi.org/10.1007/978-3-642-11212-6_4) (cit. on pp. 14, 49).
- [Rao12] M. Rao. *Trucs et bidules sur la séquence de Kolakoski*. 2012. URL: <https://www.arthy.org/kola/kola.php> (cit. on pp. 89, 92).
- [RSZ09] G. Richomme, K. Saari, and L. Zamboni. “Abelian Complexity in Minimal Subshifts”. In: *Journal of the London Mathematical Society* 83 (Nov. 2009). DOI: [10.1112/jlms/jdq063](https://doi.org/10.1112/jlms/jdq063) (cit. on p. 16).
- [RS15] M. Rigo and P. Salimov. “Another generalization of abelian equivalence: Binomial complexity of infinite words”. In: *Theoretical Computer Science* 601 (2015). WORDS 2013, pp. 47–57. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2015.07.025> (cit. on p. 16).
- [RS76] G. Rozenberg and A. Salomaa. “The Mathematical Theory of L Systems”. In: *Advances in Information Systems Science: Volume 6*. Ed. by Julius T. Tou. Boston, MA: Springer US, 1976, pp. 161–206. ISBN: 978-1-4615-8249-6. DOI: [10.1007/978-1-4615-8249-6_4](https://doi.org/10.1007/978-1-4615-8249-6_4) (cit. on pp. 14, 32).
- [SS78] A. Salomaa and M. Soittola. *Automata: Theoretic Aspects of Formal Power Series*. Berlin, Heidelberg: Springer-Verlag, 1978. ISBN: 0387902821 (cit. on p. 27).

- [Shi19] T. Shimomura. “A simple approach to minimal substitution subshifts”. In: *Topology and its Applications* 260 (2019), pp. 203–214. ISSN: 0166-8641. DOI: [10.1016/j.topol.2019.04.009](https://doi.org/10.1016/j.topol.2019.04.009) (cit. on pp. 17, 28, 49–52, 54).
- [Sin02] B. Sing. “Spektrale eigenschaften der Kolakoski-sequenzen”. In: *Diplomathesis, Universität Tübingen* (2002). DOI: <https://www.bb-math.com/bernd/pub.html#kolakoski> (cit. on pp. 18, 89, 91).
- [Sin10] B. Sing. “More Kolakoski sequences”. In: *Integers* 11B (2010), A14 (cit. on pp. 18, 86, 89, 91).
- [Sza18] M. Szabados. “Nivat’s conjecture holds for sums of two periodic configurations”. In: *SOFSEM 2018: theory and practice of computer science*. Vol. 10706. Lecture Notes in Comput. Sci. Springer, Cham, 2018, pp. 539–551. ISBN: 978-3-319-73117-9; 978-3-319-73116-2 (cit. on p. 16).
- [Wea89] W. D. Weakley. “On the number of C^∞ -words of each length”. In: *J. Comb. Theory A* 51.1 (1989), pp. 55–62. DOI: [10.1016/0097-3165\(89\)90076-9](https://doi.org/10.1016/0097-3165(89)90076-9) (cit. on pp. 18, 89, 90, 92, 94, 96, 106).

APPENDIX

A. Examples of morphisms for purely morphic sequences

A.1. Growing morphisms

Example A.1.1. Consider $\sigma_3 : 0 \mapsto 010, 1 \mapsto 11$. We first have $|\sigma_3^n(1)| = 2^n$ for all $n \geq 0$. We also have

$$|\sigma_3^{n+1}(0)| = 2|\sigma_3^n(0)| + |\sigma_3^n(1)| = 2|\sigma_3^n(0)| + 2^n$$

so a quick induction yields $|\sigma_3^n(0)| = n2^{n-1} + 2^n$ for all $n \geq 0$. Therefore σ_3 is polynomially divergent.

Example A.1.2. Consider $\sigma_4 : 0 \mapsto 0010, 1 \mapsto 11$. We have $|\sigma_4^n(1)| = 2^n$ and $|\sigma_4^n(0)| \geq 3^n$ for all $n \geq 0$. Therefore σ_4 is exponentially divergent.

A.2. Tame morphisms

Example A.2.1. Consider $\sigma_6 : 0 \mapsto 01010, 1 \mapsto 121, 2 \mapsto 2$. We have $\mathcal{B} = \{2\}$ so it is not growing, and we have $\mathcal{L}(\sigma) \cap \mathcal{B}^* = \{\varepsilon, 2\}$ so it is tame. Over $\hat{\mathcal{A}} = \{\langle 01 \rangle, \langle 10 \rangle, \langle 121 \rangle\} = \{a, b, c\}$, we have the new endomorphism

$$\hat{\sigma}_6 : a \mapsto ababa, b \mapsto cb, c \mapsto ccc.$$

We first have $|\hat{\sigma}_6^n(c)| = 3^n$ for all $n \geq 0$. We also have

$$|\sigma_6^{n+1}(b)| = |\sigma_6^n(b)| + |\sigma_6^n(c)| = |\sigma_6^n(b)| + 3^n$$

so a quick induction yields $|\sigma_6^n(b)| = \frac{3^{n+1}-1}{2}$ for all $n \geq 0$. Similarly, we have

$$|\sigma_6^{n+1}(a)| = 3|\sigma_6^n(a)| + 2|\sigma_6^n(b)| = 3|\sigma_6^n(a)| + 3^n + 1$$

so a quick induction yields $|\sigma_6^n(a)| = n3^{n-1} + \frac{3^{n+1}-1}{2}$ for all $n \geq 0$. Therefore $\hat{\sigma}_6$ is polynomially divergent and σ_6 is virtually polynomially divergent.

Example A.2.2. Consider $\sigma_7 : 0 \mapsto 0101010, 1 \mapsto 121, 2 \mapsto 2$. We have $\mathcal{B} = \{2\}$ so it is not growing, and we have $\mathcal{L}(\sigma) \cap \mathcal{B}^* = \{\varepsilon, 2\}$ so it is tame. Over $\hat{\mathcal{A}} = \{\langle 01 \rangle, \langle 10 \rangle, \langle 121 \rangle\} = \{a, b, c\}$, we have the new endomorphism

$$\hat{\sigma}_7 : a \mapsto abababa, b \mapsto cb, c \mapsto ccc.$$

A. Examples of morphisms for purely morphic sequences – A.3. Wild morphisms

Similarly to σ_6 , we have $|\hat{\sigma}_7^n(c)| = 3^n$ and $|\sigma_7^n(b)| = \frac{3^n+1}{2}$ for all $n \geq 0$. We also have $|\sigma_7^n(b)| \geq 4^n$, therefore $\hat{\sigma}_7$ is exponentially divergent and σ_7 is virtually exponentially divergent.

A.3. Wild morphisms

Example A.3.1. Consider $\sigma_8 : 0 \mapsto 01, 1 \mapsto 12, 2 \mapsto 2$. We have $\mathcal{B} = \{2\}$ so it is not growing, and $2^n \sqsubset \sigma_8^n(1)$ for all $n \geq 0$ so σ_8 is wild.

A.4. Periodic morphisms

Example A.4.1. Consider $\sigma_9 : 0 \mapsto 01, 1 \mapsto 01$. It is clearly primitive, and we have $\sigma^\infty(0) = (01)^\omega$.

Example A.4.2. Consider $\sigma_{10} : 0 \mapsto 010, 1 \mapsto 1$. We have $\mathcal{B} = \{1\}$ so it is not growing, and we have $\mathcal{L}(\sigma) \cap \mathcal{B}^* = \{\varepsilon, 1\}$ so it is tame. Over $\hat{\mathcal{A}} = \{\langle 010 \rangle\} = \{a\}$, we have the new endomorphism

$$\hat{\sigma}_{10} : a \mapsto aa$$

which is primitive so σ_{10} is virtually quasi-uniform. Lastly, we have $\sigma^\infty(0) = \tau(a^\omega) = (01)^\omega$.

Example A.4.3. Consider $\sigma_{11} : 0 \mapsto 01, 1 \mapsto 1$. We have $\mathcal{B} = \{1\}$ so it is not growing, and $1^n \sqsubset \sigma^n(0)$ so it is wild. Lastly, we have $\sigma^\infty(0) = 10^\omega$.