Canons rythmiques et pavages modulaires

Hélianthe Caure

Canons rythmiques et pavages modulaires

Hélianthe CAURE

UNIVERSITÉ PIERRE ET MARIE CURIE

École doctorale Informatique, Télécommunications et Électronique (Paris)

Institut de Recherche et Coordination Acoustique/Musique UMR STMS 9912

Pour obtenir le grade de

DOCTEUR de l'UNIVERSITÉ PIERRE ET MARIE CURIE

Spécialité **Mathématiques**

soutenue le 24 juin 2016 devant le jury composé de :

Jean-Paul ALLOUCHE	*Directeur*	Université Pierre et Marie Curie
Emmanuel AMIOT	*Examinateur*	CPGE Perpignan
Moreno ANDREATTA	*Directeur*	Université Pierre et Marie Curie
Jean-Claude BAJARD	*Examinateur*	Université Pierre et Marie Curie
Yann BUGEAUD	*Rapporteur*	Université de Strasbourg
Lucia DI VIZIO	*Examinatrice*	Université de Versailles-St Quentin
Michel RIGO	*Rapporteur*	Université de Liège

CANONS RYTHMIQUES ET PAVAGES MODULAIRES

HÉLIANTHE CAURE

UMR 9912 (STMS)
Université Pierre et Marie Curie, Paris VI
IRCAM, Music Representations team

Mai 2016 – version 0.0

DIRECTEURS :
Moreno Andreatta
Jean-Paul Allouche

Table des matières

Table des figures

1

Introduction

Ce mémoire est une contribution à l'étude des canons modulo p. De nombreux outils mathématiques et informatiques ont été employés pour l'étude des canons rythmiques mosaïques. La recherche récente s'est particulièrement attachée à trouver les canons sans périodicité interne, dits de Vuza : KOLOUNTZAKIS et MATOLCSI (2009) utilisent des méthodes exhaustives pour obtenir des listes de canons de Vuza ; ils ont aussi présenté des perturbations sur les valeurs des canons périodiques permettant de les rendre apériodiques (KOLOUNTZAKIS et collab, 2003) ; ou encore il existe des transformations conservant les propriétés d'apériodicités qui permettent de créer des grands canons de Vuza (AMIOT, 2005).

La notion de canons modulo p a été introduite par AMIOT (2004) afin d'utiliser des résultats de théorie galoisienne sur les corps finis pour étudier les canons mosaïques (AMIOT, 2005). Ils ont d'abord été considérés comme un cas particulier de canons, au même titre que les *perfect square tilings* ou les pavages par rétrogradation (JOHNSON, 2001). Puis un résultat sur les polynômes définis sur les corps finis a permis d'obtenir le premier résultat significatif 4.2.1 sur les canons modulo p. Leur intérêt principal est que tout motif rythmique permet de construire un canon rythmique modulo p, et que ce canon peut être compact modulo 2, c'est à dire que le motif pave un intervalle fini. Cette possibilité de paver modulo p a pourtant été très peu exploitée. Musicalement, les canons mosaïques ont été utilisés par des compositeurs (comme Lanza en 2001 dans *Aschenblume* pour neuf instruments, TANGIAN (2003), BLOCH (2006), LÉVY (2011), JOHNSON (2011), ou encore très récemment par Sébastien Roux dans *Cinq canons de Vuza*), mais l'étude des canons modulo p est trop récente pour que des musiciens aient pu s'y intéresser. Mathématiquement aussi, ces canons ont été très peu étudiés depuis leur introduction. Cette thèse est donc le premier travail à notre connaissance s'intéressant en profondeur au pavage modulo p, et présente des résultats variés à leur sujet, en utilisant différentes méthodes mathématiques et informatiques.

Un canon rythmique mosaïque peut se définir comme la donnée de deux motifs rythmiques. Le premier est un rythme fini, et le second représente les temps auxquels le premier rythme commence. Ces rythmes sont tels qu'à toute pulsation du temps on peut entendre un et exactement un *onset*, ou point d'attaque d'une note, du premier rythme. Un tel canon peut aussi se lire comme un pavage discret par

translation de la ligne. Les canons modulo p suivent la même définition, à ceci près qu'à chaque pulsation, on veut entendre un et exactement un *onset* modulo p.

Après une partie rappelant les notations et les définitions qui seront utilisées tout au long de cette thèse, le chapitre 3 présente un état de l'art en deux parties. La première choisit une orientation mathématique, tandis que la seconde a une lecture plus musicale. Cette dualité permet d'aborder le problème des canons rythmiques avec deux visions différentes. La section 3.1 s'intéresse particulièrement au sujet riche des pavages. Elle décrit rapidement leur étude générale, puis le cas moins étudié du pavage en une dimension. Sont ensuite rappelés les travaux utilisant des résultats d'algèbre pour l'étude du pavage, et enfin nous présentons une courte histoire des calculs de complexité. La section 3.2 est un discours présentant des résultats mathémusicaux. Après un exposé sur la modélisation du rythme, elle présente l'état des connaissances sur les canons rythmiques mosaïques. Enfin, le logiciel de programmation visuel Open Music est rapidement décrit. Les canons modulo p ayant été très peu étudiés à ce jour, ils ne sont pas (ou très peu) abordés dans cet état de l'art, mais tous les chapitres suivants de la thèse leur sont consacrés.

1.1 Complexité des problèmes de pavages

Le chapitre 4 s'intéresse aux aspects de complexité informatique du pavage. Sa première partie 4.1 traite des problèmes de pavage et de leur décidabilité et complexité. On y définit plusieurs types de pavage par translations. Le problème de pavage le plus large, c'est-à-dire d'un ensemble quelconque par des tuiles quelconques, est indécidable et justifie que nous ne nous intéressions qu'à des pavages finis. Notamment le problème de Square Tiling est présenté comme un modèle de NP-complétude en 2 dimensions. Puis le chapitre s'intéresse au pavage de la ligne par translation, ce sur quoi porte le reste de la thèse. Un théorème de Lagarias et Wang prouve que tout pavage de $\mathbb{R}$, puis de $\mathbb{Z}$, est périodique, et ainsi que les problèmes de pavage de la ligne sont décidables. Cela justifie notre définition de canons rythmiques mosaïques, qui peuvent être vus tout au long de la thèse comme des pavages périodiques de $\mathbb{Z}$ par translation avec une tuile. Ce problème de pavage est lié à un problème, nommé DIFF, sur les différences entre ensembles d'entiers, que nous montrons être NP-complet, grâce au problème des stables de théorie des graphes. Ce lien prouve que le problème de pavage par une tuile si la période est donnée est bien sûr NP, mais serait au plus NP-complet. Cependant, si la période n'est pas connue, l'état actuel des conjectures laisse à croire que le problème de pavage par une tuile serait polynomial. La

dernière partie 4.2 de ce chapitre traite toujours de la complexité du pavage périodique de $\mathbb{Z}$ par translation d'une tuile, mais en relâchant la contrainte de pavage à celle d'un recouvrement contrôlé. En effet, on ne s'intéresse plus à obtenir un *onset* sur chaque pulsation, mais un *onset* modulo p. Grâce à un théorème de théorie des corps, on peut montrer que tout motif rythmique peut paver $\mathbb{Z}$ de cette manière. Comme les coefficients d'un polyrythme modulo 2 sont des 0 ou des 1, on peut considérer tout polyrythme modulo 2 comme un rythme, et cela implique que tout motif peut paver modulo 2 de façon compacte. On dérive de cette propriété un algorithme glouton qui permet de paver avec un motif rythmique modulo 2 un certain $\mathbb{Z}_N$ en temps linéaire en N. Puis on montre que cet algorithme est optimal dans le sens où il permet d'obtenir la plus petite période N d'un tel pavage. Cet algorithme est le seul résultat qui est utilisé d'un chapitre à l'autre de la thèse. Autrement, tous les chapitres peuvent être lus de manière indépendante. C'est aussi grâce à cet algorithme que quasiment tous les résultats de cette thèse se trouvent dans les conditions où $p = 2$. En effet, que cela soit grâce à la facilité d'exploration des canons qu'offre cet algorithme en temps linéaire, ou bien en utilisant la propriété de compacité, il est plus simple de travailler modulo 2. Même si certains résultats n'ont besoin que de la compacité, et qu'il est possible pour $p > 2$ d'avoir un canon compact, on préfère se contenter d'un résultat présenté modulo 2, où l'on sait que chaque motif rythmique peut paver de façon compacte.

1.2 De la couverture au pavage : étude des donsets

Le chapitre 5 est dédié à la différence principale entre canons classiques, et canons modulo p. C'est-à-dire à la différence entre couverture et pavage dans ce cas. Pour un canon modulo p donné, on définit son ensemble de *donsets*, néologisme pour exprimer une superposition de d *onsets* à une pulsation donnée, comme le multi-ensemble des pulsations où des *onsets* sont joués en même temps, avec pour multiplicité le nombre de fois divisé par p où ils sont joués strictement plus qu'une fois. La section 5.1 commence par les définir, et on y remarque qu'il y a un lien direct entre un motif rythmique et le multi-ensemble des donsets de son canon minimal, plus direct que de les obtenir avec l'algorithme glouton. Cependant on remarque, de manière triviale avec la dualité par exemple, qu'il n'y a pas unicité des canons rythmiques permettant d'obtenir un multi-ensemble de donsets donné. Ainsi, pour trouver cette fonction entre donset et motif rythmique, il faut commencer par quotienter les motifs rythmiques qui permettent d'obtenir les mêmes donsets. C'est ce à quoi s'attache la section 5.2, dans le cas où $p = 2$. On y présente deux transformations non triviales qui, partant d'un canon modulo 2, per-

mettent d'engendrer deux canons modulo 2 plus grands, qui sont différents, et qui ont le même multi-ensemble de donsets. Ces résultats ont une limite, puisque le chapitre se termine en présentant un exemple de deux canons distincts ayant le même multi-ensemble de donsets, semblant venir de la seconde transformation, mais ne rentrant pas dans ses conditions d'application. Ainsi, nous ne savons pas à ce jour associer plus simplement qu'en le calculant un multi-ensemble de donsets aux motifs rythmiques qui permettent de l'obtenir.

1.3 Implémentation en OpenMusic

La chapitre 6 se distingue des autres par le fait qu'il est une application pratique des résultats des chapitres précédents. Comme il a été noté plus tôt, les canons rythmiques mosaïques ont souvent été utilisés comme outils de composition dans des pièces récentes. Mais la nouveauté des résultats sur les canons modulo 2, et le manque de visibilité auprès des compositeurs entraînent que ces canons sont restés dans les mains des mathématiciens. Ce petit chapitre présente donc une implémentation des résultats dans le langage de programmation visuel Open Music. La section 6.1 implémente l'algorithme glouton du chapitre 4, et peut permettre aux compositeurs de créer facilement des canons modulo 2 à exploiter. La seconde section 6.2 présente quant à elle l'implémentation des deux résultats de transformations du chapitre 5 qui permettent d'obtenir des canons différents avec les mêmes donsets. Ces résultats d'implémentations ont été présentés à la conférence internationale ICMC 2014 devant un public d'informaticiens et musiciens, et nous espérons qu'ils seront prochainement utilisés à des fins artistiques.

1.4 Accélération des algorithmes de pavages modulo 2 et similitudes des motifs des entrées

Le chapitre 7 est le chapitre contenant les résultats principaux de cette thèse. Il démontre plusieurs théorèmes nouveaux sur le pavage modulo 2 avec des motifs d'une certaine forme $A(n) = \{0, 1, n\}$ pour $n \geq 2$. La première partie 7.1 définit de nouvelles notions sur la pavage modulo p en général. On y introduit la notion de sous-couverture, qui est la sous-partie la plus petite qui permet de décrire entièrement la différence entre une couverture (modulo p) et un pavage de même taille. On y définit ensuite la notion de remplissage de ces sous-couvertures, et cela est lié au fonctionnement de l'algorithme 1 glouton. En effet, l'algorithme linéaire du chapitre 4 se comporte sur

des morceaux de taille n comme un remplissage successif de sous-couvertures. Ce lien vient de la forme de nos motifs $A(n)$ qui sont constitués de deux *onsets* consécutifs, d'un temps de silence, puis d'un dernier *onset*. L'algorithme sur ces motifs cherche à remplir cet espace de silence de manière régulière, jusqu'à ce qu'il rencontre le dernier *onset* du motif $A(n)$. Ce faisant, il crée des sous-couvertures de taille n. Cette idée de remplissage répétitif de ces espaces de silence permet de remarquer des régularités sur le motif des entrées, et d'accélérer l'algorithme glouton que nous avions précédemment. Ainsi, pour mettre en évidence ces régularités, sont définis des tableaux $T(n)$ contenant un mot décrivant le motif des entrées. Ces tableaux ont n colonnes, ainsi, lorsque le mot des entrées est écrit ligne à ligne, on peut lire sur chaque ligne le remplissage d'un espace de silence.

La première section 7.2 s'intéresse au cas particulier des motifs $A(2^k)$, car la période du canon modulo 2 compact est bien plus petite que celle du canon avec $A(n)$ avec $n \neq 2^k$. Toute cette section est dédiée à la démonstration d'un résultat permettant de paver en temps logarithmique plutôt que linéaire avec ces motifs. En effet, on parvient à construire le tableau $T(2^{k+1})$ comme une réplique de plusieurs occurrences du tableau $T(2^k)$ et de certaines de ses transformations. La preuve étant constructive, nous en déduisons ensuite les premiers résultats de dénombrement sur les canons modulo 2.

La partie 7.3 cherche à lier les motifs de la forme $A(2n)$ avec ceux de la forme $A(2n+1)$. La grande majorité de cette section est consacrée à la démonstration du théorème prouvant que le même type de construction par récurrence est possible pour les tableaux de la forme $T(2^k+1)$. Elle s'appuie sur le résultat précédent, et démontre de nombreux liens entre le motif des entrées pour paver avec un motif, et celui pour paver avec des transformations de ce motif. On parvient à montrer que le tableau $T(2^k+1)$ est (à quelques transformations près) le miroir du tableau $T(2^k)$, c'est-à-dire le symétrique du tableau selon l'axe vertical. Ainsi, on peut en déduire les mêmes résultats de dénombrement pour les motifs de la forme $A(2^k+1)$. On remarque ensuite le lien étroit entre un tableau $T(2n)$ et le miroir du tableau $T(2n+1)$ dans un cadre plus général.

La dernière section 7.4 s'intéresse elle aux liens entre les tableaux $T(2n)$ et $T(2^k)$. Les premiers ne peuvent toujours s'obtenir qu'avec l'aide de l'algorithme du chapitre 4 qui est linéaire en la taille du canon, tandis que les résultats précédents permettent d'obtenir les seconds en temps logarithmique. Le premier résultat de cette section énonce que dans les tableaux $T(2^k)$, chaque ligne ℓ contient un mot qui est la concaténation successive de son préfixe de taille $\mathcal{G}(\ell) = 2^{\lfloor \log_2(\ell) \rfloor + 1}$: son gabarit. Cela se démontre en créant une récurrence sur les sous-tableaux constitués des préfixes des lignes ayant les mêmes gabarits. Ces sous-tableaux ont la particularité d'avoir un nom-

bre pair de 1 sur chaque ligne sauf la dernière, ce qui permet d'en déduire le résultat.

On conjecture ensuite ce résultat de parité sur le nombre de 1 des préfixes de taille $\mathcal{G}(\ell)$ des lignes ℓ des tableaux $T(2n)$ dans un cas plus général. De cette conjecture se déduisent de nombreuses conjectures conditionnelles sur lesquelles se termine ce chapitre. On en déduirait que les $2^{\nu_2(m-n)}$ premières lignes des tableaux $T(m)$ et $T(n)$ sont identiques, où ν_2 est la valuation 2-adique ; puis que les lignes suivantes sont conjuguées l'une de l'autre. Ces derniers résultats laissent à penser que des transformations possibles entres les $T(2^k)$ et les $T(2n)$ permettraient d'accélérer le temps de pavage pour tous les motifs de la forme $A(n)$.

Le chapitre 3 constitue une synthèse des résultats antérieurs cette thèse, d'un point de vue mathématique puis musical, et sont reliés aux contributions de celle-ci. Le chapitre 4 revient sur la complexité du pavage et montre que le problème du pavage modulo p est immédiat. On y présente un algorithme glouton qui est utilisé tout au long de la thèse, et dont l'optimalité est démontrée dans l'article en cours de publication (CAURE, 2015). Le chapitre 5 présente la différence entre canons mosaïques et canons modulo p, et montre deux transformations permettant de créer des canons différents aux donsets identiques. Ces résultats sont publiés dans (CAURE et collab, 2014), ainsi que leur implémentation, avec celle de l'algorithme glouton qui son présentées dans le chapitre 6. Le dernier chapitre 7 est le plus important de cette thèse. La première section est en cours de publication dans (CAURE, 2015), et les autres sections sont inédites. On y démontre des accélérations des algorithmes de pavage modulo 2 grâce à des constructions par récurrence, et les premiers résultats de dénombrement sur les canons modulo 2.

2

Notations et définitions

2.1 Notations

Nous allons définir quelques notations qui peuvent ne pas être canoniques et qui seront utilisées tout au long de cette thèse.

On notera $\mathbb{R}$ l'ensemble des nombres réels, $\mathbb{Q}$ l'ensemble des nombres rationnels, $\mathbb{Z}$ est l'ensemble des entiers relatifs et $\mathbb{N} = \{0, 1, 2, \ldots\}$ est l'ensemble des entiers positifs.

L'ensemble $\mathbb{N}^* = \{1, 2, 3, \ldots\}$ est celui des entiers strictement positifs. Si $N \in \mathbb{N}^*$, on note $\mathbb{Z}_N$ le groupe cyclique à N éléments. Ses éléments sont les classes d'équivalences modulo N, et on les désigne avec l'abus de notation classique comme un entier compris entre 0 et $N-1$.

Si p est un nombre premier, alors $\mathbb{Z}_p = \mathbb{F}_p$ est le corps fini à p éléments, et non pas l'anneau des entiers p-adiques comme le voudrait la notation habituelle de théorie des nombres.

Si E est un ensemble, alors on note $\mathcal{P}(E)$ l'ensemble des parties de E. On note $E[X]$ l'ensemble des polynômes à coefficients dans E. Si E est un anneau, alors $E[X]$ l'est aussi (on utilisera notamment les anneaux $\mathbb{Z}[X]$ ou $\mathbb{F}_p[X]$) mais on peut aussi considérer l'ensemble $E[X]$ si E n'a pas de structure particulière (on utilisera souvent l'ensemble $\{0,1\}[X]$). Si $d \in \mathbb{N}^*$, on note E^d l'ensemble des d-uplets d'éléments de E. Si E est fini, on note $|E|$ son cardinal.

Pour $n_1, n_2, m \in \mathbb{Z}$ tels que $n_1 < n_2 < m$ et tels qu'il existe $l \in \mathbb{N}^*$, avec $m = n_1 + l(n_2 - n_1)$, on note $\{n_1, n_2, \ldots, m\}$ le sous-ensemble de $\mathbb{Z}$ constitué des éléments de la forme $n_1 + k(n_2 - n_1)$ pour k allant de 0 à l. Pour $n, m \in \mathbb{Z}$, on note $\{n, \ldots, m\} = \{n, n+1, \ldots, m\}$.

Si $r \in \mathbb{R}$, on note $\lfloor r \rfloor$ le plus grand entier inférieur à r.

Soit deux mots u, v sur un alphabet A, on note uv le mot concaténé.

Soit $k \in \mathbb{N}^*$, on note $(u)^k = u^k = u \cdots u$ le mot u concaténé k fois.

Soit $n \in \mathbb{N}$ et $k \in \mathbb{N}^*$, si u est un mot sur un alphabet A, on note $u[n]$ la $(n+1)$ème lettre de u, et $u[n, \ldots, n+k-1] = u[n] \cdots u[n-k+1]$ le sous-mot de u de taille k commençant par la lettre $u[n]$.

Par exemple si $u = abcdef$ alors $u[0] = a$, $u[4] = e$ et $u[0, \ldots, 3] = abcd$.

Si E est un ensemble fini, ces notations $E[n]$ et $E[n, \ldots, n+k-1]$ s'appliquent en supposant qu'on considère le mot dont les lettres sont les éléments de E concaténés dans l'ordre. Par exemple si $B = \{a, b, c, d, e, f\}$, alors de même $B[0] = a$, $B[4] = e$ et $B[0, \ldots, 3] = abcd$.

Si u est un mot fini, on note $|u|$ sa longueur qui est son nombre de lettres.

On rappelle que le produit de convolution de deux fonctions $f, g : \mathbb{Z}_N \to \mathbb{Z}$ est définie pour tout $n \in \mathbb{Z}_N$ par $f \star g(n) = \sum_{k \in \mathbb{Z}_N} f(k)g(n-k)$.

2.2 Définitions

Définition 2.2.1. Si $E \in \mathcal{P}(\mathbb{N})$, on note sa fonction indicatrice $\mathbb{1}_E : \mathbb{N} \to \{0,1\}$ définie pour tout $n \in \mathbb{N}$ par $\mathbb{1}_E(n) = 1$ si $n \in E$ et $\mathbb{1}_E(n) = 0$ sinon.

Définition 2.2.2. Un multi-ensemble E est une extension de la notion d'ensemble, où les éléments peuvent être répétés plusieurs fois. Si $e \in E$ un multi-ensemble, sa multiplicité $m(e)$ est le nombre (fini) de fois où e est répété dans E.

Définition 2.2.3. Si E est un multi-ensemble dont les éléments sont dans $\mathbb{N}$, alors on note $\mathbb{1}_E : \mathbb{N} \to \mathbb{N}$ la fonction définie pour tout $n \in \mathbb{N}$ par $\mathbb{1}_E(n) = m(n)$ si $n \in E$ et $\mathbb{1}_E(n) = 0$ sinon.

Exemple 2.2.4. On note $E = \{1,2,3,3,3,4,5,6\}$ un multi-ensemble, alors $\mathbb{1}_E(0) = 0$ et $\mathbb{1}_E(3) = 3$.

Remarque 2.2.5. Que E soit un ensemble ou un multi-ensemble dont les éléments sont dans $\mathbb{N}$, la notation $\mathbb{1}_E$ reste cohérente.

Définition 2.2.6. Soit $N \in \mathbb{N}^*$ on note $\pi_N : \mathbb{Z} \to \mathbb{Z}_N$ la fonction qui à $n \in \mathbb{Z}$ associe sa classe d'équivalence modulo N. C'est-à-dire que $\pi_N(n)$ vaut le reste la division euclidienne de n par N.

Si $E \in \mathcal{P}(\mathbb{Z})$ fini, on note $\Pi_N(E) = \{\pi_N(e) | e \in E\}$.

Définition 2.2.7. Si A, B sont deux ensembles inclus dans un groupe additif, on note $A+B$ le multi-ensemble $A+B = \{a+b | a \in A, b \in B\}$. Si $A+B$ est un ensemble, alors on dit que la somme est directe et on note $A+B = A \oplus B$.

Exemple 2.2.8. Si $A = \{0,1,3,6\}$, $B = \{0,4\}$ et $C = \{0,2\}$ des sous-ensembles de $\mathbb{Z}$, alors $A+B = \{0,1,3,4,5,6,7,10\} = A \oplus B$ tandis que $A+C = \{0,1,2,3,3,5,6,8\}$ n'est pas une somme directe.

Définition 2.2.9. Soit A un alphabet $A \in \mathcal{P}(\mathbb{N})$ qui contient la lettre 0. Alors on dit qu'un mot u sur l'alphabet A est presque nul si la suite $(u[n])_{n \in \mathbb{N}}$ est presque nulle, c'est-à-dire nulle à partir d'un certain rang.

Proposition 2.2.10. *On peut construire des bijections entre les parties finies de* $\mathbb{N}$*, les polynômes dans l'ensemble* $\{0,1\}[X]$ *et les mots qui se terminent par* 1 *sur l'alphabet* $\{0,1\}$.

Démonstration. Soit A un sous-ensemble fini de $\mathbb{N}$, on lui associe le polynôme $A(X) = \sum_{a\in A} X^a$ qui a bien ses coefficients dans $\{0,1\}$. On lui associe le mot de taille $\max(A)$ défini par $u[n] = \mathbb{1}_A(n)$. Cela définit évidemment deux bijections. □

Remarque 2.2.11. Comme on l'a vu en 3.2.1, un motif rythmique peut se définir comme un ensemble fini d'entiers contenant 0. Suivant les bijections de la proposition 2.2.10, un motif rythmique peut alors se définir de manière équivalente comme un polynôme dans l'ensemble $\{0,1\}[X]$ qui ne s'annule pas en 0, ou comme un mot fini qui commence et se termine par 1. Cependant, nous considérerons à partir de maintenant que nous pouvons ajouter autant de 0 que désiré derrière ce dernier 1, et que le mot représente toujours le même motif rythmique. Cela se justifie bien car les pulsations continuent d'exister, même si aucun *onset* n'est joué.

Définition 2.2.12. Un motif rythmique est noté de façon équivalente comme

1. un ensemble fini A d'entiers naturels contenant 0
2. un polynôme $A(X) \in \{0,1\}[X]$ qui ne s'annule pas en 0
3. un mot fini ou presque nul a de l'alphabet $\{0,1\}$ qui commence par 1.

On appellera *onset* les éléments d'un motif rythmique.

Exemple 2.2.13. Si $A = \{0,1,3,6\}$ est un motif rythmique, on peut le représenter aussi par le polynôme $A(X) = 1 + X + X^3 + X^6$ ou par les mots $a = 1101001 = 1101001000 = 11010010000\cdots$.

La même définition d'un polyrythme est possible sous trois représentations équivalentes.

Proposition 2.2.14. *On peut construire des bijections entre les multi-ensembles d'entiers naturels, les polynômes de* $\mathbb{N}[X]$ *et les mots finis de l'alphabet* $\mathbb{N}$ *qui se terminent par une lettre différente de* 0.

Démonstration. Soit A un multi-ensemble d'entiers naturels, on lui associe le polynôme $\sum_{a\in A} \mathbb{1}_A(a)X^a$. On lui associe aussi le mot de taille $\max(A)$ défini par $u[n] = \mathbb{1}_A(n)$ qui est bien sur l'alphabet $\mathbb{N}$. Cela définit évidemment deux bijections. □

Remarque 2.2.15. Nous avons vu en 3.2.1 que comme pour les rythmes, on peut toujours considérer les polyrythmes comme des multi-ensembles d'entiers naturels contenant 0. Par la propriété 2.2.14, on peut les voir de manière équivalente comme des polynômes à coefficients entiers qui ne s'annulent pas en 0, ou comme des mots finis ayant leurs lettres dans $\mathbb{Z}$ ne se terminant pas et ne commençant pas par un 0. De la même manière que pour les rythmes, nous considérons qu'après la dernière lettre non nulle, concaténer autant de 0 que désiré à un mot ne change pas le motif représenté.

Définition 2.2.16. Un motif polyrythmique est noté de façon équivalente comme

1. un multi-ensemble fini A d'entiers naturel contenant 0
2. un polynôme $A(X) \in \mathbb{Z}[X]$ qui ne s'annule pas en 0
3. un mot fini ou presque nul a sur l'alphabet $\mathbb{Z}$ qui ne commence pas par 0.

Exemple 2.2.17. Si $A = \{0,1,2,2,2,2,3,3,4,4,4,6\}$ est un multi-ensemble représentant un polyrythme, alors il est aussi représenté par le polynôme $A(X) = 1 + X + 4X^2 + 2X^3 + 3X^4 + X^6$ et par le mot $a = 1142301 = 1142301000 = 114230100000\cdots$

À un polyrytme, on peut associer un polyrythme modulo p, pour tout p premier. Il s'agit du même polyrythme, où le nombre d'*onsets* joués à chaque pulsation est considéré modulo p.

Définition 2.2.18. À un polyrythme, on associe son polyrythme modulo p :

1. Si A est un mutliensemble, A_p a les mêmes éléments que A avec leur multiplicité considérée modulo p
2. Si $A(X) \in \mathbb{Z}[X]$ est un polynôme, le polynôme $A_p(X)$ $A_p(X) \in \{0,\dots,p-1\}[X]$ est la somme des mêmes monômes et leurs coefficients sont considérés modulo p
3. Si a est un mot de $\mathbb{Z}$, a_p est le mot de l'alphabet $\{0,\dots,p-1\}$ tel que sa $i$ème lettre vaut le représentant de la classe d'équivalence modulo p de la $i$ème lettre de a.

Exemple 2.2.19. Si on reprend le polyrythme de l'exemple 2.2.17 $A = \{0,1,2,2,2,2,3,3,4,4,4,6\}$, alors son polyrythme modulo 3 peut être représenté de manière équivalente par le multi-ensemble $A_3 = \{0,1,2,3,3,6\}$, le polynôme $A_3(X) = 1 + X + X^2 + 2X^3 + X^6$ ou par le mot $a_3 = 1112001 = 1112001000 = 11200100000\cdots$

Un canon rythmique est la donnée de deux motifs rythmiques. L'un représente le motif A qui sera translaté dans le temps, appelé motif des *onsets*. Le second sera le motif B dit des entrées, c'est-à-dire qu'il représente les pulsations où sont joués les translatés du motif des *onsets*. On dit qu'un canon rythmique est mosaïque lorsque la donnée de ces deux motifs rythmiques permet de couvrir les pulsations avec un et un seul *onset* du motif des *onsets* ou d'un de ses translatés. Comme il a été montré (LAGARIAS et WANG, 1996b) qu'un tel canon est toujours périodique, on s'intéresse seulement à remplir toutes les pulsations d'une boucle de taille N. Un canon mosaïque de taille N est tel qu'il existe un couple de motifs rythmiques (A,B) tel que tout $n \in \mathbb{Z}_N$ peut s'écrire de manière unique comme $n = a + b \mod N$ avec $a \in A, b \in B$. Cela revient donc à écrire $\mathbb{Z}_N = \Pi_N(A) \oplus \Pi_N(B)$. Dans un tel cas, A et B satisfont évidemment à $|A| = |\Pi_N(A)|$, et si cela ne porte pas à confusion, on notera alors $A \oplus B = \mathbb{Z}_N$. Il a été

montré dans (AMIOT, 2011) que des définitions équivalentes existent avec les notations analogues des motifs rythmiques :

Définition 2.2.20. Soit $N \in \mathbb{N}^*$, on dit que le couple de motifs rythmiques (A, B) est un canon rythmique mosaïque de $\mathbb{Z}_N$ si de manière équivalente :

1. $A \oplus B = \mathbb{Z}_N$
2. $A(X) \cdot B(X) = 1 + X + \cdots + X^{N-1} \mod (X^N - 1)$
3. $\mathbb{1}_A \star \mathbb{1}_B = \mathbb{1}_{\mathbb{Z}_\mathbb{N}}$ où $\star$ est le produit de convolution.

On pourra aussi dire que A pave $\mathbb{Z}_N$ avec les entrées B.

Exemple 2.2.21. Le couple de motifs rythmiques $A = \{0, 1, 3, 6\}$ et $B = \{0, 4\}$ pavent $\mathbb{Z}_8$. En effet :

1. $\Pi_8(A) + \Pi_8(B) = \{0, 1, 3, 6, 4, 5, 7, 2\} = \Pi_8(A) \oplus \Pi_8(B) = \mathbb{Z}_8$
2. $A(X) \cdot B(X) = 1 + X + \cdots + X^7 \mod (X^8 - 1)$
3. Pour tout $k \in \mathbb{Z}_8$, $\mathbb{1}_A \star \mathbb{1}_B(k) = 1$.

Un canon est dit compact si les motifs (A, B) pavent un intervalle de $\mathbb{Z}$, c'est-à-dire s'il n'y a pas besoin de projection dans $\mathbb{Z}_N$:

Définition 2.2.22. (A, B) est un canon compact de $\mathbb{Z}_N$ si $A + B = \{0, \ldots, N-1\}$.

Exemple 2.2.23. Le canon de l'exemple 2.2.21 n'est pas compact, tandis que $(\{0, 1, 4, 5\}, \{0, 2\})$ est un canon compact de $\mathbb{Z}_8$.

La notion de canon modulo p a été introduite par AMIOT (2004). Elle est inspirée par la notation polynomiale d'un canon rythmique :

$$A(X) \cdot B(X) = 1 + X + \cdots + X^{N-1} \mod (X^N - 1).$$

Les deux motifs rythmiques $A(X)$ et $B(X)$ sont dans $\{0, 1\}[X]$ qui n'est pas un anneau. Ainsi, pour s'assurer que le produit de ces polynômes reste dans un espace stable, il considère les motifs rythmiques comme des polynômes de $\mathbb{F}_2[X]$. Alors, on dit qu'un couple de motifs rythmiques A, B forment un canon rythmique mosaïque modulo 2 de $\mathbb{Z}_N$ si

$$A(X) \cdot B(X) = 1 + X + \cdots + X^{N-1} \mod (X^N - 1, 2)$$

Puis la notion s'étend aisément pour tout p premier en considérant les motifs rythmiques comme des polynômes dans $\mathbb{F}_p[X]$.

Alors, un couple de motifs rythmiques (A, B) forme un canon modulo p de $\mathbb{Z}_N$ si le motif A et ses translatés joués aux entrées de B permettent d'avoir un et un seul *onset* modulo p sur chaque pulsation de $\mathbb{Z}_N$, contrairement à un canon classique qui ne permettait exactement qu'un seul *onset* sur chaque pulsation. Ainsi, si on note le multi-ensemble $C = \Pi_N(A) + \Pi_N(B)$, alors (A, B) est un canon de $\mathbb{Z}_N$ modulo p si le multi-ensemble C_p modulo p est tel que $C_p = \mathbb{Z}_N$.

Définition 2.2.24. Le couple de motifs rythmiques (A, B) est un canon de $\mathbb{Z}_N$ modulo p si de manière équivalente

1. $(\Pi_N(A) + \Pi_N(B))_p = \mathbb{Z}_N$
2. $A(X) \cdot B(X) = 1 + X + \cdots + X^{N-1} \mod (X^N - 1, p)$.

Exemple 2.2.25. Si $A = \{0, 1, 3\}$ et $B = \{0, 2, 3\}$, alors
$C = A + B = \{0, 1, 2, 3, 3, 3, 4, 5, 6\}$, donc $C_2 = \{0, 1, 2, 3, 4, 5, 6\}$, et (A, B) est un canon modulo 2 de $\mathbb{Z}_7$.
On peut aussi vérifier que

$$\begin{aligned} C(X) = A(X) \cdot B(X) &= (1 + X + X^3)(1 + X^2 + X^3) \\ &= 1 + X + X^2 + 3X + X^4 + X^5 + X^6 \end{aligned}$$

et son polynôme associé modulo 2 vaut bien

$$C_2(X) = 1 + X + X^2 + X^3 + X^4 + X^5 + X^6.$$

De la même façon que pour les canons classiques, on peut définir la notion de canon modulo p compact :

Définition 2.2.26. Un canon (A, B) modulo p de $\mathbb{Z}_N$ est dit compact si $A(X) \cdot B(X) = 1 + X + \cdots + X^{N-1} \mod (p)$, sans besoin de projection dans $\mathbb{Z}_N$.

Exemple 2.2.27. Le canon modulo 2 de l'exemple 2.2.25 est compact.

3

État de l'art et contexte historique

3.1 Problèmes du pavage en mathématiques et informatique

3.1.1 *Le pavage général*

Lorsque l'on parle de pavage en mathématiques, on pense à deux catégories de problèmes. D'abord à celui touchant les aspects géométriques : un pavage peut être vu comme une partition d'un espace, souvent le plan (FRETTLÖH et HARRISS, 2014), et on peut retrouver des traces de telles constructions dès le Xe siècle, appliquées à l'architecture du monde arabe (TENNANT et DHABI, 2003). La question est de savoir si avec un ensemble fini de formes -qu'on appelle tuiles- donné il est possible de remplir l'espace sous des opérations de translation (symétries, rotations... selon le problème). Le problème a été introduit par WANG (1965) qui a d'abord conjecturé que tout pavage du plan par un ensemble fini de tuiles finies est périodique. Mais cela a été rapidement infirmé par son élève BERGER (1966), puis par PENROSE (1979) présentant un pavage apériodique avec un ensemble de seulement 6 tuiles (réduites à 2) qui donnera le maintenant célèbre pavage de Penrose.

Paver le plan est donc un problème indécidable, et la seconde vision de ce dernier est celle des mathématiques discrètes et de l'informatique. Le pavage d'une région finie permet de mesurer la difficulté de nombreux autres problèmes. Même en n'utilisant que des motifs élémentaires, ce problème ayant une formulation très simple est extrêmement difficile à résoudre. Paver une région finie avec des polyominos verticaux ou horizontaux est un problème NP-complet (BEAUQUIER et collab, 1995), de même que le problème de Square Tiling (GAREY et JOHNSON, 1979), qui cherche à savoir si on peut paver un carré de taille donnée ayant ses frontières colorées avec des tuiles carrées ayant des couleurs sur chaque bord, en prolongeant les couleurs.

Les problèmes de pavage posent aussi des questions de combinatoire : GOLOMB (1966) s'interroge sur la classification des polyominos pouvant paver des formes finies ou non, et KERSHNER (1968) pense énumérer à tort tous les polyèdres convexes pouvant paver le plan, ce qui est en vérité un problème toujours ouvert.

Malgré la difficulté mathématique de tels problèmes, leurs formulations simples et élégantes en font des outils idéaux de pédagogie

et de vulgarisation (ANDREATTA et collab, 2006), et permettent des ponts évidents entre science et art.

3.1.2 *Le pavage en une dimension*

Une vulgarisation étant plus tangible pour le plan ou l'espace, la grande majorité des travaux sur le pavage concerne la dimension 2 ou plus, cependant le pavage de la ligne a été aussi étudié par la communauté des mathématiciens. Le plus souvent, les tuiles T sont, en dimension 1, des sous-ensembles fermés de $\mathbb{R}$, de mesure de Lebesgue $\mu(T)$ positive et finie, et de frontière de mesure nulle. On dira alors que la tuile T pave $\mathbb{R}$ par translation s'il existe un ensemble discret $\mathcal{T}$ tel que

$$\mathbb{R} = \bigcup_{t \in \mathcal{T}} (T + t)$$

et si pour tout $t \neq t' \in \mathcal{T}, \mu\left((T + t) \cap (T + t')\right) = 0$

Un résultat de structure sur $\mathcal{T}$ (LAGARIAS et WANG, 1996b) permet d'obtenir la périodicité de ces types de pavage, résultat qui est faux dès la dimension 2.

Dans le cas moins élémentaire où la tuile T serait une union infinie de composantes connexes, la structure est plus compliquée à exposer, mais des résultats de constructions existent, comme par exemple les tuiles auto-affines étudiées dans (BANDT, 1991), (GROCHENIG et HAAS, 1994), (KENYON, 1990), ou (LAGARIAS et WANG, 1996a). Différemment, dans le cadre plus compliqué où les rotations et translations sur les tuiles sont autorisées des questions sont soulevées dans (ADLER et HOLROYD, 1981), mais il y a peu de résultats.

Cependant, en restant dans le cas simple où seule la translation est autorisée, les pavages les plus étudiés en dimension 1 (TIJDEMAN, 1993) sont ceux engendrés par les tuiles appelées *clusters* par STEIN et SZABÓ (1994), qui sont des unions finies d'intervalles dont les bords sont des entiers. Dans ce cas, les pavages de $\mathbb{R}$ sont équivalents à des pavages de $\mathbb{Z}$, et une tuile T et son paveur $\mathcal{T}$ pourront être décrits comme des ensembles d'entiers.

3.1.3 *Les problèmes de pavage abordés d'un point de vue algébrique*

La conjecture de MINKOWSKI (1907) énonce que pour tout pavage de $\mathbb{R}^n$ par des hypercubes, au moins deux cubes ont une face de dimension $(n-1)$ en commun.

Cette conjecture a été démontrée par HAJÓS (1949) grâce à l'introduction de la notion de factorisation de groupes.

La conjecture s'énonce alors sous la forme de ce théorème

Théorème 3.1.1. *(HAJÓS, 1949)*
Soient G un groupe abélien fini et $a_1, \ldots, a_n \in G$.

Si le groupe G peut s'écrire comme une somme directe de sous-ensembles A_i de la forme $A_i = \left\{1, a_i, \dots, a_i^{m_i-1}\right\}$ avec $m_i > 0$ pour tout i, alors il existe i tel que A_i soit un groupe.

De Bruijn conjecture un an plus tard (DE BRUIJN, 1950) que dans le cas où G est un groupe cyclique, alors un des sous-ensembles A_i est périodique, c'est-à-dire qu'il existe $g \in G$ tel que $gA_i = A_i$. Ce résultat est pourtant faux (HAJÓS, 1950), et il faudra plusieurs années pour trouver les groupes cycliques qui peuvent se décomposer en somme directe de sous-ensembles non périodiques.

Théorème 3.1.2 ((RÉDEI, 1947),(HAJÓS, 1950),(DE BRUIJN, 1953),(SANDS, 1957),(SANDS, 1959),(SANDS, 1962))**.** *Les groupes abéliens cycliques qui admettent une décomposition en somme directe de sous-ensembles non périodiques sont ceux dont le cardinal N n'est pas de la forme*

$$N = p^\alpha, N = p^\alpha q, N = p^2q^2, N = pqr, N = p^2qr, N = pqrs$$

avec p, q, r, s nombres premiers distincts.

La théorie spectrale introduite par FUGLEDE (1974) est aussi liée aux problèmes de pavage.

On dira qu'un ensemble borné Ω est spectral si et seulement si $L^2(\Omega)$ admet une base d'exponentielles orthonormée.

La difficile conjecture de Fuglede s'énonce simplement de cette manière : Si $\Omega \subset \mathbb{R}^d$ est un ouvert borné de mesure 1, alors Ω est spectral si et seulement si Ω peut paver $\mathbb{R}^d$ par translation.

Cette conjecture reste ouverte, bien qu'elle soit démontrée ou réfutée dans certains sous-cas. Tao contredit cette conjecture en dimension $d = 5$ ou plus (TAO, 2003), puis s'inspirant de la preuve, la conjecture est montrée fausse en dimension 3 et 4 (MATOLCSI, 2005). En petites dimensions, le problème reste ouvert. Elle est cependant vraie dans certains cas particuliers, comme dans des corps finis (IOSEVICH et PETRIDIS, 2015) ou en dimension 2 lorsque l'ensemble considéré est un parallélogramme ou un hexagone régulier (IOSEVICH et collab, 2001).

3.1.4 *Une rapide histoire de la complexité*

Le théorie de la complexité est une branche de l'informatique théorique qui cherche à décider de la difficulté à résoudre un problème, et à classer les problèmes en comparant leur difficulté. La difficulté d'un problème se mesure par le nombre de ressources nécessaires à sa résolution, indépendamment de l'algorithme ou de la machine utilisés, et il sera dit décidable si ce nombre de ressources est fini.

Un problème de décision peut-être vu comme une fonction définie sur un nombre infini d'instances, et renvoyant oui ou non pour chacune de ces instances, selon qu'elles satisfassent à une propriété

désirée ou non. Ainsi, un problème de décision peut être vu comme une langage formel, et les mots du langage sont les instances pour lesquelles le problème répond oui, et celles pour lesquelles il répond non ne sont pas dans le langage.

Une machine de Turing (MINSKY, 1967) est un modèle mathématique théorique d'une machine pouvant effectuer des calculs, allant du mathématicien avec un papier et un crayon au super-ordinateur. Elle est dite déterministe si ses actions futures sont entièrement déterminées par un ensemble fini de règles. Si un algorithme peut résoudre un problème donné, alors il existe une machine de Turing qui résout ce problème. Une machine de Turing déterministe lira un mot de taille n, instance d'un problème décidable, sur un ruban et après $f(n)$ actions s'arrêtera et renverra oui ou non, décidant ainsi si ce mot appartient au langage définissant le problème de décision. C'est cette fonction f qui représente la difficulté d'un problème.

Si cette fonction est un polynôme en n, on dira que le problème est dans la classe P. Si la machine est non déterministe et qu'elle s'arrête après l'exécution de $f(n)$ actions après avoir lu une instance de taille n, et f est encore une fonction polynomiale on dira que le problème est dans la classe NP. Un des plus grands problèmes contemporains consiste à savoir si $P \neq NP$. La communauté de spécialistes n'a aucune assurance quant à cette (in ?)égalité, ni si la réponse à la question sera un jour apportée (MEER et collab, 1997).

Cependant, une classe de problèmes, dit NP-complets, a été définie et regroupe les problèmes NP assez difficiles pour que tout problème dans NP soit un sous-cas des problèmes NP-complets (GAREY et JOHNSON, 1979). Bien qu'historiquement le problème de savoir s'il existe une assignation de variables propositionnelles qui rend vraie une formule de logique propositionnelle passée en instance (SAT) (COOK, 1971) soit utilisé comme le point de départ de définition des problèmes NP, CHLEBUS (1985) définit le problème de SQUARE TILING comme un problème NP-complet à l'aide de machines de Turing, et l'utilise pour en déduire la NP-complétude de tous les problèmes classiques de cette classe.

Cette notion de complexité théorique s'étend à des algorithmes particuliers, et mesure la rapidité à laquelle il s'arrête dépendamment de la taille de ses entrées. Les instances sont alors définies plus généralement, et peuvent être des objets abstraits sur lesquels la notion de taille peut varier selon le problème de décision que l'on cherche à résoudre avec cet algorithme.

3.2 Mathématiques et Musique : une longue histoire

La pièce de Steve Reich pour deux percussionnistes *Clapping Music* (REICH et HARTENBERGER, 1980) utilise un rythme présenté sur la

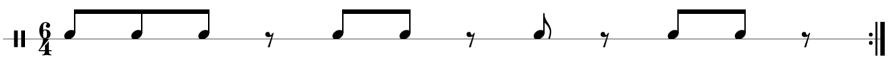


FIGURE 3.1: La séquence rythmique utilisée dans *Clapping Music*

Figure 3.1 joué en boucle pendant la pièce par un musicien, tandis que le second joue le même rythme permuté d'une pulsation toutes les 12 mesures. HAACK et collab (1998) imagine une explication au choix de ce rythme. Il y a 8 notes jouées sur les 12 temps de la mesure, soit 495 possibilités de les répartir. Seulement 2 parmi ces possibilités satisfont aux contraintes musicales suivantes : que le motif commence par une note et pas un silence, que jamais les deux musiciens ne retombent sur le même rythme avant la douzième permutation, et que le nombre de notes consécutives ne soit pas identique de part et d'autre d'un silence. Puis le rythme choisi par Reich est parmi ces 2 choix celui qui satisfait à la condition d'ensemble maximalement réparti (CLOUGH et DOUTHETT, 1991).

Cependant TOUSSAINT et collab (2002) suppose une solution moins mathématique et y voit un rythme *Yoruba* auquel est ajoutée une note entre les deux premières, que Reich aurait découvert lorsqu'il étudiait les percussions africaines dans les années 70.

3.2.1 *L'utilisation de l'algèbre comme approche à des problèmes musicaux*

L'utilisation des mathématiques en théorie de la musique remonte au moins aux gammes pythagoriciennes (GODWIN, 1992), et la tradition s'est étendue jusqu'à Euler qui a écrit un traité d'harmonie mathématique (EULER, 1739). Plus récemment, les liens entre ces deux sujets se concentrent autour de la résolutions de problèmes mathématiques soulevés par la théorie de la musique (REINER, 1985), (CLOUGH et DOUTHETT, 1991),(BLOCK et DOUTHETT, 1994), (READ, 1997),(AROM et collab, 2004), (CAURE, 2015) ; et de l'application des outils mathématiques à la formalisation de structures musicales, à l'analyse et à la composition (BALZANO, 1980),(DUNCAN, 1990),(KEITH, 1991), (CHEMILLIER et TRUCHET, 2003).

L'algèbre s'est particulièrement imposée par son utilisation par les trois grands compositeurs et théoriciens du XXe siècle Milton Babbitt, Iannis Xenakis et Anatol Vieru (ANDREATTA, 2003).

C'est le compositeur Krenek qui le premier a introduit la notion d'axiomatique dans la musique (KRENEK, 1937) pour l'étude des séries dodécaphoniques, que BABBITT (1992) explore à l'aide de la notion de groupes -de permutation et de congruence particulièrement. De telles notions lui permettent de développer des outils de combinatoire puis d'ensemble de classes de hauteurs (BABBITT, 1960).

Parallèlement, Xenakis dénonce un manque de formalisation de la musique, et développe la musique symbolique moderne (XENAKIS, 1963). Plus tard, VIERU (1993) utilise la théorie des ensembles et la notion de groupe quotient pour étudier les modes, et développe le concept de structure intervallique. Ces notions ont permis plus tard le développement de l'utilisation de la théorie des ensembles (FORTE, 1973), (LEWIN, 2010), largement employée de nos jours.

3.2.2 *La modélisation du rythme*

3.2.2.1 *Qu'est ce que le rythme ? Nos limitations.*

La plupart des études musicologiques modernes sont tournées principalement vers l'axe vertical de la musique, c'est-à-dire vers le monde des hauteurs, et ce n'est que très récemment (VINET, 2014) qu'est exploré plus en détail l'espace horizontal du temps et des rythmes. C'est Simha Arom (AROM et collab, 2004) qui a commencé à s'intéresser à la modélisation du rythme dans les années 70 pour des besoins de notation de rythmes étrangers à la musique traditionnelle occidentale. Puis a suivi une tradition de modélisation des rythmes (TOUSSAINT et collab, 2002)(TOUSSAINT et collab, 2003) pour faciliter leur étude, la génération de rythmes, l'analyse musicale, ou encore la reconnaissance automatique... De nombreux outils mathématiques et informatiques sont maintenant au service de l'étude des rythmes. Par exemple, des outils usuellement appliqués à la phylogénétique permettent des classifications des rythmes du Flamenco (GUASTAVINO et collab, 2009).

COOPER et MEYER (1963) parlent du rythme en ces termes :

> *To study rhythm is to study all of music. Rhythm both organizes, and is itself organized by, all the elements which create and shape musical processes.*
> *Just as a melody is more than simply a series of pitches, so rhythm is more than a mere sequence of durational proportions. To experience rhythm is to group separate sounds into structured patterns.*

La difficulté à définir la notion de rythme est peut-être responsable de son étude plus tardive. Des livres entiers y sont consacrés (DUMESNIL, 1979) et comme pour de nombreux domaines, la modélisation mathématique des rythmes demande des simplifications.

À la manière de RIVIÈRE (1993), nous simplifions les rythmes et ne considérons plus que les durées entre des notes successives, en oubliant toutes notions d'accent ou de timbre. De plus, notre conception de rythmes sera symbolique, et donc exacte, c'est-à-dire non sujette à des micro-variations. Nous nous intéressons de cette manière à des enchaînements d'*onsets* (attaques de départ d'une note) et du-

rées. Une note durant un temps t ou une note durant un temps $t/2$ et suivie d'un silence de temps $t/2$ sont considérées comme identiques.

Comme cela est supposé dans (VUZA, 1985), nous utilisons l'hypothèse que dans la musique occidentale, à laquelle nous nous limitons, toutes les durées sont dénotées par des nombres rationnels. C'est-à-dire que chaque durée est une fraction de la pulsation, découpage proéminent du temps, aussi appelé *tactus* (BILMES, 1993). Par une subdivision plus fine du temps, appelée *tatum* (BILMES, 1993), et mathématiquement décrite dans (VUZA, 1985), on peut supposer que chaque durée est un multiple de cette nouvelle subdivison.

Par exemple, pour l'ensemble des durées $\{1/2, 1/3, 5/2, 1/4, 8/6\}$ qui sont des fractions de la pulsation, il suffit de considérer une subdivision 12 fois plus fine du temps pour obtenir les durées suivantes : $\{6, 4, 30, 3, 16\}$.

Un rythme sera dans notre contexte ainsi défini comme un temps initial $t_0 \in \mathbb{Q}$ et un ensemble de durées entières. C'est ce que Vuza appelle un rythme régulier (VUZA, 1985).

3.2.2.2 *Les motifs rythmiques*

Devlin définit les mathématiques comme étant "*the science of patterns, and those patterns can be found anywhere you care to look for them, in the physical universe, in the living world, or even in our own minds.*" (DEVLIN, 1996) Et Feynman d'ajouter "*Mathematics is looking for patterns.*" (FEYNMAN et collab, 2003)

Musicalement aussi, BOLTON (1894) a montré qu'à l'écoute d'une pulsation régulière, l'humain ne peut s'empêcher d'entendre une répétition de motifs rythmiques.

Si un rythme est joué en boucle, une (ou plusieurs) instance(s) de ce rythme sera assimilé comme un motif basique qui se répète.

Par exemple, si notre rythme commençant à être joué à $t_0 \in \mathbb{Q}$ est défini par l'ensemble des durées $\{d_1, d_2, \ldots, d_n\}$, les *onsets* se trouveront sur les temps $t_0, t_0+d_1, t_0+d_1+d_2, \ldots, t_0+k\sum_{i=1}^{n} d_i+\sum_{i=1}^{r} d_i, \ldots$ (pour k, r entiers avec $r < n$). Alors, un auditeur entendra les *onsets* sur les temps $t_0 + k\sum_{i=1}^{n} d_i + \sum_{i=1}^{r} d_i, \ldots, t_0 + \sum_{i=1}^{r-1} d_i + k'\sum_{i=1}^{n} d_i$ (avec k, k', r entiers tels que $k' > k$ et $1 < r < n$) se répéter.

Nous définissons donc un motif rythmique comme un ensemble de pulsations entières, celles sur lesquelles sont les *onsets* qui se répètent.

Il n'y a bien sûr pas unicité de la représentation. D'abord comme le rythme est considéré de manière symbolique, deux subdivisions du temps différentes représenteraient le même rythme, comme montré sur la Figure 3.2.

Il n'y a pas non plus unicité sur le point de départ du motif. Le motif rythmique se jouant sur les pulsations entières $p_1, \ldots, p_n$ sera symboliquement identique à un motif rythmique se jouant sur toutes permutations circulaires $p_{1+s}, \ldots, p_{1+s-1}+p_n$ des pulsations entières.

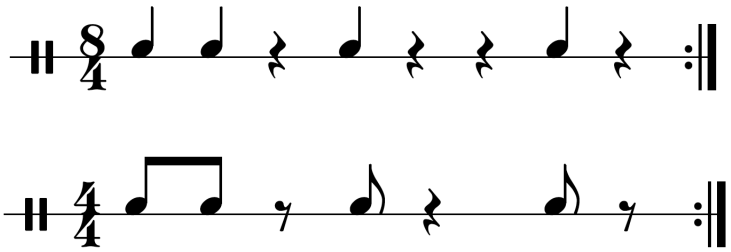


FIGURE 3.2: Deux rythmes différents qui ont pour modèle le même motif rythmique.

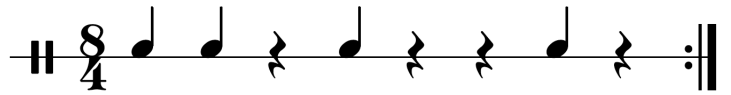


FIGURE 3.3: Un motif rythmique en notation musicale occidentale.

Lorsqu'il sera question d'énumération, deux solutions sont utilisées pour avoir un unique représentant d'un motif rythmique. Si on considère à nouveaux les durées entre les *onsets* $d_1, \ldots, d_n$, que l'on suppose divisées par leur pgcd, on peut décrire un motif rythmique par sa forme basique (AMIOT, 2004) qui classe les d_i par la permutation circulaire σ des d_i donnant le plus petit mot $d_{\sigma(1)} \cdots d_{\sigma(n)}$. Il est aussi possible de choisir le mot de Lyndon du mot $d_1 \cdots d_n$ (FRIPERTINGER, 2004).

3.2.2.3 *Différentes représentations mathématiques des motifs rythmiques*

L'écriture musicale occidentale moderne permet de décrire un motif rythmique de manière précise, mais a le défaut de ne pas être lisible par la communauté scientifique, ni par les musiciens d'autres traditions (EKWUEME, 1974), et surtout, elle n'est absolument pas maniable mathématiquement. Sur la Figure 3.3 est décrit un motif rythmique en notation occidentale traditionnelle, qui sera représenté dans différentes notations musicales ou mathématiques tout au long de cette section.

Une méthode développée par Philip Harland en 1962 à l'UCLA puis par KOETTING (1970) permet de noter très simplement des motifs rythmiques et a beaucoup été utilisée pour les notations de musiques non occidentales par des ethnomusicologues (KNIGHT, 1971), (AGAWU, 1995). Le système TUBS (Time Unit Box System), s'inspirant de la notation coréenne du XVe siècle *jeongganbo* (PYONG, 2014), utilise une (ou plusieurs) ligne(s) de boîtes. Ces boîtes représentent une unité fixe de temps. Une boîte vide signifie que rien de ne se passe, tandis

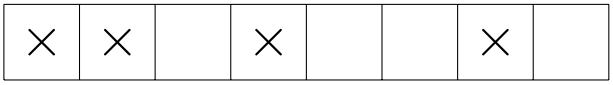


FIGURE 3.4: Un motif rythmique en notation TUBS.

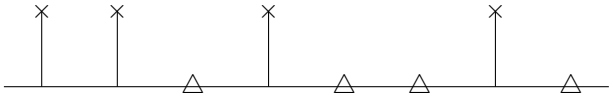


FIGURE 3.5: Un motif rythmique en notation croix-triangle.

qu'une boîte marquée d'une croix annonce qu'un *onset* est lancé au début de cet intervalle de temps. Le rythme de la Figure 3.3 est représenté en notation TUBS sur la Figure 3.4.

Douglas Eck a dérivé une variante de la notation TUBS pour des expériences psychologiques sur la perception du rythme (ECK, 2001) où un symbole est utilisé pour marquer les notes, et un autre pour les silences. Le rythme de la Figure 3.3 est représenté en notation avec les symboles croix et triangle sur la Figure 3.5.

En 1987, Gustafson a publié une nouvelle méthode de représentation du rythme (GUSTAFSON, 1987) (GUSTAFSON, 1988), combinant les avantages d'une représentation linaire comme les TUBS et des histogrammes en barre permettant de visualiser aisément la durée d'un phonème. L'idée est simplement de représenter la durée d'une note à la fois sur l'axe horizontal et sur l'axe vertical, en plaçant des carrés consécutifs sur l'axe des abscisses. Cette méthode nommée TEDAS (Temporal Elements Displayed As Squares) est illustrée sur Figure 3.6 où le rythme de la Figure 3.3 est représenté.

De la même façon, HOFMANN-ENGL (2002) a développé une méthode pour représenter un rythme comme un graphique en 2 dimensions. Notre rythme de la Figure 3.3 a pour plus petite unité de temps la noire, ou le 8ième de mesure qu'il décrit par la chaîne *chrotonique* suivante : $[12233322]\,(1/8)$ et représente comme démontré sur la Figure 3.7

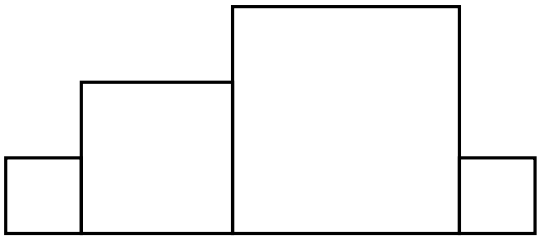


FIGURE 3.6: Un motif rythmique en notation TEDAS.

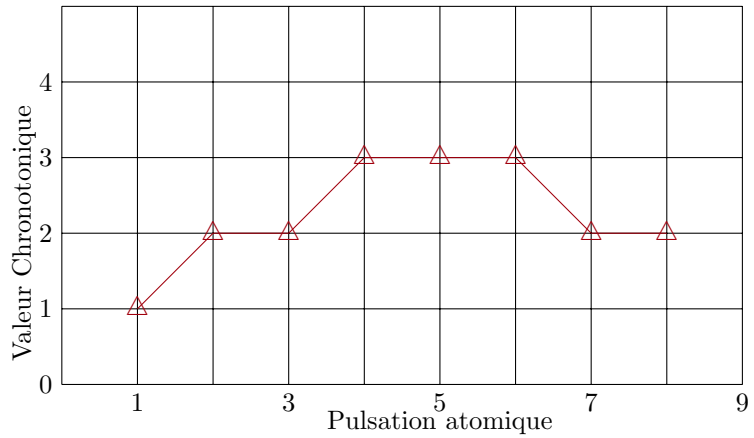


FIGURE 3.7: Un motif rythmique en notation graphique en 2 dimensions.

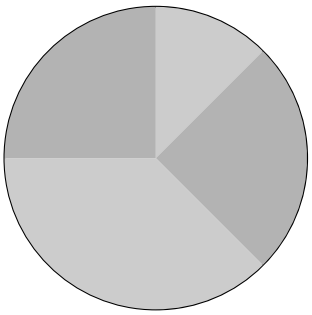


FIGURE 3.8: Un motif rythmique à la manière de Nasir al-Din al-Tusi.

Le choix de répéter un motif rythmique a inspiré des représentations géométriques circulaires ayant des traces remontant au XIIIe siècle. Un livre écrit par Nasir al-Din al-Tusi en 1252 décrit une représentation des rythmes comme un cercle découpé en parts (WRIGHT, 1978). Notre rythme de la Figure 3.3 est ainsi représenté Figure 3.8.

Plus récemment, Toussaint a utilisé une représentation des rythmes comme des polygones inscrits dans un cercle pour classifier les motifs rythmiques africains et afro-américains (TOUSSAINT et collab, 2002) (TOUSSAINT et collab, 2003). Les *onsets* du rythme sont représentés comme des points sur un cercle et sont considérés comme les sommets du polygone, comme on peut le voir sur Figure 3.9.

Empruntant la notation habituellement donnée aux hauteurs, PRESSING (1983) utilise un vecteur d'intervalles pour décrire un rythme. Chaque point du vecteur représente le nombre d'unités de temps à attendre entre chaque *onset*. Ainsi, le rythme de la Figure 3.3 s'écrirait [1232]. COYLE et SHMULEVICH (1998) et SHMULEVICH et collab (2001) préfèrent quant à eux utiliser un *vecteur des différences* de rythmes qui représente les quotients entre les placements relatifs des notes. Par exemple, notre rythme de la Figure 3.3 a pour vecteur des différences $[0.5, 2, 4/7, 7/9]$.

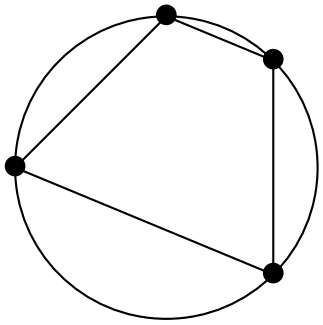


FIGURE 3.9: Un motif rythmique représenté comme un polygone.

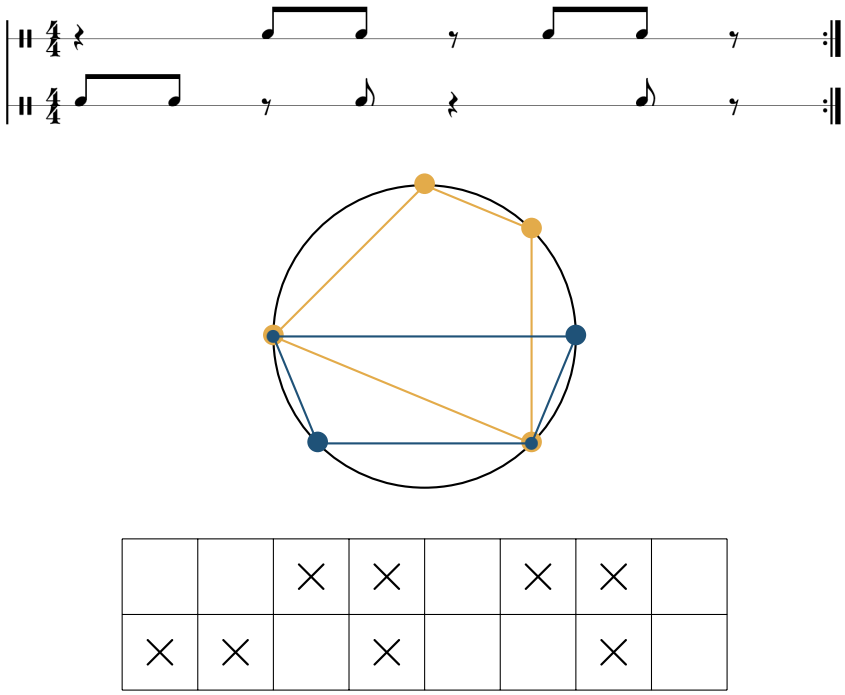
En médecine où l'étude des rythmes cardiaques est cruciale, une notation venant de l'informatique est utilisée (BETTERMANN et collab, 1999) en représentant un motif rythmique comme une suite binaire. Il s'agit d'une suite (non unique) de 0 et de 1 où chaque chiffre représente un pas dans une segmentation du temps, où à un instant donné sur cette segmentation se trouve un 1 si et seulement si un *onset* est entendu à ce temps. Le rythme de la Figure 3.3 s'écrit ainsi 11010010 ou avec un découpage du temps deux fois plus rapide 1010001000001000.

Remarque 3.2.1. De la description par des suites, dérive l'écriture sous forme de multi-ensemble (HU et TIEN, 1976), et un rythme peut se représenter comme un (multi)ensemble d'entiers. Pour une subdivision donnée du temps et un départ (le temps 0), le (multi)ensemble contient les entiers associés à une pulsation sur laquelle se trouve un *onset* à partir du départ. Le rythme de la Figure 3.3 sera donc représenté par l'ensemble $\{0,1,3,6\}$.

On définira précisément plus tard un rythme comme un ensemble d'entiers naturels qui contient 0.

La notation polynomiale (VUZA, 1991-1993) est aussi utilisée pour décrire un rythme, ce qui donne accès à tout un arsenal mathématique pour leur étude. Si un rythme est représenté par un (multi)ensemble d'entiers S (et que l'on note $m(s)$ la multiplicité des éléments $s \in S$), alors le polynôme associé à ce rythme est $\sum_{s\in S}(m(s))X^s$.

Toutes ces représentations sont faites pour des rythmes simples. Elles sont cependant facilement adaptables à un polyrythme. Pour les notations géométriques linéaires, comme sur les figures 3.3,3.4,3.5, 3.7, on utilisera la notation traditionnelle qui lie axe vertical au temps et axe horizontal aux hauteurs, en superposant les représentations. Pour les représentations géométriques circulaires, comme sur les figures 3.8 et 3.9, on préférera une distinction des différentes voix du rythme par l'utilisation de couleur différentes. Les notations mathématiques, comme celles des suites binaires, des ensembles ou des polynômes, sont aussi adaptables aux polyrythmes mais de façon



$$\{0, 1, 2, 3, 3, 5, 6, 6\}$$

$$1 + X + X^2 + 2X^3 + X^5 + 2X^6$$

FIGURE 3.10: Un motif polyrythmique représenté de haut en bas en notation traditionnelle européenne, en notation polygonale, en notation TUBS, comme un multi-ensemble et comme un polynôme.

moins évidente. Par exemple, au lieu d'une suite binaire, on peut représenter un polyrythme par une suite d'entiers où chaque nombre représente à un instant donné le nombre d'*onsets* joués simultanément. Pour les ensembles et les polynômes, il suffit de reprendre la remarque 3.2.1 précédente en lisant les parenthèses, c'est-à-dire en considérant des multi-ensembles et des polynômes à coefficients entiers. À titre d'exemple, plusieurs notations du même polyrythme sont représentées sur la Figure 3.10.

3.2.3 *Les canons rythmiques mosaïques*

Messiaen le premier construit des canons rythmiques (MESSIAEN, 2002) en utilisant des rythmes dits non rétrogradables. Cependant, une erreur de calcul sur les contraintes ne lui permet pas d'obtenir ce que nous appelons un canon rythmique mosaïque.

Un canon rythmique mosaïque est formé par un motif rythmique et certains de ses translatés dans le temps de telle manière qu'à chaque pulsation, on entende un et exactement un *onset*.

Pour la suite du chapitre pour noter un motif rythmique, nous utiliserons les notations par un multi-ensemble A ou par un polynôme $A(X)$. Nous nous limiterons à des cas de motif rythmique simple et pas de polyrythme, ainsi A sera un ensemble, et $A(X)$ un polynôme à coefficients dans $\{0,1\}$.

On rappelle que l'addition ensembliste est définie par

$$A+B=\{a+b \mid a\in A, b\in B\}.$$

Ainsi, si t_0 est une date à laquelle le motif rythmique A est translaté, ses *onsets* seront aux dates $A+\{t_0\}$, et donc si T est un ensemble de dates entières auxquelles le motif est translaté, on aura pour un canon rythmique mosaïque

$$\bigcup_{t\in T} A+\{t\}=\mathbb{Z} \text{ et pour tout } t\neq t'\in T, (A+\{t\}))\bigcap\left(A+\left\{t'\right\}\right)=\emptyset.$$

Ce qui se lit bien que toutes les pulsations de $\mathbb{Z}$ sont atteintes par un *onset* d'un translaté de A, et qu'elle ne sont atteintes qu'une seule fois. On dira alors que A et T sont en somme directe, et on notera $A+T=A\oplus T$.

On peut alors sans perte de généralité considérer que A est un ensemble fini (en mettant ses répétitions comme des translatés aux temps $t'_i\in\mathbb{Z}$ et en notant $T:=T\cup_i t'_i$) et que le temps commence à la première occurrence de A, c'est-à-dire que les éléments de A sont positifs et que $0\in A$ (Si $a=\min A$ est non nul, qui existe car A est un ensemble fini, on translate tous les éléments de A de $-a$ en gardant la propriété de canon rythmique mosaïque avec $A:=A+\{-a\}$ et $T:=T+\{-a\}$).

Ainsi, un canon rythmique mosaïque est la composition d'un sous-ensemble A de $\mathbb{N}$ fini qui contient 0 et d'un sous-ensemble T de $\mathbb{Z}$ tel que $A+T=A\oplus T=\mathbb{Z}$. On dira alors que A pave $\mathbb{Z}$ avec T.

Or Lagarias et Wang ont montré dans un cas plus général (LAGARIAS et WANG, 1996b) que tout pavage par un pavé fini est périodique. C'est-à-dire que dans nos conditions précédentes où $A\oplus T=\mathbb{Z}$, alors il existe $N\in\mathbb{N}^*$ et un ensemble fini B tel que $T=B\oplus N\mathbb{Z}$. De la même manière, on peut supposer que $0\in B\subset\mathbb{N}$.

En notant $\mathbb{Z}_N$ l'anneau des entiers modulo N, et $\pi_N:\mathbb{Z}\to\mathbb{Z}_N$ la projection des entiers dans Z_N, on définit la projection d'un ensemble fini d'entiers $\Pi_N:\mathcal{P}(\mathbb{Z})\to\mathcal{P}(\mathbb{Z}_N)$ par $\Pi_N(A)=\{\pi_N(a)\mid a\in A\}$.

On peut remarquer que si $A\oplus B\oplus N\mathbb{Z}=\mathbb{Z}$, alors A et $\Pi_N(A)$ ont même cardinal, et de même pour B.

Afin de ne pas alourdir les notations, et lorsque cela ne risque pas de prêter à confusion, nous alors noterons un élément $\pi_N(a)$ par l'entier représentant sa classe d'équivalence $a\in\{0,\dots,n-1\}$ et l'en-

semble $\Pi_N(A) = \{a' \mid \pi_N(a) = a' \in \{0, \dots, n-1\} \text{ et } a \in A\}$ par A, lorsqu'ils ont même cardinal.

Définition 3.2.2. Un canon rythmique mosaïque est la donnée d'un couple (A, B) de parties finies de $\mathbb{N}$ qui contiennent 0 tel qu'il existe $N \in \mathbb{N}^*$ avec

$$A \oplus B = \mathbb{Z}_N.$$

On dira que (A, B) est un canon rythmique mosaïque de $\mathbb{Z}_N$ ou encore que le motif rythmique A pave $\mathbb{Z}_N$ avec B. On nommera les éléments de A les *onsets*, et B sera le motif des *entrées*.

Remarque 3.2.3. La notion de somme directe est la même dans $\mathbb{Z}_N$. On a $A + B = A \oplus B$ si et seulement si

$$\bigcup_{b \in B} \Pi_N(A) + \{\pi_N(b)\} = \mathbb{Z}_N$$

et pour tout $b \neq b' \in B$, $(\Pi_N(A) + \{\pi_N(b)\}) \bigcap \left(\Pi_N(A) + \left\{\pi_N(b')\right\}\right) = \emptyset$.

Il en découle naturellement que si $A + B = A \oplus B = \mathbb{Z}_N$, alors le produit des cardinaux de A et de B vaut N.

Proposition 3.2.4. *(*AMIOT*, 2004) Si (A, B) est un canon de $\mathbb{Z}_N$, alors les polynômes $A(X)$ et $B(X)$ satisfont à*

$$A(X) \cdot B(X) = 1 + X + \cdots + X^{N-1} \mod (X^N - 1)$$

et réciproquement.

Exemple 3.2.5. Le rythme de la Figure 3.3 pave $\mathbb{Z}_8$ avec $B = \{0, 4\}$. En effet $A + B = \{0, 1, 3, 4, 5, 6, 7, 10\} = A \oplus B$ donc $\Pi_8(A \oplus B) = \{0, 1, 2, 3, 4, 5, 6, 7\}$. De même

$$A(X) \cdot B(X) = 1 + X + X^2 + X^3 + X^4 + X^5 + X^6 + X^7 \mod (X^8 - 1).$$

Nous utiliserons la notation TUBS pour représenter les canons rythmiques, en remplaçant les croix par des carrés pleins pour plus de visibilité. Comme le motif rythmique A est joué aux dates $b \in B$, le canon est un polyrythme, et chaque voix $A + \{b\}$ est représentée sur une ligne. Ainsi, cet assemblage de lignes peut se voir comme une grille (avec un pas $(1, 1)$ par exemple), et un carré plein est représenté tel que son coin inférieur droit a pour coordonnées $(a + b \mod N, i)$ avec $a \in A$ et b étant le i-ème élément de B. À titre d'exemple, le canon de l'exemple 3.2.5 est représenté en notation TUBS sur la Figure 3.11.

Définition 3.2.6. Soit $k \in \mathbb{N}^*$, si A est un motif rythmique qui pave $\mathbb{Z}_N$, alors $A + \{0, N, \dots, (k-1)N\} = A \oplus \{0, N, \dots, (k-1)N\}$ et on nomme cette somme son k-concaténé, noté $\overline{A}^k$.

Comme il a été montré par AMIOT (2005), certaines transformations sur les canons rythmiques permettent d'en engendrer d'autres.

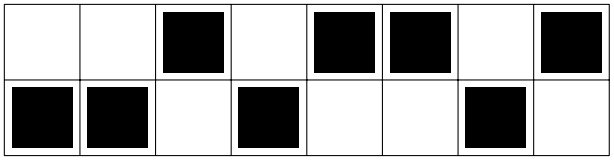


FIGURE 3.11: Un canon rythmique mosaïque en notation TUBS

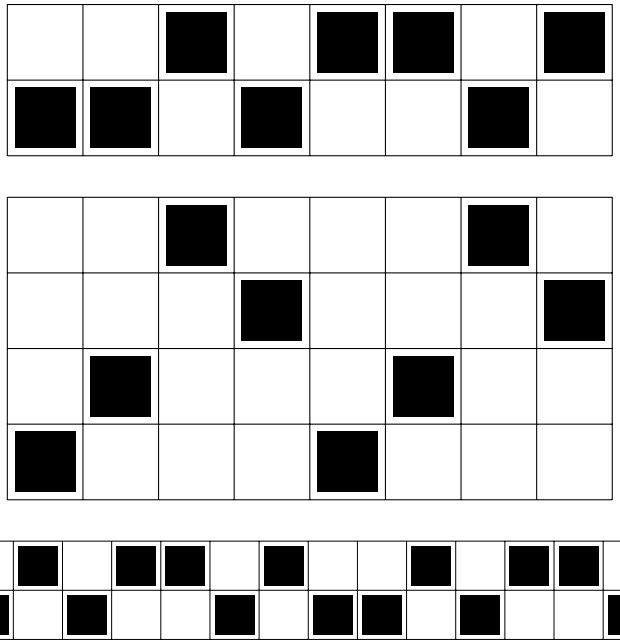


FIGURE 3.12: De haut en bas, le canon $(\{0,1,3,6\},\{0,4\})$, son dual $(\{0,4\},\{0,1,3,6\})$ et son 2-concaténé $(\{0,1,3,6,8,9,11,14\},\{0,4\})$ en notation TUBS.

Proposition 3.2.7 (dualité)**.** *(A,B) est un canon rythmique de $\mathbb{Z}_N$ si et seulement si (B,A) est un canon rythmique de $\mathbb{Z}_N$.*

Proposition 3.2.8 (concaténation)**.** *Soit $k \in \mathbb{N}^*$, si (A,B) est un canon rythmique de $\mathbb{Z}_N$, alors $(\overline{A}^k, B)$ est un canon rythmique de $\mathbb{Z}_{kN}$.*

Exemple 3.2.9. Soit $(\{0,1,3,6\},\{0,4\})$ le canon de $\mathbb{Z}_8$, alors son dual $(\{0,4\},\{0,1,3,6\})$ est un canon de $\mathbb{Z}_8$ et son 2-concaténé

$$(\{0,1,3,6,8,9,11,14\},\{0,4\})$$

est un canon de $\mathbb{Z}_{16}$. Ces trois canons sont représentés sur la Figure 3.12.

Définition 3.2.10. Un canon (A,B) de $\mathbb{Z}_N$ est dit périodique s'il existe $k \in \mathbb{N}$ tel que $0 < k < N$ et $\Pi_N(A + \{k\}) = \Pi_N(A)$ ou $\Pi_N(B + \{k\}) = \Pi_N(B)$.

Un canon non périodique est appelé canon de Vuza.

Dan Tudor Vuza nommait "canons rythmiques réguliers complémentaires de catégorie maximale" les canons rythmiques non périodiques et c'est en son honneur qu'on les appelle maintenant canon de Vuza. Le mathématicien roumain a démontré en parallèle le Théorème 3.1.2 de décomposition des groupes abéliens cycliques en facteurs non périodiques en ces termes.

Théorème 3.2.11 ((VUZA, 1991-1993))**.** *Il existe des canons de Vuza qui pave $\mathbb{Z}_N$ pour et seulement pour les N qui ne sont pas de la forme :*

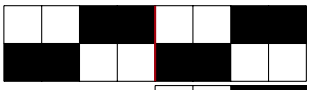
$$N = p^\alpha, N = p^\alpha q, N = p^2q^2, N = pqr, N = p^2qr, N = pqrs$$

avec p, q, r, s nombres premiers distincts.

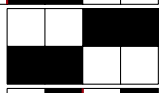
Théorème 3.2.12 ((AMIOT, 2005))**.** *Tout canon rythmique mosaïque peut être déduit par concaténation et dualité de canons de Vuza et du canon trivial* $(\{0\}, \{0\})$.

Exemple 3.2.13. Le canon rythmique $(\{0,1,4,5\}, \{0,2\})$ de $\mathbb{Z}_8$ se décompose ainsi par concaténation et dualité :

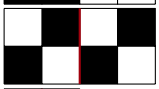
Le canon $\{0,1,4,5\} \oplus \{0,2\} = \mathbb{Z}_8$



est le 2-concaténé de $\{0,1\} \oplus \{0,2\} = \mathbb{Z}_4$



qui est le dual de $\{0,2\} \oplus \{0,1\} = \mathbb{Z}_4$



qui est le 2-concaténé de $\{0\} \oplus \{0,1\} = \mathbb{Z}_2$

dont le dual est le 2-concaténé du canon trivial :

Remarque 3.2.14. Les canons de Vuza forment une base pour les opérations de concaténation et de dualité des canons rythmiques. Il n'est bien sûr pas évident de savoir si un motif rythmique donné pourra paver ou non. Nous nous intéressons à trouver les canons de Vuza pour avoir une méthode constructive pour produire les canons rythmiques, mais aussi pour la conjecture d'une condition nécessaire et suffisante pour savoir si un motif rythmique peut paver, qui est présentée ci-après.

Définition 3.2.15. Soit A un motif rythmique, on s'intéresse aux polynômes cyclotomiques divisant le polynôme $A(X)$ et on définit

$$R_A = \{d \in \mathbb{N}^*, | \text{ le } d\text{ième polynôme cyclotomique divise } A(X)\},$$

$$S_A = \{p^a \in R_A, p \text{ premier}, a \in \mathbb{N}^*\}.$$

Définition 3.2.16. Soit A un motif rythmique, les trois conditions suivantes sont nommées conditions de Coven-Meyerowitz :

(T_0) : A peut paver

(T_1) : $A(1) = \prod_{p^\alpha \in S_A} p$

(T_2) : si $p_1^\alpha, p_2^\beta, \ldots, p_r^\gamma \in S_A$ alors $p_1^\alpha \cdot p_2^\beta \cdot \ldots \cdot p_r^\gamma \in R_A$ avec les p_i nombres premiers distincts.

Elles ont été introduites par COVEN et MEYEROWITZ (1999) lors de la recherche de conditions sur la capacité de paver d'un motif rythmique. Elles sont actuellement le meilleur espoir de condition nécessaire et suffisante pour savoir si un motif rythmique peut paver ou non.

Théorème 3.2.17 (COVEN et MEYEROWITZ (1999))**.**

1. $(T_0) \Rightarrow (T_1)$
2. $(T_1) \wedge (T_2) \Rightarrow (T_0)$
3. *Si le cardinal de A a au plus deux facteurs premiers, alors* $(T_0) \Rightarrow (T_1) \wedge (T_2)$

Proposition 3.2.18 (AMIOT (2005))**.** *La condition (T_2) est stable pour les opérations de dualité et de concaténation.*

Corollaire 3.2.19. *L'implication $(T_0) \Rightarrow (T_2)$ est vraie si et seulement si elle est vraie pour les canons de Vuza.*

Remarque 3.2.20. Ce corollaire montre l'importance des canons de Vuza. Ces canons non périodiques forment non seulement une base constructible des canons rythmiques, mais ils sont aussi au cœur d'une solution pour avoir une condition nécessaire et suffisante sur la qualité de pavabilité d'un motif rythmique.

Seulement, l'énumération des canon de Vuza n'est toujours pas à la portée des mathématiciens. La meilleure façon actuelle d'obtenir tous les canons de Vuza d'une période donnée est un algorithme d'exploration qui, même amélioré par KOLOUNTZAKIS et MATOLCSI (2009), reste exponentiel.

Ces canons construits par translation d'un motif rythmique sont ceux qui nous intéresseront dans cette thèse. Il est néanmoins intéressant de noter que d'autres se sont essayés à construire des canons rythmiques mosaïques en s'autorisant d'autres opérations sur les motifs.

Si un motif rythmique est défini par des durées entre les *onsets* $(d_1, \ldots, d_n)$, alors son rétrograde est défini par les durées $(d_n, \ldots, d_1)$. Wild a prouvé que tout motif de trois notes pave par rétrogradation et translation (WILD, 2002), phénomène qui aurait été utilisé depuis le Moyen Âge.

Soit $k \in \mathbb{N}^*$, on dit qu'on (k-)augmente un motif rythmique donné $\{0, a_1, \ldots, a_n\}$ si on considère le motif rythmique $\{0, ka_1, \ldots, ka_n\}$. Tom Johnson s'est intéressé au processus de pavage par un motif sous les opérations de translation et d'augmentation (JOHNSON, 2001), exposant de nombreux exemples. Davalan quant à lui s'est inspiré du problème du carré parfait (TUTTE, 1965) pour paver avec les k-augmentations et translations d'un motif, sans jamais utiliser deux fois un même indice k (DAVALAN, 2011).

3.2.4 *Un langage de programmation visuel : OpenMusic*

Alors que dans les années 70-80 la recherche en informatique musicale aux États-Unis s'intéressait principalement à la synthèse numérique, s'est développée en Europe à partir des années 60 la modélisation de processus musicaux pour la composition (BARBAUD, 1968), (XENAKIS, 1963), (RIOTTE et MESNAGE, 2006). Les chercheurs actuels, notamment à l'IRCAM, développent des outils de composition s'appuyant sur des modélisations formelles, comme celles présentées plus tôt dans ce chapitre par exemple.

Mais alors que les premières applications musicales de l'informatique n'étaient que des algorithmes "composant" selon un processus défini (HILLER et ISAACSON, 1957), la recherche en composition assistée par ordinateur s'attache maintenant à développer des outils qui serviraient la créativité du compositeur, en l'aidant à formaliser ou expérimenter des notions compositionnelles. Une interaction entre le compositeur et la machine permet que "*the powers of intuition and machine computation may be combined.*"(LASKE, 1981)

OpenMusic (AGON, 1998),(ASSAYAG et collab, 1997) est un environnement de composition assistée par ordinateur développé par l'équipe Représentations Musicales de l'IRCAM. C'est un langage de programmation complet, doté d'une interface visuelle.

Par exemple, un *package* d'outils mathématiques intègre des outils d'algèbre, pour la composition et l'analyse (ANDREATTA et AGON, 2005). Les *math tools* développent des outils pour l'utilisation des canons rythmiques, représentés comme polynômes, ensembles, sous-groupes...

L'avantage principal d'OpenMusic est d'être le lien entre mathématiciens et compositeurs. C'est lorsque les recherches fondamentales sont inscrites dans de tels outils de communication qu'elles peuvent être appliquées.

Le compositeur français Georges Bloch dit d'ailleurs à propos des canons rythmiques mosaïques qu'ils sont "*an intuitively evident structure that has interested numerous composers, among them Messiaen. But the construction of such a canon was beyond the means of composers working without computers, and research on the characteristics of such objects requires a musical representation tool such as OpenMusic*" (AGON et collab, 2006).

4

Complexité des problèmes de pavages

Résumé

Ce chapitre s'intéresse à la complexité de problèmes de décisions sur le pavage. Après un rappel de l'indécidabilité du problème de pavage général, et de la NP-complétude du pavage fini, nous nous penchons sur le pavage 1D qui sera l'objet de cette thèse. Le problème de pavage 1D est toujours décidable, car ces pavages sont périodiques, mais dans NP. Grâce à une propriété liant pavage 1D et différences entre ensembles d'entiers, on montre que le pavage 1D de $\mathbb{Z}$ par translation est un sous cas du problème DIFF. Ce problème est lui NP complet, et sa preuve a été améliorée dans ce chapitre, ainsi le problème de pavage 1D de $\mathbb{Z}$ est au plus NP-complet si on connait sa période, mais on ne sait rien dans le cas contraire. Le problème du pavage modulo p est ensuite démontré comme étant immédiat. En effet une propriété sur les polynômes définis sur les corps finis entraîne que tout motif pave modulo p, et même de façon compacte lorsque $p = 2$. Un algorithme glouton qui sera utilisé tout au long de cette thèse permettant de paver modulo 2 avec n'importe quel motif est décrit, et on montre ici qu'il est optimal dans le sens où il permet d'obtenir le plus petit canon compact possible.

4.1 Sur la décidabilité et la complexité des problèmes de décision liés au pavage

Définissons le problème du pavage par translation par un ensemble E muni d'une loi de composition interne $+$ et un ensemble de tuiles $T_1, \dots, T_n \subset E$. On se demande si ces tuiles et leurs translatées peuvent remplir l'espace, c'est-à-dire s'il existe un ensemble $\mathcal{T} \subset E$, dit de translations, tel chaque élément $e \in E$ appartient à une unique tuile T_i ou a une de ses translatées, et donc qu'il existe $i \in \{1, \dots, n\}$ et $t \in \mathcal{T}$ uniques tels que $e \in T_i + t$.

Définition 4.1.1. Soit un ensemble E muni d'une loi de composition interne $+$ et un ensemble de tuiles $T_1, \dots, T_n \subset E$. On dit que les

tuiles $T_1, \ldots, T_n \subset E$ pavent E par translation s'il existe $\mathcal{T} \subset E$ tel que

$$E = \bigcup_{i \in \{1,\ldots,n\}} \bigcup_{t \in \mathcal{T}} (T_i + t)$$

et pour tout $i, j \in \{1, \ldots, n\}$ et tout $t, t' \in \mathcal{T}$,

$$t \neq t' \Rightarrow (T_i + t) \cap \left(T_j + t'\right) = \emptyset$$

où $T_i + t = \{t_i + t \mid t_i \in T_i\}$.

PROBLÈME : PAVAGE

ENTRÉES : $E, T_1, \ldots T_n \subset E$

QUESTION : Les tuiles $T_1, \ldots, T_n$ pavent-elles E par translation ?

Lorsque E est infini et ses tuiles T_i sont finies, ce problème est vite très difficile. Pour $E = \mathbb{R}^2$, WANG (1965) a conjecturé qu'un pavage par translation était périodique, cependant **PAVAGE** est indécidable pour tout $\mathbb{R}^d$ avec $d \geq 2$, puisque par exemple (PENROSE, 1979) expose une solution non périodique avec $d = 2$ et $n = 6$.

On s'intéresse alors plus souvent au problème lorsque E est borné. Puis à partir de là, GOLOMB (1966) étend les capacités de pavage d'un ensemble de tuiles, et présente une hiérarchie de capacité de pavage de différents espaces de $\mathbb{R}^2$ pour un polyomino donné. Un polyomino est un ensemble de carrés unités connectés entre eux au moins par une frontière. Alors si un polyomino peut paver un rectangle, il peut paver une demi-bande prolongée à l'infinie sur un de ses côtés (c'est-à-dire l'espace borné entre les droites $x = a$, $x = b$ et $y = c$). Puis Golomb expose toute une hiérarchie présentée sur la Figure 4.1. Une flèche va d'une partie P_1 de $\mathbb{R}^2$ à une seconde partie P_2 si un polyomino peut paver P_1 implique que ce même polyomino peut paver P_2. Ce travail présente bien sur des contre-exemples qui montrent que chaque implication est stricte.

Le problème de savoir si un ensemble de tuiles peut paver une région bornée du plan est décidable, et on s'y intéressera par la suite, mais il reste très compliqué. Lorsqu'il s'agit de paver une région bornée du plan, chaque forme peut être symbolisée par une couleur, et le pavage d'une partie bornée par des tuiles quelconques finies devient équivalent au pavage d'un carré avec sa frontière colorée par des tuiles de Wang. Introduites dans (WANG, 1961), les tuiles de Wang sont des carrés dont les quatre côtés sont associés à une couleur. On pave un carré qui a sa frontière colorée avec un jeu de tuiles de Wang si on peut le remplir en prolongeant les couleurs de ses frontières, de telle manière que deux carrés partageant une frontière aient la même couleur sur cette frontière. Sur la Figure 4.2 [1], on peut voir par

1. Pour les lecteurs lisant cette thèse en noir et blanc, les trois couleurs utilisées sont rouge, bleu et jaune.

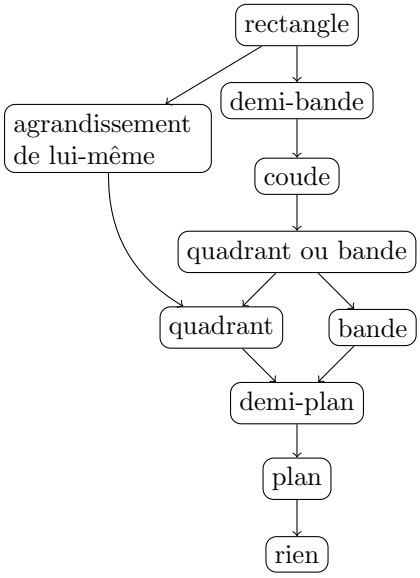


FIGURE 4.1: Si un polyomino peut paver un type d'espace P_1, et qu'une flèche va de P_1 à P_2, alors le polymonio peut paver P_2.

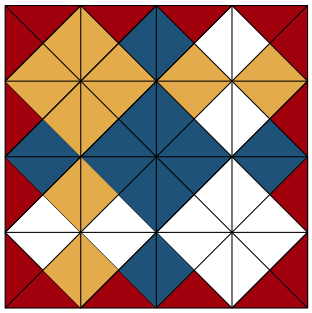


FIGURE 4.2: Un exemple de pavage par tuiles de Wang.

exemple un carré de côté 4 et dont la frontière est entièrement rouge qui est pavé par un jeu de 16 tuiles de Wang, qui sont toutes utilisées une seule fois ici. Ce problème est appelé problème de **SQUARE TILING**.

PROBLÈME : SQUARE TILING

ENTRÉES	:	$n \in \mathbb{N}^*$, la couleur des frontières du carré $n \times n$ un jeu de tuiles de Wang $T_i, \ldots, T_r$ colorées
QUESTION	:	Peut-ont paver le carré en prolongeant la couleur de sa frontière tel que 2 arrêtes communes aient toujours la même couleur ?

GAREY et JOHNSON (1979) montrent que le problème de **SQUARE TILING** est NP-complet , mais surtout il est présenté dans (CHLEBUS, 1985) comme un problème pouvant remplacer SAT comme référence

de la NP-complétude. Ainsi, tous les problèmes NP-complets classiques peuvent être réduits à **SQUARE TILING**.

Intéressons-nous maintenant au problème de **PAVAGE** en dimension 1, c'est-à-dire que nous cherchons à paver $E = \mathbb{R}$.

PROBLÈME : PAVAGE DE $\mathbb{R}$
ENTRÉES : un ensemble de tuiles $T_1, \dots, T_r \subset \mathbb{R}$ finies
QUESTION : Les tuiles $T_1, \dots, T_n$ pavent-elles $\mathbb{R}$ par translation ?

Lagarias et Wang ont montré (LAGARIAS et WANG, 1996b) que s'il existet un pavage de $\mathbb{R}$ par translation avec les tuiles $T_1, \dots, T_r$, alors il existe un pavage périodique de $\mathbb{R}$ par translation avec les tuiles $T_1, \dots, T_r$. Il en est de même avec un pavage de $\mathbb{Z}$ puisque chaque tuile $T_i \subset \mathbb{Z}$ peut être remplacée par $T_i + [0,1[$ et avoir un pavage de $\mathbb{R}$ avec ces tuiles est équivalent à avoir un pavage de $\mathbb{Z}$.

PROBLÈME : PAVAGE DE $\mathbb{Z}$
ENTRÉES : un ensemble de tuiles $T_1, \dots, T_r \subset \mathbb{Z}$ finies
QUESTION : Les tuiles $T_1, \dots, T_n$ pavent-elles $\mathbb{Z}$ par translation ?

Si k est la plus petite période d'un pavage de $\mathbb{Z}$ obtenu avec une seule tuile T finie, alors on peut borner cette période.

Définition 4.1.2. Soit T une tuile pavant $\mathbb{Z}$ par translation. On note $\Delta(T) = \max T - \min T$ son diamètre.

Théorème 4.1.3 ((KOLOUNTZAKIS et collab, 2003))**.** *Soit T une tuile finie pavant $\mathbb{Z}$ par translation de manière non triviale avec la plus petite période possible k, alors*

$$2\Delta(T) \leq k \leq 2^{\Delta(T)}.$$

Ainsi, le problème **PAVAGE DE $\mathbb{Z}$** est facilement décidable. Il suffit d'utiliser toutes les combinaisons possibles $k \in \left\{0, \dots, 2^{\Delta(T)}\right\}$ et $\mathcal{T} \subset \{0, \dots, k-1\}$ et vérifier si $\Pi_k(T + \mathcal{T}) = \mathbb{Z}_k$.

Nous allons maintenant utiliser les notations classiques de pavage périodique de $\mathbb{Z}$ par une tuile et ses translatées. La tuile est notée A, un ensemble fini de $\mathbb{N}$ qui contient 0, et elle est translatée sur les éléments de B, un ensemble fini de $\mathbb{N}$ qui contient 0. Si la période du pavage vaut N, nous noterons $A \oplus B = \mathbb{Z}_N$ comme en 2.2.20.

On note la soustraction d'un ensemble d'entiers

$$A - A = \left\{a - a' \mid a, a' \in A\right\}.$$

Proposition 4.1.4 ((AMIOT, 2011))**.**

$$A \oplus B = \mathbb{Z}_N \Leftrightarrow (A - A) \bigcap (B - B) = \{0\} \text{ et } |A||B| = N.$$

Cette propriété permet de lier le problème **PAVAGE DE $\mathbb{Z}$** à un autre problème qui est NP-complet.

PROBLÈME : DIFF
ENTRÉES : $k, N \in \mathbb{N}^*$ et deux ensembles $E, D \subset \mathbb{Z}_N$
QUESTION : Existe-t-il $A \subset E$ tel que $|A| = k$ et $A - A \subset D$?

Théorème 4.1.5. *Le problème* **DIFF** *est NP-complet.*

Démonstration. Le problème **DIFF** est bien sûr NP, et pour montrer sa NP-complétude, nous allons utiliser une réduction au problème **IND** qui est NP-complet (GAREY et JOHNSON, 1979).

PROBLÈME : IND
ENTRÉES : Un graphe simple G ayant n sommets, $k \in \mathbb{N}^*$
QUESTION : Existe-t-il un ensemble A de k sommets tel qu'aucun de ses sommets ne soit connecté par une arête ?

On note $V = \{1, \dots n\}$ l'ensemble des sommets de G.
On construit par récurrence la fonction $\phi : V \mapsto \mathbb{Z}_m$ avec m borné par un polynôme en n.
On note $\phi(1) = 0$, et si $\phi(1), \dots, \phi(r)$, $r < n$ sont construits, on choisit le plus petit $\nu \in \mathbb{Z}$ tel que

$$\nu \notin \{\phi(i) + \phi(j) - \phi(l) \mid 1 \leq i, j, l \leq r\}$$

puis on prend $m = \phi(n) + 1$.
Alors la construction de la fonction ϕ est bien polynomiale en n et a la propriété que pour tout $i, j \in \{1, \dots, n\}$, si $i < j$ les différences $\phi(j) - \phi(i)$ sont toutes deux à deux distinctes.
Alors avec $E = \{\phi(i) \mid i \in \{1, \dots, n\}\}$ et

$$D = \{\phi(i) - \phi(j) \mid i < j,\ i \text{ et } j \text{ ne sont pas connectés dans } G\},$$

et le choix $N = m$, on a bien qu'une solution $A = \{\phi(i_1), \dots, \phi(i_k)\}$ au problème **DIFF** donne une solution $\{i_1, \dots, i_k\}$ au problème **IND**.
□

Alors on remarque facilement que savoir résoudre le problème **DIFF** en temps polynomial permet de résoudre le problème d'optimisation **DIFF'** en temps polynomial :

PROBLÈME : DIFF'
ENTRÉES : $N \in \mathbb{N}^*$ et deux ensembles $E, D \subset \mathbb{Z}_N$
RECHERCHE : Quel est le plus grand k tel qu'il existe $A \subset E$ tel que $|A| = k$ avec $A - A \subset D$?

Et on peut rapprocher le problème **DIFF'** grâce à la propriété 4.1.4 au problème de pavage périodique par une seule tuile A de $\mathbb{Z}_N$.

PROBLÈME : PAVAGE DE $\mathbb{Z}_N$ PAR UNE TUILE

ENTRÉES : $N \in \mathbb{N}^*$ et une tuile $A \subset \mathbb{Z}_N$ finie

QUESTION : La tuile A pave-t-elle $\mathbb{Z}_N$ par translations ?

Passons en entrées du problème **DIFF'**

$$E = \mathbb{Z}_N \text{ et } D = (A - A)^c \cup \{0\}.$$

Si le k obtenu entier vaut $k = \dfrac{N}{|A|}$, alors par la propriété 4.1.4, A pave $\mathbb{Z}_N$ par translation. Ainsi le problème **PAVAGE DE $\mathbb{Z}_N$ PAR UNE TUILE** est au plus NP-complet, puisque c'est un cas particulier du problème **DIFF'**.

Ceci n'est cependant pas en contradiction avec la conjecture de Coven-Meyerowitz 3.2.17. En effet, la conjecture énonce qu'une tuile A pave un $\mathbb{Z}_N$ par translation si et seulement si elle satisfait aux deux conditions (T_1) et (T_2), et il existe un algorithme en temps polynomial (KOLOUNTZAKIS et MATOLCSI, 2009) permettant de savoir si A satisfait à ces deux conditions. Mais ces conditions ne disent rien de la taille N du pavage. Ainsi la conjecture se rapproche plus du problème **PAVAGE PAR UNE TUILE** :

PROBLÈME : PAVAGE PAR UNE TUILE

ENTRÉES : $A \subset \mathbb{N}$ finie

QUESTION : Existe-t-il N tel que A pave $\mathbb{Z}_N$ par translation ?

Supposant $P \neq NP$ et que la conjecture de Coven-Meyerowitz soit vraie, on aurait alors que le problème **PAVAGE PAR UNE TUILE** peut se résoudre en temps polynomial.

4.2 La complexité immédiate du problème de pavage modulo p et l'algorithme glouton du pavage modulo 2

Relâchons maintenant la contrainte de n'avoir qu'un et un seul *onset* par pulsation, pour revenir aux canons rythmiques mosaïques modulo p. La contrainte devient alors d'avoir un et un seul *onset* modulo p par pulsation.

Rappelons que les motifs rythmiques (A, B) pavent $\mathbb{Z}_N$ modulo p si

$$\begin{aligned} A(X) \cdot B(X) &= 1 + X + \cdots + X^{N-1} \mod (X^N - 1, p) \\ &= \frac{X^N - 1}{X - 1} \mod (X^N - 1, p). \end{aligned}$$

Théorème 4.2.1. *Soit p premier, et $P \in \mathbb{F}_p[X]$ tel que $P(0) \neq 0$, alors il existe $N \in \mathbb{N}$ tel que P divise $X^N - 1$.*

Démonstration. Voir par exemple (WARUSFEL, 1971). □

Comme cela a été remarqué pour la première fois par AMIOT (2005), ce théorème implique que tout canon rythmique pave modulo p. En effet, si p est premier et A est un motif rythmique fixé, alors en considérant le polynôme

$$P(X) = A(X)(X-1) \in \mathbb{F}_p[X]$$

qui ne s'annule pas en 0 puisque A est un motif rythmique, le théorème donne un polynôme $Q(X) \in \mathbb{F}_p[X]$ tel que

$$P(X) \cdot Q(X) = (X^N - 1) \mod p.$$

Puis on peut appliquer une transformation à Q pour que le polynôme représente un motif rythmique, c'est-à-dire qui a ses coefficients dans $\{0,1\}$. Pour tout monôme αX^k de $Q(X)$ tel que son coefficient $\alpha \notin \{0,1\}$ on peut itérer la transformation

$$\alpha X^k = (\alpha - 1)X^k + X^{k+N} \mod (X^N - 1)$$

jusqu'à obtenir un polynôme $\widetilde{Q} \in \{0,1\}[X]$ tel que

$$P(X) \cdot \widetilde{Q}(X) = (X^N - 1) \mod (X^N - 1, p).$$

Il est alors très important de remarquer que si le motif est considéré dans $\mathbb{F}_2[X]$, alors le polynôme obtenu $Q(X) \in \mathbb{F}_2[X]$ a déjà ses coefficients qui valent 0 ou 1, et l'égalité

$$P(X) \cdot Q(X) = (X^N - 1) \mod p$$

est vraie dans $\mathbb{F}_2[X]$, sans avoir besoin de quotienter par le polynôme $X^N - 1$. Alors les transformations pour ramener Q à un polynôme $\widetilde{Q} \in \{0,1\}[X]$ qui représente un motif rythmique ne sont pas nécessaires. Ceci implique que tout motif rythmique A pave modulo 2 et qu'il existe un polynôme $Q \in \{0,1\}[X]$ tel que

$$A(X) \cdot \widetilde{Q}(X) = 1 + X + \cdots + X^{N-1} \mod (2).$$

C'est-à-dire que tout motif rythmique admet un pavage compact modulo 2.

Exemple 4.2.2. Dans $\mathbb{F}_3[X]$, considérons le motif rythmique

$$A(X) = 1 + X + X^3.$$

Par le théorème 4.2.1, on peut obtenir l'égalité

$$\begin{aligned} X^{24} - 1 =& A(X)(X-1) \cdot (X^{20} + X^{19} + 2X^{17} + 2X^{15} + 2X^{14} + 2X^{13} + \\ & 2X^{10} + X^9 + 2X^8 + X^7 + X^6 + X^5 + 2X^4 + 2X^3 + X^2 + 1) \end{aligned}$$

Le polynôme quotient $Q(X) = X^{20} + X^{19} + 2X^{17} + 2X^{15} + 2X^{14} + 2X^{13} + 2X^{10} + X^9 + 2X^8 + X^7 + X^6 + X^5 + 2X^4 + 2X^3 + X^2 + 1$

n'a pas ses coefficients dans $\{0,1\}$, et ne représente donc pas un motif rythmique. En appliquant la transformation $2X^k = X^k + X^{k+24} \mod (X^{24}-1)$ à ses termes $(2*X^{17}, 2*X^{15}, 2*X^{14}, 2*X^{13}, 2*X^{10}, 2*X^8, 2*X^4, 2*X^3)$, on obtient alors le polynôme

$$\begin{aligned}\widetilde{Q}(X) =& X^{20}+X^{19}+X^{41}+X^{17}+X^{39}+X^{15}+X^{38}+X^{14}+X^{37}\\ &+X^{13}+X^{34}+X^{10}+X^9+X^{32}+X^8+X^7+X^6+X^5\\ &+X^{28}+X^4+X^{27}+X^3+X^2+1\end{aligned}$$

qui satisfait à

$$A(X)\cdot\widetilde{Q}(X) = 1+X+\cdots+X^{23} \mod (X^{24}-1,3).$$

Ainsi le problème de savoir si un motif rythmique pave modulo p est immédiat. Le théorème précédent a aussi inspiré un algorithme introduit dans (AMIOT, 2011) qui permet de trouver le motif des entrées B en temps linéaire en la taille du pavage N si le pavage est compact. Nous présentons maintenant l'algorithme 1 glouton dérivé du précédent, dont on montrera qu'il est optimal si le pavage est compact, ce qui est toujours le cas pour $p=2$.

Algorithme 1 : Algorithme de pavage compact modulo p

Data : $A \subset \mathbb{N}$ fini tel que $0 \in A$, p premier.
1 $B = \{0\}$, $N = \max A$
2 $C = A + B \mod p$ est le polyrythme modulo p
3 **while** $c \neq \overline{1}^n$ *pour un* n **do**
4 $i \leftarrow$ le premier indice tel que $c[i] \neq 1 \mod p$
5 **if** $i \neq (\max B + \max A + 1)$ **then**
6 $B := B \cup \{i\}$
7 $N := i + \max A$
8 $C = A + B \mod p$ est le polyrythme modulo p
9 **else**
10 break while
11 **return** (B, N)

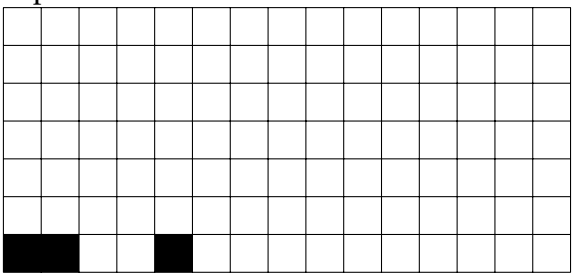
Remarque 4.2.3. La boucle **while** est parcourue au plus $N - \max(A)$ fois, donc l'algorithme est bien en temps linéaire en N.

Exemple 4.2.4. Cet algorithme se comprend facilement graphiquement. Dans la représentation TUBS, il peut s'interpréter en ces termes : on ajoute le motif rythmique A sur la première pulsation où il n'y a pas un *onset* modulo p.

Regardons l'algorithme se dérouler pour les entrées $A = \{0,1,4\}$, et $p = 2$ pour s'assurer de la compacité du pavage. À chaque étape de l'algorithme sont montrés le mot c_2 modulo 2 associé au polyrythme $C = A + B \mod 2$, ainsi que la plus petite sous-couverture $UC_2(i)$

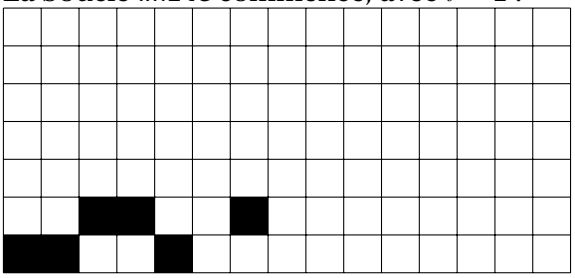
que l'algorithme essaye de remplir, notion qui sera définie plus tard en 7.1.5.

À l'étape 1, l'algorithme pose simplement les *onsets* du motif A à la pulsation 0 :



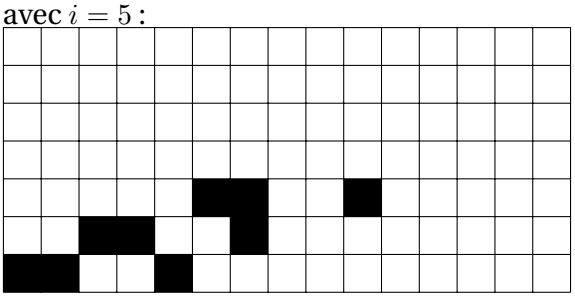
$c_2 = 11001000\cdots$
$UC_2(2) = 001$

La boucle `while` commence, avec $i = 2$:



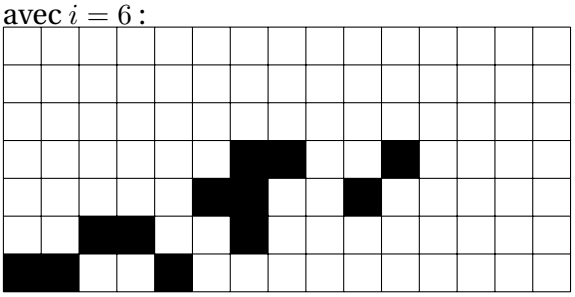
$c_2 = 1111101$
$UC_2(5) = 01$

avec $i = 5$:



$c_2 = 1111110001$
$UC_2(6) = 0001$

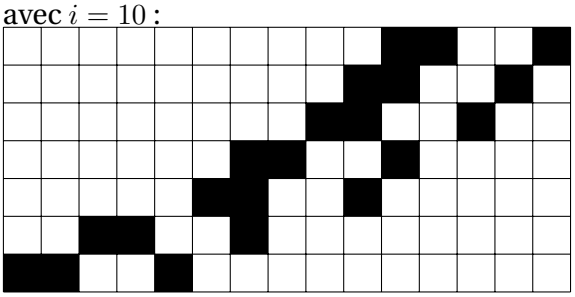
avec $i = 6$:



$c_2 = 11111111011$
$UC_2(8) = 011$

⋮

avec $i = 10$:



$c_2 = 111111111111111$
$UC_2(15) =$ mot vide

L'étape suivante donne $i = 15 = \max B + \max A + 1$ donc l'algorithme s'achève. Il renvoie B le motif des entrées obtenu, et N la taille du pavage compact.

Remarque 4.2.5. Un tel pavage n'est pas unique. Le motif rythmique $A = \{0, 1, 4\}$ peut paver modulo 2 avec les entrées $B = \{0, 2, 5, 6, 8, 9, 10\}$ obtenues par l'algorithme, mais aussi avec les entrées B_i suivantes

- $B_1 = \{0, 2, 5, 6, 8, 9, 10, 15, 17, 20, 21, 23, 24, 25\}$
- $B_2 = \{0, 2, 5, 6, 8, 9, 10, 15, 17, 20, 21, 23, 24, 25, 30, 32, 35, 36, 38, 39, 40\}$
- ...

Proposition 4.2.6. *L'algorithme 1 glouton est optimal au sens où il produit le plus petit nombre d'entrées $|B|$ (et donc le plus petit N) tel que (A, B) est un pavage compact modulo p de $\mathbb{Z}_N$.*

Démonstration. Soit p premier, et A un motif rythmique qui peut produire un pavage compact modulo p.

Supposons par l'absurde qu'il existe $\widetilde{B}$ tel que $(A, \widetilde{B})$ soit un canon compact modulo p de $\mathbb{Z}_{\widetilde{N}}$, avec $|\widetilde{B}| < |B|$ où B est le motif des entrées donné par l'algorithme 1.

Il n'est pas possible d'avoir $\widetilde{B} \not\subseteq B$, car $(A, \widetilde{B})$ étant un canon compact modulo p, lors de l'étape **4** de l'algorithme, le plus petit élément i de $B \setminus \widetilde{B}$ satisferait à $i = \max \widetilde{B} + \max A + 1$ et alors on aurait $B = \widetilde{B}$.

On peut supposer sans perte de généralité que B et $\widetilde{B}$ sont ordonnés. Soit i le plus petit indice tel que $\widetilde{B}[i] = \widetilde{b} < B[i] = b$.

Après $i - 1$ passages dans la boucle **while**, l'algorithme a produit le sous-ensemble $B[0, \dots, i-1]$ tel que les polyrythmes modulo p représentés par les mots $(A + B[0, \dots, i-1])[0, \dots, N-1]$ et $(A + \widetilde{B}[0, \dots, i-1])[0, \dots, N-1]$ soient égaux.

Comme $\widetilde{B}$ pave avec A, alors au temps $\widetilde{b}$ il ne peut pas y avoir exactement un *onset* modulo p dans ce sous mot :

$$(A + B[0, \dots, i-1])\left[\widetilde{b}\right] = (A + \widetilde{B}[0, \dots, i-1])\left[\widetilde{b}\right] \neq 1 \mod p.$$

Alors, l'étape **4** de l'algorithme, pendant le $i$ème passage dans la boucle **while**, produit le plus petit indice b' tel que dans ces sous-mots il n'y ait pas 1 *onset* $\mod p$, et il satisfait donc à $b' \leq \widetilde{b} < b$. C'est une contradiction avec le fait que le $i$ème élément de B soit b. □

Exemple 4.2.7. Comme on peut le voir dans l'exemple 4.2.4, le plus petit nombre d'entrées pour obtenir un pavage compact modulo 2 avec le motif $A = \{0, 1, 4\}$ est 7.

Remarque 4.2.8. On peut alors trouver facilement une borne supérieure à la période N d'un canon modulo p obtenu avec le motif A, et donc à la complexité de l'algorithme 1 qui est linéaire en N.

On note $I_{d,p}$ l'ensemble des polynômes irréductibles unitaires de degré d dans $\mathbb{F}_p[X]$. Alors on peut montrer que pour tout $n \in \mathbb{N}^*$,

$$X^{p^n} - X = \prod_{d|n} \prod_{R \in I_{d,p}} R.$$

Ainsi, en considérant les facteurs irréductibles de notre polynôme $P(X) = A(X)(X-1)$ et on notant n le *ppcm* de leur degrés, on a immédiatement que $P(X)$ divise $X^{p^n} - X$. Or $A(X)$, et donc $P(X)$, ne s'annule pas en 0 par définition, donc X n'est pas un facteur de P alors que $X \in I_{n,p}$. En divisant par ce terme l'égalité $P \times Q = X^{p^n} - X$, on obtient $N = p^n - 1$. Or $(X-1)$ étant irréductible dans tous les $\mathbb{F}_p[X]$, le *ppcm* des degrés des facteurs irréductibles de $A(X)$ vaut aussi n. Le degré $\deg(A)$ de $A(X)$ majore le degré d_i de chacun de ses facteurs irréductibles P_i, et ils ne peut pas y en avoir plus que $\deg(A)$. Ainsi

$$n \leq \prod_{\deg(P_i)=d_i} d_i \leq \prod_{\deg(P_i)=d_i} \deg(A) \leq \prod_{i=1}^{\deg(A)} \deg(A) = \deg(A)^{\deg(A)}.$$

Donc $N = p^n - 1 \leq p^{\deg(A)^{\deg(A)}} - 1$.

Remarque 4.2.9. Grâce au théorème 3.2.12, nous savons que les canons de Vuza sont minimaux pour les opérations de concaténation et dualité. Le canon compact (A, B) modulo p est minimal pour les opérations de dualité et concaténation si l'algorithme 1 prenant en entrée un A retourne B, et si prenant B il retourne A.

On peut dire que l'algorithme glouton renvoie les "canons de Vuza modulo p".

De plus, si (A, B) est un canon compact, et que l'algorithme 1 renvoie un motif des entrées $B' \neq B$ lorsqu'on lui donne A, alors on aura que $B' \subset B$, et le motif A' retourné par l'algorithme lorsqu'on lui donne B' est aussi tel que $A' \subset A$.

L'algorithme 1 retournant les canons de Vuza modulo p très facilement, nous espérons pouvoir trouver un lien entre ces canons et les canons de Vuza mosaïques afin de pouvoir les énumérer plus rapidement. Cependant, que nos résultats sur les canons modulo p puissent nous aider pour l'étude des canons mosaïques reste un espoir distant. Une différence principale entre ces deux types de pavage est la contrainte sur le nombre d'*onsets* qui est relâchée dans le cas du pavage modulo p, et le chapitre suivant s'intéresse à la question de pouvoir "enlever" ces *onsets* superposés. Il n'est pas assuré que nous puissions un jour retirer ces *onsets* pour revenir du pavage modulo p au pavage mosaïque, mais comprendre le lien entre motif rythmique et superpositions créées pourrait nous donner des pistes pour accélérer les algorithmes pour trouver les canons de Vuza, qui sont pour le moment exponentiels. En effet, passer de pavage mosaïque à pavage modulo 2 permet d'accélérer les algorithmes de complétion d'un temps exponentiel à un temps linéaire. Et le dernier chapitre de cette thèse utilise des résultats de récurrence qui permettent de construire un algorithme en temps logarithmique pour trouver le motif des entrées de certaines familles de motifs rythmiques. Si nous imaginons pouvoir généraliser ce résultat à tous les motifs ryth-

	Pavage mosaïque		Pavage modulo p
Rapidité des algorithmes pour trouver les Vuza de $\mathbb{Z}_N$	exponentielle	Ajout de donsets	linéaire
			Accélération par formules de récurrence
	?	Retrait de donsets	logarithmique

TABLE 4.1: Rapidité des algorithmes pour trouver les canons de Vuza mosaïques ou modulo p.

miques, et que nous puissions paver modulo 2 en temps logarithmique, alors peut être le retour à la contrainte mosaïque d'un seul *onset* par pulsation nous autorisera des algorithmes de pavage en temps moins qu'exponentiel ?

5

De la couverture au pavage : étude des donsets

Résumé

Ce chapitre s'intéresse à la première formalisation de la différence entre canons modulo p et canons mosaïques. On y définit l'ensemble des donsets, qui est celui des temps où p *onsets* sont superposés à un pavage pour créer une couverture. On y présente pour la première fois des canons différents ayant les mêmes donsets. La dualité est triviale, mais deux transformations nouvelles non évidentes sont présentées, et elles permettent à partir d'un canon modulo 2 d'obtenir deux canons modulo 2 plus grands distincts et ayant les mêmes donsets.

5.1 Définition des donsets

La distinction principale entre les canons classiques et les canons modulo p est bien sûr la superposition contrôlée des *onsets* dans le second cas. Si les canons de Vuza modulo p compacts sont très simples à obtenir comme le montre la remarque 4.2.9, ce n'est pas le cas des canons de Vuza classiques. Dans l'espoir que le pavage modulo p puisse nous donner des informations sur le pavage classique, il faut s'intéresser à leur différence fondamentale. Lorsqu'à la pulsation t, $d > 1$ *onsets* sont joués en même temps, on ne parle plus d'*onset* mais de d-*onset* ou encore de *donset*.

Définition 5.1.1. Si (A, B) est un canon modulo p de $\mathbb{Z}_N$ on appelle multi-ensemble de donsets de ce canon le multi-ensemble $D_{(A,B),p}$, sous multi-ensemble de $C = \Pi_N(A + B)$ défini par

$$D_{(A,B),p} = \bigcup_{n\in\mathbb{Z}_N} \bigcup_{\substack{i=1 \\ \mathbb{1}_C(n)=kp+1 \\ \text{avec } k\neq 0}}^{k} \{n\}\,.$$

C'est-à-dire tel que pour tout, $n \in \mathbb{Z}_{\mathbb{N}}$,

$$\mathbb{1}_{D_{(A,B),p}}(n) = \max(0, \frac{\mathbb{1}_C(n) - 1}{p}).$$

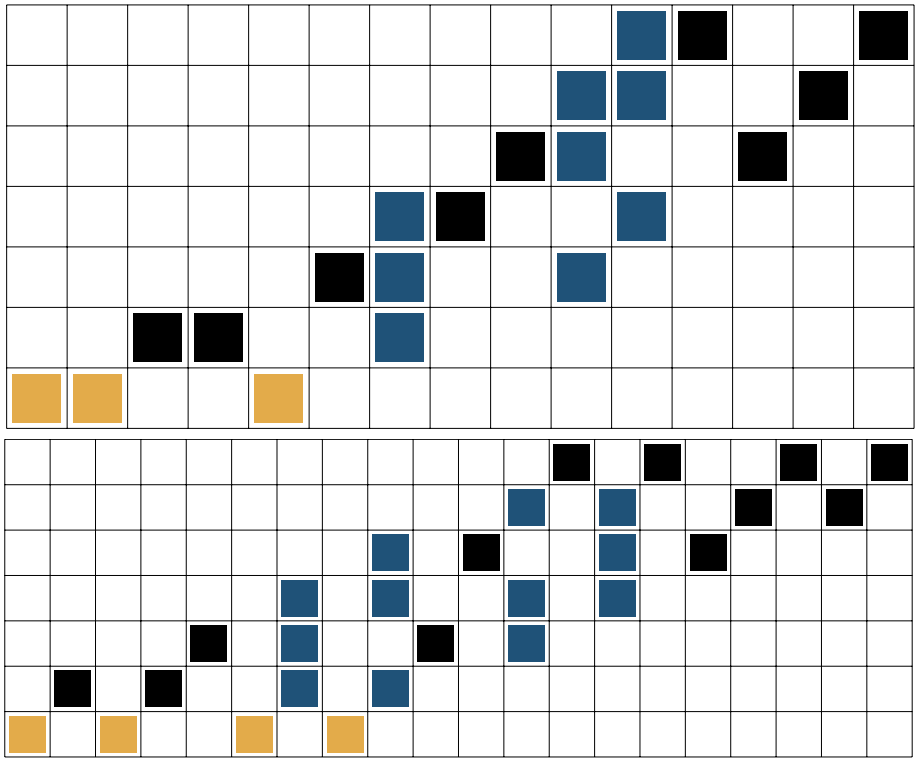


FIGURE 5.1: Les canons compacts modulo 2 obtenus avec les motifs en jaune $\{0,1,4\}$ (haut) ou $\{0,2,5,7\}$ (bas) ou leur 3-onsets mis en évidence en bleu

Exemple 5.1.2. Le canon $(\{0,1,3\},\{0,2,3\})$ de $\mathbb{Z}_7$ modulo 2 satisfait à $C = A+B = \{0,1,2,3,3,3,4,5,6,7\}$ et donc $D_{(A,B),2} = \{3\}$.

Il faudrait pour mieux comprendre le lien entre canons et donsets pouvoir exprimer une fonction simple qui à un couple associe son multi-ensemble de donsets $D\colon (A,B) \mapsto D_{(A,B),p}$. On peut imaginer que le lien est plus simple que de devoir écrire le pavage puis regarder a posteriori les donsets. On peut voir par exemple sur la Figure 5.1 des exemples de canons de petite taille N dont les donsets tombent sur le miroir du motif de base.

Seulement, on ne peut pas espérer obtenir de bijection entre $\mathcal{P}(\mathbb{N})\times\mathcal{P}(\mathbb{N})$ et son image par D. En effet, on peut trouver des canons distincts $(A,B) \neq (A',B')$ tels que $D_{(A,B),p} = D_{(A',B'),p}$. Un exemple évident est de prendre un canon et son dual.

Proposition 5.1.3. *Soit (A,B) un canon modulo p de $\mathbb{Z}_N$, alors*

$$D_{(A,B),p} = D_{(B,A),p}.$$

Démonstration. Par commutativité dans $\mathbb{Z}_N$, on a $C = \Pi_N(A+B) = \Pi_N(B+A)$. Comme (A,B) est un canon modulo p de $\mathbb{Z}_N$, le multi-ensemble C est tel que pour tout $n \in \mathbb{Z}_N$, il existe $k_n \in \mathbb{N}$ tel que

$\mathbb{1}_C(n) = k_n p + 1$. Alors pour les deux canons, les donsets correspondent aux $k_n \neq 0$ et sont sur les mêmes pulsations $\pi_N(a+b) = \pi_N(b+a)$ et sont en même quantité k_n. □

Cet exemple est assez évident, seulement il existe des transformations moins triviales pour créer des canons différents ayant les mêmes donsets.

5.2 Deux transformations modulo 2 pour avoir des donsets identiques

Théorème 5.2.1. *Si (A,B) est un canon rythmique modulo 2 de $\mathbb{Z}_N$, alors pour tout $k \in \mathbb{N}^*$ les couples $(A_k, \widetilde{B}_k)$ et $(\widetilde{A}_k, B_k)$ sont des canons modulo 2 de $\mathbb{Z}_{kN}$ et ils satisfont à*

$$D_{(A_k,\widetilde{B}_k),2} = D_{(\widetilde{A}_k,B_k),2}$$

avec $A_k = \{ka \mid a \in A\}$ et $\widetilde{A}_k = \{ka, ka+1, \ldots, ka+(k-1) \mid a \in A\}$ (et de même pour B).

Démonstration. Les polynômes $A(X)$ et $B(X)$ satisfont à

$$A(X) \cdot B(X) = c_0 + c_1 X + \cdots + c_{n-1} X^{n-1}$$

avec pour tout $i \in \{0, \ldots, N-1\}$,

$$\sum_{j=i \bmod N} c_j = 2k_i + 1.$$

Les transformations sur les motifs rythmiques peuvent s'écrire sur leur représentation polynomiale.

Si $A = \{0, a_1, \ldots, a_{n-1}\}$, alors par définition

$$A(X) = 1 + X^{a_1} + \cdots + X^{a_{n-1}}$$

puis

$$A_k(X) = 1 + X^{ka_1} + \cdots X^{ka_{n-1}} = A(X^k)$$

et alors

$$\begin{aligned}
\widetilde{A}_k(X) = &\left(1 + X^{ka_1} + \cdots X^{ka_{n-1}}\right) + \left(X + X^{ka_1+1} + \cdots X^{ka_{n-1}+1}\right) + \cdots \\
&+ \left(X^{k-1} + X^{ka_1+k-1} + \cdots X^{ka_{n-1}+k-1}\right) \\
= &\left(1 + X^{ka_1} + \cdots X^{ka_{n-1}}\right) + X\left(1 + X^{ka_1} + \cdots X^{ka_{n-1}}\right) + \cdots \\
&+ X^{k-1}\left(1 + X^{ka_1} + \cdots X^{ka_{n-1}}\right) \\
= &A_k(X)\left(\frac{X^k - 1}{X - 1}\right)
\end{aligned}$$

$$\begin{aligned}
A_k(X)\cdot\widetilde{B}_k(X) =& A_k(X)\cdot B_k(X)\frac{X^k-1}{X-1}\\
=& A(X^k)\cdot B(X^k)\frac{X^k-1}{X-1}\\
=& c_0+c_1X^k+\cdots+c_{n-1}X^{k(n-1)}\frac{X^k-1}{X-1}\\
=& c_0+c_0X+\cdots c_0X^{k-1}+c_1X^k+\cdots+c_1X^{2k-1}\\
&+c_2X^{2k}+\cdots+c_{n-1}X^{kn-1}\\
=& d_0+d_1X+\cdots+d_{kn-1}X^{kn-1}
\end{aligned}$$

avec pour tout $i\in\{0,\dots,kN-1\}$,

$$\sum_{j=i \bmod kN} d_j = \sum_{j=i \bmod kN} c_{\lfloor\frac{j}{k}\rfloor} = \sum_{j=l \bmod N} c_l = \sum_{j=l \bmod N} 2k_l+1.$$

Comme

$$\begin{aligned}
A_k(X)\cdot\widetilde{B}_k(X) &= A_k(X)\cdot B_k(X)\frac{X^k-1}{X-1}\\
&= A_k(X)\frac{X^k-1}{X-1}\cdot B_k(X)\\
&= \widetilde{A}_k(X)\cdot B_k(X)
\end{aligned}$$

Alors les couples $(A_k,\widetilde{B}_k)$ et $(\widetilde{A}_k,B_k)$ sont des canons modulo 2 de $\mathbb{Z}_{kN}$ et leur donsets tombent sur les mêmes pulsations i appartenant à l'ensemble $\{0,\dots,kN-1\}$ où $\sum_{j=i \bmod kN} k_{\lfloor\frac{j}{k}\rfloor}\neq 0$.

□

Exemple 5.2.2. Avec le canon modulo 2 $(\{0,2,3,5\},\{0,1,3,5,6\})$ de $\mathbb{Z}_{12}$, on peut créer deux canons distincts qui ont les mêmes donsets. Avec le théorème 5.2.1 et $k=2$ on crée les deux canons modulo 2 de $\mathbb{Z}_{24}$

$$(A_k,\widetilde{B}_k)=(\{0,4,6,10\},\{0,1,2,3,6,7,10,11,12,13\})$$

et

$$(\widetilde{A}_k,B_k)=(\{0,1,4,5,6,7,10,11\},\{0,2,6,10,12\}).$$

Ces trois canons sont représentés sur la Figure 5.2, et on voit bien que $D_{(A_k,\widetilde{B}_k),2}=D_{(\widetilde{A}_k,B_k),2}$.

Théorème 5.2.3. *Si (A,B) est un canon rythmique modulo 2 de $\mathbb{Z}_N$, alors pour tout d diviseur strict de N, on note pour A (et de même pour B)*

$$A'_d=A+\{0,d\}$$

et

$$\widetilde{A}'_d=A+\left\{0,d,2d\dots,\left(\frac{N}{d}-1\right)d\right\}.$$

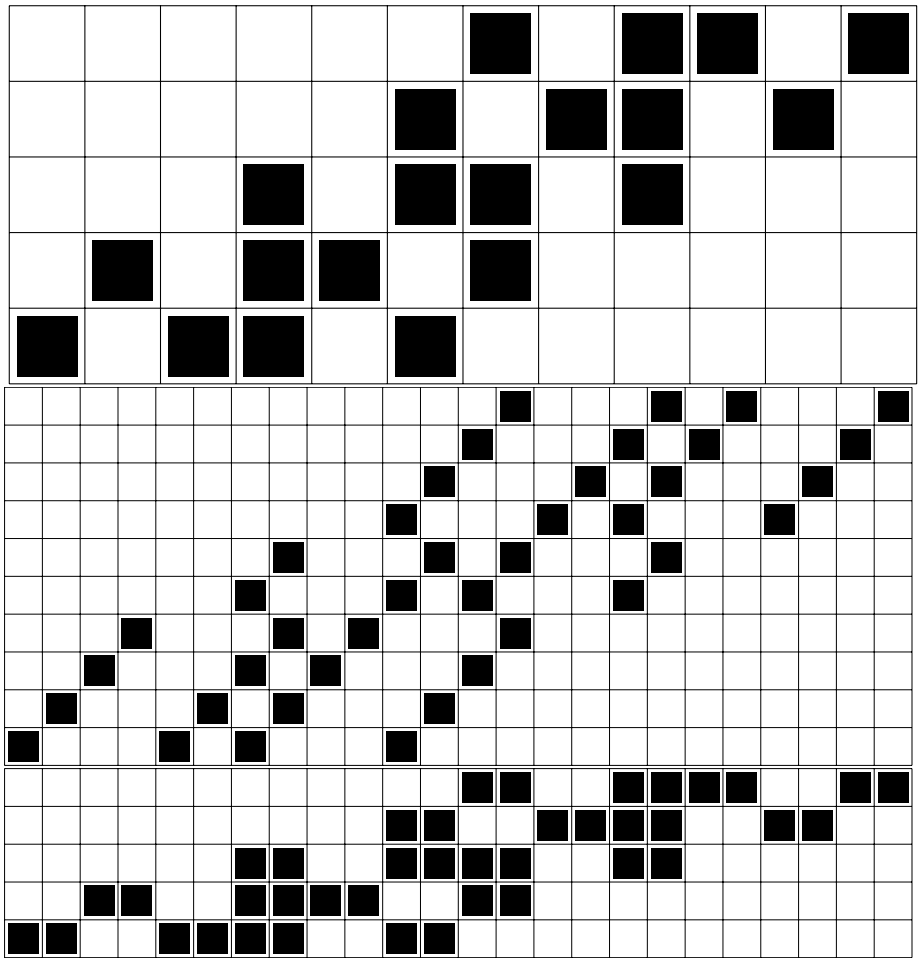


FIGURE 5.2: Le canon $(\{0,2,3,5\},\{0,1,3,5,6\})$ (haut) et ses deux transformations obtenues par le théorème 5.2.1 (bas).

et A_d et $\widetilde{A}_d$ sont les polyrythmes modulo 2 de A'_d et $\widetilde{A}'_d$.

Alors les couples $(A_d, \widetilde{B}_d)$ et $(\widetilde{A}_d, B_d)$ sont des canons modulo 2 de $\mathbb{Z}_{2N}$. Ils satisfont à

$$D_{(A_d,\widetilde{B}_d),2} = D_{(\widetilde{A}_d,B_d),2}$$

si et seulement si le polyrythme $\widetilde{A}'_d$ (et de même pour $\widetilde{B}'_d$) satisfait à $\mathbb{1}_{\widetilde{A}'_d}(n) > 1 \Rightarrow n = 0 \mod d$.

Démonstration. Les polynômes associé aux motifs transformés sont

$$A'_d(X) = A(X) + X^d A(X) = A(X)\left(1 + X^d\right)$$

et

$$\begin{aligned}\widetilde{A}'_d(X) &= A(X) + X^d A(X) + X^{2d} A(X) + \ldots + X^{\left(\frac{N}{d}-1\right)d} A(X) \\ &= A(X)\left(1 + X^d + X^{2d} + \ldots + X^{\left(\frac{N}{d}-1\right)d}\right)\end{aligned}$$

Comme $\widetilde{A}'_d$ est tel que $\mathbb{1}_{\widetilde{A}'_d}(n) > 1 \Rightarrow n = 0 \mod d$, cela signifie que les seuls monômes de $\widetilde{A}'_d(X)$ qui peuvent s'annuler, lorsqu'on considère $\widetilde{A}_d(X)$ son polyrythme modulo 2, sont ceux de la forme X^n avec $n = 0 \mod d$. A étant un motif rythmique, ils ne peuvent que provenir du produit et donc d'un terme en $X^{m_A d}$ monôme de A et d'un terme $X^{m_D d}$ du second facteur, et donc que A ne possède pas d'éléments distincts a_1, a_2 non nuls modulo d tels que $a_1 = a_2 \mod d$, car sinon, en écrivant $a_1 = a_2 + kd$ avec $k \in \mathbb{N}^*$, les termes $X^{a_1} \cdot 1 = X^{a_2} \cdot X^{kd}$ apparaissent deux fois dans le produit, et donc $\mathbb{1}_{\widetilde{A}'_d}(a_1) > 1$ avec $a_1 \neq 0 \mod d$.

Si nous considérons nos polynômes dans $\mathbb{Z}[X]$, alors on peut écrire

$$A_d(X) = A'_d(X) - \sum_{i=0}^{N} p_i X^{id}$$

$$\widetilde{A}_d(X) = \widetilde{A}'_d(X) - \sum_{i=0}^{N} \tilde{p}_i X^{id}$$

avec pour tout i, $0 \leq p_i \leq \tilde{p}_i$ pairs. Les mêmes notations s'appliquent à B et on note q_i au lieu de p_i.

$$\begin{aligned}
\widetilde{A}_d(X)B_d(X) &= \left(\widetilde{A}'_d(X) - \sum_{i=0}^{N} \tilde{p}_i X^{id}\right)\left(B'_d(X) - \sum_{i=0}^{N} q_i X^{id}\right) \\
&= \left(\widetilde{A}'_d(X)B'_d(X)\right) + \sum_{i=0}^{N}(\tilde{p}_i + q_i)X^{id} \\
&\quad - \left(\widetilde{A}'_d(X)\sum_{i=0}^{N} q_i X^{id} + B'_d(X)\sum_{i=0}^{N}\tilde{p}_i X^{id}\right) \\
&= \left(A(X)B(X)\left(1+X^d\right)\left(1+X^d+X^{2d}+\ldots+X^{\left(\frac{N}{d}-1\right)d}\right)\right) \\
&\quad + \sum_{i=0}^{N}(\tilde{p}_i + q_i)X^{id} - \left(\widetilde{A}'_d(X)\sum_{i=0}^{N} q_i X^{id} + B'_d(X)\sum_{i=0}^{N}\tilde{p}_i X^{id}\right) \\
&= \left(A(X)B(X)\left(1+2X^d+2X^{2d}+\ldots+2X^{\left(\frac{N}{d}-1\right)d}+X^N\right)\right) \\
&\quad + \sum_{i=0}^{N}(\tilde{p}_i + q_i)X^{id} - \left(\widetilde{A}'_d(X)\sum_{i=0}^{N} q_i X^{id} + B'_d(X)\sum_{i=0}^{N}\tilde{p}_i X^{id}\right)
\end{aligned}$$

Alors dans $\mathbb{F}_2[X]$, les p_i et q_i étant pairs, on obtient l'équation

$$\begin{aligned}
\widetilde{A}_d(X)B_d(X) &= A(X)B(X)\left(1+0+X^N\right)+0 \\
&= \left(1+X+\ldots+X^{N-1}\right)\left(1+X^N\right) \quad = 1+X+\ldots+X^{2N-1}
\end{aligned}$$

Donc $(\widetilde{A}_d(X), B_d(X))$ est bien un canon modulo 2 de $\mathbb{Z}_{2N}$ et comme les rôles de A et B sont interchangeables dans $\mathbb{Z}[X]$, alors $(A_d, \widetilde{B}_d)$ est aussi un canon et il a les mêmes donsets que $(\widetilde{A}_d, B_d)$.

□

Exemple 5.2.4. $(A,B) = (\{0,3,4\},\{0,1,2,4,5,8,10\})$ est un canon modulo 2 de $\mathbb{Z}_{15}$. Alors avec $d = 5$ qui divise 15 on note les polyrythmes

$$A'_5 = A + \{0,5\} = \{0,3,4,5,8,9\}$$

$$B'_5 = B + \{0,5\} = \{0,1,2,4,5,5,6,7,8,9,10,10,13,15\}$$

$$\widetilde{A}'_5 = A + \{0,5,10\} = \{0,3,4,5,8,9,10,13,14\}$$

$$\begin{aligned}
\widetilde{B}'_5 &= B + \{0,5,10\} \\
&= \{0,1,2,4,5,5,6,7,8,9,10,10,10,11,12,13,14,15,15,18,20\}
\end{aligned}$$

et leur polyrythme modulo 2 associés $A_5 = A'_5$, et

$$B_5 = \{0,1,2,4,6,7,8,9,13,15\}$$

$$\widetilde{A}_5 = \{0,3,4,5,8,9,10,13,14\}$$

$$\widetilde{B}_5 = \{0, 1, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 20\}.$$

Alors d'après le théorème 5.2.3, comme les seuls points où $\mathbb{1}_{\widetilde{B}'_5}(n) > 1$ satisfont à $n = 5, 10, 15 = 0 \mod 5$, les couples $(A_5, \widetilde{B}_5)$ et $(\widetilde{A}_5, B_5)$ sont des canons modulo 2 de $\mathbb{Z}_{30}$.

Ces trois canons sont représentés sur la Figure 5.3, et on voit bien que $D_{(A_5,\widetilde{B}_5),2} = D_{(\widetilde{A}_5,B_5),2}$.

Exemple 5.2.5. $(A, B) = (\{0, 2, 3, 4, 6\}, \{0, 1, 3, 4, 5, 7, 8\})$ est un canon modulo 2 de $\mathbb{Z}_{15}$. Alors avec $d = 5|15$ on note les polyrythmes

$$A'_5 = A + \{0, 5\} = \{0, 2, 3, 4, 5, 6, 7, 8, 9, 11\}$$
$$B'_5 = B + \{0, 5\} = \{0, 1, 3, 4, 5, 5, 6, 7, 8, 8, 9, 10, 12, 13\}$$
$$\widetilde{A}'_5 = A + \{0, 5, 10\} = \{0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16\}$$

$$\begin{aligned}\widetilde{B}'_5 &= B + \{0, 5, 10\} \\ &= \{0, 1, 3, 4, 5, 5, 6, 7, 8, 8, 9, 10, 10, 11, 12, 13, 13, 14, 15, 17, 18\}\end{aligned}$$

et leurs polyrythmes modulo 2 associés $A_5 = A'_5$ et $\widetilde{A}_5 = \widetilde{A}'_5$ et

$$B_5 = \{0, 1, 3, 4, 6, 7, 9, 10, 12, 13\}$$

$$\widetilde{B}_5 = \{0, 1, 3, 4, 6, 7, 9, 11, 12, 14, 15, 17, 18\}.$$

Comme $\mathbb{1}_{\widetilde{B}'_5}(n) > 1$ pour $n = 8 \neq 0 \mod 5$ par exemple, alors les couples $(A_5, \widetilde{B}_5)$ et $(\widetilde{A}_5, B_5)$ sont des canons modulo 2 de $\mathbb{Z}_{30}$ mais ne satisfont pas à $D_{(A_5,\widetilde{B}_5),2} = D_{(\widetilde{A}_5,B_5),2}$. Ces deux canons de $\mathbb{Z}_{30}$ sont représentés sur la Figure 5.4

Remarque 5.2.6. La plus grosse difficultés pour pouvoir associer motif rythmique et motif des donsets n'est pas de trouver des transformations, mais bien de prouver que l'on a toutes les transformations. La dualité est un exemple évident, et nous avons présenté deux transformations non triviales, seulement il reste des transformations non trouvées. Voici trois canons modulo 2 de $\mathbb{Z}_{30}$:

- $(\{0, 1, 5, 6, 10, 11\}, \{0, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18\})$
- $(\{0, 1, 2, 5, 6, 7\}, \{0, 3, 5, 6, 8, 9, 10, 11, 12, 13, 14, 16, 17, 19, 22\})$
- $(\{0, 1, 2, 5, 6, 7, 10, 11, 12\}, \{0, 3, 5, 6, 8, 9, 11, 12, 14, 17\})$

Ils sont représentés sur la Figure 5.5 et on peut voir qu'ils ont les mêmes donsets.

Les deux derniers canons viennent des transformations du canon de $\mathbb{Z}_{15}$ $(\{0, 1, 2\}, \{0, 3, 6, 9, 12\})$ avec le théorème 5.2.3 et $d = 5$.

Le premier canon quant à lui ne rentre pas dans les cas de figures présentés précédemment. Il semble venir d'une transformation de $(\{0, 1\}, \{0, 2, 4, 6, 8, 10, \text{“}12 \setminus 14\text{”}\})$ avec le théorème 5.2.3 et $d = 5$. En effet $(\{0, 1\}, \{0, 2, 4, 6, 8, 10, 12\})$ pave $\mathbb{Z}_{14}$ tandis que $(\{0, 1\}, \{0, 2, 4, 6, 8, 10, 12, 14\})$ pave $\mathbb{Z}_{16}$. Comme $\{0, 1\}$ ne peut paver

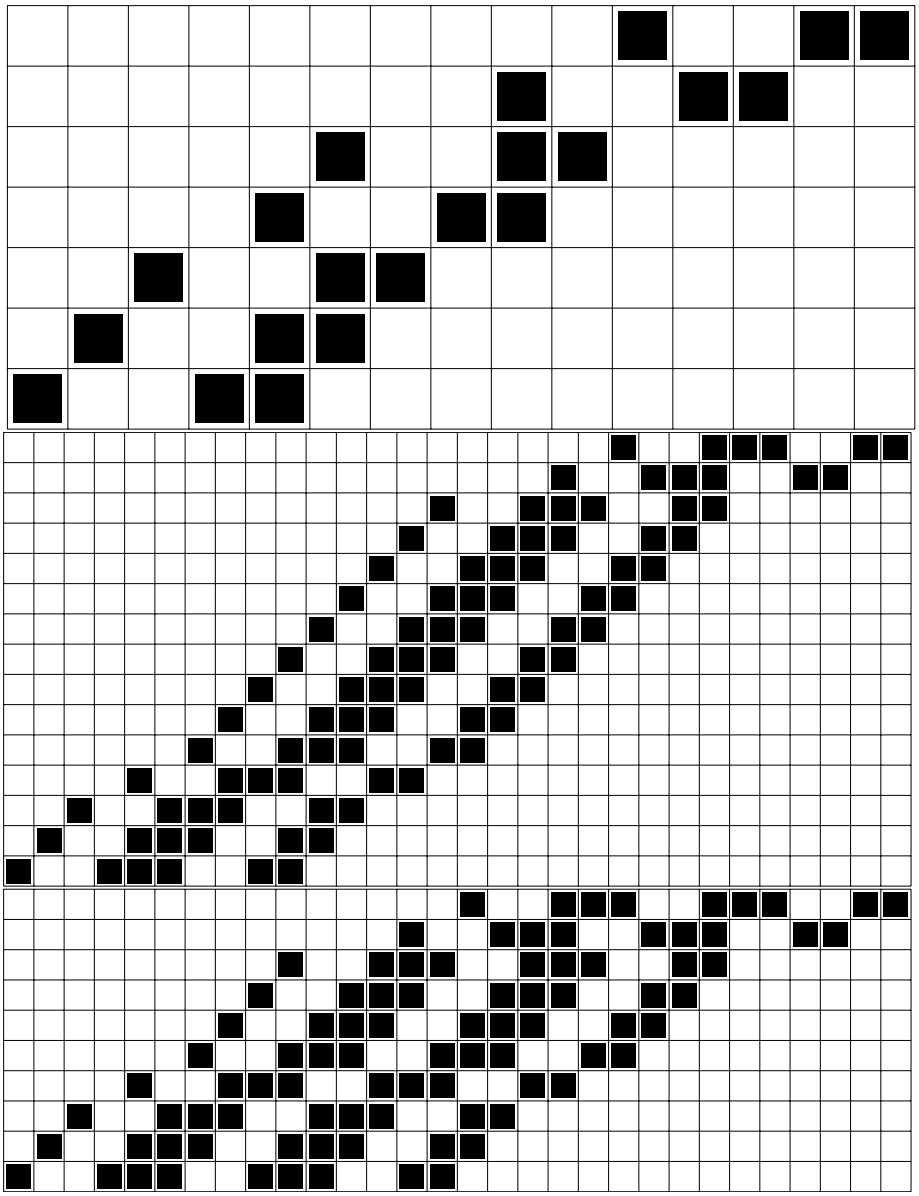


FIGURE 5.3: Le canon $\{0,3,4\}, \{0,1,2,4,5,8,10\}$ (haut) et ses deux transformations obtenues par le théorème 5.2.3 (bas).

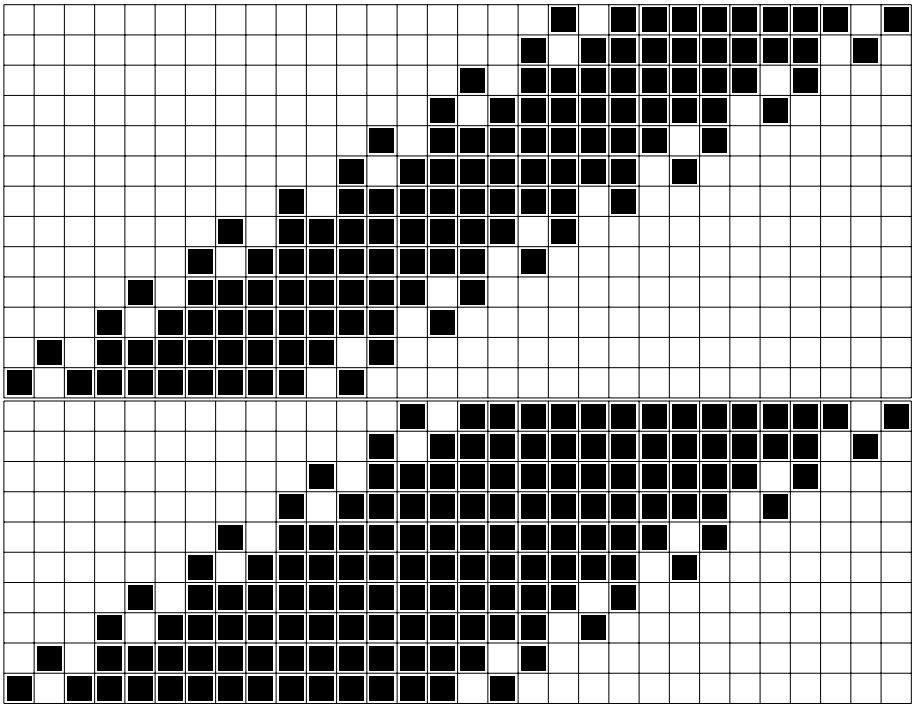


FIGURE 5.4: Les deux canons transformés de $(\{0,2,3,4,6\},\{0,1,3,4,5,7,8\})$ par le théorème 5.2.3 qui n'ont pas les mêmes donsets.

$\mathbb{Z}_{15}$, on n'aura pas $5 = d|15$ et nous sommes en dehors des conditions du théorème 5.2.3.

Cela nous laisse penser que le théorème 5.2.3 n'est peut-être qu'un cas particulier d'une autre transformation plus large. Il est malgré cela certain que les transformations liant les motifs rythmiques produisant les mêmes donsets ne sont pas toutes trouvées.

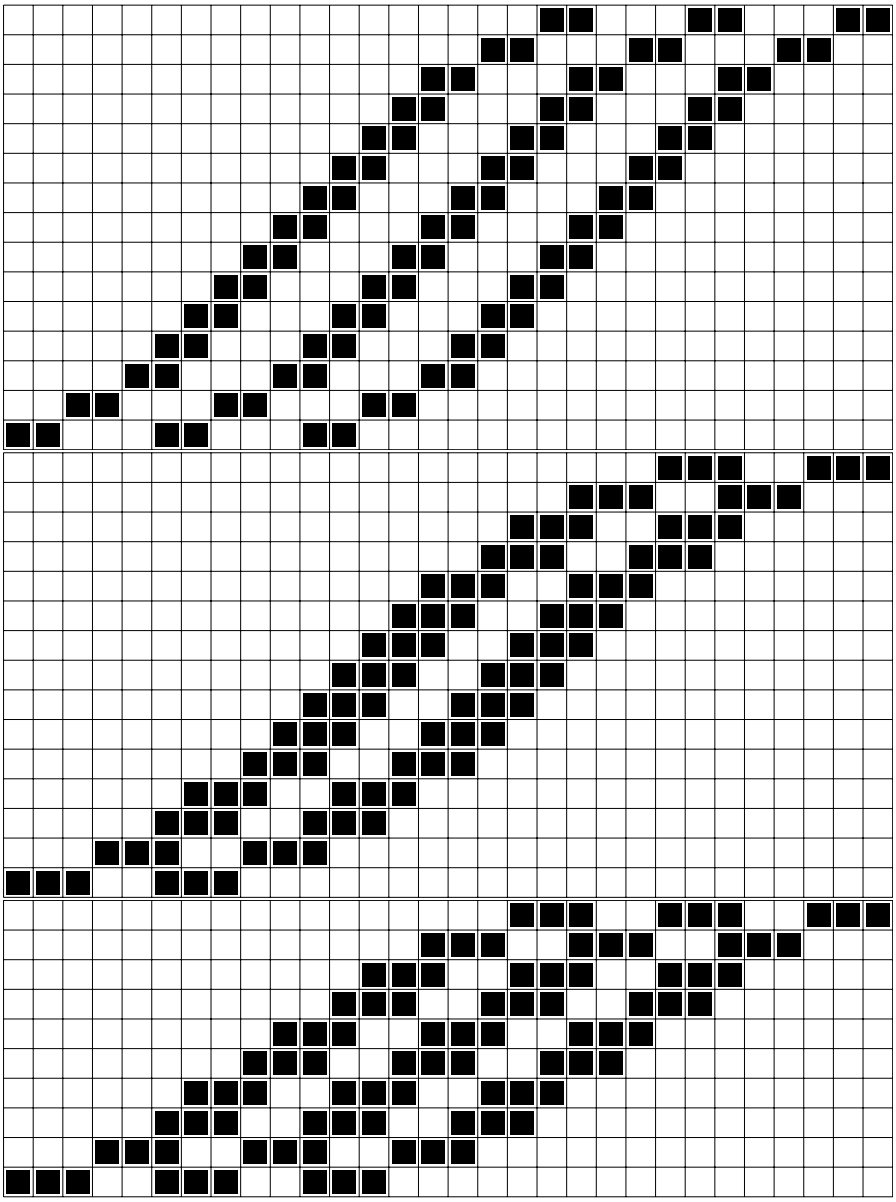


FIGURE 5.5: Trois canons de $\mathbb{Z}_{30}$ ayant les mêmes donsets.

6

Implémentation en OpenMusic

Résumé

Ce petit chapitre sert de frontière entre mathématiques et musique. Afin d'aider les compositeurs à utiliser ces nouveaux outils que sont les canons modulo p, on présente trois implémentations dans le langage OpenMusic de création et de transformations sur les canons modulo 2.

Open Music étant un langage de programmation visuelle, il permet aux non-informaticiens de travailler avec des notions complexes comme les canons rythmiques modulo 2, sans avoir à connaitre la théorie mathématique qui régit leur construction et les propriétés auxquelles ils satisfont. Ci-après sont présentés quelques *patchs* Open Music maintenant ajoutés aux MathTools. Ils peuvent permettre aux compositeurs et musiciens de construire des canons et de les transformer.

6.1 Implémentation de la construcution des pavages modulo 2

L'algorithme 1 permet de construire en temps linéaire le motif des entrées avec lequel un motif rythmique pavera modulo 2 de façon compacte. Le *patch* présenté sur la Figure 6.1 présente deux fonctions. La première `find_rythm_mod_2` permet de trouver le motif des entrées B lorsqu'on lui donne un motif rythmique A en utilisant l'algorithme 1. La fonction `make_mod_2_canon1` prend deux motifs rythmiques A, B sous forme d'ensemble d'entiers et représente musicalement le canon (A, B) modulo 2 en utilisant la contrainte que le motif A est joué pour chaque voix $b \in B$ à une tierce au dessus de la précédente.

On rappelle qu'un canon (A, B) est dit de Vuza modulo 2 si l'algorithme 1 renvoie B lorsqu'on lui donne A en entrée et A lorsqu'on lui donne B, comme cela est expliqué en remarque 4.2.9. Sur la Figure 6.2 on peut voir les deux canons modulo 2 $(\{0,1,6,7\},\{0,2,4\})$ et $(\{0,2,3,5\},\{0,1,3,5,6\})$. Il est facile de constater visuellement que le premier peut se simplifier par opérations de concaténation, tandis que le second est de Vuza.

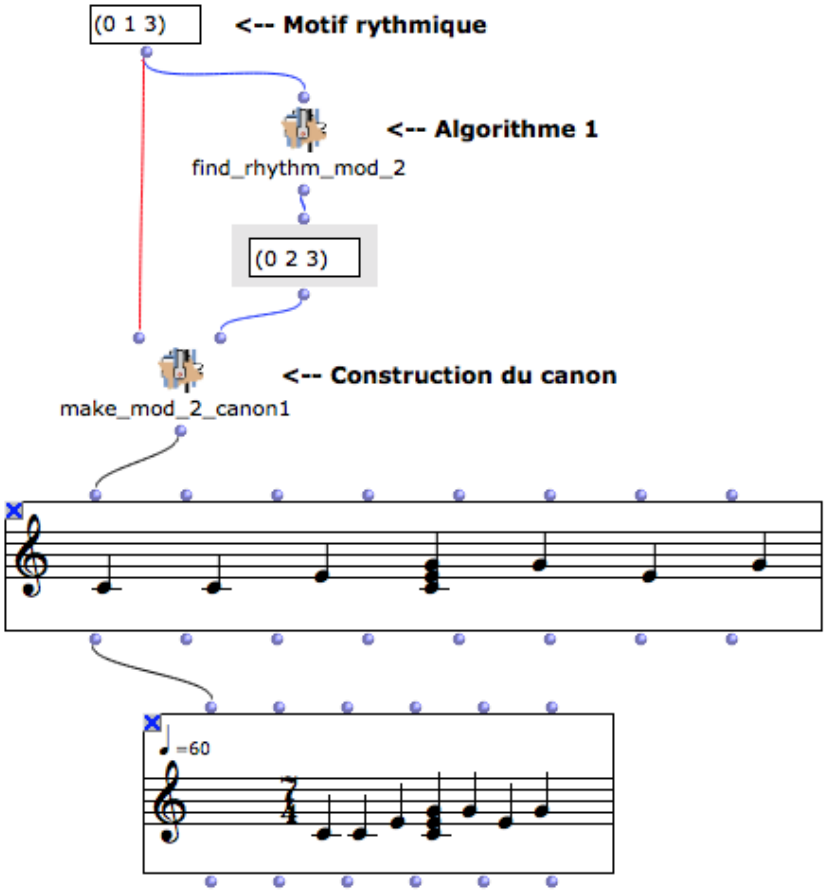


FIGURE 6.1: Un patch Open Music permettant de construire un pavage compact modulo 2.

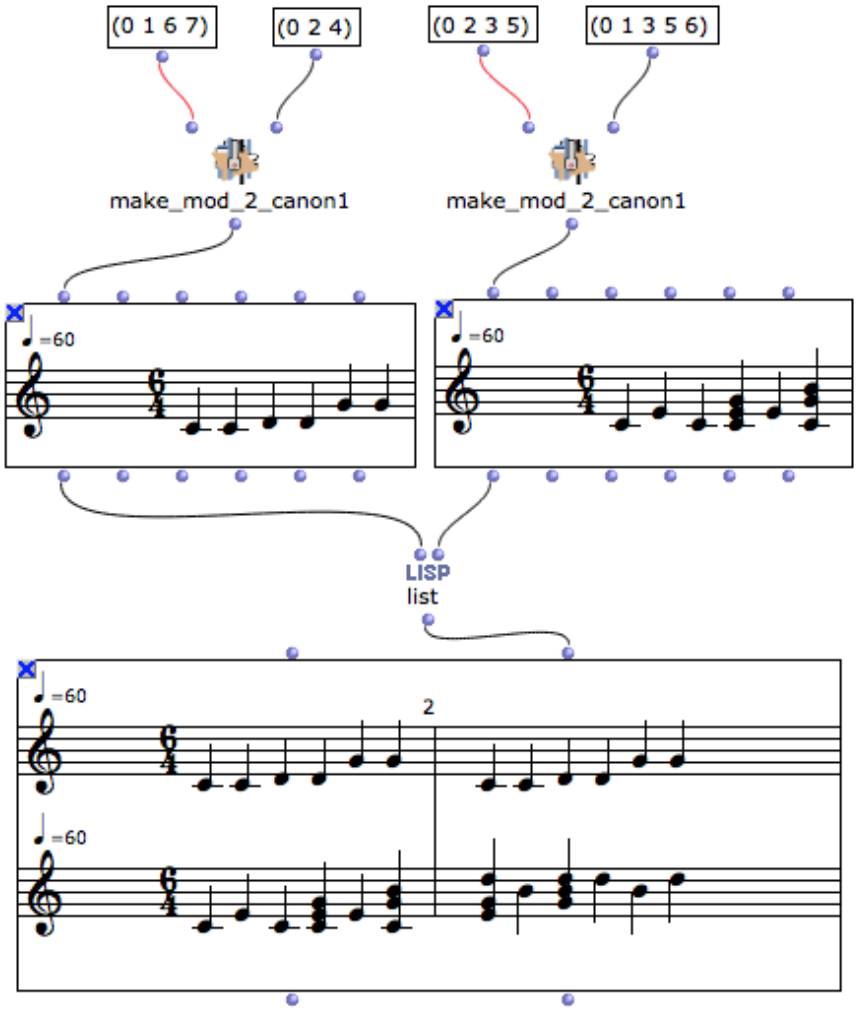


FIGURE 6.2: Deux canons modulo 2 sont représentés dans un patch Open Music, le premier n'est pas de Vuza, tandis que le second l'est.

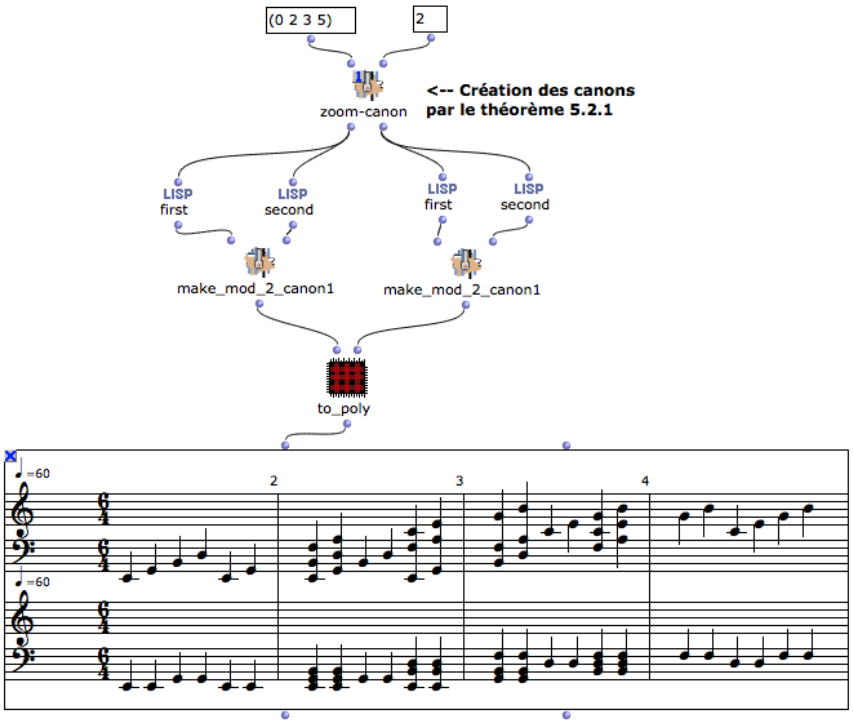


FIGURE 6.3: Un patch Open Music permettant de créer deux canons ayant les mêmes donsets grâce au théorème 5.2.1

6.2 Implémentation des transformations pour obtenir des donsets identiques

Le théorème 5.2.1 permet de créer à partir d'un canon de $\mathbb{Z}_N$ deux canons de $\mathbb{Z}_{kN}$ ayant les mêmes donsets. La Figure 6.3 présente un patch permettant de créer de tels canons. La fonction `zoom-canon` prend en entrées un motif rythmique A et un facteur k. Il crée le motif des entrées B pavant avec A modulo 2 puis renvoie les deux canons $(A_k, \widetilde{B}_k)$ et $(\widetilde{A}_k, B_k)$ ayant les mêmes donsets.

Le théorème 5.2.3 permet quant à lui de créer deux canons $(A_d, \widetilde{B}_d)$ et $(\widetilde{A}_d, B_d)$ de $\mathbb{Z}_{2N}$ à partir d'un canon (A, B) de $\mathbb{Z}_N$ et ces deux canons ont les mêmes donsets si et seulement s'ils satisfont à une propriété de polyrythmie. Sur la Figure 6.4 on peut voir un patch Open Music permettant de créer de tels canons. Premièrement, la fonction `is-transferable_p` prend en entrée un motif rythmique A, crée son motif des entrées B et la taille N de son canon compact modulo 2 minimal, puis elle vérifie s'il existe des d tels que $d|N$ et si A et B satisfont à la condition énoncée plus tôt. Elle renvoie une liste de tous ces d potentiels. La fonction `transfer-transform` prend un motif rythmique A en entrée et un d qui satisfont à la condition. Elle crée en-

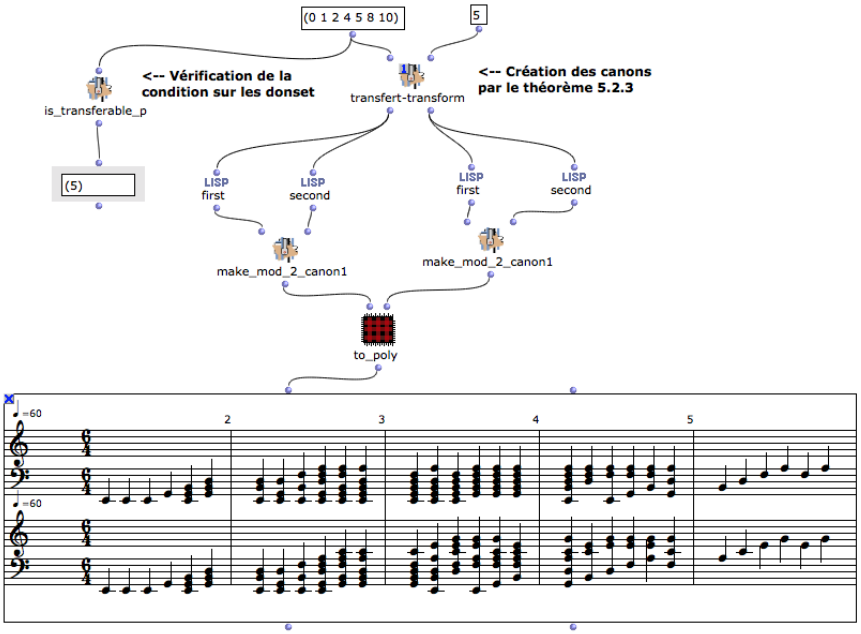


FIGURE 6.4: Un patch Open Music permettant de créer deux canons ayant les mêmes donsets grâce au théorème 5.2.3

suite le motif des entrées B puis les deux canons $(A_d, \widetilde{B}_d)$ et $(\widetilde{A}_d, B_d)$ ayant les mêmes donsets.

7

Accélération des algorithmes de pavages modulo 2 et similitudes des motifs des entrées

Résumé

Ce chapitre regroupe les résultats principaux de cette thèse. Tous les résultats sont nouveaux, qu'ils soient des récurrences sur les motifs des entrées, des dénombrements ou des liens entre motifs. Les canons modulo p ayant été très peu étudiés avant cette thèse, toutes les notions développées dans ce chapitre ont été créées pour prouver ces résultats. Si ces notions, comme par exemple la *sous-couverture* ou le *remplissage*, sont définies modulo p, elles ne seront utilisées ensuite que pour démontrer nos théorèmes et propriétés qui sont dans la situation où $p = 2$. On commence par prouver des récurrences sur les motifs des entrées permettant de paver modulo 2 avec les motifs de la forme $\{0, 1, 2^k\}$ et $\{0, 1, 2^k + 1\}$ pour tout $k \in \mathbb{N}^*$. On démontre la première en montrant que l'algorithme 1 se comporte en remplissant des sous-couvertures de taille 2^k dans la section ouvrant ce chapitre, puis on prouve la seconde dans la section suivante en utilisant le premier résultat et en exposant une symétrie entre ces deux types de motifs des entrées. Ces deux récurrences permettent de paver modulo 2 avec ces familles de motifs sans avoir à utiliser l'algorithme 1, et en accélérant significativement la vitesse pour trouver le motif des entrées. Elles permettent aussi d'obtenir des résultats de dénombrement sur la taille des canons ainsi obtenus, sur le nombre d'entrées nécessaires ainsi que sur le nombre de donsets. Le lien entre ces deux premiers résultats laisse supposer d'une symétrie entre les motifs des entrées pour paver modulo 2 avec des motifs de la forme $\{0, 1, 2n\}$ et $\{0, 1, 2n + 1\}$ pour tout $n \in \mathbb{N}^*$. D'autres liens sont ensuite présentés entre les motifs des entrées permettant de paver avec les motifs de la forme $\{0, 1, n\}$ et $\{0, 1, m\}$. Grâce à la parité conjecturée du nombre de 1 sur des sous-mots du motif des entrées toutes les m ou n lettres, on peut montrer de façon conditionnelle que les motifs des entrées sont semblables jusqu'à un certain point, puis conjugués l'un de l'autre.

7.1 Mise en place du modèle sous forme de tableaux

Nous rappelons que nous notons de façon équivalente un motif rythmique en notation ensembliste, polynomiale ou comme un mot fini ou presque nul, comme défini en 2.2.12 et 2.2.16.

Définition 7.1.1. Un motif rythmique (polyrythmique) est noté de façon équivalente comme

1. un (multi)ensemble fini A d'entiers naturels
2. un polynôme $A(X) \in \{0,1\}[X]$ ($\in \mathbb{Z}[X]$)
3. un mot fini ou presque nul a de l'alphabet $\{0,1\}$ ($\mathbb{Z}$).

Rappelons aussi que si nous cherchons à paver modulo p, nous pouvons considérer comme en 2.2.18 des polyrythmes modulo p.

Définition 7.1.2. À un polyrythme, on peut associer son polyrythme modulo p :

1. Si A est un multi-ensemble, A_p a les mêmes éléments que A avec leurs multiplicités considérées modulo p
2. Si $A(X) \in \mathbb{Z}[X]$ est un polynôme, le polynôme $A_p(X) \in \{0,\dots,p-1\}[X]$ est la somme des mêmes monômes dont les coefficients sont considérés modulo p
3. Si a est un mot de $\mathbb{Z}$, a_p est le mot de l'alphabet $\{0,\dots,p-1\}$ tel que sa $i$ème lettre vaut le représentant de la classe d'équivalence modulo p de la $i$ème lettre de a.

Dans ce chapitre nous allons nous intéresser aux pavages modulo 2 avec des motifs de la forme $A(n) = \{0,1,n\}$, et plus particulièrement aux motifs des entrées associés $B(n)$ permettant d'obtenir un pavage compact modulo 2 comme décrit dans l'algorithme 1.

Définition 7.1.3. Soit le motif rythmique $A(n) = \{0,1,n\}$, on note $B(n)$ son motif des entrées obtenu par l'algorithme 1, c'est-à-dire tel que $(A(n), B(n))$ est un canon rythmique compact modulo 2 de $\mathbb{Z}_{\max(B(n))+n-1}$. Et on note ces motifs rythmiques $a(n)$ et $b(n)$ s'ils sont vus comme des mots.

Mais commençons par présenter des notions sur les pavages modulo p :

Définition 7.1.4. Soit A et B deux motifs rythmiques quelconques, qui ne pavent pas forcément ensemble et on note $C = A + B$ le polyrythme de leur somme.

Comme A et B sont non vides (ils contiennent 0), le mot c_p commence avec un certain nombre de 1. On note m le plus petit indice tel que $c_p[m] \neq 1 \mod p$.

Comme les deux motifs A et B sont finis, le mot c_p est fini (ou presque nul), et on note M le plus grand indice tel que $c_p[M] \neq 0$.

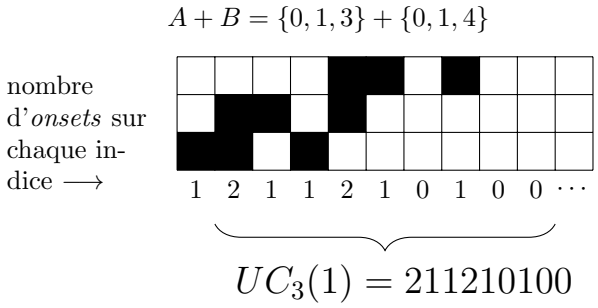


FIGURE 7.1: Représentation graphique d'une sous-couverture.

Définition 7.1.5. Avec les notations précédentes, on appelle UC_p une *sous-couverture modulo* p du couple (A, B) un sous-mot de c_p de longueur au moins $M - m + 1$ tel que pour tout n

$$UC_p[n] = c_p[n + m]$$

ou de manière équivalente ce mot commençant par un nombre fini u de 1, avec $0 \leq u < m$, ou finissant par un nombre fini de 0. Lorsque cela est nécessaire, on dit que la sous-couverture commence à l'indice $i = m - u$ et on la note $UC_p(i)$.

Exemple 7.1.6. Soient $A = \{0, 1, 3\}$ et $B = \{0, 1, 4\}$, alors

$$A + B = \{0, 1, 1, 2, 3, 4, 4, 5, 7\},$$

et une sous-couverture modulo 3 de (A, B) est

$$UC_3(1) = 2112101$$

ou encore

$$UC_3(0) = 12112101000.$$

On peut comprendre graphiquement cette sous-couverture qui est représentée sur la Figure 7.1.

Pour d'autres exemples de sous-couvertures, le lecture est invité à revenir en arrière à l'exemple 4.2.4.

Remarque 7.1.7. La notion de sous-couverture peut se lire comme une étape de l'algorithme 1. Lorsque l'ensemble des entrées est construit pour faire paver un motif rythmique, les premiers indices pavent et on avance dans la construction. L'algorithme remplit successivement de 1 les premiers indices de c_p : c'est le début du pavage et on peut ne s'intéresser qu'à la partie au delà de cet indice, nommé m dans 7.1.4. Au delà d'un certain indice, nommé M dans 7.1.4, il ne reste que des 0 et on peut paver sans se soucier des *onsets* déjà présents sur la couverture. La sous-couverture de taille minimale $M - m + 1$ contient exactement l'information nécessaire à la poursuite du pavage par l'algorithme 1. Une sous-couverture d'un couple (A, B) est définie de manière unique par son indice et sa taille.

Remarque 7.1.8. Si (A, B) est un canon modulo p, alors toutes ses sous-couvertures modulo p seront de la forme 1^k0^l, avec $k, l \in \mathbb{N}$, c'est-à-dire le mot vide si $k = l = 0$.

Définition 7.1.9. Soient A et B deux motifs rythmiques quelconques, qui ne pavent pas forcément ensemble, et UC_p la sous-couverture de longueur n et commençant à l'indice i. On dit qu'on remplit cette sous-couverture avec $B' \subset \mathbb{N}$ fini, si en notant C le multi-ensemble

$$C = \left(A + \left(B \bigcup (B' + \{i\})\right)\right)$$

on a

$$c_p\left[i, \ldots, i + n - 1\right] = 1^n.$$

Remarque 7.1.10. Le motif B' ne dépend pas de B au sens où si (A, B_1) et (A, B_2) donnent les mêmes sous-couvertures $UC_p(i)_1 = UC_p(i)_2$, alors on aura besoin du même B' pour les remplir.

Exemple 7.1.11. Soient $A = \{0, 1, 4\}$ et $B = \{0, 2, 5\}$. Alors

$$D = A + B = \{0, 1, 2, 3, 4, 5, 6, 6, 9\}$$

c'est-à-dire

$$d_2 = 1111110001\,(= 1111112001)$$

et une sous-couverture modulo 2 commençant à l'indice 6 est $UC_2(6) = 0001$. On peut remplir cette sous-couverture avec le motif des entrées $B' = \{0, 2, 3\}$.

En effet, on a

$$\begin{aligned} C =& A + \left(B \cup \left(B' + \{6\}\right)\right) \\ =& \{0, 1, 2, 3, 4, 5, 6, 6, 6, 7, 8, 9, 9, 9, 10, 10, 12, 13\} \end{aligned}$$

soit $c_2\left[6, \ldots, 9\right] = 1111$.

Remarquons, comme cela est illustré sur la Figure 7.2, que si nous avions choisi une sous-couverture plus longue, par exemple $\widetilde{UC_2}(6) = 000100$, alors on aurait dû la remplir avec $\widetilde{B}' = \{0, 2, 3, 4\}$ et non plus avec B'. En effet si nous notons les multi-ensembles sommes $\widetilde{C} = (A + (B \cup (\widetilde{B}' + \{6\})))$ et $C = (A + (B \cup (B' + \{6\})))$, alors ils satisfont à $\tilde{c}_2\left[6, \ldots, 11\right] = 111111$ et $c_2\left[6, \ldots, 11\right] = 111100 \neq 1^6$.

Remarque 7.1.12. Paver de manière compacte avec le motif A est équivalent à remplir la sous-couverture nulle 0^k, pour un $k \in \mathbb{N}$, commençant à l'indice 0.

Définition 7.1.13. Si la sous-couverture $UC_p(i)$ de (A, B) est remplie avec B', on appellera *sous-couverture suivante* de $UC_p(i)$ une sous-couverture de $(A, B \cup (B' + \{i\}))$ commençant à l'indice $i + |UC_p(i)|$.

Exemple 7.1.14. Dans l'exemple 7.1.11, une sous-couverture suivante de $UC_2(6)$ est $UC_2(10) = 0011$.

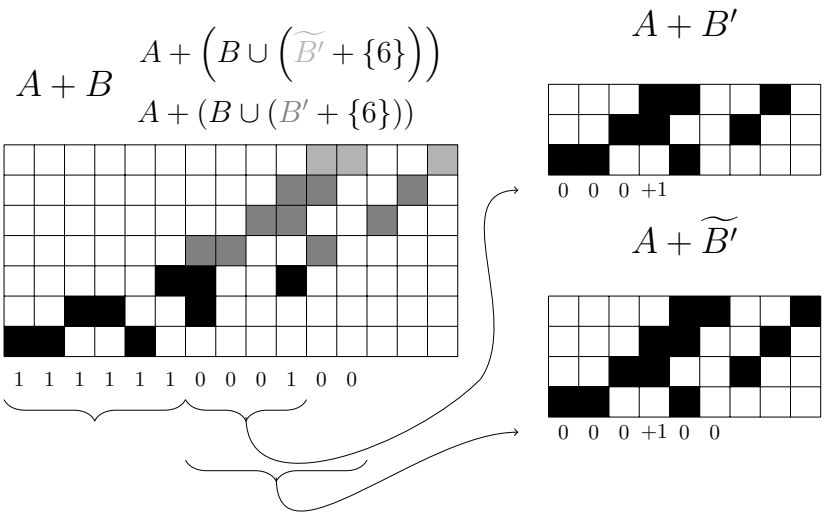


FIGURE 7.2: Deux remplissages de sous-couverture de longueur différente.

Revenons à nos motifs de la forme $A(n) = \{0, 1, n\}$, et regardons comment l'algorithme 1 crée linéairement le motif des entrées $B(n)$ pour obtenir un canon compact modulo 2 sur l'exemple $A(8)$.

À l'étape 1, l'algorithme place le motif en position 0, c'est-à-dire $B(8) = \{0\}$, et donc $(a(8) + b(8))_2 = 110000001$.

L'algorithme entre ensuite dans la boucle **while** et essaye de remplir la sous-couverture $UC_2(2) = 0000001$. Elle est remplie avec les entrées $B' = \{0, 2, 4\}$ qui nous donne après 3 passages dans la boucle les entrées

$$B(8) := B(8) \cup (\{0, 2, 4\} + \{2\}) = \{0, 2, 4, 6\}$$

c'est-à-dire

$$(a(8) + b(8))_2 = 111111111010101.$$

L'algorithme reprend ses passages dans la boucle **while** en essayant de remplir la sous-couverture suivante $UC_2(9) = 010101$, et ainsi de suite.

L'idée de l'algorithme à une échelle macroscopique est de remplir la sous-couverture actuelle, et ce faisant, une nouvelle sous-couverture est créée : la sous-couverture suivante de la définition 7.1.13. Puis on cherche à la remplir à nouveau jusqu'à obtenir un pavage compact, c'est-à-dire une sous-couverture minimale vide. Le motif des entrées $B(n)$ est construit de manière incrémentale, en remplissant les sous-couverture successives les unes après les autres lors de la boucle **while**.

Lemme 7.1.15. *Soit $i \in \mathbb{N}$, le motif rythmique $A(n) = \{0, 1, n\}$ étant de taille $\max(A(n)) - \min(A(n)) = n$, si pour remplir une sous-couverture de taille n commençant à l'indice i avec ce motif, il faut les entrées*

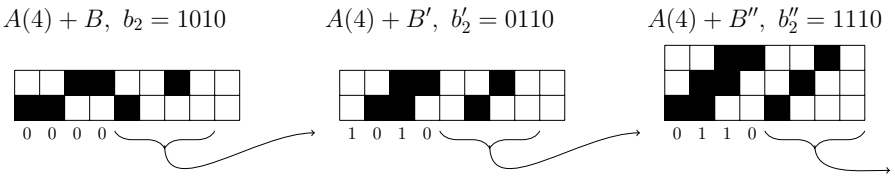


FIGURE 7.3: La sous-couverture suivante de taille n lorsqu'on pave avec $A(n)$ est le mot des entrées de remplissage.

B', alors la sous-couverture suivante de taille n (et on peut en trouver une de cette taille, quitte à la compléter par des 0) et commençant à l'indice $i+n$ est b'_2.

Démonstration. Comme la sous-couverture est de taille n, on peut supposer sans perte de généralité que tous les éléments b' de B' satisfont à $b' \leq n$. En effet s'il existe $b'' \in B'$ tel que $b'' > n$, alors pour tout $a \in A(n)$, $a + b'' + i > n + i$ et ces indices ne font pas partie de notre sous-couverture à remplir, on peut donc retirer b'' de B' sans que le remplissage de la sous-couverture ne soit changé.

Lorsque l'on remplit la sous-couverture avec B', les *onsets* a de $A(n)$ sont placés aux positions $a+b'+i$ pour tout $b' \in B'$, c'est-à-dire aux positions $b'+i$, $b'+i+1$, et $b'+i+n$. Lorsque la sous-couverture est remplie, il y a exactement un *onset* modulo 2 sur les positions avant $i+n$ puisque la sous-couverture était de taille n, et après cette position se trouvent exactement les *onsets* de la forme $b'+i+n$.

Ainsi, la sous-couverture suivante commençant à l'indice $i+n$ est formée avec les éléments $b'+i+n$ avec tous les $b' \in B'$, et donc elle vaut exactement b'_2. □

Exemple 7.1.16. Considérons $A(4) = \{0,1,4\}$ et $B' = \{0,2\}$, qui permet de remplir la sous-couverture nulle 0000 de taille 4 commençant à l'indice 0. Alors, en notant $C = A(2) + B'$ on obtient $c_2 = 11111010$, c'est-à-dire que la sous-couverture suivante commençant à l'indice $4+0=4$ et de taille 4 est 1010 qui vaut bien b'_2.

Cette étape de remplissage est représentée Figure 7.3

Sur la Figure 7.4, le canon compact modulo 2 qu'on obtient avec le motif $A(8)$ est représenté avec des séparations toutes les 8 pulsations. On peut ici facilement voir le dernier *onset* de $A(8)$ qui crée des sous-couvertures suivantes qui valent le motif utilisé pour remplir la sous-couverture 8 pulsations plus tôt.

Remarque 7.1.17. Soit $n > 1$ entier, on note $m = n$ si n est pair ou $m = n-1$ sinon, c'est-à-dire $m = 2\lfloor \frac{n}{2} \rfloor$. Une façon d'exploiter le phénomène présenté dans le lemme 7.1.15 si n est pair, est d'écrire le mot du motif des entrées $b(n)$ dans un tableau $T(n)$ ayant m colonnes en écrivant ce mot de gauche à droite puis de haut en bas. Ainsi

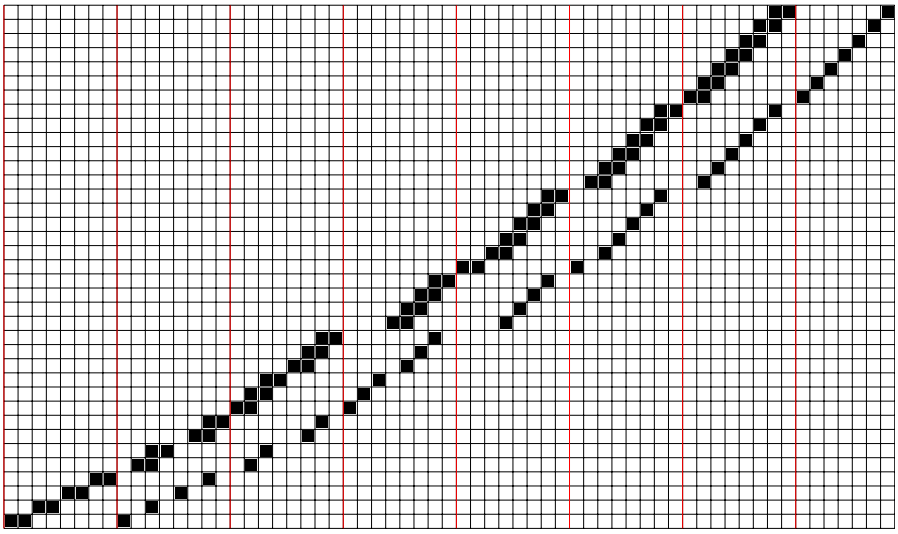


FIGURE 7.4: Canon compact modulo 2 avec le motif $A(8)$, avec séparation toutes les 8 pulsations.

chaque ligne $\ell > 0$ du tableau $T(n)$ contient un mot qui est la sous-couverture suivante de la sous-couverture obtenue avec le couple de motifs $A(n)$ et le mot b lu dans le tableau jusqu'à la ligne $\ell - 1$ (en utilisant la convention que la "ligne 0", c'est-à-dire la ligne avant le tableau, et donc avant le pavage, est la ligne nulle pour $\ell = 1$).

Définition 7.1.18. Soit$A(n)$ le motif rythmique qui pave de façon compacte modulo 2 avec $b(n)$. Le tableau $T(n)$ ayant m colonnes est défini ainsi

$$T(n) = \begin{pmatrix} b(n)[0] & b(n)[1] & \cdots & b(n)[m-1] \\ b(n)[m] & b(n)[m+1] & \cdots & b(n)[2m-1] \\ b(n)[2m] & \cdots & \cdots & \\ \cdots & & & \end{pmatrix}$$

Exemple 7.1.19. À titre d'exemple, nous allons représenter les motifs $A(n)$, $B(n)$, le mot $b(n) = b(n)_2$ et ce mot écrit dans le tableau $T(n)$ pour $n = 2, 4, 8$ et 6.

- $n = 2, A(2) = \{0,1,2\}, B(2) = \{0\}, b(2) = 1$
  $T(2) = (10)$
- $n = 4, A(4) = \{0,1,4\}, B(4) = \{0,2,5,6,8,9,10\},$
  $b(4) = 10100110111$
  $$T(4) = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$
- $n = 8, A(8) = \{0,1,8\},$

$$\begin{aligned} B(8) &= \{0,2,4,6,9,10,13,14,16,17,18,20,21,22,27,28,29,30,32, \\ &\quad 34,35,36,37,38,41,42,43,44,45,46,48,49,50,51,52,53,54\} \end{aligned}$$

$$b(8) = 10101010011001101110111000011110101111100111111011111110$$

$$T(8) = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

- $n = 6, A(6) = \{0, 1, 6\}$,

$$\begin{aligned} B(6) \quad = \quad & \{0, 2, 4, 7, 8, 11, 15, 18, 20, 21, 23, 24, 28, 29, 30, 32, 37, \\ & 38, 40, 42, 43, 44, 47, 48, 49, 50, 52, 53, 54, 55, 56\} \end{aligned}$$

$$b(6) = 10101001100100010010110110001110100001101011100111101111$$

$$T(6) = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Remarque 7.1.20. Si le pavage par un motif A n'est pas compact, la meilleure méthode que nous ayons actuellement pour connaître une approximation de sa taille est un algorithme présenté dans (AMIOT, 2011). Il retourne un multiple de la taille du plus petit canon que l'on peut obtenir avec le motif A, en incrémentant les valeurs possibles de sa taille N jusqu'à ce qu'une période N vérifie bien que $A(X)$ divise $X^N - 1 \mod p$. On ne sait à ce jour obtenir la taille exacte d'un canon modulo p que dans le cas où le motif donne un pavage compact, et il faut alors utiliser l'algorithme 1 afin d'obtenir son motif des entrées B ; on a alors $N = \max A + \max B - 1$. Comme on peut toujours paver de façon compacte modulo 2, à partir de maintenant nous ne paverons plus que modulo 2.

Par exemple, le Tableau 7.1 donne la taille N minimale du canon compact modulo 2 obtenue avec certains motifs $A(n) = \{0, 1, n\}$ grâce à l'algorithme 1. On remarque que lorsque $n = 2^k$, les motifs $A(2^k)$ permettent d'obtenir les canons modulo 2 les plus petits par rapport à n. C'est pourquoi nous allons les étudier dans la partie suivante.

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
N	3	7	15	21	63	127	63	73	889	1533	3255	7905	11811	32767	255

TABLE 7.1: Les tailles N minimales de canons compacts modulo 2 par des motifs $A(n) = \{0,1,n\}$.

7.2 La famille des motifs $\{0,1,2^k\}$

Définition 7.2.1. Nous définissons par récurrence le tableau $V(2^k)$ pour $k \in \mathbb{N}^*$:

On pose $V(2^1) = V(2) = (10)$. Si le tableau $V(2^k)$ de taille $(2^k-1, 2^k)$ est construit, alors on construit le tableau $V(2^{k+1})$ de taille $(2^{k+1}-1, 2^{k+1})$ de cette manière :

$$V(2^{k+1}) = \left(\begin{array}{c|cc} V(2^k) & V(2^k) & \\ \hline & 11\ldots1 & 0 \\ & 11\ldots1 & 0 \\ \widetilde{V(2^k)} & \vdots & \vdots \\ & 11\ldots1 & 0 \end{array}\right)$$

où $\widetilde{V(2^k)}$ est un tableau de taille $(2^k, 2^k)$ ainsi défini : sa première ligne est $0^{(2^k-1)}1$ et ses dernières lignes sont $V(2^k)$ dont la dernière colonne de 0 est remplacée par une colonne de 1 (on peut facilement montrer par récurrence que la dernière colonne de $V(2^k)$ est nulle pour tout k).

Exemple 7.2.2. Pour quelques valeurs de k, représentons à valeur d'exemple les tableaux $V(2^k)$ avec les lignes séparatrices des constructions par récurrence :

$V(2) = (10)$

$$V(4) = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ \hline 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{array}\right)$$

$$V(8) = \left(\begin{array}{ccc|cccc} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{array}\right)$$

où on peut voir $\widetilde{V(4)} = \left(\begin{array}{cccc} 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 1 & \mathit{1} \\ 0 & 1 & 1 & \mathit{1} \\ 1 & 1 & 1 & \mathit{1} \end{array}\right)$

La théorème suivant permet de lier les tableaux définis en 7.1.18 et 7.2.1, et ainsi, de passer d'une construction linéaire en la taille N du tableau $T(n)$ avec l'algorithme 1 à une construction par récurrence qui est logarithmique en la taille N dans le cas où $n = 2^k$.

Théorème 7.2.3. *Pour tout* $k \in \mathbb{N}^*$, $T(2^k) = V(2^k)$.

Avant de démontrer ce théorème, nous allons avoir besoin de quelques lemmes techniques présentés dans la partie suivante.

7.2.1 *Lemmes préliminaires*

Définition 7.2.4. Soit $k \in \mathbb{N}^*$, et B un motif rythmique. Si UC_2 est une sous-couverture de taille n avec $n < 2^k$ formée par le couple $A(2^k)$ et B, on note $\xrightarrow{k}$ la fonction sur et à valeur dans les mots finis de l'alphabet $\{0,1\}$ définie par $\xrightarrow{k}(UC_2) = b'_2$ si et seulement si UC_2 est remplie grâce au motif B' associé au mot b' de taille n (quitte à rajouter des 0 à la fin de b').

Pour une meilleure lisibilité, on notera cette fonction $UC_2 \xrightarrow{k} b'_2$.

Exemple 7.2.5. Nous avons par exemple $0^{(2n)} \xrightarrow{k} (10)^n$ si $2n \leq 2^k$. En effet, on voit bien dans 7.1.19 que la sous-couverture suivante de 0^8 (qui est la première sous-couverture du pavage) de taille 8 est $(10)^4$.

Remarque 7.2.6. Grâce au lemme 7.1.15, on sait que si $\left(A\left(2^k\right), B\left(2^k\right)\right)$ est un canon compact modulo 2, alors le motif $B\left(2^k\right)$ peut être écrit sous la forme $b(2^k)_2 = \tilde{b_1} \cdots \tilde{b_n}$ avec : pour tout $i \leq n$ la taille des b_i vaut $|\tilde{b_i}| = 2^k$ et

$$0^{2^k} \xrightarrow{k} \tilde{b_1} \xrightarrow{k} \ldots \xrightarrow{k} \tilde{b_n}.$$

La preuve de l'égalité des tableaux $T(2^k)$ et $V(2^k)$ donnera $n = 2^k - 1$.

Cela veut dire que chaque sous-couverture $\tilde{b_i}$ sera la sous-couverture suivante de $\tilde{b}_{i-1}$, comme cela est expliqué dans la remarque 7.1.17.

Ainsi, le tableau $T(2^k)$ peut s'écrire $T(2^k) = \begin{pmatrix} \tilde{b_1} \\ \tilde{b_2} \\ \cdots \\ \tilde{b_n} \end{pmatrix}$.

Exemple 7.2.7. Avec $n = 2^2$, on a $0^4 \xrightarrow{2} 1010 \xrightarrow{2} 0110 \xrightarrow{2} 1110$ et en effet $b(4)_2 = 10100110111 = 101001101110$, c'est-à-dire

$$T(4) = \begin{pmatrix} 1010 \\ 0110 \\ 1110 \end{pmatrix}.$$

Lemme 7.2.8. *Pour tout $n \in \mathbb{N}^*$, pour remplir une sous-couverture $UC_2(i)_n = 01^n0$ commençant à l'indice i et de taille $n+2$ avec le motif $a = 11$, il faut les entrées $b' = 1^{(n+1)}0$.*

Démonstration. Démontrons ce lemme par récurrence sur n.

Pour $n = 1$, pour remplir la sous-couverture $UC_2(i)_1 = 010$ avec le mot 11, il faut commencer par remplir le premier indice i, et donc par poser le motif 11 en position i, c'est-à-dire que le motif des entrées sera de la forme $b' = 1\mathbf{.}.$. La nouvelle sous-couverture est alors $100 = 1(1 + 1 \bmod 2)0$. Il faut alors poser le motif 11 à l'indice $i + 1$ pour remplir ce nouveau 0, et donc $b' = 11\mathbf{.}$. La nouvelle sous-couverture vaut alors 111, c'est donc un pavage et il ne faut plus d'entrée pour la remplir. Ainsi $b' = 110$.

Soit n fixé, supposons que le lemme soit vrai à ce rang n, essayons de remplir la sous-couverture $UC_2(i)_{n+1} = 01^{(n+1)}0$ commençant à l'indice i avec le motif 11 grâce aux entrées b'.

Cette sous-couverture commençant par un 0, pour la remplir il faut que le motif b' commence par un 1. C'est-à-dire $b' = 1\mathbf{.}^{(n+2)}$ et la nouvelle sous-couverture est

$$101^n0 = 1(1 + 1 \bmod 2)1^n0 = 1UC_2(i + 1)_n.$$

Par hypothèse de récurrence, pour paver les $n+2$ derniers indices de cette sous-couverture avec le motif 11, les $n+2$ dernières lettres de b' devront être $1^{(n+1)}0$.

On a bien $b' = 11^{(n+1)}0 = 1^{(n+2)}0$, et on conclut par récurrence. □

Lemme 7.2.9. *Pour tout $k \in \mathbb{N}^*$, $1^{(2^k-1)}0\ 1^{(2^k-1)}0 \xrightarrow{k+1} 0^{(2^k-1)}11^{(2^k-1)}0$.*

Démonstration. Notons $1^{(2^k-1)}0\ 1^{(2^k-1)}0 \xrightarrow{k+1} c$.

Le premier bloc de 1 de la sous-couverture de taille $2^k - 1$ est déjà pavé, il n'y a donc pas besoin d'entrée pour ces $2^k - 1$ premiers indices, et le mot c commence par $0^{(2^k-1)}$.

Le bloc restant de cette sous-couverture est $0\ 1^{(2^k-1)}0$, dont la longueur satisfait à $|0\ 1^{(2^k-1)}0| < 2^{k+1}$. Ainsi, seuls les deux premiers *onsets* de $A(2^{k+1}) = \{0, 1, 2^{k+1}\}$ sont utilisés pour remplir cette sous-couverture. C'est-à-dire qu'on va remplir la seconde partie $0\ 1^{(2^k-1)}0$ de cette sous-couverture avec le motif 11, et on est dans le cadre du lemme 7.2.8, ce qui signifie que $c\left[2^k, \ldots, 2^{k+1}\right] = 1^{(2^k)}0$.

On a bien $c = 0^{(2^k-1)}11^{(2^k-1)}0$. □

Lemme 7.2.10. *Pour tout $k \in \mathbb{N}^*$, tel que $n < 2^k - 2$, alors $01^n0 \xrightarrow{k} 1^{(n+1)}0$.*

Démonstration. Comme dans ce cas la taille de la sous-couverture satisfait à $|01^n0| < 2^k$, nous ne cherchons à la remplir qu'avec les deux premiers *onsets* 11 du motif $A(2^k)$, et nous sommes encore dans les conditions du lemme 7.2.8. □

Lemme 7.2.11. *Pour tout* $k \in \mathbb{N}$, $01^{(2^k-2)}0 \xrightarrow{k} 1^{(2^k-1)}0$.

Démonstration. Dans ce cas présent, nous n'avons pas

$$|01^{(2^k-2)}0| < 2^k$$

et ne sommes donc pas dans les hypothèses du lemme 7.2.8.

Cependant, nous avons exactement $|01^{(2^k-2)}0| = 2^k$. Ceci veut dire que nous sommes hors des conditions du lemme 7.2.8 seulement pour le dernier indice. Ainsi, si pour remplir cette sous-couverture il faut poser le motif $A(2^k)$ sur le premier indice (et comme elle commence par un 0 il le faudra), nous pouvons considérer la même sous-couverture avec un *onset* supplémentaire sur son dernier indice que nous cherchons à remplir avec seulement les deux premiers *onsets* de $A(2^k)$.

Nous voulons ainsi remplir la sous-couverture $01^{(2^k-2)}1$ avec 11, et par une récurrence similaire à celle présentée dans la preuve du lemme 7.2.8, nous obtenons les entrées $1^{(2^k-1)}0$. □

Lemme 7.2.12. *Pour tout* $k \in \mathbb{N}^*$*, soit des mots* u, v *sur l'alphabet* $\{0,1\}$ *tels que* $|u| = |v| = 2^k - 1$.
Si $u0 \xrightarrow{k} v0$ *alors* $u1 \xrightarrow{k} v1$.

Démonstration. Par définition d'une sous-couverture, ajouter des 0 à la fin ne la change pas, donc $u = u0$.

Notons $\tilde{V}$ (respectivement $\tilde{V}1$) le motif rythmique tel que $u = u0$ (respectivement $u1$) soit la sous-couverture $UC_2(i)$ de $A(2^k)$ avec les entrées $\tilde{V}$ (respectivement avec les entrées $\tilde{V}1$).

Pour faciliter la lecture de cette preuve, si un multi-ensemble A est associé à un mot a, nous noterons $mot(A) = a$.

Par définition, nous avons les égalités suivantes

$$mot\left(\left(A(2^k) + \left(\tilde{V} \cup (V + \{i\})\right)\right)\right)_2 \left[0, \ldots, i + 2^k - 1\right] = 1^i 1^{|U \cdot 0|}$$

et

$$mot(A(2^k) + \tilde{V})_2 = 1^i u0.$$

En notant C le multi-ensemble défini par $A(2^k) + \tilde{V} = C \cup (U + \{i\})$, c'est-à-dire un multi-ensemble dont les éléments sont à valeur dans $\{0, \ldots, i-1\}$ avec leur multiplicité qui vaut $1 \bmod 2$, nous avons

$$mot\left(\left([C \cup (U + \{i\})] \cup \left[A(2^k) + (V + \{i\})\right]\right)\right)_2 \left[0, \ldots, i + 2^k - 1\right] = 1^i 1^{|u \cdot 0|}$$

Or, comme $|V| = 2^k - 1$

$$mot\left((C \cup (U + \{i\})) \cup (\{0,1\} + (V + \{i\}))\right)_2 \left[0, \ldots, i + 2^k - 1\right] = 1^i 1^{|u \cdot 0|}$$

En retirant les i premiers indices de $1^i 1^{|u \cdot 0|}$ qui viennent de C nous avons :

$$mot\left((U + \{i\}) \cup (\{0,1\} + (V + \{i\}))\right)_2 \left[0, \ldots, i + 2^k - 1\right] = 0^i 1^{|u \cdot 0|}.$$

Soit en ne considérant plus que le suffixe de ce mot après les i premiers indices :

$$mot\,(U \cup (\{0,1\}+V))_2 \left[0,\ldots,2^k-1\right] = 1^{|u\cdot 0|} = 1^{(2^k)}.$$

Les ensembles U, V sont associés aux mots $u0 = u$ et $v0 = v$, et nous pouvons de la même manière noter $U1 = U \cup \{2^k\}$ et $V1 = V \cup \{2^k\}$ les ensembles associés aux mots $u1$ et $v1$. On a alors

$$\begin{aligned}(U1 \cup (\{0,1\}+V1)) &= \left(\left(U \cup \left\{2^k\right\}\right) \cup \left(\{0,1\} + \left(V \cup \left\{2^k\right\}\right)\right)\right)\\ &= \left(\left\{2^k\right\} \cup U \cup (\{0,1\}+V) \cup \left(\left\{2^k, 2^k+1\right\}\right)\right)\\ &= \left((U \cup (\{0,1\}+V)) \cup \left(\left\{2^k, 2^k, 2^k+1\right\}\right)\right).\end{aligned}$$

Soit

$$mot\,(U1 \cup (\{0,1\}+V1))_2 = 1^{(2^k-1)}(1+1+1)1 = 1^{(|u|+1)} = 1^{|u1|}.$$

De la même manière, en rajoutant i premiers indices en préfixe et en ajoutant un multi-ensemble $C1$ tel que $A(2^k) + \tilde{V}1 = C1 \cup (U1 + \{i\})$, nous avons

$$mot\left((C1 \cup (U1+\{i\})) \cup \left(A(2^k) + (V1+\{i\})\right)\right)_2 \left[0,\ldots,i+2^k-1\right] = 1^i 1^{|u1|}$$

ce qui se récrit

$$mot\left(A(2^k + \left(\tilde{V}1 \cup (V1+\{i\})\right)\right)_2 \left[0,\ldots,i+2^k-1\right] = 1^i 1^{|u1|}$$

Et on a bien $u1 \xrightarrow{k} v1$.

□

Lemme 7.2.13. *Pour tout $n,k \in \mathbb{N}^*$, $a \in \{0,1\}$, soient des mots u,v sur l'alphabet $\{0,1\}$ tels que $|u| = |v| = 2^k - 1$, si $ua \xrightarrow{k} v0$ alors $(ua)^n \xrightarrow{k+n-1} (v0)^n$.*

Démonstration. Comme $v0$ se termine par un 0, cela signifie que pour remplir $ua = UC_2(i)$, il ne faut pas poser le motif 11 (les deux premiers *onsets* de $A(2^k)$) à l'indice $i+2^k$. Ainsi, le remplissage de ua par V ne déborde pas sur la sous-couverture suivante de taille 2^k. Cela veut dire que lorsque l'on doit remplir la sous-couverture suivante, il n'y a pas besoin de garder en mémoire le remplissage actuel (excepté bien entendu pour le dernier *onset* de $A(2^k)$ qui crée la sous-couverture suivante). On peut donc concaténer le motif de remplissage si on souhaite remplir la n-concaténation de la sous-couverture, en prenant bien sûr soin de paver avec $\{0,1,2^{k+n-1}\}$ plutôt qu'avec $\{0,1,2^k\}$ pour ne pas rencontrer le dernier *onset* de $A(2^k)$, et donc de ne paver qu'avec les deux premiers *onsets* 11 de $A(2^{k+n-1})$ pendant le remplissage. □

7.2.2 *Preuve du Théorème 7.2.3*

Démonstration. Nous allons montrer l'égalité $V(2^k) = T(2^k)$ par récurrence sur $k \in \mathbb{N}^*$.

Si $k = 1$, nous avons bien $V(2) = (10) = T(2)$ et le motif des entrées vaut $B(2) = \{0\}$.

Si l'égalité est vraie pour un k fixé, montrons la pour $k+1$:

Grâce au lemme 7.2.13, avec $n = 2$, nous savons que si $V(2^k)$ donne le motif des entrées pour paver avec le motif $A(2^k)$, alors $(V(2^k)|V(2^k))$ donne le début du motif des entrées pour paver avec $A(2^{k+1})$. En effet, nous remplissons des sous-couverture de taille 2^{k+1} avant de rencontrer le dernier *onset* de $A(2^{k+1})$.

Le lemme 7.2.9 nous permet de construire la première ligne du bloc du bas de $V(2^{k+1})$, c'est-à-dire la première ligne de $\widetilde{V(2^k)}$ qui est $0^{(2^k-1)}1$ concaténée à $1^{(2^k-1)}0$.

Le bloc en bas à droite de $V(2^{k+1})$ s'obtient grâce au lemme 7.2.10 :

$$\begin{matrix} 11\ldots1 & 0 \\ 11\ldots1 & 0 \\ \vdots & \vdots \\ 11\ldots1 & 0 \end{matrix}$$

En effet, on peut remarquer qu'il y a toujours un 0 avant ces lignes de 1, donc nous sommes bien dans les conditions du lemme 7.2.10. Si nous ne l'étions pas, cela voudrait dire avoir une ligne pleine de 1 de taille $(2^{k+1}-1)$, c'est-à-dire un pavage compact avec le motif $A(2^{k+1})$, et cela achève de prouver l'égalité entre nos tableaux, comme on le verra plus tard avec le lemme 7.2.11.

Le lemme 7.2.12 explique pourquoi les dernières lignes de $\widetilde{V(2^k)}$ ressemblent à $V(2^k)$ sauf pour sa dernière colonne. On essaye dans ce cas de paver avec le motif 11 puisque le dernier *onset* de $A(2^{k+1})$ n'est pas pris en compte pour paver des lignes de cette taille. En utilisant l'hypothèse de récurrence $T(2^k) = V(2^k)$, la procédure nous a donné le motif des entrées $V(2^k)$ lorsqu'on a voulu remplir la sous-couverture de 0 de la ligne précédente, comme c'est expliqué dans la remarque 7.1.12. La ligne précédente à remplir est dans notre cas la première ligne de $\widetilde{V(2^k)}$, c'est-à-dire $0^{(2^k-1)}1$, et les sous-couvertures suivantes obtenues sont celles écrites dans les lignes de $V(2^k)$ avec un 1 en dernier indice remplaçant le 0.

Finalement, le lemme 7.2.11 permet de terminer la construction. La dernière ligne de $V(2^{k+1})$ vaut $1^{(2^{k+1}-1)}$ ce qui permet d'obtenir un pavage compact modulo 2 avec le motif $A(2^{k+1})$ puisque son dernier *onset* se trouve alors juste derrière cette ligne de 1. En effet, cette suite de $2^{k+1}-1$ entrées consécutives qui donne un pavage modulo 2 est suivie par $2^{k+1}-1$ *onsets* (qui ne sont pas des *donsets*), ce qui termine ce pavage compact.

On peut bien conclure par récurrence que pour tout $k \in \mathbb{N}^*$, le tableau $V(2^k)$ construit par récurrence comme dans le définition 7.2.1 est égal au tableau $T(2^k)$ contenant le mot $b(2^k)$ écrit ligne à ligne comme dans la définition 7.1.18.

□

Cette construction par récurrence du motif des entrées permet de construire les pavages compacts modulo 2 minimaux avec les motifs de la forme $A(2^k) = \{0,1,2^k\}$ en temps logarithmique par rapport à la taille du pavage, tandis que l'algorithme 1 permettait lui de les obtenir en temps linéaire, seulement. La structure de ces pavages particuliers permet aussi d'obtenir les premiers résultats de combinatoire sur les pavages modulo p. On peut déduire du tableau $T(2^k)$ la taille exacte du pavage ainsi que des informations sur les entrées et les *donsets*.

Corollaire 7.2.14. *Pour tout $k \in \mathbb{N}^*$, le canon formé du couple $(A(2^k), B(2^k))$ est le plus petit canon compact modulo 2 de $\mathbb{Z}_N$ avec le motif $A(2^k)$,*

$$\textit{et sa taille vaut } N = 4^k - 1$$

$$\textit{il a } |B(2^k)| = 4^k - 3^k \textit{ entrées et}$$

$$|D_{(A(2^k),B(2^k)),2}| = 4^k - \frac{3^{k+1}-1}{2} \textit{ donsets.}$$

Démonstration. Si on note $b(k) = |B(2^k)| = 4^k - 3^k$ le nombre d'entrées pour ce canon, alors ce nombre correspond au nombre de 1 dans le tableau $T(2^k)$ et on peut les compter par récurrence.

Nous avons $b(1) = 1$ et $b(2) = 7$.

La construction du tableau $T(2^{k+1})$ nous donne l'égalité

$$b(k+1) = 3 \times b(k) + 4^k.$$

En effet, le terme $3 \times b(k)$ vient du tableau $T(2^k)$ écrit trois fois dans le tableau $T(2^{k+1})$, et le terme en 4^k vient du bloc de 1 en bas à droite de taille $(2^k, 2^k)$, incluant la dernière colonne de $\widetilde{T(2^k)}$.

Cette égalité sur $b(k)$ se récrit $b(k+2) - 7b(k+1) + 12b(k) = 0$, et donc

$$b(k) = \frac{b(2) - 3b(1)}{4} \times 4^k - \frac{4b(1) - b(2)}{3} \times 3^4 = 4^k - 3^k.$$

Afin d'obtenir la taille du canon N, il suffit de savoir que le canon se termine lorsque le motif $A(2^k)$ est placé sur le dernier 1 de $b(2^k)$, c'est-à-dire

$$N = (2^k - 1) \times 2^k - 1 + 2^k = 4^k - 1.$$

En effet, le premier produit vient de la taille $(2^k, 2^k - 1)$ du tableau $T(2^k)$, et le -1 suivant du fait que le dernier 1 se trouve en pénultième position dans le tableau. Le dernier terme $+2^k$ vient de l'ajout du motif $A(2^k)$ sur cet indice.

Finalement, pour énumérer les *donsets*, nous allons avoir besoin du lemme suivant.

Lemme 7.2.15. *Chaque k-onset du canon modulo 2 de taille minimal construit avec le motif A avec $|A| = n$ est tel que $k \leq n$.*

Démonstration. Soit $A = \{a_1, \ldots, a_n\}$ et B tels que (A, B) soit un canon modulo 2 minimal de $\mathbb{Z}_N$ et supposons qu'il existe un k-*onset* au temps t tel que $k > n$.

Ce k-*onset* est construit à partir de différents *onsets* de A superposés : $a_{j_1}, \ldots, a_{j_k}$, et on note b_{j_i} la voix des entrées de B qui permet d'obtenir l'*onset* a_{j_i} au temps t.

Comme $k > n$, par le principe des tiroirs, il existe $j_i \neq j_l$ tel que $a_{j_i} = a_{j_l}$. Ainsi au temps $b_{j_i} = b_{j_l} = t - a_{j_i}$ deux occurrences du motif rythmique A sont jouées simultanément. C'est-à-dire qu'à chaque temps $b_{j_i} + a_m = b_{j_l} + a_m$, $m = 1, \ldots, k$, deux *onsets* de A sont superposés, ce qui représente zéro *onset* modulo 2.

Il est clair que nous aurons le même nombre d'*onsets* modulo 2 à chaque pulsation de $\Pi_N(A + B)$ et de $\Pi_N(A + (B \setminus \{b_{j_i}, b_{j_l}\}))$.

C'est-à-dire que $(A, B \setminus \{b_{j_i}, b_{j_l}\})$ est un canon modulo 2 de $\mathbb{Z}_N$ plus petit que (A, B), ce qui contredit sa minimalité. □

Nous pouvons maintenant énumérer les *donsets*. Le motif rythmique $A(2^k)$ n'ayant que trois éléments, le lemme 7.2.15 nous dit que tous les *donsets* de $(A(2^k), B(2^k))$ ont au plus 3 *onsets* superposés, et comme nous pavons modulo 2, cela veut dire qu'il y a un *onsets* ou trois à chaque pulsation.

Ainsi, si l'on note $D(2^k)$ l'ensemble des indices où se situent un *donset* qui n'est pas un *onset* (c'est-à-dire qui a trois *onsets* superposés), nous avons la formule

$$|A(2^k)| \times |B(2^k)| - 2|D(2^k)| = N.$$

En effet, le facteur 2 multipliant $\sharp D_k$ vient du fait que chaque *donset* étant un 3-*onset* a exactement 2 *onsets* superposés au dessus des *onsets* d'un pavage classique. Le résultat s'ensuit : $|D_k| = 4^k - \dfrac{3^{k+1} - 1}{2}$. □

7.3 La famille des motifs $\{0, 1, 2^k + 1\}$

Nous allons construire dans cette partie le tableau $T(2^n + 1)$ en fonction du tableau $T(2^n)$. Ainsi grâce au théorème 7.2.3, nous pourrons aussi obtenir les entrées qui permettent de paver avec le motif $A(2^n + 1)$ en temps logarithmique.

Définition 7.3.1. Nous définissons par récurrence pour $k \in \mathbb{N}^*$ le tableau $W(2^k + 1)$:

Soit $W(2^1 + 1) = W(3) = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$. Si le tableau $W(2^k + 1)$ de taille $(2^k, 2^k)$ est construit, alors on construit le tableau $W(2^{k+1} + 1)$ de taille $(2^{k+1}, 2^{k+1})$ de cette manière :

$$W(2^{k+1}+1) = \left(\begin{array}{c|c} W(2^k+1) & \widetilde{W(2^k+1)} \\ \hline 11\ldots1 & \\ 11\ldots1 & \\ \vdots & W(2^k+1) \\ 11\ldots1 & \end{array}\right)$$

où $\widetilde{W(2^k + 1)}$ est le tableau de taille $(2^k, 2^k)$ ainsi défini : ses $2^k - 1$ premières lignes sont les mêmes que celles de $W(2^k + 1)$ et la dernière ligne de 1 est remplacée par une ligne de 0 (on peut facilement montrer par récurrence que la dernière ligne de $W(2^k)$ vaut 1^{2^k} pour tout k).

Exemple 7.3.2. Pour quelques valeurs de k, représentons à titre d'exemple les tableaux $W(2^k+1)$ avec les lignes séparatrices des constructions par récurrence :

$$W(3) = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

$$W(5) = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ \hline 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{array}\right)$$

$$W(9) = \left(\begin{array}{cccc|cccc} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}\right)$$

où on peut voir $\widetilde{W(5)} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ \mathit{0} & \mathit{0} & \mathit{0} & \mathit{0} \end{pmatrix}$

Théorème 7.3.3. *Pour tout* $k \in \mathbb{N}^*$, $T(2^k+1) = W(2^k+1)$.

Pour démontrer ce théorème, nous allons avoir besoin de quelques notions supplémentaires et de lemmes techniques.

7.3.1 *Lemmes préliminaires*

En utilisant les conventions de notations par exemple utilisées dans (CROCHEMORE et collab, 2001), nous introduisons les notions de mot *miroir* ou *renversé*, et de mot *conjugué* :

Définition 7.3.4. Soit un mot u de taille n, son mot miroir est le mot noté u^{R} défini comme le mot u lu de droite à gauche :

$$u^{\mathrm{R}} = u[n-1] \cdots u[0].$$

Définition 7.3.5. Soit u un mot sur l'alphabet $\{0,1\}$ de taille n, alors on note $\overline{u}$ son mot conjugué, défini par

$$\overline{u} = \overline{u[0]} \cdots \overline{u[n-1]}$$

où $\overline{0} = 1$ et $\overline{1} = 0$.

Lemme 7.3.6. *Soient deux sous-couvertures* $UC_2(i)$ *et* UC_2'*, et les motifs des entrées* b *et* b' *pour les remplir avec le motif* 11*. Si la dernière lettre de* b *est un* 0*, alors le motif des entrées pour remplir la sous-couverture* $UC_2UC_2'(i)$ *avec le motif* 11 *est* bb'*.*

Démonstration. On note b'' le motif des entrées pour remplir $UC_2UC_2'(i)$ avec le motif 11. Comme on peut remplir $UC_2(i)$ avec b, on a $b'' = b\mathbf{.}^{|UC_2'|}$. Alors les indices entre i et $i+|UC_2|-1$ sont remplis, et comme b se termine par un 0, le dernier indice sur lequel le motif 11 est posé est $i+|UC_2|-2$. Ainsi il n'y a aucun *onset* venant du remplissage avec b sur les indices plus grands que $i+|UC_2|$. On peut donc remplir la suite de la sous-couverture entre les indices $i+|UC_2|$ et $i+|UC_2|+|UC_2'|-1$ avec b' car ce motif des entrées permet bien de remplir UC_2'. C'est-à-dire $b'' = bb'$. □

Lemme 7.3.7. *Soit une sous-couverture* $UC_2(i) = u_1 \cdots u_n$ *de taille* $n \in \mathbb{N}^*$*, et* b *le motif des entrées pour la remplir avec le motif* 11*. Alors la sous-couverture* $UC_2'(i) = \overline{u_1}u_2 \cdots u_n$ *de taille* n *est remplie par le motif* 11 *avec le motif des entrées* $\overline{b}$*.*

Démonstration. Montrons-le par récurrence sur n.

Si $n=1$, $UC_2(i)=1$ ou $UC_2(i)=0$. Dans le premier cas, la sous-couverture est déjà remplie, donc $b=0$, dans le second, il suffit de poser le motif 11 sur l'indice i et alors $b=1$, et les deux cas sont bien conjugués l'un de l'autre.

Si le lemme est vrai pour un n fixé, montrons le pour la sous-couverture $UC_2(i)=u_1\cdots u_{n+1}$ de taille $n+1$. Soit b le motif des entrées nécessaires pour la remplir avec le motif 11, et b' le motif pour remplir la sous-couverture $UC_2'(i)=\overline{u_1}u_2\cdots u_{n+1}$.

Si $u_1=0$, alors il faut poser le motif 11 sur l'indice i pour remplir $UC_2(i)$, c'est-à-dire $b[0]=1$. Alors $UC_2'(i)=1u_2\cdots u_{n+1}$, et comme l'indice i est rempli, il ne faut pas poser le motif 11 sur indice i et $b'[0]=0$. Dans le cas de ces deux sous-couvertures, une fois le motif 11 posé sur l'indice i ou non, alors son deuxième *onset* est sur l'indice $i+1$ ou non, et $UC_2(i)$ devient $1\overline{u_2}u_3\cdots u_{n+1}$ tandis que $UC_2'(i)$ devient $1u_2u_3\cdots u_{n+1}$. Les deux sous-couvertures suivantes de taille n commençant à l'indice $i+1$ satisfont à l'hypothèse de récurrence, et s'il faut $b[1,\dots,n]$ pour remplir $UC_2(i+1)$, alors pour remplir $UC_2'(i+1)$ il faut $\overline{b}[1,\dots,n]$. Donc

$$b'=0\overline{b[1,\dots,n]}=\overline{b[0]b[1,\dots,n]}=\overline{b}.$$

Si $u_1=1$, alors $UC_2'(i)[0]=u_1'=0$, donc le raisonnement précédent s'applique en inversant les rôles de UC_2 et UC_2'. □

Lemme 7.3.8. *Soient deux sous-couvertures $UC_2(i)$ et UC_2', et les motifs des entrées b et b' pour les remplir avec le motif 11. Si la dernière lettre de b est un 1, alors le motif des entrées pour remplir la sous-couverture $UC_2UC_2'(i)$ avec le motif 11 est $b\overline{b'}$.*

Démonstration. On note b'' le motif des entrées pour remplir $UC_2UC_2'(i)$ avec le motif 11. Comme on peut remplir $UC_2(i)$ avec b, on a $b''=b.^{|UC_2'|}$. Alors les indices entre i et $i+|UC_2|-1$ sont remplis, et comme b se termine par un 1, le dernier indice sur lequel le motif 11 est posé est $i+|UC_2|-1$. Ainsi le deuxième *onset* du motif est superposé au premier indice de $UC_2'=u_1'\cdots u_n'$, c'est-à-dire que cette sous-couverture devient $\overline{u_1'}u_2'\cdots u_n'$.

Pour remplir la seconde partie de la sous-couverture $UC_2UC_2'(i)$, on applique le lemme 7.3.7 et il faut donc les entrées $\overline{b'}$. Donc on a bien $b''=b\overline{b'}$. □

Lemme 7.3.9. *Soit $n\in\mathbb{N}^*$ et $m,k,l\in\mathbb{N}$. Pour paver une $UC_2(i)$ de la forme $00^{2k}1^n00^{2l}1^m$ avec le motif 11, il faut un motif des entrées de la forme $b'=(10)^k11^n0(10)^l0^m=(10)^k1^{(n+1)}0(10)^l0^m$.*

Démonstration. Pour remplir la sous-couverture

$$UC_2(i)=0^{2k}01^n0^{(2l+1)}1^m$$

de taille $2k+n+2l+m+2$ commençant à l'indice i il faut le motif des entrées noté b'.

Pour remplir les $2k$ premiers 0 de $UC_2(i)$ avec 11, on peut facilement montrer par récurrence qu'il faut les entrées $(10)^k$, c'est-à-dire

$$b' = (10)^k \text{.}^{(1+n+2l+1+m)}$$

La sous-couverture est alors de la forme $1^{2k}01^n0^{(2l+1)}1^m$. Pour la remplir à partir de l'indice $i+2k$ jusqu'à l'indice $i+2k+n+2$, on est dans le cas du lemme 7.2.8, et le motif des entrées vaut donc

$$b' = (10)^k 1^{(n+1)} 0 \text{.}^{(2l+m)} .$$

Si $l=0$, alors la sous-couverture est remplie, et le motif des entrées est complètement décrit et est de la forme : $b' = (10)^k 1^{(n+1)} 00^m$.

Sinon, de la même façon qu'avec les 0^{2k} au début de la couverture, il faudra que le motif des entrées pour remplir cette partie de la sous-couverture

$$UC_2(i)\,[i+2k+n+3,\ldots,i+2k+n+3+2l] = 0^{2l}$$

soit de la forme $(10)^l$. Comme le motif des entrées pour remplir la première partie de la sous-couverture se termine par un 0, on peut utiliser le lemme 7.3.6 et concaténer ces motifs des entrées. C'est-à-dire que le motif des entrées vaut

$$b' = (10)^k 1^{(n+1)} 0 (10)^l \text{.}^{(m)} .$$

La sous-couverture est donc remplie jusqu'à l'indice $i+2k+n+1+2l$ et le motif est posé pour la dernière fois sur l'indice $i+2k+n+2l+2$. Cela implique que ce dernier motif 11 n'a pas d'*onset* sur les m derniers indices de la sous-couverture qui sont déjà remplis. Donc le motif des entrées est complètement décrit et est de la forme

$$b' = (10)^k 1^{(n+1)} 0 (10)^l 0^m.$$

□

Lemme 7.3.10. *Soit $m \in \mathbb{N}^*$ et $n,k,l \in \mathbb{N}$. Pour paver une UC_2 de la forme $1^m00^{2k}1^n00^{2l}$ avec le motif 11, il faut un motif des entrées de la forme $b' = 0^m1(01)^k1^n0(10)^l$.*

Démonstration. La sous-couverture commençant par un bloc de 1 de taille m, le motif des entrées est donc de la forme

$$b' = 0^m \text{.}^{(2k+n+2l+2)} .$$

Pour remplir le reste de la sous-couverture, on applique le même raisonnement qu'au lemme 7.3.9 avec $m=0$ et on obtient immédiatement

$$b' = 0^m (10)^k 1^{n+1} 0 (10)^l.$$

□

Lemme 7.3.11. *Soit UC_2 une sous-couverture de taille n contenant k fois la lettre 1. Si le motif des entrées pour la remplir avec le motif 11 est b, alors le motif b' pour remplir la sous-couverture UC_2^{R} est :*

- $b' = (b\,[0, \ldots, n-2])^{\mathrm{R}}\, b\,[n-1]$, *si k est de même parité de n*
- $b' = \left(\overline{b\,[0, \ldots, n-2]}\right)^{\mathrm{R}}\, b\,[n-1]$, *sinon.*

Démonstration.

- Si k est de même parité de n, alors UC_2 contient un nombre pair de 0.
  Si la sous-couverture commence par un 1, on peut écrire

  $$UC_2 = 1^{m_1}00^{2k_1}1^{n_1}00^{2l_1}\cdots 1^{m_r}00^{2k_r}1^{n_r}00^{2l_r}$$

  avec $r \in \mathbb{N}$ et pour tout $i \in \{1, \ldots, r\}$, $m_i \in \mathbb{N}^*$ et $n_i, k_i, l_i \in \mathbb{N}$.
  Alors chaque bloc de la forme $1^{m_i}00^{2k_i}1^{n_i}00^{2l_i}$ est rempli avec des entrées de la forme

  $$b_i = 0^{m_i}1(01)^{k_i}1^{n_i}0(10)^{l_i}$$

  par le lemme 7.3.10. Et comme ces entrées finissent par un 0, on peut les concaténer grâce au lemme 7.3.6. Donc UC_2 est rempli avec le motif

  $$b = b_1 \cdots b_r.$$

  La sous-couverture miroir UC_2^{R} vaut

  $$UC_2^{\mathrm{R}} = 0^{2l_r}01^{n_r}0^{2k_r}01^{m_r}\cdots 0^{2l_1}01^{n_1}0^{2k_1}01^{m_1}.$$

  Alors chaque bloc de la forme $00^{2l_i}1^{n_i}0^{2k_i}01^{m_i}$ est rempli par le motif des entrées

  $$b_i'^{\mathrm{R}} = (10)^{l_i}1^{(n_i+1)}0(10)^{k_i}0^{m_i}$$

  par le lemme 7.3.9. Et comme ces entrées finissent par un 0, on peut les concaténer grâce au lemme 7.3.6. Donc UC_2^{R} est rempli avec le motif

  $$b' = b_r'^{\mathrm{R}} \cdots b_1'^{\mathrm{R}}.$$

  Si on regarde le mot miroir des $b_i = 0^{m_i}1(01)^{k_i}1^{n_i}0(10)^{l_i}$, on a

  $$\begin{aligned} bi^{\mathrm{R}} &= (01)^{l_i}01^{n_i}(10)^{k_i}10^{m_i} \\ &= 0(10)^{(l_i-1)}101^{n_i}10(10)^{(k_i-1)}100^{(m_i-1)} \\ &= 0(10)^{l_i}1^{(n_i+1)}0(10)^{k_i}0^{(m_i-1)} \end{aligned}$$

  Alors

  $$\begin{aligned} b^{\mathrm{R}} &= (b_1 \cdots b_r)^{\mathrm{R}} \\ &= b_r^{\mathrm{R}} \cdots b_1^{\mathrm{R}} \\ &= 0b_r'^{\mathrm{R}} \cdots b_2'^{\mathrm{R}}(10)^{l_1}1^{(n_1+1)}0(10)^{k_1}0^{(m_1-1)} \end{aligned}$$

car il est évident que pour deux mots u, v, $(uv)^{\mathrm{R}} = v^{\mathrm{R}}u^{\mathrm{R}}$.
Soit

$$(b\,[0,\dots,n-2])^{\mathrm{R}} = b_r'^{\mathrm{R}} \cdots b_2'^{\mathrm{R}} (10)^{l_1} 1^{(n_1+1)} 0 (10)^{k_1} 0^{(m_1-1)}$$

Et donc

$$\begin{aligned}(b\,[0,\dots,n-2])^{\mathrm{R}}\, b\,[n-1] &= b_r'^{\mathrm{R}} \cdots b_2'^{\mathrm{R}} (10)^{l_1} 1^{(n_1+1)} 0 (10)^{k_1} 0^{(m_1-1)} 0 \\ &= b'.\end{aligned}$$

Maintenant, si la sous-couverture commence par un 0, nous avons quasiment le même raisonnement.
On peut écrire

$$UC_2 = 00^{2k_1} 1^{n_1} 00^{2l_1} 1^{m_1} \cdots 00^{2k_r} 1^{n_r} 00^{2l_r} 1^{m_r}$$

avec $r \in \mathbb{N}$ et pour tout $i \in \{1,\dots,r\}$, $n_i \in \mathbb{N}^*$ et $m_i, k_i, l_i \in \mathbb{N}$.
Alors chaque bloc de la forme $00^{2k_i} 1^{n_i} 00^{2l_i} 1^{m_i}$ est rempli avec des entrées de la forme

$$b_i = (10)^{k_i} 1^{(n_i+1)} 0 (10)^{l_i} 0^{m_i}$$

par le lemme 7.3.9. Et comme ces entrées finissent par un 0, on peut les concaténer grâce au lemme 7.3.6. Donc UC_2 est rempli avec le motif

$$b = b_1 \cdots b_r.$$

La sous-couverture miroir UC_2^{R} vaut

$$UC_2^{\mathrm{R}} = 1^{m_r} 0^{2l_r} 01^{n_r} 0^{2k_r} 0 \cdots 1^{m_1} 0^{2l_1} 01^{n_1} 0^{2k_1} 0.$$

Alors chaque bloc de la forme $1^{m_i} 00^{2l_i} 1^{n_i} 0^{2k_i} 0$ est rempli par le motif des entrées

$$b_i'^{\mathrm{R}} = 0^{m_i} 1 (01)^{l_i} 1^{n_i} 0 (10)^{k_i}$$

par le lemme 7.3.10. Et comme ces entrées finissent par un 0, on peut les concaténer grâce au lemme 7.3.6. Donc UC_2^{R} est rempli avec le motif

$$b' = b_r'^{\mathrm{R}} \cdots b_1'^{\mathrm{R}}.$$

Si on regarde le mot miroir des $b_i = (10)^{k_i} 1^{(n_i+1)} 0 (10)^{l_i} 0^{m_i}$, on a

$$\begin{aligned} bi^{\mathrm{R}} &= 0^{m_i} (01)^{l_i} 01^{(n_i+1)} (01)^{k_i} \\ &= 0^{(m_i+1)} 1 (01)^{l_i-1} 011^{n_i} 0 (10)^{(k_i-1)} 1 \\ &= 00^{m_i} 1 (01)^{l_i} 1^{n_i} 0 (10)^{(k_i-1)} 1 \end{aligned}$$

Alors

$$\begin{aligned} b^{\mathrm{R}} &= (b_1 \cdots b_r)^{\mathrm{R}} \\ &= b_r^{\mathrm{R}} \cdots b_1^{\mathrm{R}} \\ &= 0 b_r'^{\mathrm{R}} \cdots b_2'^{\mathrm{R}} 0^{m_1} 1 (01)^{l_1} 1^{n_1} 0 (10)^{(k_1-1)} 1 \end{aligned}$$

Soit

$$(b\,[0,\dots,n-2])^{\mathrm{R}} = b_r'^{\mathrm{R}} \cdots b_2'^{\mathrm{R}} 0^{m_1} 1 (01)^{l_1} 1^{n_1} 0 (10)^{(k_1-1)} 1$$

Et donc

$$\begin{aligned}(b\,[0,\dots,n-2])^{\mathrm{R}}\, b\,[n-1] &= b_r'^{\mathrm{R}} \cdots b_2'^{\mathrm{R}} 0^{m_1} 1 (01)^{l_1} 1^{n_1} 0 (10)^{(k_1-1)} (10) \\ &= b'.\end{aligned}$$

– Si k n'est pas de même parité que n, alors UC_2 contient un nombre impair de 0.
Nous n'allons montrer que le cas où UC_2 commence par un 1, car l'autre cas se démontre de manière identique, comme lorsque k est de même parité que n.
En notant le mot $u_{i,j} = 1^{m_{i,j}} 00^{2k_{i,j}} 1^{n_{i,j}} 00^{2l_{i,j}}$, on peut écrire

$$UC_2 = u_{1,1} \cdots u_{r,1} 0^{2k} 0 u_{1,2} \cdots u_{s,2}$$

avec $r, s \in \mathbb{N}$ et pour tout $i \in \{1, \dots, \max(r,s)\}$ et tout $j \in \{1,2\}$, $m_{i,j} \in \mathbb{N}^*$ et $k, n_{i,j}, k_{i,j}, l_{i,j} \in \mathbb{N}$.
Les premiers blocs des $u_{i,1}$ sont remplis avec des entrées de la forme

$$b_{i,1} = 0^{m_{i,1}} 1 (01)^{k_{i,1}} 1^{n_{i,1}} 0 (10)^{l_{i,1}}$$

par le lemme 7.3.10, et le bloc suivant 0^{2k} est rempli avec les entrées $(10)^k$. Comme ces entrées finissent pas un 0, on peut les concaténer grâce au lemme 7.3.6. Enfin, pour remplir le 0 supplémentaire, il suffit de poser le motif une fois sur cet indice, c'est-à-dire d'avoir le motif des entrées 1.
Le deuxième bloc des $u_{i,2}$ seul serait rempli par les

$$b_{i,2} = 0^{m_{i,2}} 1 (01)^{k_{i,2}} 1^{n_{i,2}} 0 (10)^{l_{i,2}}$$

concaténés pour les mêmes raisons.
Seulement, pour remplir $u_{1,1} \cdots u_{r,1} 0^{2k} 0$ il faut les entrées

$$b_{1,1} \cdots b_{r,1} (10)^k 1$$

on est dans le cas du lemme 7.3.8 car le motif se termine par un 1.
Donc pour remplir UC_2, il faut le motif des entrées

$$b = b_{1,1} \dots b_{r,1} (10)^k 1 \overline{b_{1,2} \cdots b_{s,2}}.$$

La sous-couverture miroir UC_2^{R} vaut

$$UC_2^{\mathrm{R}} = u_{s,2}^{\mathrm{R}} \cdots u_{1,2}^{\mathrm{R}} 0^{2k} 0 u_{r,1}^{\mathrm{R}} \cdots u_{1,1}^{\mathrm{R}}$$

où

$$u_{i,j}^{\mathrm{R}} = 00^{2l_{i,j}} 1^{n_{i,j}} 00^{2k_{i,j}} 1^{m_{i,j}}.$$

Pour remplir un bloc du type $u_{i,j}^{\mathrm{R}}$ il faut les entrées

$$b_{i,j}'^{\mathrm{R}} = (10)^{l_{i,j}} 11^{n_{i,j}} 0 (10)^{k_{i,j}} 0^{m_{i,j}}$$

et pour le bloc 0^{2k} il faut les entrées $(10)^k$. Comme précédemment, ces motifs se terminent tous par un 0 donc on peut les concaténer grâce au lemme 7.3.6, sauf le 0 qui se remplit avec un 1, donc le deuxième bloc pour $j = 1$ suit le lemme 7.3.8.
Ainsi, pour remplir la sous-couverture miroir UC_2^{R}, il faut le motif des entrées

$$b' = b_{s,2}'^{\mathrm{R}} \cdots b_{1,2}'^{\mathrm{R}} (10)^k 1 \overline{b_{r,1}'^{\mathrm{R}} \cdots b_{1,1}'^{\mathrm{R}}}.$$

Si on regarde le mot miroir des $b_{i,j} = 0^{m_{i,j}} 1 (01)^{k_{i,j}} 1^{n_{i,j}} 0 (10)^{l_{i,j}}$ on a

$$\begin{aligned} b_{i,j}^{\mathrm{R}} &= (01)^l 01^{n_{i,j}} (10)^{k_{i,j}} 10^{m_{i,j}} \\ &= 0 (10)^{l_{i,j}} 1^{(n_{i,j}+1)} 0 (10)^{k_{i,j}} 0^{(m_{i,j}-1)} \end{aligned}$$

Alors

$$\begin{aligned} b^{\mathrm{R}} &= (b_{1,1} \dots b_{r,1} (10)^k 1 \overline{b_{1,2} \dots b_{s,2}})^{\mathrm{R}} \\ &= \overline{b_{s,2}}^{\mathrm{R}} \dots \overline{b_{1,2}}^{\mathrm{R}} 1 (01)^k b_{r,1}^{\mathrm{R}} \dots b_{1,1}^{\mathrm{R}} \\ &= 1 \overline{b_{s,2}}'^{\mathrm{R}} \dots \overline{b_{1,2}}'^{\mathrm{R}} (01)^k 0 b_{r,1}'^{\mathrm{R}} \dots (10)^{l_{1,1}} 11^{n_{1,1}} 0 (10)^{k_{1,1}} 0^{(m_{1,1}-1)} \\ &= \overline{1 b_{s,2}'^{\mathrm{R}} \dots b_{1,2}'^{\mathrm{R}} (10)^k 1 \overline{b_{r,1}}'^{\mathrm{R}} \dots \overline{(10)^{l_{1,1}} 11^{n_{1,1}} 0 (10)^{k_{1,1}} 0^{(m_{1,1}-1)}}} \end{aligned}$$

Soit

$$(b[0, \dots, n-2])^{\mathrm{R}} = \overline{b_{s,2}'^{\mathrm{R}} \dots b_{1,2}'^{\mathrm{R}} (10)^k 1 \overline{b_{r,1}}'^{\mathrm{R}} \dots \overline{(10)^{l_{1,1}} 11^{n_{1,1}} 0 (10)^{k_{1,1}} 0^{(m_{1,1}-1)}}}$$

Et donc

$$(b[0, \dots, n-2])^{\mathrm{R}} \overline{b[n-1]} = \overline{0 b_{s,2}'^{\mathrm{R}} \dots b_{1,2}'^{\mathrm{R}} (10)^k 1 \overline{b_{r,1}}'^{\mathrm{R}} \dots \overline{(10)^{l_{1,1}} 11^{n_{1,1}} 0 (10)^{k_{1,1}} 0^{(m_{1,1}-1)}} 1} =$$
$$\overline{0 b_{s,2}'^{\mathrm{R}} \dots b_{1,2}'^{\mathrm{R}} (01)^k 0 \overline{b_{r,1}}'^{\mathrm{R}} \dots \overline{(10)^{l_{1,1}} 11^{n_{1,1}} 0 (10)^{k_{1,1}} 0^{(m_{1,1}-1)} 0}} = \overline{b'}.$$

On a bien $b' = \overline{(b[0, \dots, n-2])^{\mathrm{R}} b[n-1]}$.

□

Lemme 7.3.12. *Pour tout $k \in \mathbb{N}^*$, chaque ligne du tableau $T(2^k)$ contient un nombre pair de 1, sauf la dernière qui en contient exactement $2^k - 1$.*

Démonstration. Montrons le par récurrence sur k.

Lorsque $k = 1$, $T(2) = (10)$ et c'est évident.

Si le résultat est vrai pour un k fixé, montrons le sur le tableau $T(2^{k+1})$.

On rappelle que le tableau $T(2^{k+1})$ est de taille $(2^{k+1}, 2^{k+1} - 1)$ et satisfait à $T(2^{k+1}) =$

$$\left(\begin{array}{c|cc} T(2^k) & T(2^k) & \\ \hline & 11\ldots1 & 0 \\ & 11\ldots1 & 0 \\ \widetilde{T(2^k)} & \vdots & \vdots \\ & 11\ldots1 & 0 \end{array}\right)$$

où $\widetilde{T(2^k)}$ est un tableau de taille $(2^k, 2^k)$ défini tel que sa première ligne soit $0^{(2^k-1)}1$ et que ses dernières lignes soient $T(2^k)$ dont la dernière colonne de 0 est remplacée par une colonne de 1.

Les $2^k - 1$ première lignes contiennent évidemment un nombre pair de 1 puisqu'elles sont composées de deux mots identiques concaténés : $(T(2^k)|T(2^k))$.

La ligne suivante est $0^{(2^k-1)}11^{2^k-1}0$, et a bien un nombre pair de 1.

Les lignes suivantes, sauf la dernière, sont la concaténation d'une ligne ℓ de $T(2^k)$, d'un 1 et de $(2^k - 1)$ fois la lettre 1. Par hypothèse de récurrence, ℓ a un nombre pair de 1, donc ces lignes de $T(2^{k+1})$ aussi.

La dernière ligne de $T(2^{k+1})$ est la concaténation de la dernière ligne de $T(2^k) = 1^{2^k-1}$, d'un 1 et de $1^{2^k-1}0$. Elle vaut

$$1^{2^k-1}11^{2^k-1}0 = 1^{2^{k+1}-1}0.$$

□

7.3.2 *Preuve du théorème 7.3.3*

Démonstration. De la même manière que pour démontrer le théorème 7.2.3, nous allons montrer que les lignes de $W(2^k + 1)$ sont les sous-couvertures de taille 2^k formées par $A(2^k + 1)$ et le mot $b(2^k + 1)$ lu jusqu'à la ligne précédente dans $W(2^k + 1)$.

C'est-à-dire, en utilisant le lemme 7.1.15, et en écrivant

$$b(2^k + 1) = b_1 \ldots b_{n+1}$$

avec les mots b_i de taille 2^k tels que pour tout i, b_i est la sous-couverture de $A(2^k + 1)$ et $b_1 \ldots b_{i-1}$ (avec la convention $b_0 = 0^{2^k}$ comme l'explique la remarque 7.1.12), on veut montrer que

$$W(2^k + 1) = \begin{pmatrix} b'_1 \\ b'_2 \\ \cdots \\ b'_{n+1} \end{pmatrix}$$

de le même manière que lorsqu'on note

$$T(2^k) = \begin{pmatrix} \tilde{b_1} \\ \tilde{b_2} \\ \dots \\ \tilde{b_n} \end{pmatrix}.$$

Comme pour tout i, les sous-couvertures successives b_i sont de taille $|b_i| = 2^k < \max A(2^k+1)$, alors on ne les pavera qu'avec les deux premiers *onsets* du motif, c'est-à-dire, avec le motif 11.

En reprenant la convention $b_0 = 0^{2^k}$, on sait que pour remplir cette première sous-couverture qui commence à l'indice 0 avec le motif 11, il faut les entrées $b'_1 = (10)^k$. Donc par le lemme 7.1.15, la sous-couverture suivante commençant à l'indice (2^k+1) vaut $(10)^k$.

Seulement $T(2^k+1)$ a 2^k colonnes, et non 2^k+1 colonnes, donc nous cherchons la sous-couverture b_1 de taille 2^k commençant à l'indice 2^k. On a donc

$$b_1 = 0(10)^{k-1}1 = (01)^k = \tilde{b_1}^{\mathrm{R}}.$$

Remarquons maintenant grâce au lemme 7.3.12 que toutes les lignes de $T(2^k)$, sauf la dernière, ont leur taille qui est de même parité que le nombre de 1 qu'elles contiennent, car elles sont de taille 2^k.

Donc par le lemme 7.3.11, on a automatiquement que

$$b'_2 = \tilde{b_2}\left[0,\dots,2^k-2\right]^{\mathrm{R}}\tilde{b_2}\left[2^k-1\right] = \tilde{b_2}\left[0,\dots,2^k-2\right]^{\mathrm{R}}0.$$

Seulement, nous cherchons à nouveau à remplir la sous-couverture de taille 2^k commençant à l'indice 2×2^k, c'est-à-dire

$$b_2 = 0b'_2\left[0,\dots,2^k-2\right] = \tilde{b_2}^{\mathrm{R}}.$$

Donc par le lemme 7.3.11, on a automatiquement que

$$b'_3 = \tilde{b_3}\left[0,\dots,2^k-2\right]^{\mathrm{R}}\tilde{b_3}\left[2^k-1\right] = \tilde{b_2}\left[0,\dots,2^k-2\right]^{\mathrm{R}}1.$$

De la même manière, on montre que pour $i\in\{0,n\}$ toute ligne b'_i de $W(2^k+1)$ vaut

$$b'_i = \tilde{b_i}\left[0,\dots,2^k-2\right]^{\mathrm{R}}\tilde{b_i}\left[2^k-1\right],$$

et que la sous-couverture à remplir suivante de taille 2^k commençant à l'indice $i\times 2^k$ vaut $\tilde{b_i}^{\mathrm{R}}$.

Et comme toutes ces lignes finissent par un 0, on peut concaténer ces motifs b'_i pour obtenir le mot $b(2n+1)$ par le lemme 7.3.6.

Pour $i = n+1$, on ne peut plus utiliser le lemme 7.3.11 de la même manière, puisque $\tilde{b_n}$ a un nombre impair de 1, et on est donc dans le second cas du lemme. On peut décider de la convention $\tilde{b_{n+1}} = 0^{2^k-1}$**.** (en comptant le 0 en dernière position sur la ligne n de $T(2^k)$),

puisque le pavage $(A(2^k), B(2^k)$ est compact et qu'on peut ajouter 2^k 0 au mot $b(2^k)$ et obtenir le même canon. Et comme le pavage est périodique, après ces 0, si on voulait continuer le pavage, le mot suivant serait à nouveau $b(2^k)$, c'est-à-dire qu'en complétant $b(2^k)$ par 2^k lettres, alors $\tilde{b}_{n+1} = 0^{2^k}1$.

Alors, par le second cas du lemme 7.3.11,

$$\begin{aligned} b'_{n+1} &= \overline{\tilde{b_{n+1}}\,[0, \dots, 2^k - 2]}^{\mathrm{R}}\, \tilde{b_{n+1}}\left[2^k - 1\right] \\ &= \overline{0^{2^k-1}}1 = 1^{2^k} \end{aligned}$$

Pour une autre façon de comprendre le ligne b'_{n+1}, il suffit de regarder la sous-couverture précédente de taille $2^k + 1$ qui vaut

$$b'_{n-1}\left[2^k - 1\right] b'_n = 01^{2^k-1}0$$

par le lemme 7.3.12. Elle se remplit avec le motif $A(2^k + 1)$ par les entrées $1^{2^k}0$, ce qui se montre de la même façon que le lemme 7.2.11.

Donc, le tableau $T(2^k + 1) = \begin{pmatrix} b'_1 \\ b'_2 \\ \cdots \\ b'_{n+1} \end{pmatrix}$ est tel que pour $i \leq n$,

$$b'_i = \tilde{b_i}\left[0, \dots, 2^k - 2\right]^{\mathrm{R}} \tilde{b_i}\left[2^k - 1\right]$$

et $b'_{n+1} = 1^{2^k}$.

Si on considère le tableau $T'(2^k + 1)$ qui vaut $T(2^k + 1)$ sans sa dernière ligne $T'(2^k + 1) = \begin{pmatrix} b'_1 \\ b'_2 \\ \cdots \\ b'_n \end{pmatrix}$.

Alors, ce tableau vaut exactement le tableau $T(2^k)$ dont on a retiré la dernière colonne, transformé chaque ligne en son miroir, puis remis la dernière colonne. Or on peut remarquer que $W(2^k + 1)$ vaut exactement le tableau $T(2^k)$ auquel on a appliqué ces transformations, puis ajouté une dernière ligne de 1.

On a alors bien que pour tout $k \in \mathbb{N}^*$,

$$W(2^k + 1) = \left(\frac{T'(2^k + 1)}{1 \cdots 1}\right) = T(2^k + 1).$$

□

Exemple 7.3.13. Pour voir ce lien entre $T(2^k)$ et $T(2^k + 1)$, nous allons les représenter pour $k = 3$ ainsi que $T'(2^k + 1)$:

$$T(8) = \left(\begin{array}{ccccccc|c} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{array}\right)$$

$$T(9) = \left(\begin{array}{cccccccc} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}\right)$$

$$\text{et } T'(9) = \left(\begin{array}{ccccccc|c} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{array}\right)$$

où on voit bien que la partie dans la dernière colonne de $T'(9)$ est le même tableau que $T(8)$ sans sa dernière colonne où toutes les lignes sont en miroir.

Cette construction par récurrence du motif des entrées permet également de construire les pavages compacts modulo 2 minimaux avec les motifs de la forme $A(2^k+1) = \{0, 1, 2^k+1\}$ en temps logarithmique par rapport à la taille du pavage, tandis que l'algorithme 1 permettait lui de les obtenir en temps linéaire seulement. La structure de ces pavages particuliers permet d'obtenir de la même manière des résultats combinatoires.

Corollaire 7.3.14. *Pour tout $k \in \mathbb{N}^*$, le canon formé du couple $(A(2^k+1), B(2^k+1))$ est le plus petit canon compact modulo 2 de $\mathbb{Z}_N$,*

$$\textit{sa taille vaut } N = 4^k + 2^k + 1$$

$$\textit{il a } |B(2^k)| = 4^k - 3^k + 2^k \textit{ entrées et}$$

$$|D_{(A(2^k+1),B(2^k+1)),2}| = 4^k + 2^k - \frac{3^{k+1}+1}{2} \textit{ donsets.}$$

Démonstration. Si on note $b(k) = |B(2^k+1)|$ le nombre d'entrées pour ce canon, alors ce nombre correspond au nombre de 1 dans le tableau $T(2^k+1)$ et on peut les compter par récurrence.

Nous avons $b(1) = 3$, $b(2) = 11$ et $b(3) = 45$.

La construction du tableau $T(2^{k+1} + 1)$ nous donne l'égalité

$$b(k + 1) = 3 \times b(k) - 2^k + 4^k.$$

En effet, le terme $3 \times b(k)$ vient du tableau $T(2^k + 1)$ écrit trois fois dans le tableau $T(2^{k+1} + 1)$, et le terme -2^k vient de la ligne de 1 changée en une ligne de 0 dans $\widetilde{T(2^k + 1)}$, puis le terme en 4^k vient du bloc de 1 en bas à gauche de taille $(2^k, 2^k)$.

Cette égalité sur $b(k)$ se récrit

$$b(k + 3) = 9b(k + 2) - 26b(k + 1) + 24b(k)$$

et donc

$$b(k) = 4^k - 3^k + 2^k.$$

Afin d'obtenir la taille du canon N, il suffit de savoir que le canon se terminera lorsque le motif $A(2^k + 1)$ est placé sur le dernier 1 de $B(2^k + 1)$, c'est-à-dire

$$N = (2^k) \times 2^k + 2^k + 1 = 4^k + 2^k + 1.$$

En effet, le premier produit vient de la taille $(2^k, 2^k)$ du tableau $T(2^k + 1)$, dont le dernier 1 se trouve en dernière position, et le terme $+2^k + 1$ vient de l'addition du motif $A(2^k + 1)$ sur cet indice.

Finalement, pour énumérer les *donsets*, nous utilisons le lemme 7.2.15 qui nous donne

$$|A(2^k + 1)| \times |B(2^k + 1)| - 2\sharp D(2^k + 1) = N$$

car $|A(2^k + 1)| = 3$.

Puis s'ensuit le résultat

$$|D_{(A(2^k+1),B(2^k+1),2}| = 4^k + 2^k - \frac{3^{k+1} + 1}{2}.$$

□

Les corollaires des théorèmes 7.2.3 et 7.3.3 nous donnent les tailles des canons compacts minimaux pour les motifs de la forme $\{0, 1, 2^k\}$ et $\{0, 1, 2^k + 1\}$. Le premier canon s'obtient par la construction d'un tableau représentant les entrées par récurrence sur k, et le second se déduit de ce tableau par quelques transformations immédiates. Ces deux familles de motifs sont particulières au sens où les canons compacts minimaux qu'ils permettent de construire sont bien plus petits que pour tous les autres motifs $\{0, 1, n\}$. On peut voir sur la Figure 7.5 les tailles N des canons compacts minimaux obtenues par l'algorithme 1 en échelle logarithmique par rapport à n, dernier *onset* du motif $A(n)$ avec lequel il est construit. Dans le cas où $n = 2^k$ ou $n = 2^k + 1$, les tailles sont représentées en jaune et liées par une ligne

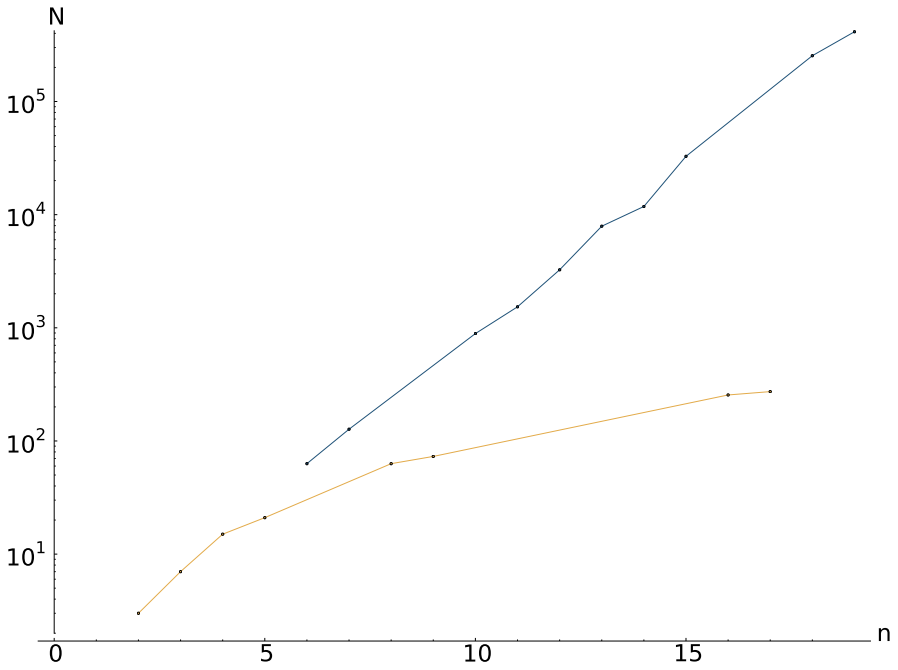


FIGURE 7.5: Les tailles N minimales de canons compacts modulo 2 par des motifs $A(n) = \{0, 1, n\}$, en jaune lorsque $n = 2^k$ ou $n = 2^k + 1$ et en bleu sinon.

jaune, et on remarque une croissance de N en linéaire en n, dans le cas contraire, les tailles sont en bleu et liées par une ligne bleue, et on voit une croissance exponentielle en n.

Observation 7.3.15. Lorsqu'on regarde les tableaux de type $T(2^k)$, on voit des successions de 1 d'une ligne à l'autre. On peut souvent remarquer que si à la ligne ℓ on a le motif 010 en indice i, on aura à la ligne $\ell + 1$ le motif 0110 en indice $i - 1$, etc. jusqu'à ce qu'on ait des 0 sous ces 1. Ces croissances de nombre de 1 successifs forment des triangles rectangles orientés vers la gauche. Ces croissances peuvent se comprendre grâce au lemme 7.2.8. Comme ces motifs de la forme 01^n0 viennent des pavages avec le motif 11, on peut en fait les trouver dans tous les tableaux $T(2n)$. Ils sont par exemple mis en évidence Figure 7.6 où sont représentées les 40 premières lignes du tableau $T(34)$. On peut voir ces successions de triangles ici mis en évidence en rouge, et le lecteur est encouragé à les regarder dans les tableaux $T(2^k)$ afin de comprendre la régularité de ces motifs, et de voir apparaître des triangles à la Sierpiński. Il est intéressant de noter que, contrairement aux tableaux $T(2^k)$, il est possible que certains triangles "débordent" d'un côté du tableau à l'autre. Par exemple, on peut remarquer tout en bas à droite du tableau sur la Figure 7.6 la fin d'un triangle qui commence tout en bas à gauche. Ceci est cohérent car nos tableaux $T(n)$ ont une structure de tore. Les bords droit

et gauche sont collés puisqu'il s'agit du même mot, et le haut du tableau se colle au bas puisque le pavage est compact.

Comme le tableau $T(2^k+1)$ est obtenu par un miroir du tableau $T(2^k)$, les triangles de 1 apparaissent de la même façon, mais sont orientés vers la droite. Et on peut remarquer pareillement ces triangles pour tous les motifs de la forme $T(2n+1)$. Par exemple sur la Figure 7.7 sont représentées les 40 premières lignes du tableau $T(35)$. On remarque aussi des similarités troublantes entre $T(34)$ et le miroir de $T(35)$. Les parties vides de triangle dans $T(34)$ apparaissent dans le miroir de $T(35)$ comme des parties qui contiennent les plus petits triangles possible et en plus grand nombre. Aussi, à partir de la 30ème ligne, la densité des triangles change brutalement pour les deux. En annexe se trouvent de plus grands exemples pour le lecteur intéressé.

Cela laisse à penser que, de même que montré dans la preuve de 7.3.3, les miroirs des tableaux $T(2n+1)$ et les tableaux $T(2n)$ sont similaires à quelques transformations près. Bien sûr, les mêmes transformations qu'entre $T(2^k)$ et $T(2^k+1)$ ne s'appliquent pas puisque les lignes des $T(2n)$ ne se terminent pas forcément par un zéro et on ne peut plus appliquer la preuve. Néanmoins, cela reste une piste intéressante à creuser, et on pourrait alors ne s'intéresser qu'aux $T(2n)$.

7.4 Similitudes sur les motifs des entrées pour la famille des motifs $\{0,1,n\}$

Définition 7.4.1. Le *gabarit* est la fonction $\mathcal{G} : \mathbb{N}^* \to \mathbb{N}^*$ définie par $\mathcal{G}(n) = 2^{\lfloor \log_2(n) \rfloor + 1}$. On prolonge par convention la fonction à 0 par $\mathcal{G}(0) = 1$.

Nous allons nous intéresser aux préfixes de taille $\mathcal{G}(n)$ des mots écrits sur la ligne n du tableau $T(2^k)$.

Définition 7.4.2. Soit $k \geq n$, on définit $TR(n)$ le sous-tableau de $T(2^k)$ de taille $(\frac{\mathcal{G}(n)}{2}, \mathcal{G}(n))$ commençant à la ligne 2^{n-1} et à la colonne 1.

Exemple 7.4.3. Dans le tableau $T(16)$ nous avons encadré les sous-tableaux $TR(1)$, $TR(2)$, $TR(3)$ et $TR(4)$.

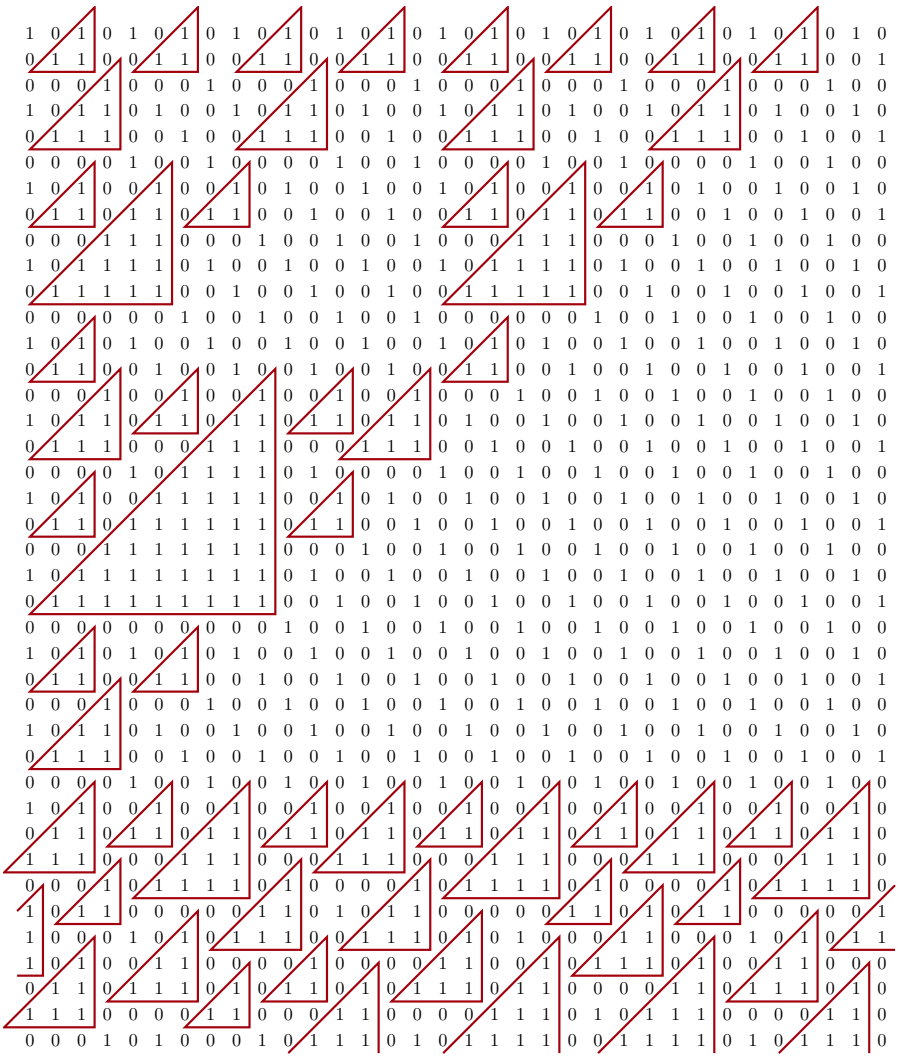


FIGURE 7.6: Les 40 premières lignes de $T(34)$, avec mise en valeur en rouge des croissances des motifs 01^n0.

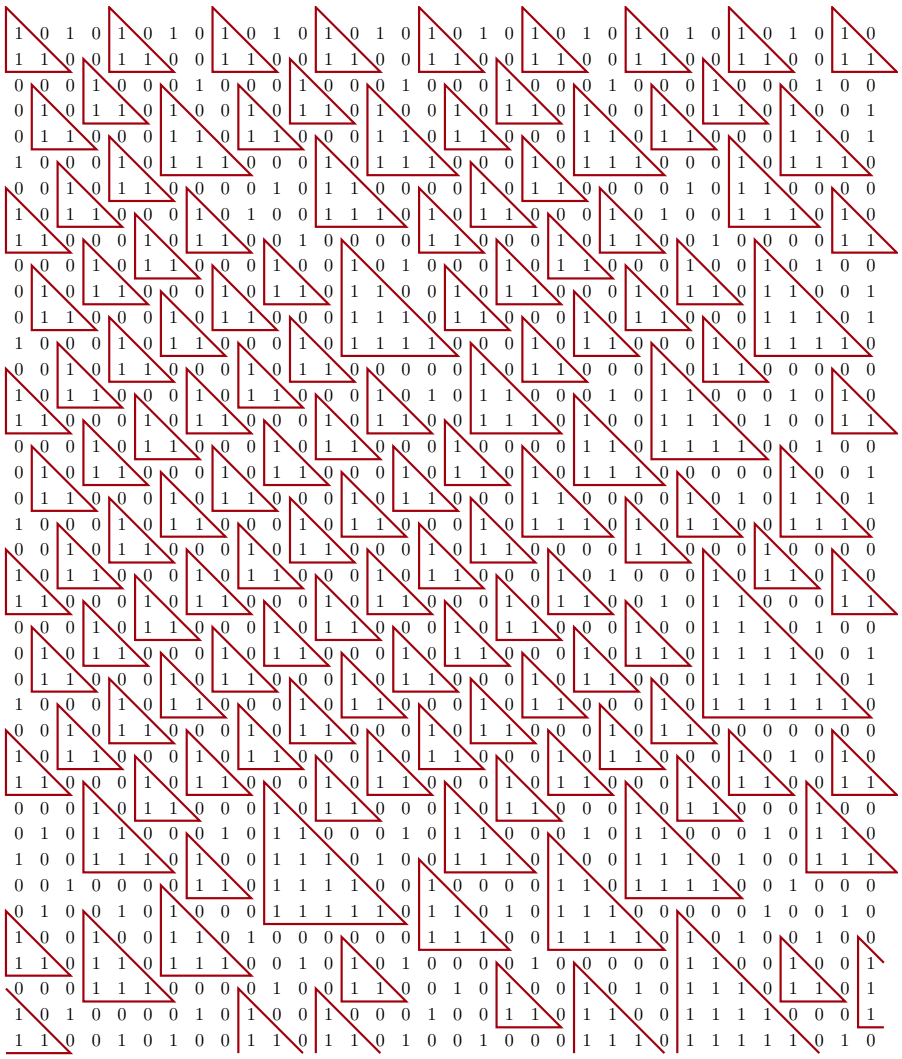


FIGURE 7.7: Les 40 premières lignes de $T(35)$, avec mise en valeur en rouge des croissances des motifs 01^n0 en miroir.

$$T(16) = \left(\begin{array}{cccccccccccccccc}
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{array}\right)$$

Par exemple :

$$TR(3) = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Proposition 7.4.4. *Pour tout $n \in \mathbb{N}^*$, les $TR(n)$ satisfont à*

$$TR(n+1) = \left(\begin{array}{ccc|c}
 & 0 & & 1\cdots10 \\
\widetilde{TR(n)} & \vdots & \widetilde{TR(n)} & 1\cdots10 \\
 & 0 & & 1\cdots10 \\
\hline
 & 1 & \cdots & 1\cdots10 \\
\widetilde{TR(n)} & \vdots & \vdots & 1\cdots10 \\
 & 1 & \cdots & 1\cdots10
\end{array}\right)$$

où $\widetilde{TR(n)}$ vaut $TR(n)$ privé de ses $\mathcal{G}(n)-\mathcal{G}(n-1)+1$ dernières colonnes.

Exemple 7.4.5.

$$T(4) = \left(\begin{array}{ccc|c|ccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
\hline
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{array}\right)$$

avec $\widetilde{TR(3)}$ $\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$

Démonstration. Commençons par remarquer que $TR(n)$ est la moitié inférieure du tableau $T(2^n)$.

Or on a l'égalité $T(2^{n+1}) = \left(\begin{array}{c|cc} T(2^n) & T(2^n) & \\ \hline & 11\ldots1 & 0 \\ & 11\ldots1 & 0 \\ \widetilde{T(2^n)} & \vdots & \vdots \\ & 11\ldots1 & 0 \end{array}\right)$

où $\widetilde{T(2^n)}$ est défini tel que sa première ligne soit $0^{(2^k-1)}1$ et que ses dernières lignes soient $T(2^n)$ dont la dernière colonne de 0 est remplacée par une colonne de 1.

C'est-à-dire $TR(n+1) = \left(\begin{array}{c|cc} & 11\ldots1 & 0 \\ \widetilde{T(2^n)} & \vdots & \vdots \\ & 11\ldots1 & 0 \end{array}\right)$

et donc que $\widetilde{TR(n)}$ vaut $\widetilde{T(2^{n-1})}$ privé de sa dernière colonne qu'on note $\widetilde{T'(2^{n-1})}$.

On a donc $TR(n+1) = \left(\begin{array}{c|c|cc} 0\cdots0 & 1 & 11\ldots1 & 0 \\ T'(2^n) & \vdots & \vdots & \vdots \\ & 1 & 11\ldots1 & 0 \end{array}\right)$

où $T'(2^n)$ vaut $T(2^n)$ privé de sa dernière colonne.

Soit $TR(n+1) = \left(\begin{array}{cc|c|cc} 0\cdots0 & 0\cdots0 & 1 & 11\ldots1 & 0 \\ T(2^{n-1}) & T'(2^{n-1}) & \vdots & \vdots & \vdots \\ \hline & 11\ldots1 & \vdots & \vdots & \vdots \\ \widetilde{T(2^{n-1})} & 11\ldots1 & \vdots & \vdots & \vdots \\ & 11\ldots1 & 1 & 11\ldots1 & 0 \end{array}\right)$

C'est-à-dire $TR(n+1) = \left(\begin{array}{c|c|cc|cc} 0\cdots0 & 0 & 0\cdots0 & 1 & 11\ldots1 & 0 \\ T'(2^{n-1}) & \vdots & T'(2^{n-1}) & \vdots & \vdots & \vdots \\ & 0 & & \vdots & \vdots & \vdots \\ \hline & 1 & 11\ldots1 & \vdots & \vdots & \vdots \\ \widetilde{T'(2^{n-1})} & \vdots & 11\ldots1 & \vdots & \vdots & \vdots \\ & 1 & 11\ldots1 & 1 & 11\ldots1 & 0 \end{array}\right)$

$$\text{Et donc } TR(n+1) = \left(\begin{array}{c|c|c|c|c} & 0 & & 1 & 11\dots1\ \ 0 \\ \widetilde{T'(2^{n-1})} & \vdots & \widetilde{T'(2^{n-1})} & \vdots & \vdots\ \ \vdots \\ & 0 & & \vdots & \vdots\ \ \vdots \\ \hline & 1 & 11\dots1 & \vdots & \vdots\ \ \vdots \\ \widetilde{T'(2^{n-1})} & \vdots & 11\dots1 & \vdots & \vdots\ \ \vdots \\ & 1 & 11\dots1 & 1 & 11\dots1\ \ 0 \end{array}\right)$$

Comme $\widetilde{TR(n)} = \widetilde{T'(2^{n-1})}$ on a bien l'égalité recherchée.

□

Proposition 7.4.6. *Pour tout n, $TR(n)$ a un nombre pair de 1 sur chacune de ses lignes sauf sur la dernière.*

Démonstration. On le démontre facilement par récurrence. $TR(1) = (10)$ a bien un nombre impair de 1 sur sa dernière (et unique) ligne.

Si cela est vrai pour un n fixé, on utilise la récurrence de la propriété 7.4.4

$$TR(n+1) = \left(\begin{array}{c|c|c|c} & 0 & & 1\cdots10 \\ \widetilde{TR(n)} & \vdots & \widetilde{TR(n)} & 1\cdots10 \\ & 0 & & 1\cdots10 \\ \hline & 1 & \cdots & 1\cdots10 \\ \widetilde{TR(n)} & \vdots & \vdots & 1\cdots10 \\ & 1 & \cdots & 1\cdots10 \end{array}\right)$$

où $\widetilde{TR(n)}$ vaut $TR(n)$ privé de ses $\mathcal{G}(n) - \mathcal{G}(n-1) + 1$ dernières colonnes.

Les $\mathcal{G}(n) - \mathcal{G}(n-1) + 1$ dernières colonnes de $TR(n)$ n'influencent pas la parité de son nombre de 1 ligne à ligne, puisque chaque ligne de $TR(n)$ se termine par $1^{\mathcal{G}(n)-\mathcal{G}(n-1)}0$ et

$$\mathcal{G}(n) - \mathcal{G}(n-1) = 2^{\lfloor \log_2(n) \rfloor + 1} - 2^{\lfloor \log_2(n-1) \rfloor + 1} = 2^{\lfloor \log_2(n-1) \rfloor + 1}$$

est bien entendu pair.

On peut donc se contenter de regarder la parité du nombre de 1 sur les lignes de $\widetilde{TR(n+1)}$. Le premier bloc se compose de deux lignes identiques de $\widetilde{TR(n)}$ concaténées de part et d'autre d'un 0, donc on a bien un nombre pair de 1.

Les lignes du bloc du dessous sont les lignes de $\widetilde{TR(n)}$ concaténées à $1^{2^{\lfloor \log_2(n-1) \rfloor + 1}}$, donc ont même parité que $\widetilde{TR(n)}$, c'est-à-dire qu'elles ont toutes un nombre pair de 1 sauf la dernière. □

Lemme 7.4.7. *Soit UC_2 une sous-couverture de taille n contenant k fois la lettre 1, et b le motif des entrées pour la remplir avec le motif 11.*

– *Si n est de même parité que k, alors b se termine par un 0.*

– *Si n n'est pas de même parité que k, alors b se termine par un 1.*

Démonstration.

– Si n est de même parité que k, comme vu dans la preuve du lemme 7.3.11, si UC_2 commence par un 1, alors elle sera de la forme

$$UC_2 = 1^{m_1}00^{2k_1}1^{n_1}00^{2l_1}\cdots 1^{m_r}00^{2k_r}1^{n_r}00^{2l_r}$$

avec $r \in \mathbb{N}$ et pour tout $i \in \{1,\dots,r\}$, $m_i \in \mathbb{N}^*$ et $n_i, k_i, l_i \in \mathbb{N}$.
Elle sera remplie par $b = b_1 \cdots b_r$, avec

$$b_i = 0^{m_i}1(01)^{k_i}1^{n_i}0(10)^{l_i}$$

et donc se termine par un 0.
De même, si UC_2 commence par un 0, alors elle peut s'écrire sous la forme

$$UC_2 = 00^{2k_1}1^{n_1}00^{2l_1}1^{m_1}\cdots 00^{2k_r}1^{n_r}00^{2l_r}1^{m_r}$$

avec $r \in \mathbb{N}$ et pour tout $i \in \{1,\dots,r\}$, $n_i \in \mathbb{N}^*$ et $m_i, k_i, l_i \in \mathbb{N}$.
Elle sera remplie par $b = b_1 \cdots b_r$, avec

$$b_i = (10)^{k_i}1^{(n_i+1)}0(10)^{l_i}0^{m_i}$$

et donc se termine par un 0.

– Si n n'est pas de même parité que k, de la même façon que dans la preuve du lemme 7.3.11, si par exemple UC_2 commence par un 1, on peut écrire

$$UC_2 = u_{1,1}\cdots u_{r,1}0^{2k}0u_{1,2}\dots u_{s,2}$$

avec

$$u_{i,j} = 1^{m_{i,j}}00^{2k_{i,j}}1^{n_{i,j}}00^{2l_{i,j}}$$

et $r, s \in \mathbb{N}$ et pour tout $i \in \{1,\dots,\max(r,s)\}$ et tout $j \in \{1,2\}$, $m_{i,j} \in \mathbb{N}^*$ et $k, n_{i,j}, k_{i,j}, l_{i,j} \in \mathbb{N}$.
Alors, elle sera remplie par

$$b = b_{1,1}\cdots b_{r,1}(10)^k 1\overline{b_{1,2}\dots b_{s,2}}$$

avec

$$b_{i,j} = 0^{m_{i,j}}1(01)^{k_{i,j}}1^{n_{i,j}}0(10)^{l_{i,j}}$$

et donc se termine par un 1.

□

Théorème 7.4.8. *Pour tout $n \in \mathbb{N}^*$, le mot sur la ligne n du tableau $T(2^k)$ est la concaténation autant de fois que nécessaire pour remplir la ligne de son préfixe de taille $\mathcal{G}(n)$.*

Démonstration. On suppose que $k \geq n$, sinon sa ligne ℓ n'est pas assez grande pour avoir un préfixe strict de taille $\mathcal{G}(n)$ et le théorème est évident puisque le mot est lui même, concaténé 1 fois.

Soit $n \in \mathbb{N}^*$, on note $u(n)$ le mot écrit sur la ligne n, et

$$u'(n) = u(n)\,[0, \ldots, \mathcal{G}(n) - 1]$$

son préfixe de taille $\mathcal{G}(n)$.

Montrons ce résultat par récurrence sur $n \in \mathbb{N}^*$.

La première ligne de $T(2^k)$ vaut $(10)^k$ sur laquelle est bien écrit un mot constitué de concaténations successives de son préfixe 10 de taille $\mathcal{G}(1) = 2$.

Si cela est démontré jusqu'à la ligne $n-1$, montrons le pour la $n$ième ligne de $T(2^k)$.

Par définition des tableaux TR, le préfixe $u'(n)$ est le mot écrit sur une des lignes de $TR(m)$, avec $m = \lfloor \log_2(n) \rfloor + 1$.

Si n n'est pas de la forme 2^l, alors le mot $u_0 = u'(n-1)$ écrit comme préfixe sur sa ligne précédente $n-1$ est aussi écrit dans le même $TR(m)$. Comme il n'est pas sur la dernière ligne de $TR(m)$, u_0 a un nombre pair de 1 par le lemme 7.4.6 et est de taille $\mathcal{G}(n-1)$ paire. Donc pas le lemme 7.4.7, le mot pour le remplir avec le motif 11 se termine par un 0. Or, par la remarque 7.1.17 et la définition 7.1.18 du tableau $T(2^k)$, ce mot est exactement $u'(n)$.

Sur la ligne $n-1$, le mot

$$u(n-1) = u'(n-1)u'(n-1)\cdots u'(n-1)$$

est une concaténation, dont chaque bloc se remplit avec le motif des entrées $u'(n)$ qui se termine par un 0. Donc par le lemme 7.3.6, le motif des entrées qui remplit $u(n-1)$ est le mot concaténé

$$u(n) = u'(n)u'(n)\cdots u'(n).$$

Si n est de la forme $n = 2^l$, alors le mot $u'(n-1)$ est écrit sur la dernière ligne du tableau $TR(m-1)$ et est de taille deux fois plus petite que le mot $u'(n)$. On notera $u''(n)$ le préfixe de $u'(n)$ de taille $\mathcal{G}(n-1)$. Par le lemme 7.4.6 le mot $u'(n-1)$ a donc un nombre impair de 1 mais est de taille $\mathcal{G}(n-1)$ paire. Donc par le lemme 7.4.7, le mot $u''(n)$ se termine par un 1.

Ainsi, pour paver le mot $u'(n-1)u'(n-1)$ on sait par le lemme 7.3.8 qu'il faudra le mot $u'(n) = u''(n)\overline{u''(n)}$ qui lui se termine par un 0. Donc par le lemme 7.3.6, la ligne n contient bien le mot

$$u''(n)\overline{u''(n)}u''(n)\overline{u''(n)}\cdots u''(n)\overline{u''(n)}.$$

□

Remarque 7.4.9. Cette preuve permet de remarquer que la première ligne des tableaux $TR(m)$ contient toujours un mot de la forme $u\overline{u}$.

Remarque 7.4.10. On définit la suite $a(n)$ comme étant le nombre de 1 du mot $u'(n)$. Alors on peut remarquer que cette suite vérifie la récurrence

$$a(n+1) = 2a(\lfloor \frac{n}{2} \rfloor) + (n \bmod 2)(2^{\lfloor \log_2(n) \rfloor - a(\lfloor \frac{n}{2} \rfloor)})$$

grâce à la proposition 7.4.4. C'est-à-dire que la suite $a(n)$ est la suite A073138 de l'OEIS. Cela veut dire que $a(n)$ est en fait le plus grand nombre possible en écriture binaire ayant exactement le même nombre de 1 et de 0 que n en écriture binaire.

Conjecture 7.4.11. Pour tout $n \geq 2$ et tout $k \in \mathbb{N}^*$, le mot préfixe de taille $\mathcal{G}(m)$ écrit sur la ligne m du tableau $T(n)$ a un nombre pair de 1 si $m \neq 2^k - 1$ et un nombre impair sinon.

De cette conjecture se déduisent de nombreuses propriétés conditionnelles que nous allons exposer ci-après.

Conjecture 7.4.12. Pour tout $n \geq 2$, et $m \in \mathbb{N}^*$, le mot sur la ligne m du tableau $T(n)$ est la concaténation autant de fois que nécessaire pour remplir la ligne de son préfixe de taille $\mathcal{G}(m)$.

Démonstration sous réserve de démonstration de la conjecture 7.4.11. La preuve est exactement la même que celle du théorème 7.4.8 en supposant la conjecture 7.4.11 à la place de la propriété 7.4.6. □

Remarque 7.4.13. Comme pour n quelconque, le tableau $T(n)$ n'a pas un nombre de colonnes qui est une puissance de 2, la concaténation de ces préfixes ne tombe pas rond. C'est-à-dire qu'une ligne contient un mot qui est la concaténation $\lfloor \frac{n}{\mathcal{G}(m)} \rfloor$ fois d'un mot $u'(m)$ de taille $\mathcal{G}(m)$, puis de son préfixe de taille d avec d le reste de la division euclidienne de n par $\mathcal{G}(m)$.

Exemple 7.4.14. Par exemple, regardons la 4ième ligne de $T(38)$ qui vaut

$$10110100101101001011010010110100101101.$$

Comme $\mathcal{G}(4) = 8$ alors $u'(4) = 10110100$ qui est concaténé 4 fois, puis le mot 101101 de taille 6 est concaténé une dernière fois, et on a bien $38 = 8 \times 4 + 6$.

Pour la même raison qu'évoquée en remarque 7.4.9, on voit que le mot $u'(4)$ étant à la ligne $4 = 2^2$ il s'écrit sous la forme

$$u'(4) = 1011\overline{1011}.$$

Définition 7.4.15. Soient $m, n \in \mathbb{N}$ tels que $m \geq n \geq 2$, on note $d(m,n) = 2^{\nu_2(m-n)}$ où ν_2 est la valuation 2-adique définie par

$$\begin{array}{rccl} \nu_2 : & \mathbb{Z} & \longrightarrow & \mathbb{N} \cup \{\infty\} \\ & n & \longmapsto & \begin{cases} \infty & \text{si } n = 0 \\ \max\{k \in \mathbb{N} \mid 2^k \text{ divise } n\} & \text{sinon} \end{cases} \end{array}$$

Conjecture 7.4.16. Soient $m, n \in \mathbb{N}$ tels que $m \geq n \geq 2$, alors les $d(m,n)$ premières lignes du tableau $T(n)$ sont des préfixes des $d(m,n)$ premières lignes du tableau $T(m)$.

Démonstration sous réserve de démonstration de la conjecture 7.4.11. Pour tout $k \in \mathbb{N}^*$, on note les divisions euclidiennes

$$m = q_{m,k} \times \mathcal{G}(k) + d_{m,k}$$

et

$$n = q_{n,k} \times \mathcal{G}(k) + d_{n,k}.$$

Alors

$$m - n = (q_{m,k} - q_{n,k}) \times \mathcal{G}(k) + d_{m,k} - d_{n,k}$$

et par définition pour tout $k \leq d(m,n)$, $d_{m,k} = d_{n,k}$.

Donc, par la conjecture 7.4.12, sur chaque ligne $k \leq d(m,n)$ du tableau $T(m)$ (resp. $T(n)$) se trouve un mot $u'(k)_m$ (resp. $u'(k)_n$) de taille $\mathcal{G}(k)$ concaténé $q_{m,k}$ (resp. $q_{n,k}$) fois suivi de son préfixe de taille $d_{m,k}$.

On va montrer par récurrence sur $k \leq d(m,n)$ que les mots $u'(k)_m$ et $u'(k)_n$ sont égaux.

Pour les deux tableaux $T(n)$ et $T(m)$, sur la première ligne est bien sûr écrite une concaténation de 10 puisqu'il s'agit de remplir la sous-couverture nulle avec le motif 11.

Si les mots $u'(k-1)_m$ et $u'(k-1)_n$ sont égaux, alors la ligne $k-1$ de $T(m)$ (resp $T(n)$) contient le mot $u'(k-1)_m$ concaténé $q_{m,k-1}$ (resp $q_{n,k-1}$) fois suivi de son préfixe de taille $d_{m,k-1}$.

Si son préfixe de taille $d_{m,k-1}$ se termine par un 0, alors par le lemme 7.3.6 et la définition 7.1.18, pour remplir la sous-couverture $u'(k-1)_m$ de taille paire avec le motif 11, il faut le mot $u'(k)_m$ qui se termine par un 0 si $u'(k-1)_m$ a un nombre pair de 1, et qui se termine par un un 1 sinon.

Si le préfixe de taille $d_{m,k-1}$ se termine par un 1, alors par le lemme 7.3.6 et la définition 7.1.18, pour remplir la sous-couverture $u'(k-1)_m$ de taille paire avec le motif 11 il faut le mot $\overline{u'(k)_m}$ qui se termine par un 1 si $u'(k-1)_m$ a un nombre pair de 1, et qui se termine par un un 0 sinon.

Dans le cas où $u'(k)_m$ ou $\overline{u'(k)_m}$ se termine par un 0, alors ces motifs se concatènent par le lemme 7.3.6 et un préfixe de la ligne k du tableau $T(m)$ vaut $(u'(k)_m)^{q_{m,k}}$ ou $(\overline{u'(k)_m})^{q_{m,k}}$; et un préfixe du tableau $T(n)$ vaut $(u'(k)_m)^{q_{n,k}}$ ou $(\overline{u'(k)_m})^{q_{n,k}}$.

Dans le cas où $u'(k)_m$ ou $\overline{u'(k)_m}$ se termine par un 1, alors par le lemme 7.3.8 les motifs $u'(k)_m\overline{u'(k)_m}$ et $\overline{u'(k)_m}u'(k)_m$ se terminent par un 0 et remplissent le motif $u'(k-1)_m u'(k-1)_m$. Alors par le lemme 7.3.6, ils se concatènent et un préfixe de la ligne k du tableau $T(m)$

vaut $(u'(k)_m\overline{u'(k)_m})^{q_{m,k}/2}$ ou $(\overline{u'(k)_m}u'(k)_m)^{q_{m,k}/2}$; et un préfixe du tableau $T(n)$ vaut $(u'(k)_m\overline{u'(k)_m})^{q_{n,k}/2}$ ou $(\overline{u'(k)_m}u'(k)_m)^{q_{n,k}/2}$. On remarque que dans ce second cas, c'est lorsque $\mathcal{G}(k)$ double et donc que k est une puissance de 2, et donc le mot $u'(k)_m$ considéré est deux fois plus long que le mot $u'(k-1)_m$ et est donc $u'(k)_m\overline{u'(k)_m}$ ou $\overline{u'(k)_m}u'(k)_m$.

On a bien montré que pour tout $k \leq d(m,n)$, la ligne k du tableau $T(n)$ contient le mot $u'(k)_m$ concaténé un certain nombre de fois, suivi du mot qui est son préfixe de taille $d_{m,k}$. Cette ligne est bien un préfixe de la ligne k du tableau $T(m)$, puisqu'elle contient elle même le mot $u'(k)_m$ concaténé un nombre de fois plus grand comme $m \leq n$ suivi de son préfixe de $d_{m,k}$.

□

Exemple 7.4.17. Si on s'intéresse aux tableaux $T(34)$ et $T(46)$, alors $46-34=12$ et $d(46,34)=4$. Alors les 5 premières lignes de $T(34)$ et $T(46)$ sont :

$$\begin{pmatrix}
10101010101010101010101010101010\\
01100110011001100110011001100110\\
00010001000100010001000100010001\\
10110100101101001011010010110100\\
01110010011100100111001001110010
\end{pmatrix}$$

$$\begin{pmatrix}
1010101010101010101010101010101010101010101010\\
0110011001100110011001100110011001100110011001\\
0001000100010001000100010001000100010001000100\\
1011010010110100101101001011010010110100101101\\
1000110110001101100011011000110110001101100011
\end{pmatrix}$$

Et on voit bien que les 4 premières lignes de $T(34)$ sont des préfixes des 4 premières lignes de $T(46)$ tandis que leurs 5èmes lignes sont différentes.

Conjecture 7.4.18. Soient $m,n \in \mathbb{N}$ tels que $m \geq n \geq 2$, alors la ligne $d(m,n)+1$ du tableau $T(n)$ contient un mot dont le conjugué est un préfixe du mot sur la ligne $d(m,n)+1$ du tableau $T(m)$.

Démonstration sous réserve de démonstration de la conjecture 7.4.11. En reprenant les notations de la preuve de la conjecture 7.4.16, alors la ligne $k=d(m,n)$ est la première ligne telle que $d_{m,k} \neq d_{n,k}$.

On peut écrire par la définition 7.4.15

$$m-n = 2^k \times r \text{ avec } 2 \not| r. \qquad (1)$$

Sur la ligne $k=2^{\nu_2(m-n)}$ on sait que le mot $u'(k)_m$ de taille $\mathcal{G}(k)=2^{k+1}$ qui est concaténé peut s'écrire sous la forme $u\overline{u}$ comme vu dans la preuve de la conjecture 7.4.16.

Alors on a $m = 2^{k+1}q_m + r_m$ et $n = 2^{k+1}q_n + r_n$ avec $r_m, r_n < 2^{k+1}$. Les restes r_m et r_n sont d'ailleurs les tailles des préfixes p_m et p_n des mots $u'(k)_m$ et $u'(k)_n$ qui complète la ligne.

On peut donc écrire

$$m - n = 2^{k+1}(q_m - q_n) + r_m - r_n = 2^k(2(q_m - q_n)) + r_m - r_n.$$

Donc par l'égalité (1), $|r_m - r_n|$ est un multiple de 2^k. Or $r_m - r_n \neq 0$ puisque $p_m \neq p_n$ et on sait que l'un est préfixe de l'autre par la conjecture 7.4.16. Et comme $r_m, r_n < 2^{k+1}$, $|r_m - r_n| < 2^{k+1}$, et alors $|r_m - r_n| = 2^k$

Et comme la ligne k contient les mots u et $\overline{u}$ de taille 2^k concaténés successivement, si par exemple p_m est préfixe de p_n alors on aura $p_n = u\overline{p_m}$ ou $p_n = \overline{up_m}$.

Donc p_n et p_m finissent par des lettres conjuguées. Quitte à échanger leur rôles, on peut supposer que p_n se termine par un 0 et p_m par un 1.

Pour remplir les mots $u'(k)_m$ il faut les mots u qui terminent par un 0 par la conjecture 7.4.11. Comme la ligne k de $T(n)$ se termine par un 0, on peut les concaténer, et la ligne $T(n)$ contient la concaténation de u autant de fois que nécessaire pour remplir la ligne $k+1$. Comme la ligne k de $T(m)$ se termine par un 1, alors par le lemme 7.3.8, il faudra le mot $\overline{u}$ autant de fois que nécessaire pour remplir la ligne.

Et donc les lignes $d(m,n) + 1 = k + 1$ de $T(m)$ et $T(n)$ sont bien conjuguées l'une de l'autre, à taille près.

□

Exemple 7.4.19. Dans l'exemple 7.4.17, la $d(46,34) + 1 = 5$ième ligne du tableau $T(34)$ contient le mot

$$01110010011100100111001001110010 01$$

dont le conjugué

$$10001101100011011000110110001101 10$$

est bien un préfixe de la 5ème ligne de $T(46)$.

8

Conclusions et perspectives

8.1 Conclusion

Ce mémoire de thèse s'est attaché à développer différents points de vue pour l'étude des canons modulo p. Cette nouvelle façon de paver la ligne était jusqu'à récemment considérée comme un exemple particulier, et rarement abordée plus en détails qu'une simple présentation du concept. Cependant, le pavage modulo 2 offre de nombreux avantages pour leur étude. Le fait que tout canon puisse être compact permet d'appréhender la complétion du pavage de manière directe, sans avoir besoin de back tracking comme pour les canons mosaïques, et ainsi de construire des algorithmes de pavage très intuitifs qui sont pourtant extrêmement rapides. Paver modulo 2 permet aussi de créer des motifs complexes ayant de nombreuses régularités internes. Ces similitudes parfois fractales, ou encore les propriétés sur la répartition des 1 et des 0, ont parfois une apparence magique, comme de nombreux résultats de théorie des nombres, mais expriment surtout la richesse de ce sujet et les années d'amusement qui nous sont promises pour son exploration.

Plusieurs pistes sont explorées dans ce mémoire. Le chapitre 4 justifie l'intérêt de l'étude des canons modulo p. Quand les canons mosaïques sont présentés comme un problème difficile, les approcher modulo p est une alternative pour les étudier. En effet, les problèmes de savoir si un motif a la qualité de paver de manière classique sont pour le moment considérés dans NP. Si le manque de résultats à ce jour ne nous permet pas d'affirmer leur NP-complétude, il nous limite aussi pour leur exploration. Durant les dix dernières années, la méthode la plus efficace pour trouver les canons de Vuza a été d'améliorer des détails pour accélérer légèrement l'attaque en force brute. Des semaines de calculs ont permis d'établir des listes de canons mosaïques non périodiques pour les plus petites périodes, mais il n'existe toujours pas de solution pour obtenir les canons de Vuza d'une grande période donnée. L'étude des canons mosaïques s'est avérée être une science de conjectures qui se révèlent souvent fausses. C'est pourquoi l'existence d'un algorithme glouton qui permet de trouver l'équivalent des canons de Vuza modulo 2 apporte un grand espoir. Les générer facilement rend leur étude agréable, et nous espérons que les informations obtenues sur ces canons modulo 2 pourront nous permettre de répondre à certaines conjectures sur les ca-

nons classiques. Le chapitre 5 est rédigé dans ce but. Comment d'un canon modulo 2 pouvons nous revenir à un canon mosaïque ? Nous sommes encore loin d'avoir la réponse, mais ce chapitre en pose la première pierre. En effet il s'intéresse à trouver un lien entre motif rythmique et la différence entre couverture et pavage du canon compact modulo 2 qu'il permet de créer. Les superpositions de groupes de 2 *onsets* au dessus du pavage créant une couverture forment la différence principale avec le pavage mosaïque. Comprendre quand tombent ces donsets et en quelle quantité est la première étape dans l'optique de les "supprimer". En effet seule une maîtrise du lien entre motif rythmique et donsets générés permettrait d'utiliser les canons modulo 2 pour obtenir des informations sur les canons mosaïques. Le chapitre s'interroge donc sur l'unicité sous certaines transformations des motifs rythmiques générant des multiensembles de donsets donnés.

Si nous sommes encore loin d'espérer un retour des canons modulo p au canons mosaïques, certains résultats sur les canons modulo 2 nous permettent déjà de comprendre leur régularité et d'accélérer les algorithmes de pavage. En effet, si trouver le motif des entrées pour un pavage mosaïque se fait en temps exponentiel, libérer la contrainte sur le nombre d'*onsets* par pulsation permet de le calculer en temps linéaire. Nous pouvons imaginer que si nous parvenons à revenir à la contrainte d'avoir exactement un *onset* par pulsation, l'algorithme de complétion redeviendrait exponentiel. Mais des résultats de récurrences permettent d'obtenir le motif des entrées modulo 2 en temps logarithmique. Ainsi l'espoir est que la combinaison d'un algorithme de retour, et d'une accélération de pavage modulo 2 offre une amélioration sur la vitesse de complétion des pavages mosaïques. Ces résultats récents permettent aussi d'obtenir la taille précise et le nombre de donsets pour le pavage modulo 2 de certaines familles de motifs rythmiques. Ces résultats de cardinalité seront utiles de deux manière dans l'espoir d'un retour au pavage mosaïque. Premièrement la connaissance du nombre de donsets obtenus par un motif rythmique sera nécessaire pour revenir aux canons mosaïques. La taille, ou période, d'un pavage -tant modulo p que mosaïque- obtenu avec un motif est pour le moment impossible à prévoir tant que le pavage complet n'est pas déterminé. Mis à part ces familles de motifs modulo 2, aucune relation directe entre motif rythmique qui peut paver et période minimale de son canon rythmique n'a été mise en valeur, et le retour au pavage mosaïque pourrait nous apprendre beaucoup sur ce lien. En effet, lorsque nous avons un motif rythmique avec lequel nous cherchons à paver de manière classique, connaître sa période donnerait directement la taille de son motif des entrées, faisant gagner un temps précieux aux algorithmes de complétion. C'est le chapitre 7 qui présente ces récurrences et cardinalités. Le motif des entrées modulo 2 est complètement construc-

tible en temps logarithmique pour les familles de motifs de la forme $\{0,1,2^k\}$ ou $\{0,1,2^k+1\}$, pour $k \in \mathbb{N}^*$, en utilisant un modèle de tableau. La structure des tableaux modélisant de la même manière les motifs des entrées pour les familles de motifs de la forme $\{0,1,n\}$, pour $n \geq 2$, présente un grand intérêt dans l'étude du pavage modulo 2. Les lignes sont composées d'un sous-mot concaténé jusqu'à complétion de la ligne. Ainsi, pour chacun de ces tableaux, la première ligne est composée de la concaténation d'un mot de taille 2, puis les deux lignes suivantes de la concaténation d'un mot de taille 4, et les quatre lignes suivantes d'un mot de taille 8, ect... Résultat aussi étonnant que ces répétitions, ces tableaux ont un nombre, dépendant seulement de la différence entre la valeur du dernier *onset* des motifs rythmiques, de leur premières lignes qui sont égales. Ces nombreux liens entre motifs des entrées associés aux motifs de la forme $\{0,1,n\}$ laissent espérer que nous pourrons un jour lier la construction par récurrence des tableaux avec $n = 2^k(+1)$ au cas plus général. Cela montre encore la richesse de l'étude du pavage modulo 2. Si plusieurs pistes sont abordées dans cette thèse, il reste encore de nombreux chemins à explorer pour comprendre plus en profondeur les canons modulo 2. Nous allons présenter ci-après quelques idées que l'auteure de cette thèse aurait aimé creuser plus en profondeur lors de sa thèse, et qui pourront être utiles à tous les mathématiciens souhaitant poursuivre son travail, y compris elle même. Elles touchent principalement aux tableaux des mots des entrées comme définis au chapitre 7, et le retour de pavage modulo p à pavage mosaïque.

8.2 Perspectives sur les tableaux du chapitre 7

Nous avons pu montrer la similitude entre les tableaux $T(2^k)$ et $T(2^k+1)$ pour tout $k \in \mathbb{N}^*$. Ils sont à de légères transformations près les symétriques par rapport à l'axe vertical l'un de l'autre. Et nous avons pu remarquer des ressemblances remarquables entre le miroir du tableau $T(2n+1)$ et le tableau $T(2n)$. Nous pensons qu'il serait possible de trouver une transformation permettant de déduire le motif $b(2n+1)$ à partir du motif $b(2n)$. Ainsi, nous pourrions limiter notre étude de pavage modulo 2 aux motifs de la forme $A(2n)$.

Lorsque nous utilisons la table du XNOR

	1	0
1	1	0
0	0	1

et que nous regardons les mots, de taille $\mathcal{G}(n)$ pour chaque ligne n, par lesquels il faut multiplier lettres à lettres les préfixes de $T(2)$ pour arriver à ceux de $T(4)$, puis ceux de $T(4)$ pour arriver à $T(6)$, ect... on

trouve des tableaux $T(2-4)$,$T(6-8)$, ect... ressemblant fortement aux tableaux $T(2^k)$, quitte à concaténer les préfixes pour arriver au nombre de colonnes voulues. De même si nous répétons le processus pour passer des tableaux $T(2-4)$ à $T(6-8)$, ect... Ces similitudes qui ne sont pour le moment que des observations, laissent penser que les tableaux de la forme $T(2n)$ pour $n \in \mathbb{N}^*$ sont en fait générables de manière automatique par un tout petit jeu de règles sur les mots de l'alphabet $\{0,1\}$.

Nous espérons pouvoir trouver ces règles, pour ainsi comprendre les transformations qui permettent de passer des tableaux $T(2^k)$ aux tableaux $T(2n)$ quelconque en temps assez rapide. Ainsi, avec ce qui a été dit plus tôt, nous espérons combiner les résultats des théorèmes du chapitre 7 avec ces transformations pour obtenir les pavages pour toute la famille des motifs de la forme $A(n)$ en temps moins que linéaire.

À la suite, nous présumons qu'une même logique puisse s'appliquer pour toute la famille des motifs de la forme

$$A(k,n) = \{0,1,2,\ldots,k,n\}$$

pour $k,n \in \mathbb{N}$, $2 \leq k < n$.

8.3 Sur les mots $b(n)$

Comme nous l'avons remarqué sur la Figure 7.6, la répartition des 1 et des 0 des mots $b(n)$ permet de créer des motifs complexes à un niveau macroscopique. Une représentation visuelle de ces mots dans les tableaux $T(n)$ en coloriant les 1 ou 0 par des carrés noirs ou blancs évoque les dessins tracés par les règles 0 à 255 générés par des automates cellulaires en une dimension (WOLFRAM, 1983). Par exemple, pour $n = 2^k$, le mot $b(n)$ trace une partie du motif obtenu avec la règle 90. La différence principale avec ces motifs lorsque $n \neq 2^k$ est le changement brutale de motif par blocs, comme on peut le voir en annexe. Nous soupçonnons que ces changements viennent du nombre de 1 dans l'écriture en base 2 de n. Il serait intéressant de voir si la répartition des 1 dans n écrit en base 2 peut être associée à un jeu de règles dont l'automate dessinerait $b(n)$.

Plus simplement, il reste à prouver la conjecture 7.4.11 précisant la parité du nombre de 1 sur les lignes du tableau $T(n)$ pour que s'en déduisent les derniers résultats du dernier chapitre.

8.4 Sur le retour des canons modulo p aux canons mosaïques

Nous l'avons vu, si nous souhaitons pouvoir utiliser les résultats sur les canons modulo p pour obtenir des informations sur le pa-

vage mosaïque, il faut commencer par maîtriser le lien entre motif rythmique et donsets générés. Cela passera comme expliqué dans le chapitre 5 par la recherche de toutes les transformations qui associent deux canons distincts ayant le même multiset de *donsets*. La difficulté ne viendra pas tant de trouver des transformations que de s'assurer que nous les ayons toutes. Une autre vision de ce problème serait de s'intéresser s'il est possible de savoir quels sont les multisets pouvant représenter un multiset de donsets. Par exemple, il est évident avec nos notations que 0 ne pourra jamais être un élément d'un tel multiset (à moins que de façon triviale on n'empile $p+1$ fois le motif au temps 0). Même si nous ne savons pas quels motifs permettent d'obtenir un tel multiset de donset, savoir lesquels sont générables nous apporterait probablement des idées de solutions pour un retour aux canons mosaïques.

Un autre problème peu abordé dans cette thèse est celui que tous nos résultats se font modulo 2 et donc concernent des canons compacts. Si nous définissons les canons de Vuza modulo 2, de la même manière que les canons de Vuza classiques, comme ceux étant minimaux par concaténation, il est très facile de les obtenir. En effet, l'algorithme 1 qui est linéaire a la particularité de produire des motifs des entrées minimaux par concaténation ; et un second passage dans l'algorithme de ce motif des entrées permet d'avoir un motif des *onsets* lui même minimal par concaténation. C'est-à-dire que les canons de Vuza modulo 2 s'obtiennent en temps linéaire, contrairement aux Vuza mosaïques qui s'obtiennent en temps exponentiel. Mais s'intéresser aux doublons n'est pas suffisant ici. En effet, les canons de Vuza mosaïques ne sont jamais compacts, tandis que l'algorithme 1 ne produit *que* des canons compacts. Cela n'a pas encore été exploré, mais il serait très intéressant de s'attacher à la construction de canon modulo 2 non compact minimaux par concaténation. La question est de savoir s'il existe de tels canons étant strictement modulo 2 et non mosaïques, et sans superposition triviale du motif sur la même pulsation. Nous n'avons pour le moment pas d'exemple d'un motif rythmique qui produirait un canons strictement modulo 2 plus petit que son canon compact minimal. Savoir s'ils existent ou non changerait complètement la notion de canon de Vuza modulo p.

8.5 Sur des pistes variées

La plupart de nos résultats sont dans le cas où $p = 2$ car la facilité de paver de manière compacte nous a permis une plus grande familiarisation avec ces canons. En effet, lorsque $p \neq 2$ le pavage modulo p n'a plus de raison d'être compact (mais peut l'être bien entendu), cependant trouver le motif des entrées reste plus simple que dans le cas mosaïque. Cela revient à effectuer une division dans

$\mathbb{F}_p[X]$ après avoir utilisé un algorithme en temps linéaire qui permet de trouver un multiple de la plus petite période possible. Puis, une simplification par déconcaténation et quelques transformations sur le polynôme quotient obtenu nous donnent le pavage modulo p de ce motif. Cela est bien entendu plus compliqué que pour le pavage modulo 2, et la perte de compacité rajoute une difficulté aux preuves. Cependant, il serait intéressant de voir si d'autres familles de motifs rythmiques permettent d'obtenir des résultats par récurrence sur les motifs des entrées. On pense bien évidemment aux familles $\{0, 1, p^k\}$ pour $k \in \mathbb{N}^*$ modulo p.

Une piste à explorer serait d'appliquer les outils souvent utilisés sur les canons mosaïques pour étudier les canons modulo p. La transformée de Fourier discrète est au cœur d'un livre sur les mathémusiques en cours de publication (AMIOT, 2016) dont un chapitre entier est consacré aux canons rythmiques tant le nombre de résultats qu'elle a permis d'obtenir est conséquent. Les canons quasi-périodiques ont aussi été étudiés, ils sont définis comme les pavages en plus grande dimension modélisant les quasi-cristaux. Il n'y a pour le moment pas de définition de ce que pourrait être un pavage modulo p quasi-périodique, mais il serait intéressant de créer ce parallèle.

En annexe est présentée une liste des canons modulo 2 pour les premières périodes possibles. On remarque par exemple que pour les périodes $N = 2^k$ pour $k \in \mathbb{N}^*$ nous n'obtenons pas de canon modulo 2 strict, et donc pas de canon de Vuza modulo 2. Aussi, la suite des périodes N suivant A071642 de l'OEIS ne peut bien sûr admettre que le canon trivial modulo 2, ce qui est évident si on utilise la notation polynomiale du pavage. Tout un travail de combinatoire reste à faire pour savoir combien de canons, de Vuza ou non, modulo 2 il existe pour chaque période N.

Les polynômes cyclotomiques de $\mathbb{Z}[X]$ jouent un rôle fondamental dans l'étude des canons mosaïques. Ils sont par exemple à la base des conditions de Coven-Meyerowitz qui sont à ce jour le meilleur espoir de CNS pour savoir si un motif rythmique peut paver ou non. Aucun travail n'a pour le moment été fait sur le lien entre motif pavant modulo 2 une certaine période, et les polynômes cyclotomiques de $\mathbb{F}_2[X]$ qui divisent son polynôme caractéristique.

Finalement, pour obtenir leur place comme outil classique de mathémusique, il ne manque aux canons modulo p que de trouver un compositeur qui les sacrerait dans une de ses pièces. Leur étude est encore très récente, et ni mathématiciens ni musiciens n'ont pour le moment eu l'occasion de se plonger dans le monde merveilleux du pavage modulo 2. Nul doute que de nombreux résultats mathématiques et autres applications musicales n'attendent que leur popularisation.

9 Annexes

9.1 Liste des canons compacts modulo 2 de $\mathbb{Z}_N$ pour N de 2 à 32

Pour N allant de 2 à 32, sont présentés ci-dessous tous les canons (A, B) compacts modulo 2 de $\mathbb{Z}_N$. Les ensembles A et B sont notés sous forme de liste, ainsi que leur cardinal, et la liste D représente le multi-ensemble de donsets $D_{(A,B),2}$.

Les canons qui ne sont pas de Vuza sont en jaune. C'est à dire les canons qui ne sont **pas** tels que l'algorithme 1 renvoie B lorsqu'il prend A en entrée et réciproquement. En particulier, tous les canons n'ayant pas de donsets $D = []$ ne seront pas de Vuza.

Les canons issus de la transformation du théorème 5.2.1 sont en bleu, et le canon dont ils sont issus est noté.

Les canons issus de la transformation du théorème 5.2.3 (ainsi que ceux ayant des donsets identiques et semblant venir d'une transformation de même type) sont en rouge.

N=2

A =[0, 1] de taille 2 B =[0] de taille 1 D =[] de taille 0

N = 3

A =[0, 1, 2] de taille 3 B =[0] de taille 1 D =[] de taille 0

N = 4

A =[0, 1] de taille 2 B =[0, 2] de taille 2 D =[] de taille 0
A =[0, 1, 2, 3] de taille 4 B =[0] de taille 1 D =[] de taille 0

N = 5

A =[0, 1, 2, 3, 4] de taille 5 B =[0] de taille 1 D =[] de taille 0

N =6

A =[0, 3] de taille 2 B =[0, 1, 2] de taille 3 D =[] de taille 0
A =[0, 2, 4] de taille 3 B =[0, 1] de taille 2 D =[] de taille 0
A =[0, 1, 2, 3, 4, 5] de taille 6 B =[0] de taille 1 D =[] de taille 0

N =7

A =[0, 1, 3] de taille 3 B =[0, 2, 3] de taille 3 D =[3] de taille 1
A =[0, 1, 2, 3, 4, 5, 6] de taille 7 B =[0] de taille 1 D =[] de taille 0

N =8

A =[0, 4] de taille 2 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0
A =[0, 1, 4, 5] de taille 4 B =[0, 2] de taille 2 D =[] de taille 0
A =[0, 2, 4, 6] de taille 4 B =[0, 1] de taille 2 D =[] de taille 0
A =[0, 1, 2, 3, 4, 5, 6, 7] de taille 8 B =[0] de taille 1 D =[] de taille 0

N = 9
A =[0, 3, 6] de taille 3 B =[0, 1, 2] de taille 3 D =[] de taille 0
A =[0, 1, 2, 3, 4, 5, 6, 7, 8] de taille 9 B =[0] de taille 1 D =[] de taille 0

N =10
A =[0, 5] de taille 2 B =[0, 1, 2, 3, 4] de taille 5 D =[] de taille 0
A =[0, 2, 4, 6, 8] de taille 5 B =[0, 1] de taille 2 D =[] de taille 0
A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9] de taille 10 B =[0] de taille 1 D =[] de taille 0

N =11
A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10] de taille 11 B =[0] de taille 1 D =[] de taille 0

N =12
A =[0, 1, 3, 4] de taille 4 B =[0, 2, 3, 4, 5, 7] de taille 6 D =[3, 4, 5, 6, 7, 8] de taille 6
A =[0, 2, 3, 5] de taille 4 B =[0, 1, 3, 5, 6] de taille 5 D =[3, 5, 6, 8] de taille 4
A =[0, 6] de taille 2 B =[0, 1, 2, 3, 4, 5] de taille 6 D =[] de taille 0
A =[0, 1, 6, 7] de taille 4 B =[0, 2, 4] de taille 3 D =[] de taille 0
A =[0, 4, 8] de taille 3 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0
A =[0, 1, 2, 6, 7, 8] de taille 6 B =[0, 3] de taille 2 D =[] de taille 0
A =[0, 3, 6, 9] de taille 4 B =[0, 1, 2] de taille 3 D =[] de taille 0
A =[0, 1, 4, 5, 8, 9] de taille 6 B =[0, 2] de taille 2 D =[] de taille 0
A =[0, 2, 4, 6, 8, 10] de taille 6 B =[0, 1] de taille 2 D =[] de taille 0
A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11] de taille 12 B =[0] de taille 1 D =[] de taille 0

N =13
A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12] de taille 13 B =[0] de taille 1 D =[] de taille 0

N =14
A =[0, 1, 2, 4] de taille 4 B =[0, 3, 4, 5, 6, 8, 9] de taille 7 D =[4, 5, 6, 7, 8, 9, 10] de taille 7
A =[0, 2, 3, 4] de taille 4 B =[0, 1, 3, 4, 5, 6, 9] de taille 7 D =[3, 4, 5, 6, 7, 8, 9] de taille 7
A =[0, 2, 6] de taille 3 B =[0, 1, 4, 5, 6, 7] de taille 6 D =[6, 7] de taille 2

A =[0, 4, 6] de taille 3 B =[0, 1, 2, 3, 6, 7] de taille 6 D =[6, 7] de taille 2

Zoom x2 de A = [0,2,3] B = [0,1,3] D = [3]

A =[0, 1, 3, 7, 8, 10] de taille 6 B =[0, 2, 3] de taille 3 D =[3, 10] de taille 2

A =[0, 2, 3, 7, 9, 10] de taille 6 B =[0, 1, 3] de taille 3 D =[3, 10] de taille 2

A =[0, 2, 4, 6, 8, 10, 12] de taille 7 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 7] de taille 2 B =[0, 1, 2, 3, 4, 5, 6] de taille 7 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] de taille 14 B =[0] de taille 1 D =[] de taille 0

N =15

A =[0, 2, 3, 4, 6] de taille 5 B =[0, 1, 3, 4, 5, 7, 8] de taille 7 D =[3, 4, 5, 6, 7, 7, 8, 9, 10, 11] de taille 10

A =[0, 1, 2, 3, 6] de taille 5 B =[0, 4, 6, 7, 8] de taille 5 D =[6, 7, 8, 9, 10] de taille 5

A =[0, 3, 4, 5, 6] de taille 5 B =[0, 1, 2, 4, 8] de taille 5 D =[4, 5, 6, 7, 8] de taille 5

A =[0, 1, 4] de taille 3 B =[0, 2, 5, 6, 8, 9, 10] de taille 7 D =[6, 9, 10] de taille 3

A =[0, 3, 4] de taille 3 B =[0, 1, 2, 4, 5, 8, 10] de taille 7 D =[4, 5, 8] de taille 3

A =[0, 5, 10] de taille 3 B =[0, 1, 2, 3, 4] de taille 5 D =[] de taille 0

A =[0, 3, 6, 9, 12] de taille 5 B =[0, 1, 2] de taille 3 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14] de taille 15 B =[0] de taille 1 D =[] de taille 0

N =16

A =[0, 8] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7] de taille 8 D =[] de taille 0

A =[0, 1, 8, 9] de taille 4 B =[0, 2, 4, 6] de taille 4 D =[] de taille 0

A =[0, 2, 8, 10] de taille 4 B =[0, 1, 4, 5] de taille 4 D =[] de taille 0

A =[0, 1, 2, 3, 8, 9, 10, 11] de taille 8 B =[0, 4] de taille 2 D =[] de taille 0

A =[0, 4, 8, 12] de taille 4 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0

A =[0, 1, 4, 5, 8, 9, 12, 13] de taille 8 B =[0, 2] de taille 2 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14] de taille 8 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] de taille 16 B =[0] de taille 1 D =[] de taille 0

N =17

A =[0, 3, 4, 5, 8] de taille 5 B =[0, 1, 2, 4, 6, 7, 8] de taille 7 D =[4, 5, 6, 7, 8, 9, 10, 11, 12] de taille 9

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16] de taille 17 B =[0] de taille 1 D =[] de taille 0

N =18

A =[0, 1, 3, 4, 6, 7] de taille 6 B =[0, 2, 3, 4, 5, 6, 7, 8, 10] de taille 9 D =[3, 4, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 13, 14] de taille 18

A =[0, 9] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8] de taille 9 D =[] de taille 0

A =[0, 1, 2, 9, 10, 11] de taille 6 B =[0, 3, 6] de taille 3 D =[] detaille 0

A =[0, 6, 12] de taille 3 B =[0, 1, 2, 3, 4, 5] de taille 6 D =[] de taille 0

A =[0, 1, 6, 7, 12, 13] de taille 6 B =[0, 2, 4] de taille 3 D =[] de taille 0

A =[0, 1, 2, 6, 7, 8, 12, 13, 14] de taille 9 B =[0, 3] de taille 2 D =[] de taille 0

A =[0, 3, 6, 9, 12, 15] de taille 6 B =[0, 1, 2] de taille 3 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16] de taille 9 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17] de taille 18 B =[0] de taille 1 D =[] de taille 0

N =19

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18] de taille 19 B =[0] de taille 1 D =[] de taille 0

N =20

A =[0, 1, 5, 6] de taille 4 B =[0, 2, 4, 5, 6, 7, 8, 9, 11, 13] de taille 10 D =[5, 6, 7, 8, 9, 10, 11, 12, 13, 14] de taille 10

A =[0, 2, 5, 7] de taille 4 B =[0, 1, 4, 6, 8, 11, 12] de taille 7 D =[6, 8, 11, 13] de taille 4

A =[0, 10] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9] de taille 10 D =[] de taille 0

A =[0, 1, 10, 11] de taille 4 B =[0, 2, 4, 6, 8] de taille 5 D =[] de taille 0

A =[0, 1, 2, 3, 4, 10, 11, 12, 13, 14] de taille 10 B =[0, 5] de taille 2 D =[] de taille 0

A =[0, 5, 10, 15] de taille 4 B =[0, 1, 2, 3, 4] de taille 5 D =[] de taille 0

A =[0, 4, 8, 12, 16] de taille 5 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0

A =[0, 1, 4, 5, 8, 9, 12, 13, 16, 17] de taille 10 B =[0, 2] de taille 2 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16, 18] de taille 10 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] de taille 20 B =[0] de taille 1 D =[] de taille 0

N =21

A =[0, 2, 3, 4, 5, 6, 8] de taille 7 B =[0, 1, 3, 4, 6, 8, 9, 11, 12] de taille 9 D =[3, 4, 5, 6, 6, 7, 8, 8, 9, 9, 10, 11, 11, 12, 12, 13, 14, 14, 15, 16, 17] de taille 21

A =[0, 1, 2, 4, 6] de taille 5 B =[0, 3, 4, 5, 7, 8, 9, 13, 14] de taille 9 D =[4, 5, 6, 7, 8, 9, 9, 10, 11, 13, 14, 15] de taille 12

A =[0, 2, 4, 5, 6] de taille 5 B =[0, 1, 5, 6, 7, 9, 10, 11, 14] de taille 9 D =[5, 6, 7, 9, 10, 11, 11, 12, 13, 14, 15, 16] de taille 12

A =[0, 1, 3, 6, 8] de taille 5 B =[0, 2, 3, 6, 8, 11, 12] de taille 7 D =[3, 6, 8, 9, 11, 12, 14] de taille 7

A =[0, 2, 5, 7, 8] de taille 5 B =[0, 1, 4, 6, 9, 10, 12] de taille 7 D =[6, 8, 9, 11, 12, 14, 17] de taille 7

A =[0, 1, 2, 4, 5, 8, 9] de taille 7 B =[0, 3, 4, 9, 11] de taille 5 D =[4, 5, 8, 9, 11, 12, 13] de taille 7

A =[0, 1, 4, 5, 7, 8, 9] de taille 7 B =[0, 2, 7, 8, 11] de taille 5 D =[7, 8, 9, 11, 12, 15, 16] de taille 7

A =[0, 1, 5] de taille 3 B =[0, 2, 4, 5, 8, 9, 10, 12, 13, 14, 15] de taille 11 D =[5, 9, 10, 13, 14, 15] de taille 6

A =[0, 4, 5] de taille 3 B =[0, 1, 2, 3, 5, 6, 7, 10, 11, 13, 15] de taille 11 D =[5, 6, 7, 10, 11, 15] de taille 6

A =[0, 3, 9] de taille 3 B =[0, 1, 2, 6, 7, 8, 9, 10, 11] de taille 9 D =[9, 10, 11] de taille 3

A =[0, 6, 9] de taille 3 B =[0, 1, 2, 3, 4, 5, 9, 10, 11] de taille 9 D =[9, 10, 11] de taille 3

Zoom x3 de A = [0,1,3] B = [0,2,3] D = [3]

A =[0, 1, 3, 7, 8, 10, 14, 15, 17] de taille 9 B =[0, 2, 3] de taille 3 D =[3, 10, 17] de taille 3

A =[0, 2, 3, 7, 9, 10, 14, 16, 17] de taille 9 B =[0, 1, 3] de taille 3 D =[3, 10, 17] de taille 3

A =[0, 7, 14] de taille 3 B =[0, 1, 2, 3, 4, 5, 6] de taille 7 D =[] detaille 0

A =[0, 3, 6, 9, 12, 15, 18] de taille 7 B =[0, 1, 2] de taille 3 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20] de taille 21 B =[0] de taille 1 D =[] de taille 0

N =22

A =[0, 11] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10] de taille 11 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20] de taille 11 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21] de taille 22 B =[0] de taille 1 D =[] de taille 0

N =23

A =[0, 1, 5, 6, 7, 9, 11] de taille 7 B =[0, 2, 4, 5, 6, 10, 11] de taille 7 D =[5, 6, 7, 9, 10, 11, 11, 11, 12, 13, 15, 16, 17] de taille 13

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22] de taille 23 B =[0] de taille 1 D =[] de taille 0

N =24

A =[0, 1, 3, 4, 6, 7, 9, 10] de taille 8 B =[0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13] de taille 12 D =[3, 4, 5, 6, 6, 7, 7, 8, 8, 9, 9, 9, 10, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 16, 16, 17, 17, 18, 19, 20] de taille 36

A =[0, 2, 3, 4, 5, 6, 7, 9] de taille 8 B =[0, 1, 3, 4, 6, 7, 8, 10, 11, 13, 14] de taille 11 D =[3, 4, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 10, 11, 11, 12, 12, 13, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 19, 20] de taille 32

A =[0, 1, 2, 4, 5, 6] de taille 6 B =[0, 3, 4, 6, 7, 8, 9, 10, 11, 13, 14, 17] de taille 12 D =[4, 5, 6, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 17, 18, 19] de taille 24

A =[0, 1, 2, 4, 5, 6, 8, 9, 10] de taille 9 B =[0, 3, 4, 6, 7, 9, 10, 13] de taille 8 D =[4, 5, 6, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 17, 18, 19] de taille 24

A =[0, 2, 3, 5, 6, 8, 9, 11] de taille 8 B =[0, 1, 3, 5, 7, 9, 11, 12] de taille 8 D =[3, 5, 6, 7, 8, 9, 9, 10, 11, 11, 12, 12, 13, 14, 14, 15, 16, 17, 18, 20] de taille 20

A =[0, 1, 3, 5, 7, 8] de taille 6 B =[0, 2, 3, 5, 6, 9, 10, 12, 13, 15] de taille 10 D =[3, 5, 6, 7, 8, 9, 10, 10, 11, 12, 13, 13, 14, 15, 16, 17, 18, 20] de taille 18

A =[0, 3, 4, 7, 8, 11] de taille 6 B =[0, 1, 2, 4, 5, 7, 8, 10, 11, 12] de taille 10 D =[4, 5, 7, 8, 8, 9, 10, 11, 11, 12, 12, 13, 14, 15, 15, 16, 18, 19] de taille 18

A =[0, 3, 4, 7] de taille 4 B =[0, 1, 2, 4, 5, 7, 9, 11, 12, 14, 15, 16] de taille 12 D =[4, 5, 7, 8, 9, 11, 12, 14, 15, 16, 18, 19] de taille 12

A =[0, 2, 6, 8] de taille 4 B =[0, 1, 4, 5, 6, 7, 8, 9, 10, 11, 14, 15] de taille 12 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17] de taille 12

A =[0, 1, 2, 3, 6, 7, 8, 9] de taille 8 B =[0, 4, 6, 8, 10, 14] de taille 6 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17] de taille 12]

Zoom x2 de A = [0,1,3,4] B = [0,2,3,4,5,7] D = [3,4,5,6,7,8]

A =[0, 2, 3, 5, 12, 14, 15, 17] de taille 8 B =[0, 1, 3, 5, 6] de taille 5 D =[3, 5, 6, 8, 15, 17, 18, 20] de taille 8

A =[0, 4, 6, 10] de taille 4 B =[0, 1, 2, 3, 6, 7, 10, 11, 12, 13] de taille 10 D =[6, 7, 10, 11, 12, 13, 16, 17] de taille 8

A =[0, 1, 4, 5, 6, 7, 10, 11] de taille 8 B =[0, 2, 6, 10, 12] de taille 5 D =[6, 7, 10, 11, 12, 13, 16, 17] de taille 8

Zoom x2 de A = [0,2,3,5] B = [0,1,3,5,6] D =[]

A =[0, 1, 3, 5, 6, 12, 13, 15, 17, 18] de taille 10 B =[0, 2, 3, 5] de taille 4 D =[3, 5, 6, 8, 15, 17, 18, 20] de taille 8

A =[0, 1, 3, 4, 12, 13, 15, 16] de taille 8 B =[0, 2, 3, 4, 5, 7] de taille 6 D =[3, 4, 5, 6, 7, 8, 15, 16, 17, 18, 19, 20] de taille 12

A =[0, 2, 3, 4, 5, 7, 12, 14, 15, 16, 17, 19] de taille 12 B =[0, 1, 3, 4] de taille 4 D =[3, 4, 5, 6, 7, 8, 15, 16, 17, 18, 19, 20] de taille 12

A =[0, 2, 12, 14] de taille 4 B =[0, 1, 4, 5, 8, 9] de taille 6 D =[] de taille 0

A =[0, 1, 2, 12, 13, 14] de taille 6 B =[0, 3, 6, 9] de taille 4 D =[] de taille 0

A =[0, 3, 12, 15] de taille 4 B =[0, 1, 2, 6, 7, 8] de taille 6 D =[] de taille 0

A =[0, 1, 8, 9, 16, 17] de taille 6 B =[0, 2, 4, 6] de taille 4 D =[] de taille 0

A =[0, 1, 12, 13] de taille 4 B =[0, 2, 4, 6, 8, 10] de taille 6 D =[] de taille 0

A =[0, 6, 12, 18] de taille 4 B =[0, 1, 2, 3, 4, 5] de taille 6 D =[] de taille 0

A =[0, 2, 8, 10, 16, 18] de taille 6 B =[0, 1, 4, 5] de taille 4 D =[] de taille 0

A =[0, 4, 8, 12, 16, 20] de taille 6 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0

A =[0, 2, 4, 12, 14, 16] de taille 6 B =[0, 1, 6, 7] de taille 4 D =[] de taille 0

A =[0, 1, 2, 3, 12, 13, 14, 15] de taille 8 B =[0, 4, 8] de taille 3 D =[] de taille 0

A =[0, 8, 16] de taille 3 B =[0, 1, 2, 3, 4, 5, 6, 7] de taille 8 D =[] de taille 0

A =[0, 1, 6, 7, 12, 13, 18, 19] de taille 8 B =[0, 2, 4] de taille 3 D =[] de taille 0

A =[0, 3, 6, 9, 12, 15, 18, 21] de taille 8 B =[0, 1, 2] de taille 3 D =[] de taille 0

A =[0, 12] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11] de taille 12 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 12, 13, 14, 15, 16, 17] de taille 12 B =[0, 6] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19] de taille 12 B =[0, 4] de taille 2 D =[] de taille 0

A =[0, 1, 2, 6, 7, 8, 12, 13, 14, 18, 19, 20] de taille 12 B =[0, 3] de taille 2 D =[] de taille 0

A =[0, 1, 4, 5, 8, 9, 12, 13, 16, 17, 20, 21] de taille 12 B =[0, 2] de taille 2 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22] de taille 12 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23] de taille 24 B =[0] de taille 1 D =[] de taille 0

N =25

A =[0, 5, 10, 15, 20] de taille 5 B =[0, 1, 2, 3, 4] de taille 5 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24] de taille 25 B =[0] de taille 1 D =[] de taille 0

N =26

A =[0, 13] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12] de taille 13 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24] de taille 13 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25] de taille 26 B =[0] de taille 1 D =[] de taille 0

N =27

A =[0, 9, 18] de taille 3 B =[0, 1, 2, 3, 4, 5, 6, 7, 8] de taille 9 D =[] de taille 0

A =[0, 1, 2, 9, 10, 11, 18, 19, 20] de taille 9 B =[0, 3, 6] de taille 3 D =[] de taille 0

A =[0, 3, 6, 9, 12, 15, 18, 21, 24] de taille 9 B =[0, 1, 2] de taille 3 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26] de taille 27 B =[0] de taille 1 D =[] de taille 0

N =28

A =[0, 1, 2, 4, 7, 8, 9, 11] de taille 8 B =[0, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16] de taille 14 D =[4, 5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 15, 16, 16, 16, 17, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 23, 24] de taille 42

A =[0, 2, 3, 4, 7, 9, 10, 11] de taille 8 B =[0, 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 16] de taille 14 D =[3, 4, 5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 15, 16, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 23] de taille 42

A =[0, 1, 2, 5, 6, 7, 9] de taille 7 B =[0, 3, 5, 6, 8, 11, 12, 13, 14, 15, 17, 18] de taille 12 D =[5, 6, 7, 8, 9, 10, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 23, 24] de taille 28

A =[0, 2, 3, 4, 7, 8, 9] de taille 7 B =[0, 1, 3, 4, 5, 6, 7, 10, 12, 13, 15, 18] de taille 12 D =[3, 4, 5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 17, 18, 19, 20, 21, 22] de taille 28

A =[0, 1, 3, 4, 5, 6, 8, 11] de taille 8 B =[0, 2, 3, 4, 8, 10, 11, 14, 15, 16] de taille 10 D =[3, 4, 5, 6, 7, 8, 8, 9, 10, 11, 11, 12, 13, 14, 14, 15, 15, 16, 16, 17, 18, 19, 19, 20, 21, 22] de taille 26

A =[0, 3, 5, 6, 7, 8, 10, 11] de taille 8 B =[0, 1, 2, 5, 6, 8, 12, 13, 14, 16] de taille 10 D =[5, 6, 7, 8, 8, 9, 10, 11, 11, 12, 12, 13, 13, 14, 15, 16, 16, 17, 18, 19, 19, 20, 21, 22, 23, 24] de taille 26

A =[0, 1, 2, 5, 7, 8, 9, 12] de taille 8 B =[0, 3, 5, 7, 9, 11, 13, 14, 15] de taille 9 D =[5, 7, 8, 9, 10, 11, 12, 12, 13, 14, 14, 15, 15, 16, 16, 17, 18, 19, 20, 21, 22, 23] de taille 22

A =[0, 1, 3, 4, 5, 9, 10, 12] de taille 8 B =[0, 2, 3, 4, 6, 7, 12, 14, 15] de taille 9 D =[3, 4, 5, 6, 7, 7, 8, 9, 10, 11, 12, 12, 13, 14, 15, 15, 16, 16, 17, 18, 19, 24] de taille 22

A =[0, 2, 3, 7, 8, 9, 11, 12] de taille 8 B =[0, 1, 3, 8, 9, 11, 12, 13, 15] de taille 9 D =[3, 8, 9, 10, 11, 11, 12, 12, 13, 14, 15, 15, 16, 17, 18, 19, 20, 20, 21, 22, 23, 24] de taille 22

A =[0, 3, 4, 5, 7, 10, 11, 12] de taille 8 B =[0, 1, 2, 4, 6, 8, 10, 12, 15] de taille 9 D =[4, 5, 6, 7, 8, 9, 10, 11, 11, 12, 12, 13, 13, 14, 15, 15, 16, 17, 18, 19, 20, 22] de taille 22

A =[0, 1, 2, 5, 7, 10] de taille 6 B =[0, 3, 5, 7, 8, 9, 12, 15, 16, 17] de taille 10 D =[5, 7, 8, 9, 10, 10, 12, 13, 14, 15, 16, 17, 17, 18, 19, 22] de taille 16

A =[0, 3, 5, 8, 9, 10] de taille 6 B =[0, 1, 2, 5, 8, 9, 10, 12, 14, 17] de taille 10 D =[5, 8, 9, 10, 10, 11, 12, 13, 14, 15, 17, 17, 18, 19, 20, 22] de taille 16

A =[0, 2, 6, 7, 9, 13] de taille 6 B =[0, 1, 4, 5, 6, 8, 11, 12, 13, 14] de taille 10 D =[6, 7, 8, 10, 11, 12, 13, 13, 14, 14, 15, 17, 18, 19, 20, 21] de taille 16

A =[0, 4, 6, 7, 11, 13] de taille 6 B =[0, 1, 2, 3, 6, 8, 9, 10, 13, 14] de taille 10 D =[6, 7, 8, 9, 10, 12, 13, 13, 14, 14, 15, 16, 17, 19, 20, 21] de taille 16

A =[0, 1, 2, 5] de taille 4 B =[0, 3, 5, 9, 10, 11, 12, 13, 15, 16, 18, 20, 21, 22] de taille 14 D =[5, 10, 11, 12, 13, 14, 15, 16, 17, 18, 20,21, 22, 23] de taille 14

A =[0, 3, 4, 5] de taille 4 B =[0, 1, 2, 4, 6, 7, 9, 10, 11, 12, 13, 17, 19, 22] de taille 14 D =[4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17, 22] de taille 14

A =[0, 4, 6, 8] de taille 4 B =[0, 1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 18, 19] de taille 14 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] de taille 14

A =[0, 2, 6, 8, 10, 12, 18] de taille 7 B =[0, 1, 4, 5, 6, 7, 8, 9] de taille 8 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] de taille 14

Zoom x2 de A = [0,2,3,4] B = [0,1,3,4,5,6,9] D = [3,4,5,6,7,8,9]

A =[0, 1, 7, 8] de taille 4 B =[0, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 15, 17, 19] de taille 14 D =[7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20] de taille 14

A =[0, 1, 3, 4, 5, 6, 9, 14, 15, 17, 18, 19, 20, 23] de taille 14 B =[0, 2, 3, 4] de taille 4 D =[3, 4, 5, 6, 7, 8, 9, 17, 18, 19, 20, 21, 22, 23] de taille 14

A =[0, 3, 4, 5, 6, 8, 9, 14, 17, 18, 19, 20, 22, 23] de taille 14 B =[0, 1, 2, 4] de taille 4 D =[4, 5, 6, 7, 8, 9, 10, 18, 19, 20, 21, 22, 23, 24] de taille 14

A =[0, 1, 2, 3, 6, 11, 12] de taille 7 B =[0, 4, 6, 7, 8, 11, 13, 15] de taille 8 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] de taille 14

A =[0, 1, 6, 9, 10, 11, 12] de taille 7 B =[0, 2, 4, 7, 8, 9, 11, 15] de taille 8 D =[8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21] de taille 14

A =[0, 2, 4, 8] de taille 4 B =[0, 1, 6, 7, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19] de taille 14 D =[8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21] de taille 14

A =[0, 6, 8, 10, 12, 16, 18] de taille 7 B =[0, 1, 2, 3, 4, 5, 8, 9] de taille 8 D =[8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21] de taille 14

Zoom x2 de A = [0,3,4,5,6,8,9] B = [0,1,2,4] D = [4,5,6,7,8,9,10]

A =[0, 2, 3, 4, 14, 16, 17, 18] de taille 8 B =[0, 1, 3, 4, 5, 6, 9] de taille 7 D =[3, 4, 5, 6, 7, 8, 9, 17, 18, 19, 20, 21, 22, 23] de taille 14

A =[0, 1, 2, 4, 14, 15, 16, 18] de taille 8 B =[0, 3, 4, 5, 6, 8, 9] de taille 7 D =[4, 5, 6, 7, 8, 9, 10, 18, 19, 20, 21, 22, 23, 24] de taille 14

A =[0, 1, 3, 6] de taille 4 B =[0, 2, 3, 6, 9, 11, 13, 14, 15, 16, 19, 20, 21] de taille 13 D =[3, 6, 9, 12, 14, 15, 16, 17, 19, 20, 21, 22] de taille 12

A =[0, 3, 5, 6] de taille 4 B =[0, 1, 2, 5, 6, 7, 8, 10, 12, 15, 18, 19, 21] de taille 13 D =[5, 6, 7, 8, 10, 11, 12, 13, 15, 18, 21, 24] de taille 12

A =[0, 2, 7, 9] de taille 4 B =[0, 1, 4, 5, 7, 9, 11, 13, 14, 17, 18] de taille 11 D =[7, 9, 11, 13, 14, 16, 18, 20] de taille 8

A =[0, 4, 12] de taille 3 B =[0, 1, 2, 3, 8, 9, 10, 11, 12, 13, 14, 15] de taille 12 D =[12, 13, 14, 15] de taille 4

A =[0, 8, 12] de taille 3 B =[0, 1, 2, 3, 4, 5, 6, 7, 12, 13, 14, 15] de taille 12 D =[12, 13, 14, 15] de taille 4

Zoom x4 de A = [0,1,3] B = [0,2,3] D = [3]

A =[0, 1, 4, 5, 12, 13] de taille 6 B =[0, 2, 8, 10, 12, 14] de taille 6 D =[12, 13, 14, 15] de taille 4

Zoom x2 de A = [0,2,6] B = [0,1,4,5,6,7] D = [6,7] et le second est dans le x4

A =[0, 1, 8, 9, 12, 13] de taille 6 B =[0, 2, 4, 6, 12, 14] de taille 6 D =[12, 13, 14, 15] de taille 4

Zoom x2 de A = [0,4,6] B = [0,1,2,3,6,7] D = [6,7] et le second est dans le x4

A =[0, 1, 2, 3, 6, 7, 14, 15, 16, 17, 20, 21] de taille 12 B =[0, 4, 6] de taille 3 D =[6, 7, 20, 21] de taille 4

A =[0, 1, 4, 5, 6, 7, 14, 15, 18, 19, 20, 21] de taille 12 B =[0, 2, 6] de taille 3 D =[6, 7, 20, 21] de taille 4

A =[0, 1, 3, 7, 8, 10, 14, 15, 17, 21, 22, 24] de taille 12 B =[0, 2, 3] de taille 3 D =[3, 10, 17, 24] de taille 4

A =[0, 2, 3, 7, 9, 10, 14, 16, 17, 21, 23, 24] de taille 12 B =[0, 1, 3] de taille 3 D =[3, 10, 17, 24] de taille 4

A =[0, 1, 3, 14, 15, 17] de taille 6 B =[0, 2, 3, 7, 9, 10] de taille 6 D =[3, 10, 17, 24] de taille 4

A =[0, 2, 3, 14, 16, 17] de taille 6 B =[0, 1, 3, 7, 8, 10] de taille 6 D =[3, 10, 17, 24] de taille 4

A =[0, 2, 6, 14, 16, 20] de taille 6 B =[0, 1, 4, 5, 6, 7] de taille 6 D =[6, 7, 20, 21] de taille 4

A =[0, 4, 6, 14, 18, 20] de taille 6 B =[0, 1, 2, 3, 6, 7] de taille 6 D =[6, 7, 20, 21] de taille 4

A =[0, 4, 8, 12, 16, 20, 24] de taille 7 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0

A =[0, 1, 14, 15] de taille 4 B =[0, 2, 4, 6, 8, 10, 12] de taille 7 D =[] de taille 0

A =[0, 7, 14, 21] de taille 4 B =[0, 1, 2, 3, 4, 5, 6] de taille 7 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 14, 15, 16, 17, 18, 19, 20] de taille 14 B =[0, 7] de taille 2 D =[] de taille 0

A =[0, 14] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] de taille 14 D =[] de taille 0

A =[0, 1, 4, 5, 8, 9, 12, 13, 16, 17, 20, 21, 24, 25] de taille 14 B =[0, 2] de taille 2 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26] de taille 14 B =[0, 1] de taille 2 D =[] de taille 0

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27] de taille 28 B =[0] de taille 1 D =[] de taille 0

N =29

A =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28] de taille 29 B =[0] de taille 1 D =[] de taille 0

N =30

A =[0, 1, 3, 4, 6, 7, 9, 10, 12, 13] de taille 10 B =[0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16] de taille 15 D =[3, 4, 5, 6, 6, 7, 7, 8, 8, 9, 9, 9, 10, 10, 10, 11, 11, 11, 12, 12, 12, 12, 13, 13, 13, 13, 14, 14, 14, 14, 15, 15, 15, 15, 16, 16, 16, 16, 17, 17, 17, 17, 18, 18, 18, 19, 19, 19, 20, 20, 20, 21, 21, 22, 22, 23, 23, 24, 25, 26] de taille 60

A =[0, 1, 3, 4, 6, 7, 9, 11, 12, 14, 15, 17, 18] de taille 13 B =[0, 2, 3, 4, 5, 6, 7, 8, 9, 11] de taille 10 D =[3, 4, 5, 6, 6, 7, 7, 8, 8, 9, 9, 9, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 14, 14, 14, 15, 15, 15, 16, 16, 17, 17, 17, 18, 18, 18, 19, 19, 20, 20, 20, 21, 21, 22, 22, 23, 23, 24, 25, 26] de taille 50

A =[0, 1, 2, 3, 5, 7, 8, 11] de taille 8 B =[0, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18] de taille 15 D =[5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 15, 16, 16, 16, 17, 17, 17, 18, 18, 18, 19, 19, 19, 20, 20, 21, 21, 22, 23, 24, 25] de taille 45

A =[0, 3, 4, 6, 8, 9, 10, 11] de taille 8 B =[0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18] de taille 15 D =[4, 5, 6, 7, 8, 8, 9, 9, 10, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 15, 16, 16, 16, 17, 17, 17, 18, 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 24] de taille 45

A =[0, 1, 2, 4, 5, 6, 7, 8, 9, 13] de taille 10 B =[0, 3, 4, 6, 8, 9, 11, 13, 14, 15, 16] de taille 11 D =[4, 5, 6, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 13, 14, 14, 15, 15, 15, 16, 16, 16, 17, 17, 17, 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 24] de taille 40

A =[0, 4, 5, 6, 7, 8, 9, 11, 12, 13] de taille 10 B =[0, 1, 2, 3, 5, 7, 8, 10, 12, 13, 16] de taille 11 D =[5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 16, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 21, 22, 23, 24, 25] de taille 40

A =[0, 1, 2, 3, 5, 7, 9, 10, 12] de taille 9 B =[0, 4, 5, 6, 7, 9, 10, 11, 12, 14, 16, 17] de taille 12 D =[5, 6, 7, 7, 8, 9, 9, 10, 10, 11, 11, 12, 12, 12, 13, 13, 14, 14, 14, 15, 15, 16, 16, 16, 17, 17, 17, 18, 18, 19, 19, 19, 20, 21, 21, 22, 23, 24, 26] de taille 39

A =[0, 2, 3, 5, 7, 9, 10, 11, 12] de taille 9 B =[0, 1, 3, 5, 6, 7, 8, 10, 11, 12, 13, 17] de taille 12 D =[3, 5, 6, 7, 8, 8, 9, 10, 10, 10, 11, 11, 12, 12, 12,

13, 13, 13, 14, 14, 15, 15, 15, 16, 16, 17, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 22, 23, 24] de taille 39

A =[0, 1, 3, 5, 7, 9, 10] de taille 7 B =[0, 2, 3, 5, 6, 8, 9, 10, 11, 13, 14, 16, 17, 19] de taille 14 D =[3, 5, 6, 7, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 23, 24, 26] de taille 34

A =[0, 1, 4, 5, 6, 9] de taille 6 B =[0, 2, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18, 19, 20] de taille 15 D =[6, 7, 8, 9, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 23, 24, 25] de taille 30

A =[0, 1, 4, 5, 6, 9, 10, 11, 14] de taille 9 B =[0, 2, 6, 7, 8, 9, 11, 13, 14, 15] de taille 10 D =[6, 7, 8, 9, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 22, 23, 24, 25] de taille 30

A =[0, 1, 2, 3, 4, 6, 7, 11, 12] de taille 9 B =[0, 5, 6, 8, 10, 11, 12, 13, 15, 17] de taille 10 D =[6, 7, 8, 9, 10, 11, 11, 12, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 17, 18, 18, 19, 19, 20, 21, 22, 23, 24] de taille 30

A =[0, 1, 4, 6, 7, 8, 9, 10, 12] de taille 9 B =[0, 2, 5, 8, 9, 11, 13, 14, 16, 17] de taille 10 D =[6, 8, 9, 9, 10, 11, 12, 12, 13, 14, 14, 15, 15, 16, 17, 17, 17, 18, 18, 19, 20, 20, 21, 21, 22, 23, 23, 24, 25, 26] de taille 30

A =[0, 1, 2, 5, 6, 7] de taille 6 B =[0, 3, 5, 6, 8, 9, 10, 11, 12, 13, 14, 16, 17, 19, 22] de taille 15 D =[5, 6, 7, 8, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 21, 22, 23, 24] de taille 30

A =[0, 1, 5, 6, 10, 11] de taille 6 B =[0, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18] de taille 15 D =[5, 6, 7, 8, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 21, 22, 23, 24] de taille 30

A =[0, 1, 2, 5, 6, 7, 10, 11, 12] de taille 9 B =[0, 3, 5, 6, 8, 9, 11, 12, 14, 17] de taille 10 D =[5, 6, 7, 8, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 21, 22, 23, 24] de taille 30

A =[0, 1, 2, 3, 5, 6, 8] de taille 7 B =[0, 4, 5, 7, 9, 10, 12, 13, 16, 18, 20, 21] de taille 12 D =[5, 6, 7, 8, 9, 10, 10, 11, 12, 12, 13, 13, 14, 15, 15, 16, 17, 18, 18, 19, 20, 21, 21, 22, 23, 24, 26] de taille 27

A =[0, 1, 5, 6, 8, 9, 10, 11, 12] de taille 9 B =[0, 2, 4, 5, 6, 7, 9, 11, 12, 17] de taille 10 D =[5, 6, 7, 8, 9, 10, 10, 11, 11, 12, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 17, 18, 18, 19, 20, 21, 22, 23] de taille 30

A =[0, 3, 4, 5, 8, 9] de taille 6 B =[0, 1, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 20] de taille 15 D =[4, 5, 6, 7, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 20, 21, 22, 23] de taille 30

A =[0, 3, 4, 5, 8, 9, 10, 13, 14] de taille 9 B =[0, 1, 2, 4, 6, 7, 8, 9, 13, 15] de taille 10 D =[4, 5, 6, 7, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 20, 21, 22, 23] de taille 30

A =[0, 2, 3, 6, 7, 9] de taille 6 B =[0, 1, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 17, 19, 20] de taille 15 D =[3, 6, 7, 8, 9, 9, 10, 10, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 19, 19, 20, 20, 21, 22, 23, 26] de taille 30

A =[0, 2, 3, 4, 5, 6, 8, 11, 12] de taille 9 B =[0, 1, 3, 4, 6, 8, 9, 12, 15, 17] de taille 10 D =[3, 4, 5, 6, 6, 7, 8, 8, 9, 9, 10, 11, 11, 12, 12, 12, 13, 14, 14, 15, 15, 16, 17, 17, 18, 19, 20, 20, 21, 23] de taille 30

A =[0, 2, 3, 5, 6, 7, 8] de taille 7 B =[0, 1, 3, 5, 8, 9, 11, 12, 14, 16, 17, 21] de taille 12 D =[3, 5, 6, 7, 8, 8, 9, 10, 11, 11, 12, 13, 14, 14, 15, 16, 16, 17, 17, 18, 19, 19, 20, 21, 22, 23, 24] de taille 27

A =[0, 1, 3, 4, 5, 10, 12] de taille 7 B =[0, 2, 3, 4, 6, 7, 9, 10, 12, 13, 16, 17] de taille 12 D =[3, 4, 5, 6, 7, 7, 8, 9, 10, 10, 11, 12, 12, 13, 13, 14, 14, 15, 16, 16, 17, 17, 18, 19, 20, 21, 22] de taille 27

A =[0, 2, 7, 8, 9, 11, 12] de taille 7 B =[0, 1, 4, 5, 7, 8, 10, 11, 13, 14, 15, 17] de taille 12 D =[7, 8, 9, 10, 11, 12, 12, 13, 13, 14, 15, 15, 16, 16, 17, 17, 18, 19, 19, 20, 21, 22, 22, 23, 24, 25, 26] de taille 27

A =[0, 1, 2, 7, 8, 9, 11, 13] de taille 8 B =[0, 3, 6, 7, 9, 10, 11, 15, 16] de taille 9 D =[7, 8, 9, 10, 11, 11, 12, 13, 14, 15, 16, 16, 17, 17, 18, 18, 19, 20, 22, 23, 24] de taille 21

A =[0, 2, 4, 5, 6, 11, 12, 13] de taille 8 B =[0, 1, 5, 6, 7, 9, 10, 13, 16] de taille 9 D =[5, 6, 7, 9, 10, 11, 11, 12, 12, 13, 13, 14, 15, 16, 17, 18, 18, 19, 20, 21, 22] de taille 21

A =[0, 2, 3, 4, 9, 11, 14] de taille 7 B =[0, 1, 3, 4, 5, 6, 10, 13, 14, 15] de taille 10 D =[3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 14, 15, 15, 16, 17, 17, 18, 19, 24] de taille 20

A =[0, 3, 5, 10, 11, 12, 14] de taille 7 B =[0, 1, 2, 5, 9, 10, 11, 12, 14, 15] de taille 10 D =[5, 10, 11, 12, 12, 13, 14, 14, 15, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 26] de taille 20

A =[0, 2, 6, 8, 10, 14, 16] de taille 7 B =[0, 1, 4, 5, 6, 7, 8, 9, 12, 13] de taille 10 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 14, 15, 15, 16, 17, 18, 19, 20, 21, 22, 23] de taille 20

A =[0, 4, 6, 8, 12] de taille 5 B =[0, 1, 2, 3, 6, 7, 8, 9, 10, 11, 14, 15, 16, 17] de taille 14 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 14, 15, 15, 16, 17, 18, 19, 20, 21, 22, 23] de taille 20

Zoom x2 de A = [0,1,3,4,5,7,8] B = [0,2,3,4,6] D= [3,4,5,6,7,7,8,9,10,11]

A =[0, 1, 2, 6, 7, 9, 10] de taille 7 B =[0, 3, 8, 11, 12, 14, 15, 16, 17, 19] de taille 10 D =[9, 10, 12, 13, 14, 15, 16, 17, 17, 18, 18, 19, 20, 21, 21, 22, 23, 24, 25, 26] de taille 20

A =[0, 1, 3, 4, 8, 9, 10] de taille 7 B =[0, 2, 3, 4, 5, 7, 8, 11, 16, 19] de taille 10 D =[3, 4, 5, 6, 7, 8, 8, 9, 10, 11, 11, 12, 12, 13, 14, 15, 16, 17, 19, 20] de taille 20

A =[0, 2, 3, 4, 6, 15, 17, 18, 19, 21] de taille 10 B =[0, 1, 3, 4, 5, 7, 8] de taille 7 D =[3, 4, 5, 6, 7, 7, 8, 9, 10, 11, 18, 19, 20, 21, 22, 22, 23, 24, 25, 26] de taille 20

A =[0, 1, 3, 4, 5, 7, 8, 15, 16, 18, 19, 20, 22, 23] de taille 14 B =[0, 2, 3, 4, 6] de taille 5 D =[3, 4, 5, 6, 7, 7, 8, 9, 10, 11, 18, 19, 20, ,21, 22, 22, 23, 24, 25, 26] de taille 20

A =[0, 1, 4, 7, 8] de taille 5 B =[0, 2, 5, 6, 7, 8, 10, 11, 13, 14, 15, 16, 19, 21] de taille 14 D =[6, 7, 8, 9, 10, 11, 12, 13, 14, 14, 15, 15, 16, 17, 18, 19, 20, 21, 22, 23] de taille 20

A =[0, 4, 5, 7, 8, 9] de taille 6 B =[0, 1, 2, 3, 5, 6, 9, 12, 16, 17, 20] de taille 11 D =[5, 6, 7, 8, 9, 9, 10, 10, 11, 12, 13, 14, 16, 17, 20, 21, 24, 25] de taille 18

A =[0, 2, 3, 5, 8, 11] de taille 6 B =[0, 1, 3, 5, 6, 8, 9, 10, 16, 17, 18] de taille 11 D =[3, 5, 6, 8, 8, 9, 10, 11, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21] de taille 18

A =[0, 3, 6, 8, 9, 11] de taille 6 B =[0, 1, 2, 8, 9, 10, 12, 13, 15, 17, 18] de taille 11 D =[8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 18, 19, 20, 21, 21, 23, 24, 26] de taille 18

A =[0, 2, 5, 7, 8, 13] de taille 6 B =[0, 1, 4, 6, 9, 10, 12, 13, 14, 15, 16] de taille 11 D =[6, 8, 9, 11, 12, 13, 14, 14, 15, 16, 17, 17, 18, 19, 20, 21, 22, 23] de taille 18

A =[0, 5, 6, 8, 11, 13] de taille 6 B =[0, 1, 2, 3, 4, 6, 7, 10, 12, 15, 16] de taille 11 D =[6, 7, 8, 9, 10, 11, 12, 12, 13, 14, 15, 15, 16, 17, 18, 20, 21, 23] de taille 18

A =[0, 1, 2, 4, 5, 9] de taille 6 B =[0, 3, 4, 8, 11, 14, 15, 17, 18, 19, 20] de taille 11 D =[4, 5, 8, 9, 12, 13, 15, 16, 17, 18, 19, 19, 20, 20, 21, 22, 23, 24] de taille 18

A =[0, 1, 3, 5] de taille 4 B =[0, 2, 3, 5, 6, 7, 10, 11, 12, 13, 14, 16, 19, 23, 24] de taille 15 D =[3, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 19, 24] de taille 15

A =[0, 2, 4, 5] de taille 4 B =[0, 1, 5, 8, 10, 11, 12, 13, 14, 17, 18, 19, 21, 22, 24] de taille 15 D =[5, 10, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24, 26] de taille 15

A =[0, 1, 3, 7] de taille 4 B =[0, 2, 3, 8, 9, 10, 11, 12, 13, 14, 16, 19, 20, 21, 22] de taille 15 D =[3, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23] de taille 15

A =[0, 4, 6, 7] de taille 4 B =[0, 1, 2, 3, 6, 8, 9, 10, 11, 12, 13, 14, 19, 20, 22] de taille 15 D =[6, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20, 26] de taille 15

A =[0, 2, 3, 6, 8, 13, 14] de taille 7 B =[0, 1, 3, 6, 8, 11, 13, 15] de taille 8 D =[3, 6, 8, 9, 11, 13, 14, 14, 15, 16, 17, 19, 21] de taille 13

A =[0, 4, 7, 10, 13, 15, 16] de taille 7 B =[0, 1, 2, 3, 7, 10, 11, 13] de taille 8 D =[7, 10, 11, 13, 14, 15, 16, 17, 17, 18, 20, 23, 26] de taille 13

A =[0, 1, 6, 8, 11, 12, 14] de taille 7 B =[0, 2, 4, 7, 9, 12, 14, 15] de taille 8 D =[8, 10, 12, 13, 14, 15, 15, 16, 18, 20, 21, 23, 26] de taille 13

A =[0, 1, 3, 6, 9, 12, 16] de taille 7 B =[0, 2, 3, 6, 10, 11, 12, 13] de taille 8 D =[3, 6, 9, 11, 12, 12, 13, 14, 15, 16, 18, 19, 22] de taille 13

A =[0, 2, 4, 5, 7, 9] de taille 6 B =[0, 1, 5, 7, 10, 13, 15, 19, 20] de taille 9 D =[5, 7, 9, 10, 12, 14, 15, 17, 19, 20, 22, 24] de taille 12

A =[0, 3, 5, 8, 10, 13] de taille 6 B =[0, 1, 2, 5, 8, 11, 14, 15, 16] de taille 9 D =[5, 8, 10, 11, 13, 14, 15, 16, 18, 19, 21, 24] de taille 12

A =[0, 2, 4, 8, 16] de taille 5 B =[0, 1, 6, 7, 8, 9, 10, 11, 12, 13] de taille 10 D =[8, 9, 10, 11, 12, 13, 14, 15, 16, 17] de taille 10

A =[0, 6, 8, 10, 12] de taille 5 B =[0, 1, 2, 3, 4, 5, 8, 9, 16, 17] de taille 10 D =[8, 9, 10, 11, 12, 13, 14, 15, 16, 17] de taille 10

Zoom x2 de A = [0,1,2,4,8] B = [0,3,4,5,6] D = [4,5,6,7,8]

A =[0, 8, 12, 14, 16] de taille 5 B =[0, 1, 2, 3, 4, 5, 6, 7, 12, 13] de taille 10 D =[12, 13, 14, 15, 16, 17, 18, 19, 20, 21] de taille 10

A =[0, 2, 4, 6, 12] de taille 5 B =[0, 1, 8, 9, 12, 13, 14, 15, 16, 17] de taille 10 D =[12, 13, 14, 15, 16, 17, 18, 19, 20, 21] de taille 10

Zoom x2 de A = [0,4,6,7,8] B = [0,1,2,3,6] D = [6,7,8,9,10]

A =[0, 1, 2, 3, 6, 15, 16, 17, 18, 21] de taille 10 B =[0, 4, 6, 7, 8] de taille 5 D =[6, 7, 8, 9, 10, 21, 22, 23, 24, 25] de taille 10

A =[0, 3, 4, 5, 6, 15, 18, 19, 20, 21] de taille 10 B =[0, 1, 2, 4, 8] de taille 5 D =[4, 5, 6, 7, 8, 19, 20, 21, 22, 23] de taille 10

A =[0, 1, 2, 4, 8, 15, 16, 17, 19, 23] de taille 10 B =[0, 3, 4, 5, 6] de taille 5 D =[4, 5, 6, 7, 8, 19, 20, 21, 22, 23] de taille 10

A =[0, 4, 6, 7, 8, 15, 19, 21, 22, 23] de taille 10 B =[0, 1, 2, 3, 6] de taille 5 D =[6, 7, 8, 9, 10, 21, 22, 23, 24, 25] de taille 10

A =[0, 1, 2, 4, 5, 8, 10, 15, 16, 17, 19, 20, 23, 25] de taille 14 B =[0, 3, 4] de taille 3 D =[4, 5, 8, 19, 20, 23] de taille 6

A =[0, 2, 5, 6, 8, 9, 10, 15, 17, 20, 21, 23, 24, 25] de taille 14 B =[0, 1, 4] de taille 3 D =[6, 9, 10, 21, 24, 25] de taille 6

A =[0, 2, 8] de taille 3 B =[0, 1, 4, 5, 10, 11, 12, 13, 16, 17, 18, 19, 20, 21] de taille 14 D =[12, 13, 18, 19, 20, 21] de taille 6

A =[0, 1, 2, 3, 8, 9] de taille 6 B =[0, 4, 10, 12, 16, 18, 20] de taille 7 D =[12, 13, 18, 19, 20, 21] de taille 6

Zoom x2 de A = [0,1,4] B = [0,2,5,6,8,9,10] D = [6,9,10]

A =[0, 6, 8] de taille 3 B =[0, 1, 2, 3, 4, 5, 8, 9, 10, 11, 16, 17, 20, 21] de taille 14 D =[8, 9, 10, 11, 16, 17] de taille 6

A =[0, 1, 6, 7, 8, 9] de taille 6 B =[0, 2, 4, 8, 10, 16, 20] de taille 7 D =[8, 9, 10, 11, 16, 17] de taille 6

Zoom x2 de A = [0,3,4] B = [0,1,2,4,5,8,10] D = [4,5,8]

A =[0, 1, 4, 15, 16, 19] de taille 6 B =[0, 2, 5, 6, 8, 9, 10] de taille 7 D =[6, 9, 10, 21, 24, 25] de taille 6

A =[0, 3, 4, 15, 18, 19] de taille 6 B =[0, 1, 2, 4, 5, 8, 10] de taille 7 D =[4, 5, 8, 19, 20, 23] de taille 6

A =[0, 1, 2, 15, 16, 17] de taille 6 B =[0, 3, 6, 9, 12] de taille 5 D =[] de taille 0

A =[0, 1, 10, 11, 20, 21] de taille 6 B =[0, 2, 4, 6, 8] de taille 5 D =[] de taille 0

A =[0, 1, 2, 3, 4, 15, 16, 17, 18, 19] de taille 10 B =[0, 5, 10] de taille 3 D =[] de taille 0

A =[0, 10, 20] de taille 3 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9] de taille 10 D =[] de taille 0

A =[0, 1, 6, 7, 12, 13, 18, 19, 24, 25] de taille 10 B =[0, 2, 4] de taille 3 D =[] de taille 0

A =[0, 1, 2, 3, 4, 10, 11, 12, 13, 14, 20, 21, 22, 23, 24] de taille 15 B =[0, 5] de taille 2 D =[] de taille 0

A =[0, 5, 10, 15, 20, 25] de taille 6 B =[0, 1, 2, 3, 4] de taille 5 D =[] de taille 0

A =[0, 6, 12, 18, 24] de taille 5 B =[0, 1, 2, 3, 4, 5] de taille 6 D =[] de taille 0

A =[0, 15] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14] de taille 15 D =[] de taille 0

A =[0, 1, 2, 6, 7, 8, 12, 13, 14, 18, 19, 20, 24, 25, 26] de taille 15 B =[0, 3] de taille 2 D =[] de taille 0

A =[0, 3, 6, 9, 12, 15, 18, 21, 24, 27] de taille 10 B =[0, 1, 2] de taille 3 D =[] de taille 0

A =[0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28] de taille 15 B =[0, 1] de taille 2 D =[] de taille 0

N =31

A =[0, 1, 2, 3, 5, 7, 8, 9, 10, 11, 15] de taille 11 B =[0, 4, 5, 6, 7, 8, 10, 12, 13, 14, 15] de taille 11 D =[5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 15, 15, 15, 16, 16, 16, 17, 17, 17, 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 25] de taille 45

A =[0, 1, 3, 4, 5, 7, 9, 11, 12, 13, 15] de taille 11 B =[0, 2, 3, 4, 6, 8, 10, 11, 12, 14, 15] de taille 11 D =[3, 4, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, 11, 12, 12, 13, 13, 13, 14, 14, 15, 15, 15, 15, 15, 16, 16, 17, 17, 17, 18, 18, 19, 19, 19, 20, 21, 21, 22, 23, 23, 24, 25, 26, 27] de taille 45

A =[0, 1, 3, 4, 5, 6, 7, 9, 10] de taille 9 B =[0, 2, 3, 4, 7, 8, 10, 12, 13, 16, 17, 18, 20] de taille 13 D =[3, 4, 5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 13, 14, 14, 15, 16, 16, 17, 17, 17, 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 25, 26, 27] de taille 43

A =[0, 1, 2, 3, 5, 6, 8, 9, 10] de taille 9 B =[0, 4, 5, 7, 10, 11, 12, 13, 16, 17, 20] de taille 11 D =[5, 6, 7, 8, 9, 10, 10, 11, 12, 12, 13, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 25, 26] de taille 34

A =[0, 1, 2, 4, 5, 7, 8, 9, 10] de taille 9 B =[0, 3, 4, 7, 8, 9, 10, 13, 15, 16, 20] de taille 11 D =[4, 5, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 17, 18, 18, 19, 20, 20, 21, 22, 23, 24, 25] de taille 34

A =[0, 2, 3, 4, 6, 7, 8, 9, 10] de taille 9 B =[0, 1, 3, 4, 5, 8, 9, 11, 14, 15, 20] de taille 11 D =[3, 4, 5, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 17, 17, 18, 18, 19, 20, 21, 22, 23, 24] de taille 34

A =[0, 1, 2, 3, 4, 6, 7, 8, 10] de taille 9 B =[0, 5, 6, 9, 11, 12, 15, 16, 17, 19, 20] de taille 11 D =[6, 7, 8, 9, 10, 11, 12, 12, 13, 13, 14, 15, 15, 16, 16, 17, 17, 18, 18, 19, 19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 25, 26, 27] de taille 34

A =[0, 2, 3, 5, 7, 8, 10] de taille 7 B =[0, 1, 3, 5, 6, 7, 10, 13, 14, 15, 17, 19, 20] de taille 13 D =[3, 5, 6, 7, 8, 8, 9, 10, 10, 11, 12, 13, 13, 14, 15, 15, 16, 17, 17, 18, 19, 20, 20, 21, 22, 22, 23, 24, 25, 27] de taille 30

A =[0, 2, 4, 7, 8, 9, 11, 12, 15] de taille 9 B =[0, 1, 6, 9, 10, 11, 13, 14, 15] de taille 9 D =[8, 9, 10, 11, 12, 13, 13, 14, 15, 15, 16, 17, 17, 18, 18, 19, 20, 21, 21, 22, 22, 23, 24, 25, 26] de taille 25

A =[0, 3, 4, 6, 7, 8, 11, 13, 15] de taille 9 B =[0, 1, 2, 4, 5, 6, 9, 14, 15] de taille 9 D =[4, 5, 6, 7, 8, 8, 9, 9, 10, 11, 12, 12, 13, 13, 14, 15, 15, 16, 17, 17, 18, 19, 20, 21, 22] de taille 25

A =[0, 2, 4, 5, 6, 9, 10] de taille 7 B =[0, 1, 5, 6, 7, 10, 11, 13, 14, 18, 20] de taille 11 D =[5, 6, 7, 9, 10, 10, 11, 11, 12, 13, 14, 15, 15, 16, 16, 17, 18, 19, 20, 20, 22, 23, 24] de taille 23

A =[0, 3, 5, 6, 8, 9, 10] de taille 7 B =[0, 1, 2, 5, 6, 7, 11, 13, 16, 17, 20] de taille 11 D =[5, 6, 7, 8, 9, 10, 10, 11, 11, 12, 13, 14, 15, 16, 16, 17, 19, 20, 21, 22, 23, 25, 26] de taille 23

A =[0, 1, 2, 4, 5, 7, 10] de taille 7 B =[0, 3, 4, 7, 9, 13, 14, 15, 18, 19, 20] de taille 11 D =[4, 5, 7, 8, 9, 10, 11, 13, 14, 14, 15, 16, 17, 18, 19, 19, 20, 20, 21, 22, 23, 24, 25] de taille 23

A =[0, 1, 4, 5, 6, 8, 10] de taille 7 B =[0, 2, 6, 7, 9, 10, 13, 14, 15, 19, 20] de taille 11 D =[6, 7, 8, 10, 10, 11, 12, 13, 14, 14, 15, 15, 16, 17, 18, 19, 19, 20, 20, 21, 23, 24, 25] de taille 23

A =[0, 1, 2, 3, 5] de taille 5 B =[0, 4, 5, 6, 8, 10, 13, 15, 16, 17, 18, 21, 22, 24, 25] de taille 15 D =[5, 6, 7, 8, 9, 10, 11, 13, 15, 16, 17, 18, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27] de taille 22

A =[0, 4, 5, 6, 10] de taille 5 B =[0, 1, 2, 3, 5, 8, 9, 10, 11, 12, 15, 17, 18, 19, 20] de taille 15 D =[5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25] de taille 22

A =[0, 1, 2, 4, 5] de taille 5 B =[0, 3, 4, 8, 9, 10, 11, 14, 16, 18, 19, 21, 22, 23, 25] de taille 15 D =[4, 5, 8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20, 21, 22, 23, 23, 24, 25, 26, 27] de taille 22

A =[0, 1, 3, 4, 5] de taille 5 B =[0, 2, 3, 4, 6, 7, 9, 11, 14, 15, 16, 17, 21, 22, 25] de taille 15 D =[3, 4, 5, 6, 7, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 22, 25, 26] de taille 22

A =[0, 2, 3, 4, 5] de taille 5 B =[0, 1, 3, 4, 7, 8, 9, 10, 12, 15, 17, 19, 20, 21, 25] de taille 15 D =[3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 12, 13, 14, 15, 17, 19, 20, 21, 22, 23, 24, 25] de taille 22

A =[0, 1, 6, 7, 10] de taille 5 B =[0, 2, 4, 10, 11, 12, 15, 16, 18, 19, 20] de taille 11 D =[10, 11, 12, 16, 17, 18, 19, 20, 21, 22, 25, 26] de taille 12

A =[0, 1, 3, 9, 10] de taille 5 B =[0, 2, 3, 7, 10, 11, 13, 14, 16, 18, 20] de taille 11 D =[3, 10, 11, 12, 13, 14, 16, 17, 19, 20, 21, 23] de taille 12

A =[0, 3, 4, 9, 10] de taille 5 B =[0, 1, 2, 4, 5, 8, 9, 10, 16, 18, 20] de taille 11 D =[4, 5, 8, 9, 10, 11, 12, 13, 14, 18, 19, 20] de taille 12

A =[0, 1, 7, 9, 10] de taille 5 B =[0, 2, 4, 6, 7, 9, 10, 13, 17, 18, 20] de taille 11 D =[7, 9, 10, 11, 13, 14, 16, 17, 18, 19, 20, 27] de taille 12

A =[0, 4, 5, 13, 15] de taille 5 B =[0, 1, 2, 3, 5, 6, 7, 10, 11, 14, 15] de taille 11 D =[5, 6, 7, 10, 11, 14, 15, 15, 16, 18, 19, 20] de taille 12

A =[0, 1, 3, 7, 15] de taille 5 B =[0, 2, 3, 8, 9, 10, 11, 12, 13, 14, 15] de taille 11 D =[3, 9, 10, 11, 12, 13, 14, 15, 15, 16, 17, 18] de taille 12

A =[0, 2, 10, 11, 15] de taille 5 B =[0, 1, 4, 5, 8, 9, 10, 12, 13, 14, 15] de taille 11 D =[10, 11, 12, 14, 15, 15, 16, 19, 20, 23, 24, 25] de taille 12

A =[0, 8, 12, 14, 15] de taille 5 B =[0, 1, 2, 3, 4, 5, 6, 7, 12, 13, 15] de taille 11 D =[12, 13, 14, 15, 15, 16, 17, 18, 19, 20, 21, 27] de taille 12

A =[0, 1, 2, 6, 7, 12, 15] de taille 7 B =[0, 3, 8, 9, 13, 14, 15] de taille 7 D =[9, 10, 14, 15, 15, 15, 16, 20, 21] de taille 9

A =[0, 1, 6, 8, 11, 13, 15] de taille 7 B =[0, 2, 4, 7, 9, 14, 15] de taille 7 D =[8, 10, 13, 15, 15, 15, 17, 20, 22] de taille 9

A =[0, 2, 5] de taille 3 B =[0, 1, 4, 6, 7, 9, 10, 11, 12, 14, 16, 20, 23, 24, 25] de taille 15 D =[6, 9, 11, 12, 14, 16, 25] de taille 7

A =[0, 3, 5] de taille 3 B =[0, 1, 2, 5, 9, 11, 13, 14, 15, 16, 18, 19, 21, 24, 25] de taille 15 D =[5, 14, 16, 18, 19, 21, 24] de taille 7

N =32

A =[0, 1, 2, 3, 4, 5, 6, 7, 16, 17, 18, 19, 20, 21, 22, 23] de taille 16 B =[0, 8] de taille 2 D =[] de taille 0

A =[0, 2, 8, 10, 16, 18, 24, 26] de taille 8 B =[0, 1, 4, 5] de taille 4 D =[] de taille 0

A =[0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19, 24, 25, 26, 27] de taille 16 B =[0, 4] de taille 2 D =[] de taille 0

A =[0, 4, 8, 12, 16, 20, 24, 28] de taille 8 B =[0, 1, 2, 3] de taille 4 D =[] de taille 0

A =[0, 16] de taille 2 B =[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] de taille 16 D =[] de taille 0

A =[0, 1, 16, 17] de taille 4 B =[0, 2, 4, 6, 8, 10, 12, 14] de taille 8 D =[] de taille 0

A =[0, 2, 16, 18] de taille 4 B =[0, 1, 4, 5, 8, 9, 12, 13] de taille 8 D =[] de taille 0

A =[0, 1, 2, 3, 16, 17, 18, 19] de taille 8 B =[0, 4, 8, 12] de taille 4 D =[] de taille 0

A =[0, 4, 16, 20] de taille 4 B =[0, 1, 2, 3, 8, 9, 10, 11] de taille 8 D =[] de taille 0

A =[0, 1, 4, 5, 16, 17, 20, 21] de taille 8 B =[0, 2, 8, 10] de taille 4 D =[] de taille 0

A =[0, 2, 4, 6, 16, 18, 20, 22] de taille 8 B =[0, 1, 8, 9] de taille 4 D =[] de taille 0

A =[0, 8, 16, 24] de taille 4 B =[0, 1, 2, 3, 4, 5, 6, 7] de taille 8 D =[] de taille 0

A =[0, 1, 8, 9, 16, 17, 24, 25] de taille 8 B =[0, 2, 4, 6] de taille 4 D =[] de taille 0

A =[0, 2, 8, 10, 16, 18, 24, 26] de taille 8 B =[0, 1, 4, 5] de taille 4 D =[] de taille 0

9.2 Algorithmes en python

9.2.1 *Algorithme 1*

L'algorithme `pavagedansF2` prend en entrée un motif rythmique A sous forme d'une liste d'entiers contenant 0, et renvoie la liste B sous sa forme b_2 et la taille N du canons compact (A, B) modulo 2 de $\mathbb{Z}_N$ obtenu avec l'algorithme 1.

```
def delisteazero(A):
    B = [0 for n in range(0,max(A)+1)]
    for i in A:
        B[i] = 1
    return(B)


def pavagedansF2(A): #liste qui contient 0
    A.sort()
    entree = [0]
    doublon = []
    k = A[(len(A)-1)]
    N = k
    verif = [0 for n in range(0, k+1)]
    for i in A:
        verif[i] = 1
    while ((0) in verif) and (N <3000):
        i = verif.index(0)
        aajouter = i+k +1 - len(verif)
        verif.extend([0 for n in range(0, aajouter)])
        for j in A:
            if verif[j+i] == 0:
                verif[j+i] = 1
            else:
                verif[j+i] = 0
                doublon.append(j+i)
        entree.append(i)
        N = k+i
    print(N +1)
    print(delisteazero(entree))
```

9.2.2 *Algorithme retournant le gabarit*

L'algorithme `gabarit` prend entrées un motif rythmique $A(n) = \{0, 1, n\}$, et un numéro de ligne r, et renvoie le préfixe de taille du gabarit $\mathcal{G}(r)$ de r sur la ligne r du tableau $T(n)$.

```
def tailledeligne(n):
    return(2^(floor(log((n+1),2))+1))


def gabarit(A,r):
    A.sort()
    entree = [0]
    doublon = []
    k = A[(len(A)-1)]
    N = k
    verif = [0 for n in range(0, k+1)]
    for i in A:
        verif[i] = 1
    while ((0) in verif) and (N <(r+3)*k):
        i = verif.index(0)
        aajouter = i+k +1 - len(verif)
        verif.extend([0 for n in range(0, aajouter)])
        for j in A:
            if verif[j+i] == 0:
                verif[j+i] = 1
            else:
                verif[j+i] = 0
                doublon.append(j+i)
        entree.append(i)
        N = k+i
    return(delisteazero(entree)[(r-1)*k:(r-1)*k+tailledeligne(r-1)])
```

9.3
Routine Tikz pour représenter les canons en notation TUBS

En utilisant le *package* `incgraph`, cette routine permet de tracer les canons (A, B) de $\mathbb{Z}_N$ en notation TUBS, en notant A et B comme des listes d'entiers contenant 0, en appelant `tracer{A}{B}{N}`.

```
\newcommand{\tracer}[3]{
\begin{tikzpicture}

\newcounter{i}
\setcounter{i}{0}


\foreach \b in #2 {
\foreach \a in #1 {

\fill  ({int(mod(\a+\b, #3))+0.1}, {\value{i}+0.1}) rectangle ++(0.8,0.8);
%\fill  ({int(mod(\a+\b, #3))}, {\value{i}}) rectangle ++(1,1);

}

\stepcounter{i}

}

\draw (0,0) grid[xstep=1,ystep=1] (#3,\value{i});


%\foreach \j in {0,8,...,#3}{
%
%\draw[ultra thick,color=red] (\j,0) -- (\j,{\value{i}});
%
%}

\end{tikzpicture}
}

\begin{inctext}[paper=graphics]
\tracer{{  0,1,3,6 }}{{0,4 }}{8}
\end{inctext}
```

9.4 Représentation des triangles de l'observation 7.3.15

Nous allons ici présenter plusieurs tableaux les premières lignes des tableaux $T(n)$ pour quelques n. Cela va permettre de remarquer les changements par bloc de la densité des triangle de 1. Les tableaux sont représentés avec des points de différentes tailles. Un point plus petit symbolise un 0 et un point plus gros un 1. Les points sont rouges lorsque le 1 fait partie d'un triangle.

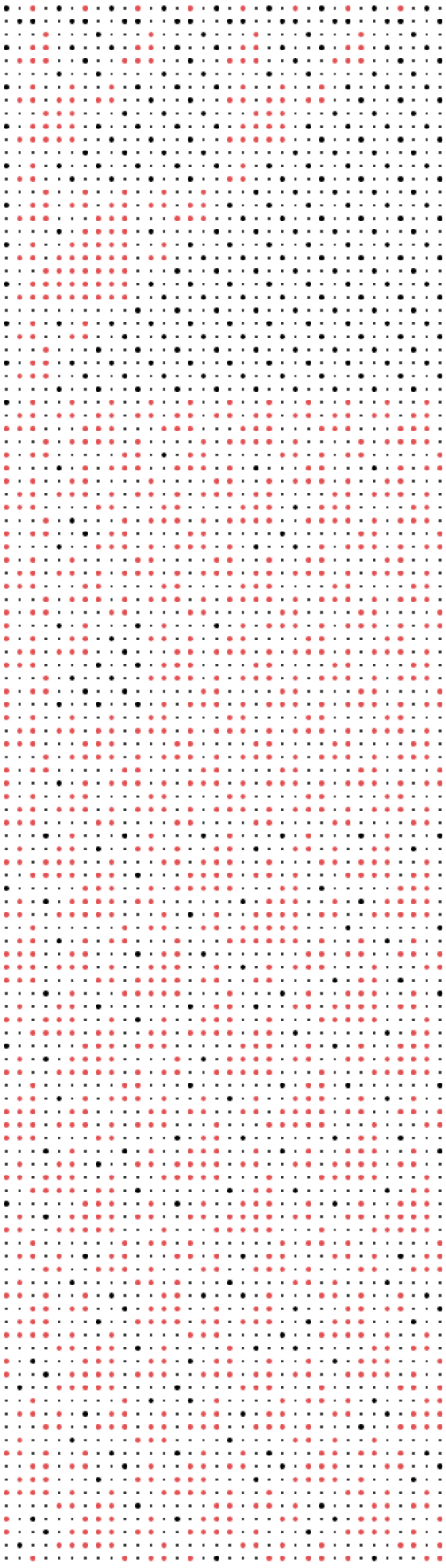


FIGURE 9.1: Les premières lignes du tableau $T(34)$ avec mis en valeur des triangles.

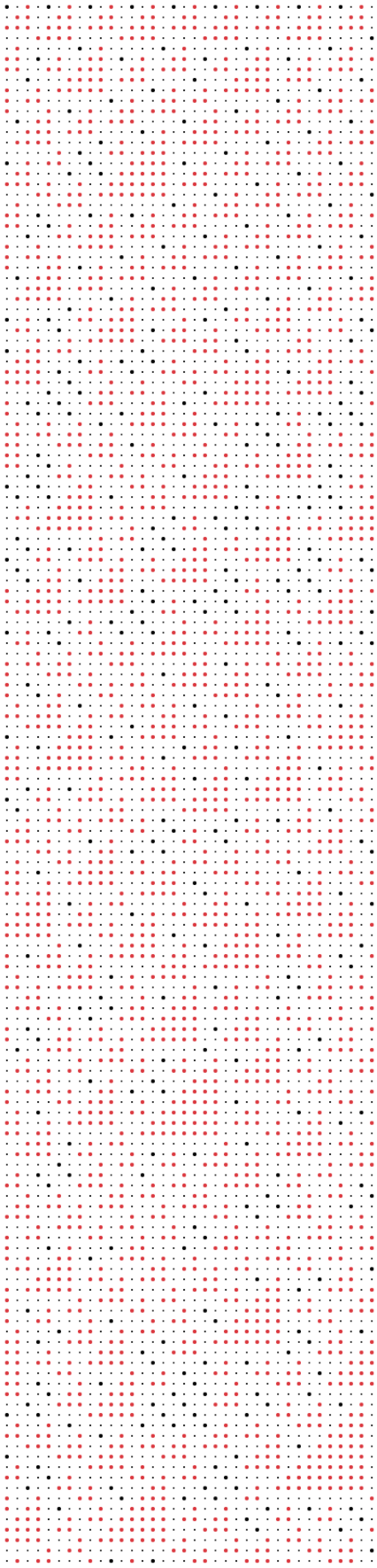


FIGURE 9.2: Les premières lignes du tableau $T(36)$ avec mis en valeur des triangles.

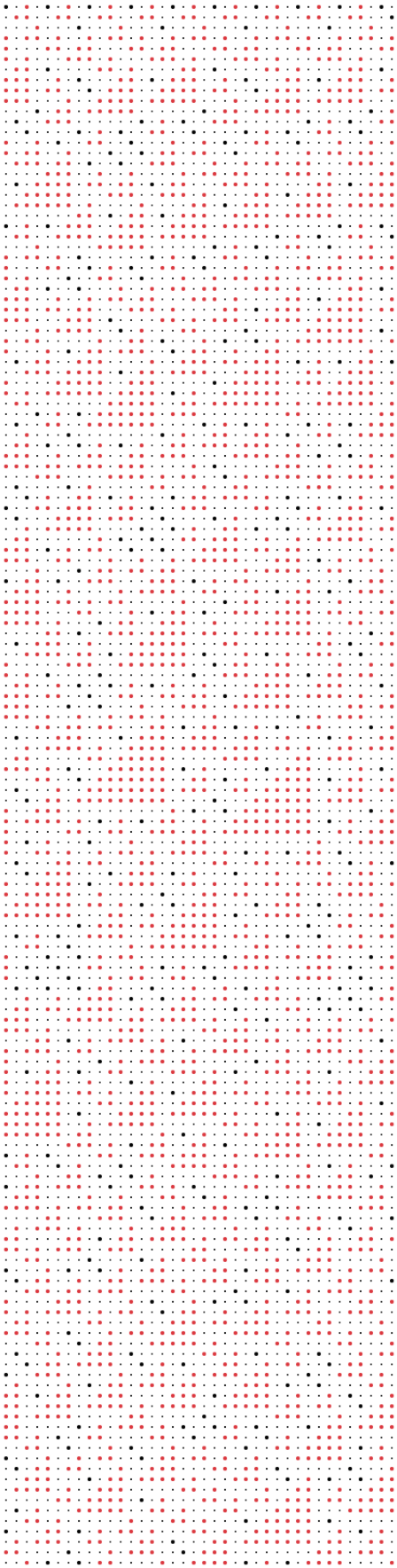


FIGURE 9.3: Les premières lignes du tableau $T(38)$ avec mis en valeur des triangles.

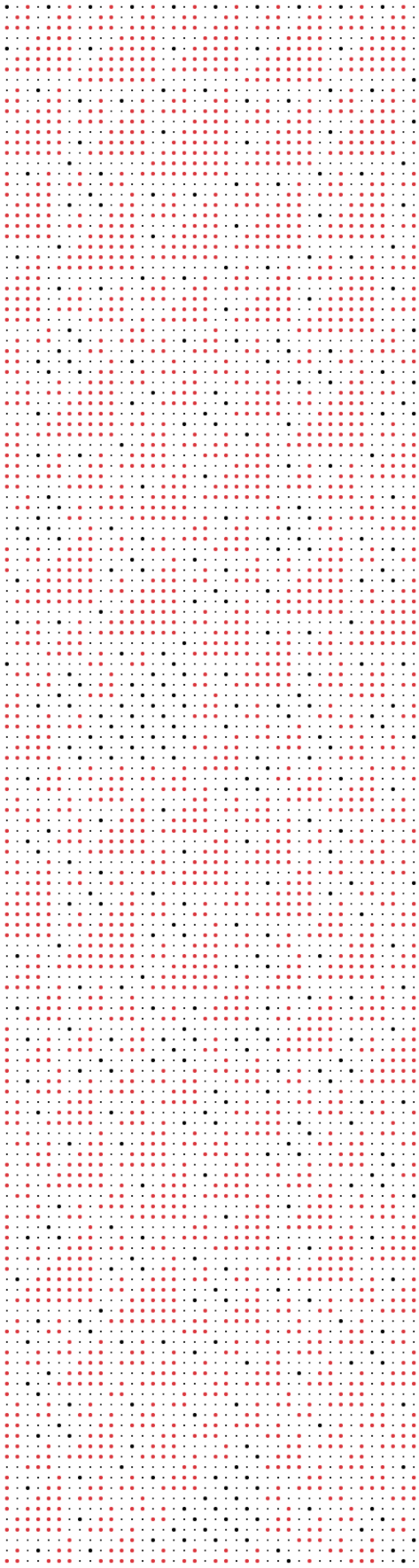


FIGURE 9.4: Les premières lignes du tableau $T(40)$ avec mis en valeur des triangles.

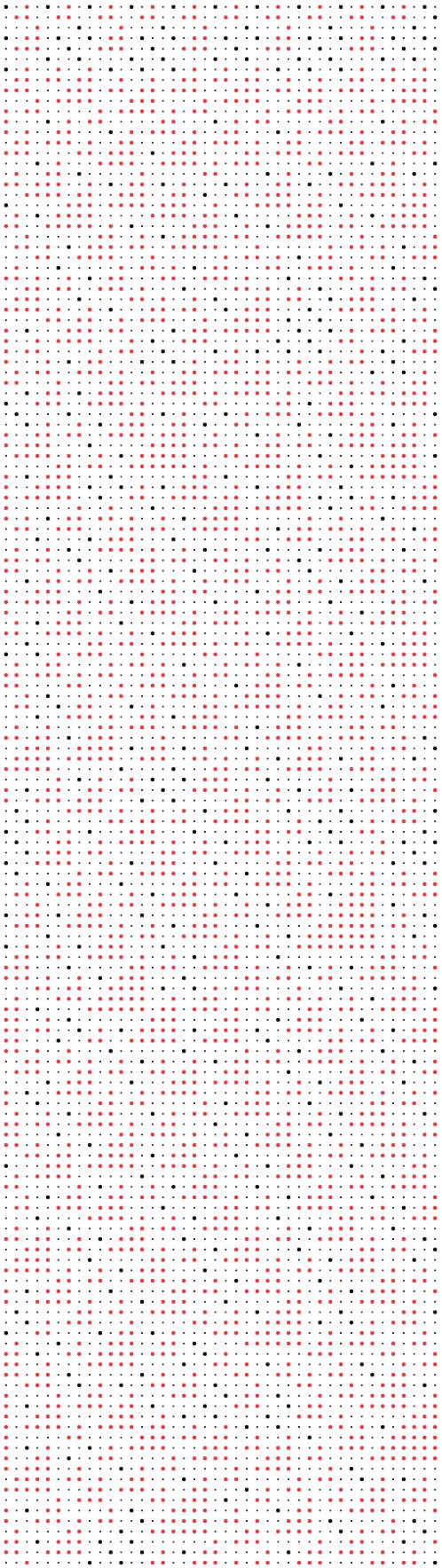


FIGURE 9.5: Les premières lignes du tableau $T(42)$ avec mis en valeur des triangles.

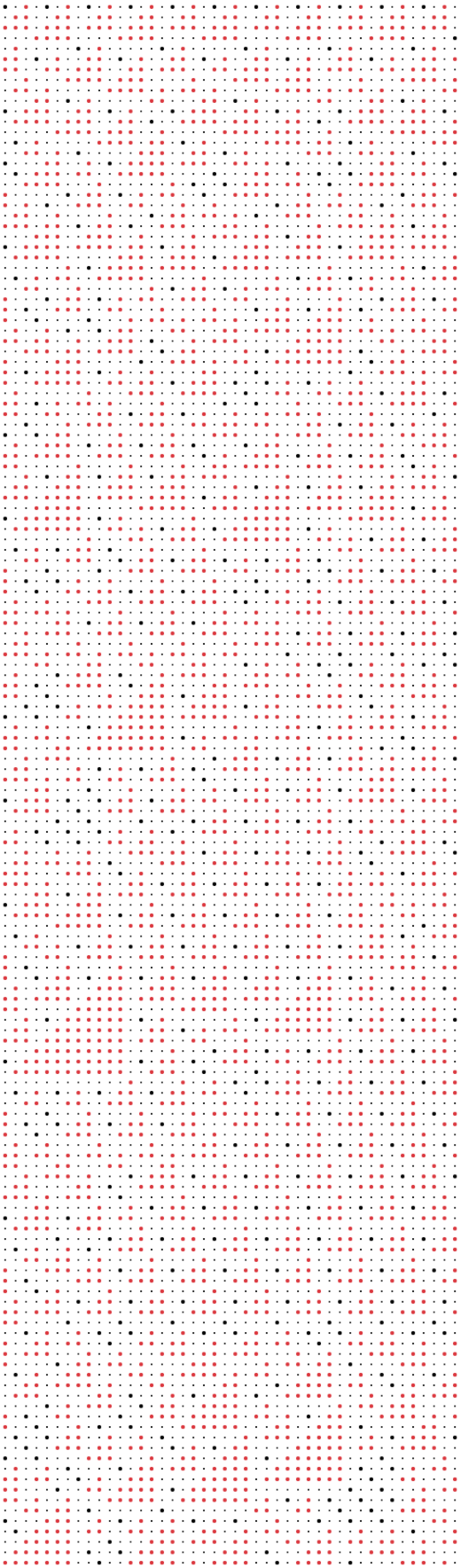


FIGURE 9.6: Les premières lignes du tableau $T(44)$ avec mis en valeur des triangles.

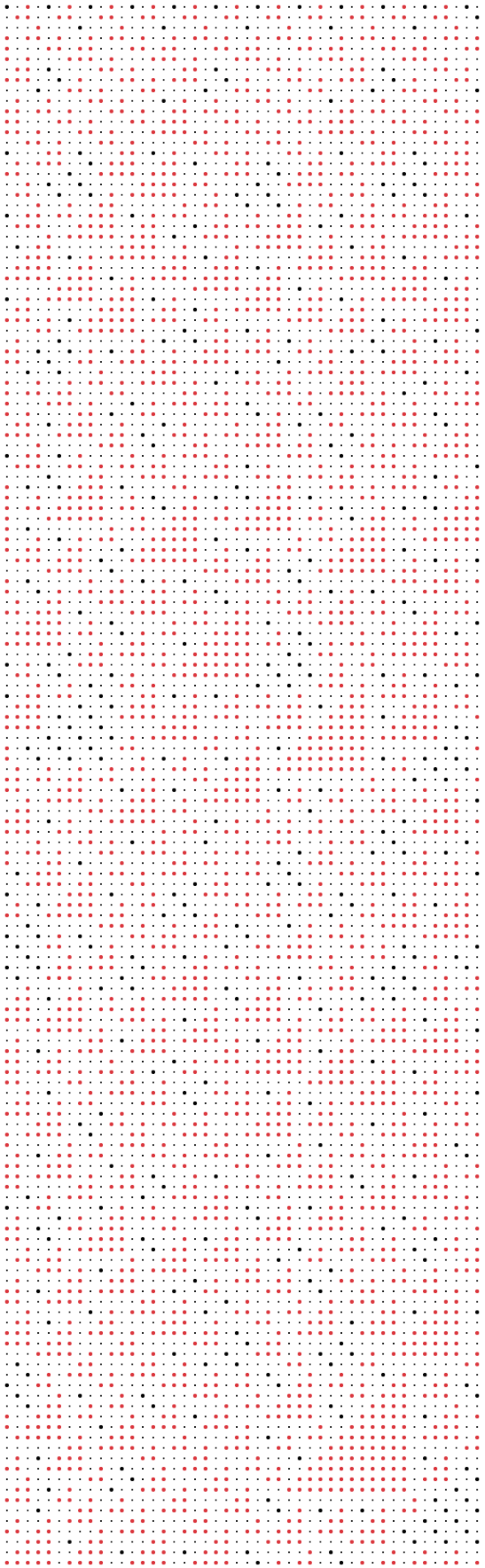


FIGURE 9.7: Les premières lignes du tableau $T(46)$ avec mis en valeur des triangles.

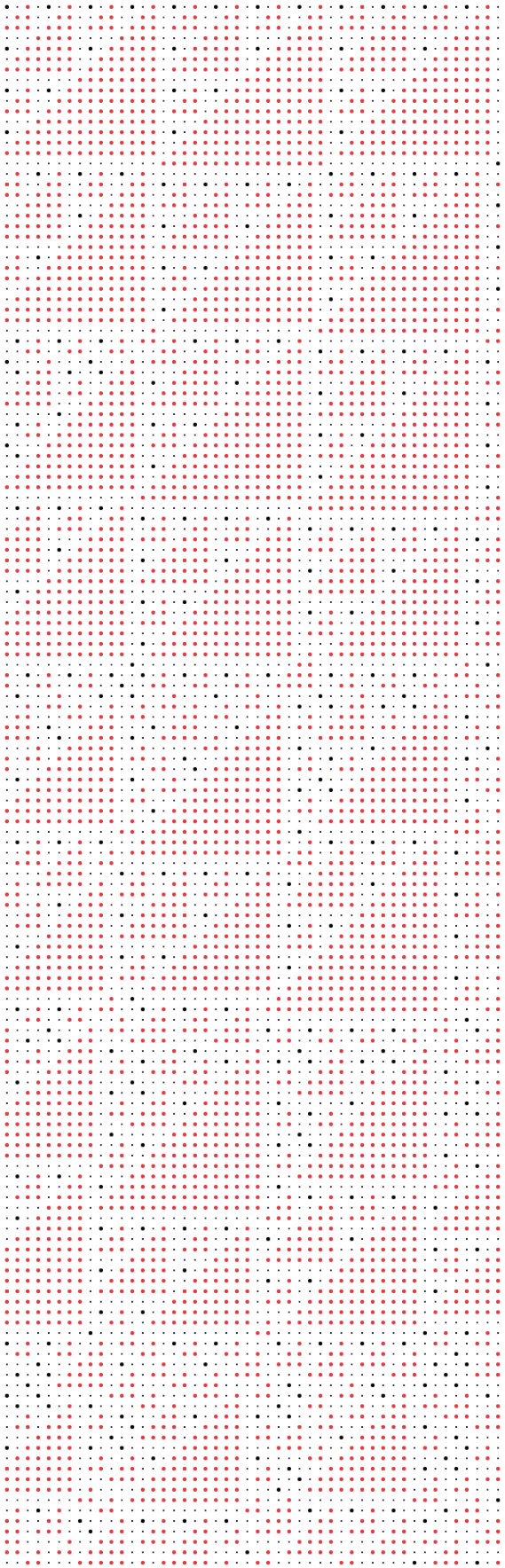


FIGURE 9.8: Les premières lignes du tableau $T(48)$ avec mis en valeur des triangles.

9.5 Table des symboles couramment utilisés

Symbole	Signification
$\mathbb{Z}$	Ensemble des entiers relatifs
$\mathbb{N}$	Ensemble des entiers positifs
$\mathbb{N}^*$	Ensemble des entiers strictement positifs
$\mathbb{Z}_N$	Groupe cyclique à N éléments
$\mathbb{F}_p$	Corps fini à p éléments
$\mathcal{P}(E)$	Ensemble des parties de E
$E[X]$	Ensemble des polynômes à coefficients dans E
$\lvert E\rvert$	Cardinal de E
$\lfloor r\rfloor$	Partie entière de r
$\mathbb{1}_E$	Fonction indicatrice de A
$\oplus$	Somme directe
$\mid$	Symbole de divisibilité
σ	Somme
A, B	Motifs (poly)rythmiques sous forme de (multi-)ensemble
$A(X)$, $B(X)$	Motifs (poly)rythmiques sous forme de polynôme
a, b	Motifs (poly)rythmiques sous forme de mot
A_p	Motif polyrythmique modulo p
$A', \widetilde{A}$	transformés de motif rythmique
(A, B)	Canon rythmique
$UC_p(i)$	Sous-couverture modulo p commençant à l'indice i
$D_{(A,B),p}$	Multi-ensemble de donsets du canon (A, B)
$A(n)$	Motif $\{0, 1, n\}$
$P(X)$, $Q(X)$	Polynômes
p, q, r, s, N, n, t, k	Entiers
T, V, W	Tableaux
ℓ	Ligne d'un tableau
$(u)^k$	Mot u concaténé k fois
$u[n]$	$(n+1)$ème lettre du mot u
$u[n, \ldots, n+k-1]$	Sous-mot de u de taille k commençant par la lettre $u[n]$
$\lvert u\rvert$	Longueur du mot u
u^{R}	Mot miroir de u
$\overline{u}$	Mot conjugué de u

Bibliographie

ANDREW ADLER et FRED HOLROYD. 1981, «Some Results on one-dimensional Tilings», *Geometriae Dedicata*, vol. 10, nº 1, p. 49–58. (Cité en page 14.)

KOFI AGAWU. 1995, «The Invention of “African Rhythm"», *Journal of the American Musicological Society*, vol. 48, nº 3, p. 380–395. (Cité en page 20.)

CARLOS AGON. 1998, *OpenMusic : Un langage visuel pour la composition musicale assistée par ordinateur*, thèse de doctorat. (Cité en page 30.)

CARLOS AGON, JEAN BRESSON et GÉRARD ASSAYAG. 2006, «The OM Composer's Book», *IRCAM–Editions Delatour France*, vol. 1. (Cité en page 30.)

EMMANUEL AMIOT. 2004, «Why Rhythmic Canons are Interesting ?», *Perspectives of Mathematical and Computer-Aided Music Theory*, p. 190–209. (Cité en pages 1, 11, 20, and 26.)

EMMANUEL AMIOT. 2005, «Rhythmic Canons and Galois Theory», *Colloquium on Mathematical Music Theory*. (Cité en pages 1, 26, 28, 29, and 37.)

EMMANUEL AMIOT. 2011, «Structures, Algorithms and Algebraic Tools for Rythmic Canons», *Perspectives of New Music*, vol. 49, nº 2, p. 93–142. (Cité en pages 11, 34, 38, and 68.)

EMMANUEL AMIOT. 2016, *Music through Fourier Space - Discret Fourier Transform in Music Theory*, Computational Music Sciences Series, Springer. (Cité en page 108.)

MORENO ANDREATTA. 2003, *Méthodes algébriques en musique et musicologie du XXe siecle : aspects théoriques, analytiques et compositionnels*, thèse de doctorat, École des Hautes Etudes en Sciences Sociales. (Cité en page 17.)

MORENO ANDREATTA et CARLOS AGON. 2005, «Algebraic Models in Music Theory, Analysis and Composition : Towards a Formalized Computational Musicology», dans *Understanding And Creating Music*. (Cité en page 30.)

MORENO ANDREATTA, CARLOS AGON, THOMAS NOLL et EMMANUEL AMIOT. 2006, «Towards Pedagogability of Mathematical Music

Theory», dans *Bridges. Mathematical Connections in Art, Music and Science*, p. 277–284. (Cité en page 14.)

SIMHA AROM, MARTIN THOM, BARBARA TUCKETT et RAYMOND BOYD. 2004, *African Polyphony and Polyrhythm : Musical Structure and methodology*, Cambridge University Press. (Cité en pages 17 and 18.)

GERARD ASSAYAG, CARLOS AGON, JOSHUA FINEBERG et PETER HANAPPE. 1997, «An Object Oriented Visual Environment for Musical Composition», dans *International Computer Music Conference*, p. 364–367. (Cité en page 30.)

MILTON BABBITT. 1960, «Twelve-tone Invariants as Compositional Determinants», *The Musical Quarterly*, vol. 46, n° 2, p. 246–259. (Cité en page 17.)

MILTON BABBITT. 1992, *The Function of Set Structure in the 12-tone system*, thèse de doctorat, Princeton University. (Cité en page 17.)

GERALD BALZANO. 1980, «The Group-Theoretic Description of 12-fold and Microtonal Pitch Systems», *Computer music journal*, vol. 4, p. 66–84. (Cité en page 17.)

CHRISTOPH BANDT. 1991, «Self-Similar Sets 5. Integer Matrices and Fractal Tilings of $\mathbb{R}^n$», *Proceedings of the American Mathematical Society*, p. 549–562. (Cité en page 14.)

PIERRE BARBAUD. 1968, *La musique, discipline scientifique*, Dunod. (Cité en page 30.)

DANIÈLE BEAUQUIER, MAURICE NIVAT, ERIC RÉMILA et MIKE ROBSON. 1995, «Tiling Figures of the Plane with Two Bars», *Computational Geometry*, vol. 5, n° 1, p. 1–25. (Cité en page 13.)

ROBERT BERGER. 1966, *The Undecidability of the Domino Problem*, vol. 66, American Mathematical Society. (Cité en page 13.)

H BETTERMANN, D AMPONSAH, D CYSARZ et P VAN LEEUWEN. 1999, «Musical Rhythms in Heart Period Dynamics : a Cross-cultural and Interdisciplinary Approach to Cardiac Rhythms», *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 277, n° 5, p. H1762–H1770. (Cité en page 23.)

JEFFREY ADAM BILMES. 1993, *Timing is of the Essence : Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm*, thèse de doctorat, Massachusetts Institute of Technology. (Cité en page 19.)

GEORGES BLOCH. 2006, «Vuza Canons into the Museum», *The OM Composer ?s Book*, vol. 1. (Cité en page 1.)

STEVEN BLOCK et JACK DOUTHETT. 1994, «Vector Products and Intervallic Weighting», *Journal of Music Theory*, vol. 38, nº 1, p. 21–41. (Cité en page 17.)

THADDEUS BOLTON. 1894, «Rhythm», *The American Journal of Psychology*, vol. 6, nº 2, p. 145–238. (Cité en page 19.)

NICOLAAS GOVERT DE BRUIJN. 1950, «On Bases for the Set of Integers», *Publ. Math. Debrecen*, vol. 1, p. 232–242. (Cité en page 15.)

NICOLAAS GOVERT DE BRUIJN. 1953, «On the Factorization of Finite Abelian Groups», *Indag. Math. Kon. Ned. Akad. Wetensch., Amsterdam*, vol. 15, p. 258–264. (Cité en page 15.)

HÉLIANTHE CAURE. 2015, «From Covering to Tiling modulo p (Modulo p Vuza Canons : Generalities and Resolution of a Case modulo 2.)», *arXiv preprint arXiv :1505.06930*. (Cité en pages 6 and 17.)

HÉLIANTHE CAURE, CARLOS AGON et MORENO ANDREATTA. 2014, «Modulus p Rhythmic Tiling Canons and some Implementations in OpenMusic visual Programming Language», dans *Proceedings ICMC| SMC| 2014*, p. 1077–1082. (Cité en page 6.)

MARC CHEMILLIER et CHARLOTTE TRUCHET. 2003, «Computation of Words satisfying the "Rhythmic oddity Property" (after Simha Arom's Works)», *Information Processing Letters*, vol. 86, nº 5, p. 255–261. (Cité en page 17.)

BOGDAN CHLEBUS. 1985, «From Domino Tilings to a new Model of Computation», dans *Computation Theory*, Springer, p. 24–33. (Cité en pages 16 and 33.)

JOHN CLOUGH et JACK DOUTHETT. 1991, «Maximally Even Sets», *Journal of Music Theory*, vol. 35, nº 1/2, p. 93–173. (Cité en page 17.)

STEPHEN COOK. 1971, «The Complexity of Theorem-Proving Procedures», dans *Proceedings of the third annual ACM symposium on Theory of computing*, ACM, p. 151–158. (Cité en page 16.)

GROSVENOR COOPER et LEONARD MEYER. 1963, *The Rhythmic Structure of Music*, vol. 118, University of Chicago Press. (Cité en page 18.)

ETHAN COVEN et AARON MEYEROWITZ. 1999, «Tiling the Integers with Translates of one Finite Set», *Journal of Algebra*, vol. 212, nº 1, p. 161–174. (Cité en page 29.)

EDWARD COYLE et ILYA SHMULEVICH. 1998, «A System for Machine Recognition of Music Patterns», dans *Acoustics, Speech and Signal Processing*, vol. 6, IEEE International Conference on, p. 3597–3600. (Cité en page 22.)

MAXIME CROCHEMORE, CHRISTOPHE HANCART, THIERRY LECROQ et collab. 2001, *Algorithmique du texte*, vol. 3, Vuibert Paris. (Cité en page 78.)

JEAN-PAUL DAVALAN. 2011, «Perfect Rhythmic Tilings», *Perspectives of New Music*, vol. 49, n° 2, p. 144–197. (Cité en page 29.)

KEITH DEVLIN. 1996, *Mathematics : The Science of Patterns : The Search for Order in Life, Mind and the Universe*, Macmillan. (Cité en page 19.)

RENÉ DUMESNIL. 1979, *Rythme musical*, Slatkine. (Cité en page 18.)

ANDREW DUNCAN. 1990, «Combinatorial Music Theory», dans *Audio Engineering Society Convention*, Audio Engineering Society. (Cité en page 17.)

DOUGLAS ECK. 2001, «A Positive-evidence Model for Classifying Rhythmical patterns», dans *Journal of New Music Research*, Citeseer. (Cité en page 21.)

LAZ EN EKWUEME. 1974, «Concepts of African Musical Theory», *Journal of Black Studies*, p. 35–64. (Cité en page 20.)

LEONHARD EULER. 1739, *Tentamen novae theoriae musicae ex certissimis harmoniae principiis dilucide expositae*, ex typographia Academiae scientiarum. (Cité en page 17.)

RICHARD FEYNMAN, FERNANDO MORINIGO et WILLIAM WAGNER. 2003, «Feynman Lectures on Gravitation», . (Cité en page 19.)

ALLEN FORTE. 1973, *The Structure of Atonal Music*, vol. 304, Yale University Press. (Cité en page 18.)

DIRK FRETTLÖH et EDMUND HARRISS. 2014, «Tilings Encyclopedia», . (Cité en page 13.)

HARALD FRIPERTINGER. 2004, «Tiling Problems in Music Theory», *Perspectives in Mathematical and Computational Music Theory*, vol. 1, p. 153. (Cité en page 20.)

BENT FUGLEDE. 1974, «Commuting Self-adjoint Partial Differential Operators and a Group Theoretic Problem», *Journal of Functional Analysis*, vol. 16, n° 1, p. 101–121. (Cité en page 15.)

MICHAEL GAREY et DAVID JOHNSON. 1979, «Computers and Intractability : a Guide to the Theory of NP-Completeness», *WH Free. Co., San Fr.* (Cité en pages 13, 16, 33, and 35.)

JOSCELYN GODWIN. 1992, *The Harmony of the Spheres : The Pythagorean Tradition in Music*, Inner Traditions/Bear and Co. (Cité en page 17.)

SOLOMON GOLOMB. 1966, «Tiling with Polyominoes», *Journal of Combinatorial Theory*, vol. 1, n° 2, p. 280–296. (Cité en pages 13 and 32.)

KARLHEINZ GROCHENIG et ANDREW HAAS. 1994, «Self-similar Lattice Tilings», *Journal of Fourier Analysis and Applications*, vol. 1, n° 2, p. 131–170. (Cité en page 14.)

CATHERINE GUASTAVINO, FABRICE MARANDOLA, GODFRIED TOUSSAINT, FRANCISCO GOMEZ MARTIN et RAFA ABSAR. 2009, «Perception of Rhythmic Similarity in Flamenco Music : Comparing Musicians and Non-Musicians.», . (Cité en page 18.)

KJELL GUSTAFSON. 1987, «A New Method for Displaying Speech Rhythm, with Illustrations from some Nordic Languages», *Nordic Prosody IV*, p. 105–114. (Cité en page 21.)

KJELL GUSTAFSON. 1988, «The Graphical Representation of Rhythm», *Progress Reports from Oxford Phonetics*, vol. 3, p. 6–26. (Cité en page 21.)

JOEL HAACK et collab. 1998, «The Mathematics of Steve Reich's», dans *Bridges : Mathematical Connections in Art, Music, and Science*, Bridges Conference, p. 87–92. (Cité en page 17.)

GYÖRGY HAJÓS. 1949, «Sur la factorisation des groupes abéliens», *Časopis Pest. Mat. Fys*, vol. 74, p. 157–162. (Cité en page 14.)

GYÖRGY HAJÓS. 1950, «Sur le probleme de factorisation des groupes cycliques», *Acta Mathematica Hungarica*, vol. 1, n° 2, p. 189–195. (Cité en page 15.)

LEJAREN HILLER et LEONARD MAXWELL ISAACSON. 1957, *Illiac Suite, for string quartet*, vol. 30, New Music Edition. (Cité en page 30.)

LUDGER HOFMANN-ENGL. 2002, «Rhythmic Similarity : A Theoretical and Empirical Approach», dans *The Seventh International Conference on Music Perception and Cognition*, p. 564–567. (Cité en page 21.)

TC HU et BN TIEN. 1976, «Generating Permutations with Nondistinct Items», *The American Mathematical Monthly*, vol. 83, n° 8, p. 629–631. (Cité en page 23.)

ALEX IOSEVICH, NETS KATZ et TERENCE TAO. 2001, «Fuglede Conjecture holds for Convex Planar Domains», *arXiv preprint math :0104087*. (Cité en page 15.)

ALEX IOSEVICH et GIORGIS PETRIDIS. 2015, «Tiling Sets and Spectral Sets over Finite Fields», *arXiv preprint arXiv :1509.01090*. (Cité en page 15.)

TOM JOHNSON. 2001, «Tiling the Line (pavage de la ligne). Self-Replicating Melodies, Rhythmic Canons, and an Open Problem», *Les Actes des 8eme Journées d'Informatique Musicale*, p. 147–152. (Cité en pages 1 and 29.)

TOM JOHNSON. 2011, «Tiling in My Music», *Perspectives of New Music*, vol. 49, nº 2, p. 9–21. (Cité en page 1.)

MICHAEL KEITH. 1991, *From Polychords to Polya : Adventures in Musical Combinatorics*, Vinculum Press. (Cité en page 17.)

RICHARD KENYON. 1990, «Self-Similar Tilings», cahier de recherche. (Cité en page 14.)

BRANDON KERSHNER. 1968, «On Paving the Plane», *The American Mathematical Monthly*, vol. 75, nº 8, p. 839–844. (Cité en page 13.)

RODERIC KNIGHT. 1971, «Towards a Notation and Tablature for the Kora and its Application to Other Instruments», *African Music*, vol. 5, nº 1, p. 23–36. (Cité en page 20.)

JAMES KOETTING. 1970, «Analysis and Notation of West African Drum Ensemble Music», *Selected Reports in Ethnomusicology*, vol. 1, nº 3, p. 115–46. (Cité en page 20.)

MIHAIL KOLOUNTZAKIS et MÁTÉ MATOLCSI. 2009, «Algorithms for Translational Tiling», *Journal of Mathematics and Music*, vol. 3, nº 2, p. 85–97. (Cité en pages 1, 29, and 36.)

MIHAIL KOLOUNTZAKIS et collab. 2003, «Translational Tilings of the Integers with Long Periods», *Journal of Combinatorics*, vol. 10. (Cité en pages 1 and 34.)

ERNST KRENEK. 1937, *Über Neue Musik*, Verlag der Ringbuchhändlung. (Cité en page 17.)

JEFFREY LAGARIAS et YANG WANG. 1996a, «Self-affine Tiles in $\mathbb{R}^n$», *Advances in Mathematics*, vol. 121, nº 1, p. 21–49. (Cité en page 14.)

JEFFREY LAGARIAS et YANG WANG. 1996b, «Tiling the Line with Translates of one Tile», *Inventiones mathematicae*, vol. 124, p. 341–365. (Cité en pages 10, 14, 25, and 34.)

OTTO LASKE. 1981, «Composition Theory in Koenig's Project One and Project Two», *Computer Music Journal*, vol. 5, nº 4, p. 54–65. (Cité en page 30.)

FABIEN LÉVY. 2011, «Three Uses of Vuza Canons», *Perspectives of New Music*, vol. 49, nº 2, p. 23–31. (Cité en page 1.)

DAVID LEWIN. 2010, *Musical Form and Transformation : Four Analytic Essays*, Oxford University Press. (Cité en page 18.)

MÁTÉ MATOLCSI. 2005, «Fuglede's Conjecture fails in Dimension 4», *Proceedings of the American Mathematical Society*, vol. 133, n° 10, p. 3021–3026. (Cité en page 15.)

KLAUS MEER, CHRISTIAN MICHAUX et collab. 1997, «A Survey on Real Structural Complexity Theory», *Bulletin of the Belgian Mathematical Society Simon Stevin*, vol. 4, n° 1, p. 113–148. (Cité en page 16.)

OLIVIER MESSIAEN. 2002, *Traité de rythme, de couleur, et d'ornithologie :(1949-1992) : en sept tomes*, vol. B, A. Leduc. (Cité en page 24.)

HERMANN MINKOWSKI. 1907, *Diophantische Approximationen : eine Einführung in die Zahlentheorie*, vol. 2, BG Teubner. (Cité en page 14.)

MARVIN MINSKY. 1967, *Computation : Finite and Infinite Machines*, Prentice-Hall, Inc. (Cité en page 16.)

ROGER PENROSE. 1979, «Pentaplexity a Class of non-periodic Tilings of the Plane», *The mathematical intelligencer*, vol. 2, n° 1, p. 32–37. (Cité en pages 13 and 32.)

JEFF PRESSING. 1983, «Cognitive Isomorphisms between Pitch and Rhythm in World Musics : West Africa, the Balkans and Western Tonality», *Studies in Music*, vol. 17, p. 38–61. (Cité en page 22.)

CHUN IN PYONG. 2014, «The relation and Influence between the Musics and Cultures for India and Korea», *Asian Musicology*, p. 109–149. (Cité en page 20.)

RONALD READ. 1997, «Combinatorial Problems in the Theory of Music», *Discrete Mathematics*, vol. 167, p. 543–551. (Cité en page 17.)

LÁSZLÓ RÉDEI. 1947, «Zwei Lückensätze über Polynome in endlichen Primkörpern mit Anwendung auf die endlichen Abelschen Gruppen und die Gaussischen Summen», *Acta Mathematica*, vol. 79, n° 1, p. 273–290. (Cité en page 15.)

STEVE REICH et RUSS HARTENBERGER. 1980, *Clapping Music*, Universal Edition. (Cité en page 16.)

DAVID REINER. 1985, «Enumeration in Music Theory», *The American Mathematical Monthly*, vol. 92, n° 1, p. 51–54. (Cité en page 17.)

ANDRÉ RIOTTE et MARCEL MESNAGE. 2006, «Formalismes et modèles musicaux», *Editions Delatour France/IRCAM*, vol. 2. (Cité en page 30.)

HERVÉ RIVIÈRE. 1993, «On Rhythmical marking in Music», *Ethnomusicology*, vol. 37, n° 2, p. 243–250. (Cité en page 18.)

ARTHUR SANDS. 1957, «On the Factorisation of Finite Abelian Groups», *Acta Mathematica Hungarica*, vol. 8, n° 1, p. 65–86. (Cité en page 15.)

ARTHUR SANDS. 1959, «The Factorization of Abelian Groups», *The Quarterly Journal of Mathematics*, vol. 10, n° 1, p. 81–91. (Cité en page 15.)

ARTHUR SANDS. 1962, «On a Problem of L. Fuchs», *Journal of the London Mathematical Society*, vol. 1, n° 1, p. 277–284. (Cité en page 15.)

ILYA SHMULEVICH, OLLI YLI-HARJA, EDWARD COYLE, DIRK-JAN POVEL et KJELL LEMSTRÖM. 2001, «Perceptual Issues in Music Pattern Recognition : Complexity of Rhythm and Key Finding», *Computers and the Humanities*, vol. 35, n° 1, p. 23–35. (Cité en page 22.)

SHERMAN STEIN et SÁNDOR SZABÓ. 1994, *Algebra and Tiling : Homomorphisms in the service of Geometry*, 25, Cambridge University Press. (Cité en page 14.)

ANDRANIK TANGIAN. 2003, «Constructing Rhythmic Canons», *Perspectives of New Music*, p. 66–94. (Cité en page 1.)

TERENCE TAO. 2003, «Fuglede's Conjecture is False in 5 and Higher Dimensions», *arXiv preprint math/0306134*. (Cité en page 15.)

RAYMOND TENNANT et ABU DHABI. 2003, «Islamic Constructions : The Geometry needed by Craftsmen», dans *BRIDGES/ISAMA International Conference*, p. 459–463. (Cité en page 13.)

ROBERT TIJDEMAN. 1993, *Decomposition of the Integers as a Direct Sum of two Subsets*, Rijksuniversiteit Leiden. Mathematisch Instituut. (Cité en page 14.)

GODFRIED TOUSSAINT et collab. 2002, «A Mathematical Analysis of African, Brazilian, and Cuban Clave Rhythms», dans *Proceedings of BRIDGES : Mathematical Connections in Art, Music and Science*, p. 157–168. (Cité en pages 17, 18, and 22.)

GODFRIED TOUSSAINT et collab. 2003, «Classification and Phylogenetic Analysis of African Ternary Rhythm Timelines», dans *Proceedings of BRIDGES : Mathematical Connections in Art, Music and Science*, p. 25–36. (Cité en pages 18 and 22.)

WILLIAM THOMAS TUTTE. 1965, «The Quest of the Perfect Square», *The American Mathematical Monthly*, vol. 72, n° 2, p. 29–35. (Cité en page 29.)

ANATOL VIERU. 1993, *The Book of Modes*, Editura Muzicala a Uniunii Compozitorilor şi Muzicologilor din România. (Cité en page 18.)

HUGUES VINET, éd.. 2014, *Produire le temps*, Hermann, 73-106 p.. (Cité en page 18.)

DAN TUDOR VUZA. 1985, «Sur le rythme périodique», *Revue Roumaine de Linguistique-Cahiers de Linguistique Théorique et Appliquée*, vol. 22, n° 1, p. 173–188. (Cité en page 19.)

DAN TUDOR VUZA. 1991-1993, «Supplementary Sets and Regular Complementary Unending Canons», *Perspectives of New Music*. In four parts. (Cité en pages 23 and 28.)

HAO WANG. 1961, «Proving Theorems by Pattern Recognition», *Bell system technical journal*, vol. 40, n° 1, p. 1–41. (Cité en page 32.)

HAO WANG. 1965, «Games, Logic and Computers», dans *Computation, Logic, Philosophy*, Springer, p. 195–217. (Cité en pages 13 and 32.)

ANDRÉ WARUSFEL. 1971, *Structures algébriques finies : groupes, anneaux, corps*, vol. 10, Hachette. (Cité en page 37.)

JOHN WILD. 2002, «Tessellating the Chromatic», *Perspectives of New Music*. (Cité en page 29.)

STEPHEN WOLFRAM. 1983, «Statistical Mechanics of Cellular Automata», *Reviews of modern physics*, vol. 55, n° 3, p. 601. (Cité en page 106.)

OWEN WRIGHT. 1978, *The Modal System of Arab and Persian Music AD 1250-1300*, Oxford University Press. (Cité en page 22.)

IANNIS XENAKIS. 1963, «Musiques formelles», *La revue musicale, volumes*, vol. 253. (Cité en pages 18 and 30.)