



Systèmes de particules et collisions discrètes dans les automates cellulaires

Gaétan Richard

► To cite this version:

Gaétan Richard. Systèmes de particules et collisions discrètes dans les automates cellulaires. Autre [cs.OH]. Université de Provence - Aix-Marseille I, 2008. Français. NNT : . tel-00347778

HAL Id: tel-00347778

<https://theses.hal.science/tel-00347778>

Submitted on 16 Dec 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

AIX-MARSEILLE UNIVERSITÉ
ÉCOLE DOCTORALE DE MATHÉMATIQUES ET INFORMATIQUE E.D. 184

THÈSE

présentée pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ DE PROVENCE

Spécialité : Informatique

par

Gaétan RICHARD

sous la direction de Nicolas OLLINGER

Titre :

**Systèmes de particules et collisions discrètes dans
les automates cellulaires**

soutenue publiquement le 4 décembre 2008

JURY

Julien CASSAIGNE	CNRS & Université Aix-Marseille II	<i>Membre</i>
Bruno DURAND	Université Aix-Marseille I	<i>Membre</i>
Enrico FORMENTI	Université de Nice Sophia Antipolis	<i>Président</i>
Luciano MARGARA	Università di Bologna (Italie)	<i>Rapporteur</i>
Jacques MAZOYER	Université de Lyon	<i>Membre / Co-rapporteur</i>
Nicolas OLLINGER	Université Aix-Marseille I	<i>Membre / Directeur de thèse</i>
Alexander SHEN	CNRS & Université Aix-Marseille I	<i>Membre / Rapporteur</i>
Véronique TERRIER	Université de Caen	<i>Membre / Co-rapporteur</i>

Remerciements

Une thèse est certes un accomplissement personnel mais il n'est rendu possible qu'avec l'aide des personnes qui vous accompagnent. Au moment de clore ce mémoire, je me rends compte à quel point leur présence m'a été précieuse au cours de ces dernières années. Cette partie de la thèse est donc destinée à les remercier et je m'excuse à l'avance auprès des personnes que j'ai oubliées ou que je n'ai pas explicitement mentionnées.

Je remercie tout d'abord ma famille qui m'a toujours soutenu tout au long de mes études. J'ai également une pensée particulière pour tous les enseignants que j'ai eu et qui m'ont donné le goût de l'informatique et des sciences en général. En particulier, je souhaite remercier Jacques MAZOYER et Marianne DELORME qui m'ont les premiers initié aux « arcanes » de l'informatique fondamentale à l'ENS Lyon.

Je remercie également les personnes qui m'ont accompagné lors de ces années de thèse au sein du LIF et tout particulièrement les membres de l'équipe Escape dont j'ai partagé le quotidien pendant ces quelques années. Au delà du simple aspect scientifique, j'ai pu profiter d'un environnement agréable dans et en dehors du travail. Qu'il me soit permis de remercier tout particulièrement : Laurent pour son aléatoire, Alexis pour sa distribution, Victor pour ses captures de films, Jérémie pour ses choix de films, Emmanuel pour sa complexité, Guillaume pour son équipe, Grégory pour ses castors et Bruno pour son efficacité dans la résolution des problèmes, ... et Martin que je n'ai pas oublié.

Je remercie également tous les chercheurs rencontrés au cours de ma thèse et qui m'ont permis de progresser dans ma recherche et de m'intégrer dans la communauté scientifique. J'espère pouvoir continuer longtemps à participer aux rencontres FRAC et retrouver les membres (officiels ou non) du feu projet Sycomore.

Je me dois également de remercier Luciano MARGARA, Jacques MAZOYER, Véronique TERRIER et Alexander SHEN pour avoir relu, chacun à leur manière, ma thèse. Je remercie également tous les relecteurs anonymes qui m'ont permis d'en améliorer le contenu tant sur le plan scientifique que de la rédaction. Un grand merci aussi à Julien CASSAIGNE, Bruno DURAND et Enrico FORMENTI pour avoir accepté de siéger dans mon jury de thèse. J'adresse

également une petite pensée pour tous ceux qui ont agit dans l'ombre pour permettre la bonne réalisation de cette thèse.

Et pour finir, je souhaite tout particulièrement remercier mon directeur de thèse, Nicolas OLLINGER qui a su me guider tout au long de cette thèse et sans lequel je n'aurais sans doute jamais pu mener celle-ci à son terme. Par son perfectionnisme et son dynamisme, il m'a appris à me surpasser dans la réalisation de ma recherche. J'espère pouvoir bénéficier encore longtemps de ses conseils avisés et continuer à travailler avec lui.

Table des matières

Introduction	1
1 Automates cellulaires, particules et collisions	5
1.1 Automates cellulaires	6
1.1.1 Présentation	6
1.1.2 Propriétés	9
1.1.3 Nilpotence et trace	13
1.2 Particules et collisions	16
1.2.1 Définitions	16
1.2.2 Propriétés	18
1.2.3 Généralité des définitions	20
1.3 Robustesse des définitions	22
1.3.1 Automates de cartes	23
1.3.2 Cas de $\mathbb{Z}$	25
1.3.3 Cas de $\mathbb{Z}^2$	27
2 Systèmes symboliques de particules et collisions	33
2.1 Particules et collisions syntaxiques	34
2.1.1 Géométrie discrète	34
2.1.2 Patrons	36
2.1.3 Patrons et décidabilité	41
2.2 Ligatures	43
2.2.1 Représentation symbolique	43
2.2.2 Schémas de ligature	44
2.2.3 Pondérations	46
2.3 Validité des schémas de ligature finis	48
2.3.1 Résolution	48
2.3.2 Un exemple complet	52
2.3.3 Expressivité des schémas de ligature finis	54
3 Extensions	57
3.1 Familles régulières de signaux	58
3.1.1 Construction de signaux	58

3.1.2	Schémas de signaux	59
3.2	Schémas de ligature infinis	62
3.2.1	Représentation des schémas de ligature infinis	62
3.2.2	Jeux de faces réguliers	64
3.2.3	Jeux de faces ordonnés	66
4	Applications aux automates cellulaires universels	73
4.1	Universalités	74
4.1.1	Simulation et groupage	74
4.1.2	Liens avec les systèmes PaCo	76
4.2	Universalité Turing de l'automate cellulaire 110	78
4.2.1	Simulation et système PaCo	79
4.2.2	Jeu de faces associé aux calculs	86
4.3	Un intrinsèque universel à 4 états	89
4.3.1	Construction du système PaCo	89
4.3.2	Jeu de faces finiment pondérées	92
	Perspectives	97
	Bibliographie	103

Introduction

UN axe fondateur de l'informatique fondamentale est de comprendre la notion de « calcul ». Un des premier pas dans cette direction a été l'introduction, dans les années 1930, de deux modèles de calcul : les machines de Turing par A. TURING [49] et les fonctions récursives par A. CHURCH [12]. Ces deux approches ont mis en évidence une notion commune et robuste de fonction calculable. Par la suite, de nombreux autres modèles de calcul ont été introduits (systèmes de Post par E. POST [44], machines de Minsky par M. MINSKY [39]...). Il est désormais admis que cette notion de fonction calculable est la même dans tous les systèmes de calcul « raisonnables ». On appelle couramment cette assertion thèse de Church-Turing-Post. Tous ces systèmes ont pour particularité d'être exprimables par des modèles séquentiels dans lesquels l'opération élémentaire du calcul est localisée à un endroit unique.

Dans les années 1940, sur une idée de S. ULAM, J. VON NEUMANN a introduit un système pour modéliser l'auto-reproduction [50] : les automates cellulaires. Ce modèle possède la particularité d'être un modèle de calcul simple et fortement parallèle : il est composé d'entités élémentaires appelées cellules, réparties de façon uniforme, interagissant de façon locale et synchrone. Très vite, il est apparu que ces interactions élémentaires pouvaient donner lieu à des comportements très complexes. Un exemple de tels comportements est observé dans le célèbre *jeu de la vie* introduit par J. CONWAY dans les années 1970 [23].

Ce contraste entre un comportement local simple et un comportement global complexe se rencontre très souvent dans de nombreux phénomènes physiques, chimiques, sociaux, etc. Ces phénomènes sont regroupés sous l'appellation générique de systèmes complexes et ont été étudiés dans des domaines variés. Ces systèmes complexes peuvent être modélisés à l'aide de variantes d'automates cellulaires dans lesquelles on introduit des événements probabilistes [19], quantiques [51]... Bien qu'étant le modèle de base, les automates cellulaires sont également utilisés dans ces modélisations et présentent des comportements variés et complexes. Afin de mieux comprendre les tenants et aboutissants de ces phénomènes, de nombreuses études ont été menées sur le modèle des automates cellulaires (voir ces quelques réf-

rences représentatives : [7, 9, 15, 22, 28, 29, 36]). Ces études peuvent être divisées en deux branches principales : l'étude du modèle en tant que tel et la modélisation de systèmes complexes.

Dans le cadre de l'étude du modèle, de nombreux travaux ont été réalisés pour comprendre les liens entre le comportement local et le comportement global des automates cellulaires. Ces travaux ont donné lieu à une abondante littérature. Pour le lecteur intéressé par ces aspects, un point de départ peut être l'état de l'art effectué par J. KARI [31]. Cette approche permet, en outre, d'appréhender la notion de complexité parallèle du calcul.

Un autre intérêt des automates cellulaires est leur capacité à modéliser des phénomènes réels présents dans de nombreuses disciplines (informatique, biologie, sociologie, chimie, physique, ...). Les expériences et simulations ont mis à jour un certain nombre de phénomènes complexes intéressants. L'un de ces phénomènes, qui va tout particulièrement nous intéresser dans ce mémoire est l'auto-organisation : dans de nombreux automates cellulaires, même si le comportement local est complètement défini, il est impossible de prévoir le comportement global ; cependant, il existe des cas où l'on voit apparaître un comportement prédictible dans un niveau intermédiaire. Un exemple permettant d'intuiter ce phénomène est celui des fourmis. On peut observer que les fourmis interagissent entre elles de façon simple et que le comportement global de la fourmilière est extrêmement complexe. Néanmoins, on peut aussi observer à un niveau intermédiaire des colonies de fourmis ayant un comportement simple et prédictible.

Pour comprendre et analyser ces phénomènes, un premier pas est de réussir à obtenir de tels comportements dans un cadre plus formel. Il se trouve que les automates cellulaires « simples » sont capables d'exhiber des comportements reconnus comme étant de l'auto-organisation. En particulier, S. WOLFRAM a proposé une classification de certains automates cellulaires à l'aide de leur comportement sur des configurations aléatoires [52]. Cette classification met en valeur, de façon empirique, le phénomène d'auto-organisation au travers des particules et des collisions. Comme le titre du mémoire l'indique, ces objets tiennent une place importante dans la suite.

Les particules et collisions ont déjà été étudiées sous plusieurs approches différentes et complémentaires. Une première approche, centrée sur la combinatoire de ces objets, cherche à les énumérer et poser des contraintes. Sur ce sujet, on peut citer, entre autres, l'article de J. CRUTCHFIELD *et al.* [27] qui borne le nombre de collisions possibles en fonction des particules présentes, ou la thèse de B. MARTIN (II) [37] qui s'intéresse à l'existence d'une structure de groupe sur les fonds. Il existe également une approche algorithmique qui se base sur une vision symbolique de ces objets (*les signaux*) et cherche ensuite à les utiliser pour bâtir des constructions algorithmiques subtiles. Dans ce domaine, on peut trouver par exemple les travaux de J. MAZOYER et V. TERRIER [38, 47]. On retrouve également une utilisation, le plus souvent informelle, de ces objets pour les constructions de petits automates

cellulaires complexes. L'exemple le plus significatif d'une telle utilisation est la construction explicite de M. COOK [14] montrant l'universalité de l'automate cellulaire 110.

Malgré ces nombreuses études, il n'existe à l'heure actuelle aucune vision globale des particules et collisions prenant à la fois en compte les aspect combinatoires et algorithmiques. Ce mémoire propose une formalisation des ces objets en accord avec l'intuition obtenue lors de l'observation et développe un ensemble d'outils formels pour manipuler ces objets dans le cadre de constructions algorithmiques. Il est organisé en quatre chapitres.

Le premier chapitre est principalement consacré à la présentation des particules et collisions. Son début est dédié à une série de rappels sur les automates cellulaires qui met en valeur les différents cas (expérimentaux et algorithmiques) dans lesquels les particules et collisions apparaissent. Au passage, nous présentons une nouvelle caractérisation de la nilpotence des automates cellulaires à l'aide de la notion de trace (correspondant aux colonnes des diagrammes espace-temps) dont la preuve repose sur l'aspect bidimensionnel des diagrammes espace-temps. À l'aide de cette vision bidimensionnelle, nous donnons les définitions de fonds, particules et collisions correspondant intuitivement aux objets bipériodiques, unipériodiques et réguliers du plan. Après avoir discuté de la généralité de ces définitions, nous montrerons que ces définitions sont robustes en les mettant en relation avec des colorations simples du plan obtenues en prenant une extension des langages réguliers en dimension 2.

Le deuxième chapitre est dédié à l'étude de l'algorithmique des particules et collisions. Après avoir introduit des notions élémentaires de géométrie discrète, nous donnons de nouvelles définitions (dans une vision syntaxique) de fond, particules et collisions. Ces définitions, dont nous montrons l'équivalence avec celles du chapitre précédent, permettent de déduire des nombreux petits résultats sur la caractérisation des particules et collisions, résultats qui nous seront utiles par la suite. À l'aide de ces nouvelles présentations, nous développons un outils syntaxique de construction d'objets à l'aide d'une nouvelle opération : le schéma de ligature. Nous montrons que ces schémas de ligature peuvent s'interpréter comme des colorations et qu'il existe, dans le cas où le schéma fait intervenir un nombre fini d'éléments, un algorithme décidant si l'opération aboutit à un ensemble de colorations valides. Pour finir, nous étudions plus en détails la complexité de cet algorithme.

Le troisième chapitre propose des extensions du résultat du chapitre précédent afin de faciliter son utilisation. Nous proposons d'abord des méthodes pour construire des systèmes de particules et collisions complexes à partir d'un système de base. Cette extension a pour principal objectif de faciliter l'encodage d'informations dans de tels systèmes. La deuxième partie du

chapitre propose des pistes pour contourner la restriction de la finitude des schémas de ligature. Après avoir discuté des problèmes induits par la description de schémas de ligature infinis, nous développons deux méthodes permettant d'établir les liens entre interprétations et schémas de ligature infinis : les jeux de faces finiment pondérés et les jeux de faces ordonnés.

Dans un dernier chapitre, nous concluons en nous intéressant aux applications des méthodes présentées dans ce mémoire dans le cadre des automates cellulaires universels. Après avoir rappelé les notions d'universalité et regardé les liens entre ces notions et les systèmes de particules et collisions, nous donnons deux exemples complets d'utilisation de ces systèmes. Un premier concerne une nouvelle présentation, par le biais de particules et collisions, de l'universalité Turing de l'automate cellulaire 110. Le second consiste en la construction d'un automate cellulaire intrinsèquement universel de rayon 1 et à 4 état qui est le plus petit connu à ce jour.

Chapitre 1

Automates cellulaires, particules et collisions

ES automates cellulaires étant l'objet principal de notre étude, ce chapitre est consacré à leur présentation. Après une sélection de résultats sur ce modèle, une nouvelle caractérisation de la nilpotence est présentée (Thm. 1.1.8). Viennent ensuite les définitions des notions de particule et collision — objets centraux de cette thèse. Après une rapide étude de quelques propriétés de ces objets, l'accent est mis sur la portée de ces définitions et leur robustesse qui est caractérisée par l'intermédiaire de trois théorèmes (1.3.9, 1.3.11 et 1.3.12).

Sommaire

1.1	Automates cellulaires	6
1.1.1	Présentation	6
1.1.2	Propriétés	9
1.1.3	Nilpotence et trace	13
1.2	Particules et collisions	16
1.2.1	Définitions	16
1.2.2	Propriétés	18
1.2.3	Généralité des définitions	20
1.3	Robustesse des définitions	22
1.3.1	Automates de cartes	23
1.3.2	Cas de $\mathbb{Z}$	25
1.3.3	Cas de $\mathbb{Z}^2$	27

1.1 Automates cellulaires

1.1.1 Présentation

Introduits par J. VON NEUMANN, sur une idée de S. ULAM, pour modéliser les phénomènes d'auto-reproduction [50], les automates cellulaires sont un modèle simple de phénomènes régis par une dynamique locale et uniforme.

Définition 1.1.1. *Un automate cellulaire est un triplet $\mathcal{A} = (Q, r, f)$ où*

- Q est un ensemble fini d'états ;
- $r \in \mathbb{N}$, le rayon ;
- $f : Q^{2r+1} \rightarrow Q$ la règle locale de transition.

Cet objet agit sur l'espace $Q^{\mathbb{Z}}$ des *configurations* — les suites biinfinies d'éléments de Q — par application locale et uniforme de la règle locale de transition. Il en résulte la *fonction globale* $F : Q^{\mathbb{Z}} \rightarrow Q^{\mathbb{Z}}$ définie, pour tout $c \in Q^{\mathbb{Z}}$ et $i \in \mathbb{Z}$, par $F(c)(i) = f(c(i+v) \mid v \in \llbracket -r, r \rrbracket)$. L'étude de ces modèles porte principalement sur propriétés dynamiques de objets obtenus lors d'applications successives de la fonction globale. La fonction ainsi obtenue est dénotée par F^t où $t \in \mathbb{N}$ est le *temps*. En particulier, pour une configuration *initiale* $c \in Q^{\mathbb{Z}}$, l'*orbite* $O_c \in Q^{\mathbb{N} \times \mathbb{Z}}$ de cette configuration est définie, pour tout $t \in \mathbb{N}$, par $O_C(t) = F^t(c)$.

Un sous-ensemble basique d'automates cellulaires est celui obtenu en fixant le rayon à 1 et le nombre d'états à 2. À isomorphisme près, on obtient les 256 *automates cellulaires élémentaires*. Ces automates sont usuellement désignés à l'aide d'un nombre entre 0 et 255 correspondant à la valeur obtenu en regardant les valeurs prises par leur règle locale de transition comme un mot binaire. La figure 1.1 présente, par exemple, la règle locale de l'automate cellulaire 110 présent tout au long de ce mémoire.

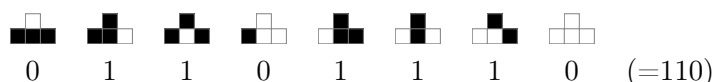


Fig. 1.1 – Règle locale de transition de l'automate cellulaire 110

Parmi les configurations $c \in Q^{\mathbb{Z}}$, on distingue les configurations *périodiques* de période $u \in \mathbb{N}^+$ vérifiant, pour tout $i \in \mathbb{Z}$, $c(i) = c(u+i)$. De même, les configurations *ultimement périodiques* de période $u \in \mathbb{N}^+$ et de *défaut* $k \in \mathbb{N}$ sont les éléments $c \in Q^{\mathbb{Z}}$ vérifiant, pour tout $i \in \mathbb{Z}$ tel que $|i| > k$, $c(i) = c(i+u)$. Une telle configuration ultimement périodique est caractérisée par trois mots finis $g, d \in Q^u$ et $m \in Q^{2k+1}$ vérifiant, pour tout $i \in \mathbb{Z}$:

$$c(i) = \begin{cases} g(i \bmod u) & \text{si } i < -k \\ m(i - k/2) & \text{si } -k \leq i \leq k \\ d(i \bmod u) & \text{si } i > k \end{cases}$$

Dans la suite, une telle décomposition d'un mot ultimement périodique c est notée ${}^\omega gmd^\omega$. Les configurations périodiques étant un cas particulier de configurations ultimement périodiques, la notation précédente est étendue par ${}^\omega m^\omega$ dans le cas des configurations périodiques. À cause de l'uniformité de la fonction globale, il est clair que l'image par cette fonction d'une configuration ultimement périodique reste ultimement périodique et que l'image d'une configuration périodique reste périodique. En appelant *quiescent* un état $q \in Q$ vérifiant $f(q^{2^r+1}) = q$, une configuration $c \in Q^\mathbb{Z}$ *q-finie* est une configuration égale à q partout sauf en un nombre fini de positions (*i.e.*, vérifiant qu'il existe un entier $k \in \mathbb{N}$ tel que $c(i) = q$ pour tout $i \in \mathbb{Z}$ tel que $|i| > k$). Dans la suite, la mention de l'état est omise même si cette condition est essentielle car elle permet d'assurer que l'image d'une configuration finie par application de la fonction globale reste finie.

Les configurations sont représentées à l'aide de lignes horizontales constituées de carrés colorés représentant chacun un élément de Q . Cette représentation est étendue aux orbites en empilant les itérés successifs de la configuration initiale par application de la fonction globale comme présenté sur la figure 1.2. Cette représentation donne naissance à une coloration du plan dans lequel le temps est orienté de bas en haut.

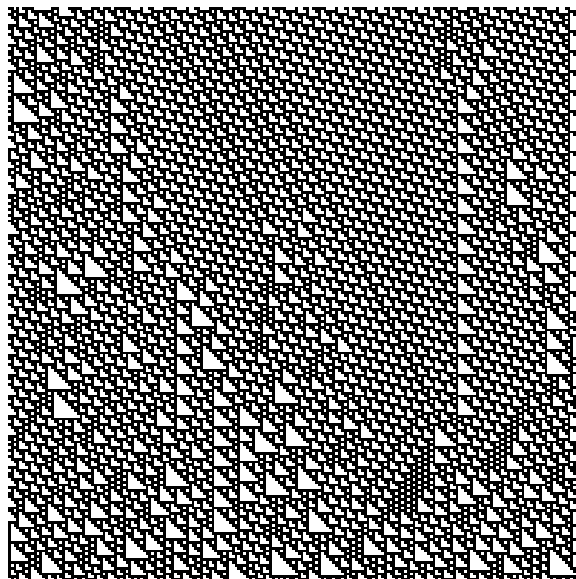


Fig. 1.2 – Exemple d'orbite de l'automate cellulaire 110

Dans cette thèse, nous choisissons de nous placer dans le cadre bidimensionnel. Ce choix permet, entre autre, de se retrouver dans le cadre connu et étudié de la géométrie discrète et de pouvoir en utiliser les éléments. Dans ce cadre, pour un alphabet fini Σ , une *coloration* est un élément de $\Sigma^{\mathbb{Z}^2}$. Il est alors possible d'étendre la notion d'orbite de la façon suivante :

Définition 1.1.2. *Un diagramme espace-temps est une coloration $\Delta \in Q^{\mathbb{Z}^2}$ vérifiant que, pour tout $t \in \mathbb{Z}$, $\Delta(t+1) = F(\Delta(t))$.*

Comme les automates cellulaires sont définis de manière locale, le diagramme espace-temps peut également être défini comme un pavage $\Delta \in Q^{\mathbb{Z}^2}$ du plan suivant l'ensemble de contraintes locales définies par, pour tout $i, t \in \mathbb{Z}^2$, $\Delta(t+1)(i) = f(\Delta(t)(i+v) \mid v \in \llbracket -r, r \rrbracket)$.

Dans ce cadre bidimensionnel, un *vecteur* est un élément de $\mathbb{Z}^2$. Deux vecteurs $u, v \in \mathbb{Z}^2$ sont *colinéaires* s'il existe deux entiers $n, n' \in \mathbb{Z}^*$ vérifiant $nu = n'v$. La notion de périodicité est étendue à l'ensemble des colorations : une coloration $c \in \Sigma^{\mathbb{Z}^2}$ est *périodique* de période $u \in \mathbb{Z}^2 \setminus \{(0,0)\}$ si, pour tout $i \in \mathbb{Z}^2$, $c(i+u) = c(i)$. Elle est *bipériodique* si elle possède deux périodes non colinéaires.

Dans un cadre plus général, il est possible de définir la notion de *système dynamique discret* comme étant un couple (E, g) où E est un espace compact et $g : E \rightarrow E$ une fonction continue et d'étudier les propriétés (topologiques par exemple) des itérés de g .

Sur l'espace des configurations, la topologie de Cantor induite par la distance $d : Q^{\mathbb{Z}} \times Q^{\mathbb{Z}} \rightarrow \mathbb{R}^+$ est définie, pour toutes colorations $w, w' \in Q^{\mathbb{Z}}$, par $d(c, c') = 2^{-\min_{i \in \mathbb{Z}} \{|i| \mid c(i) \neq c'(i)\}}$ (cf. Fig. 1.3). Muni de cette topologie, l'espace des configurations est un compact. De plus, la localité de la fonction globale implique que celle-ci est continue pour cette topologie ce qui fait du couple $(Q^{\mathbb{Z}}, F)$ un système dynamique discret.

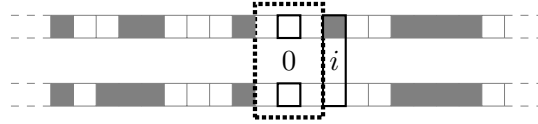


Fig. 1.3 – Deux configurations à distance 2^{-i}

Une telle vision permet d'introduire de nombreux et puissants outils de démonstration et d'analyse. En particulier, l'*ensemble limite* défini par $\Omega_F = \bigcap_{n \in \mathbb{N}} F^n(Q^{\mathbb{Z}})$ correspond à l'ensemble des configurations pouvant être atteint après un nombre arbitraire d'étapes. Par compacité, on remarque que cet ensemble est forcément non vide et qu'il correspond aux configurations ayant une chaîne infinie d'antécédents. Cette dernière propriété prenant ici un relief particulier puisqu'elle permet de faire un lien fort entre orbites et diagrammes espace-temps : les restrictions de diagramme espace-temps à un demi-plan $\mathbb{N} \times \mathbb{Z}$ correspondent exactement aux orbites des éléments de l'ensemble limite. Cette caractérisation apporte une justification au choix d'utiliser les diagrammes espace-temps puisqu'il caractérisent la dynamique limite de l'automate cellulaire (*i.e.*, sans les phénomènes transitoires).

Pour tenir compte de l'uniformité de la règle globale des automates cellulaires dans la vision des systèmes dynamiques discrets, il est possible d'introduire le *shift* $\sigma : Q^{\mathbb{Z}} \rightarrow Q^{\mathbb{Z}}$ défini, pour tout $c \in Q^{\mathbb{Z}}$ et $i \in \mathbb{Z}$, par

$\sigma(c)(i) = c(i + 1)$. À l'aide de celui-ci, il est possible de caractériser complètement les automates cellulaires en tant que systèmes dynamiques :

Théorème 1.1.3 (G. HEDLUND, 1969 [26]). *Les fonctions de transition globales des automates cellulaires sur l'alphabet Q sont les fonctions continues de $Q^{\mathbb{Z}} \rightarrow Q^{\mathbb{Z}}$ qui commutent avec le shift.*

Ce théorème permet d'assimiler les automates cellulaires avec les fonctions globales de transitions.

1.1.2 Propriétés

Dans cette section, nous choisissons de présenter une sélection de résultats obtenus sur les automates cellulaires en les regroupant de façon thématique.

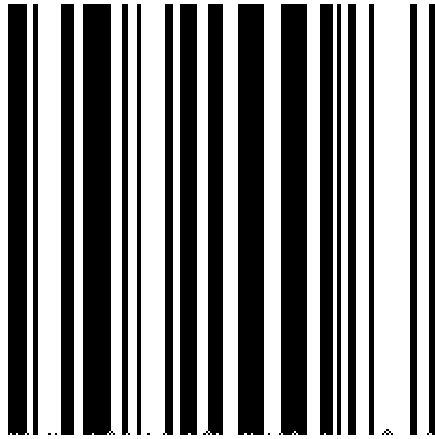
Introduits dans ce but, les automates cellulaires ont été très souvent utilisés pour la modéliser des phénomènes gouvernés par des règles locales [54]. Dans cette optique, de nombreuses observations ont été effectuées sur les automates cellulaires afin d'appréhender l'ensemble des comportements exhibés par ces objets. De ce point de vue, un des travaux les plus connus est la classification informelle proposée par S. WOLFRAM [52]. Cette classification empirique repose sur l'observation du comportement apparent des automates cellulaires élémentaires, sur des configurations initiales aléatoires suivant une loi uniforme, et comporte quatre classes (voir Fig. 1.4) :

- Classe I : « *evolution leads to a homogeneous state* » ;
- Classe II : « *evolution leads to a set of separated simple stable or periodic structures* » ;
- Classe III : « *evolution leads to a chaotic pattern* » ;
- Classe IV : « *evolution leads to complex localised structures, sometimes long-lived* » .

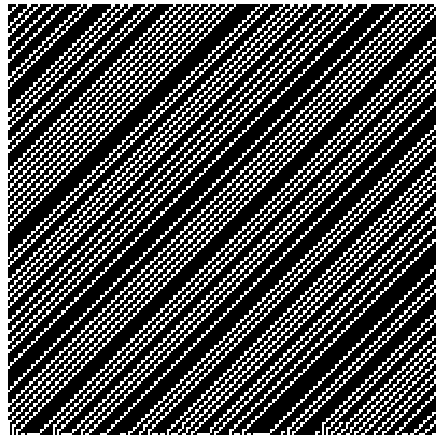
Par la suite, d'autres classifications plus formelles ont été proposées. Que ce soit pour étudier le comportement asymptotique des automates cellulaires comme la classification de K. ČULÍK II et S. YU [16] basée sur l'ensemble limite ; pour étudier la notion de chaos comme la classification de B. DURAND *et al.* [17] basée sur la complexité de Kolmogorov ; ou pour étudier la complexité algorithmique des automates cellulaires [45, 43, 48] en se basant sur la notion de groupage.

La classification de Wolfram met en valeur, par l'intermédiaire de sa classe IV, l'existence d'automates cellulaires présentant spontanément des structures ayant un comportement complexe mais prédictible. Ces structures permettent d'encoder et de manipuler de l'information dans l'évolution des automates cellulaires et sont les éléments essentiels développés dans ce mémoire sous le nom de particules et collisions..

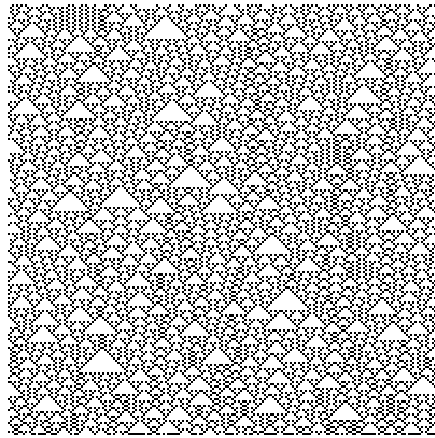
Ces objets sont également des éléments fondamentaux dans l'étude des automates cellulaires selon le point de vue algorithmique. En effet, les automates cellulaires peuvent être vus et étudiés comme un modèle de calcul



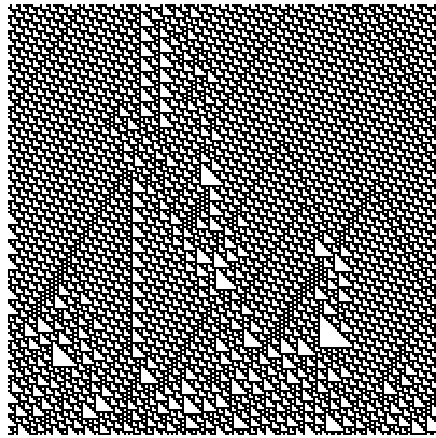
Règle 232 (classe I)



Règle 215 (Classe II)



Règle 18 (classe III)



Règle 110 (Classe IV)

Fig. 1.4 – Exemple de classes selon S. WOLFRAM

massivement parallèle. Pour cela, il suffit de distinguer un état particulier $q_f \in Q$ qui servira d'état d'arrêt. On considère alors l'automate comme un reconnaiseur de langage sur un alphabet $Q' \subset Q \setminus \{q_f\}$ en définissant que l'automate *accepte* le mot $w \in Q'^*$ si l'état d'arrêt apparaît dans l'orbite de configuration finie $c \in Q^{\mathbb{Z}}$ contenant w . Il est possible de définir une notion de temps (nombre d'itérations nécessaires) et d'espace (taille maximale de la portion non quiescente du mot) pour un tel calcul. Une classe de complexité temporelle particulière et intéressante dans ce cadre est le temps réel. Un langage est *reconnu en temps réel* par un automate cellulaire de rayon 1 si pour tout mot w , l'état d'arrêt apparaît en position 0 et au temps $|w|$ si et seulement si le mot appartient au langage. Intuitivement, ce temps correspond au nombre minimal d'étapes nécessaire pour faire parvenir l'information complète du mot à cette position 0.

Problème ouvert 1.1.4 (A. R. SMITH III, 1971 [46]). *Le temps réel sur automate cellulaire est-il égal au temps linéaire ?*

La notion de temps réel est équivalent à la notion de Fisher conductibilité définie de la façon suivante : un langage $L \subset \mathbb{N}$ est *Fisher-constructible* s'il existe un automate cellulaire de rayon 1 et une configuration finie $c \in Q^{\mathbb{Z}}$ telle que $O(t)(O) = q_f$ si et seulement si $t \in L$. Cette notion tire son origine d'une des première construction de ce type réalisée par P. FISHER qui marque l'ensemble des nombre premier [21]. La figure 1.5 présente un automate à 8 état du à I. KOREC [35] réalisant une telle construction.

Dans de telles constructions, on retrouve les structures encodant l'information que nous avons déjà mentionnées en discutant des classifications. Dans ce cadre, ces objets sont souvent associés à la notion de *signal* telle qu'étudiée par J. MAZOYER and V. TERRIER [38].

Pour conclure l'aperçu des propriétés connues des automates cellulaires, il est impossible de faire l'impasse sur les liens entre règle locale de transition et fonction globale. En effet, même si le premier détermine entièrement le second, il existe des cas dans lesquels des propriétés de la fonction globale ne peuvent pas être décidées en fonction de la règle locale.

Un résultat historique sur les propriétés de la fonction globale est celui liant la surjectivité et l'injectivité sur les configurations finies :

Théorème 1.1.5 (E. MOORE, 1962 [40] et J. MYHILL, 1963 [42]). *Un automate cellulaire est surjectif si et seulement si il est injectif sur les configurations finies.*

Ce résultat peut-être complété en le diagramme de la figure 1.6 par d'autres résultats classiques et bien connu pour donner un aperçu des liens entre injectivité et surjectivité suivant l'espace des configurations sous-jacent.

Il est également possible de montrer que certaines propriétés de ce type peuvent être décidées à partir de la règle locale.

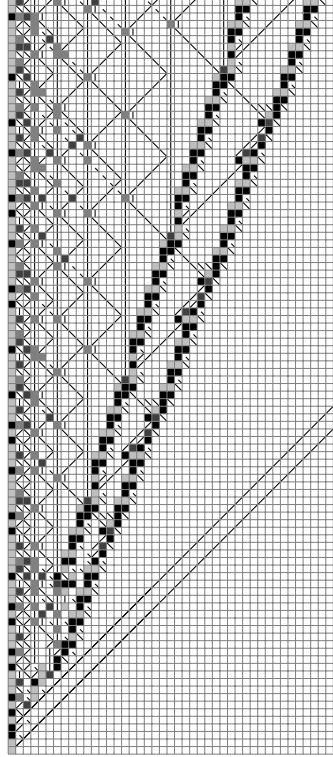


Fig. 1.5 – Marquage des nombres premiers par automate cellulaire

$$\begin{array}{c}
 F \text{ reversible} \leftrightarrow F \text{ injective} \leftrightarrow F|_P \text{ injective} \\
 \downarrow \\
 F|_F \text{ surjective} \\
 \downarrow \\
 F \text{ surjective} \leftrightarrow F|_P \text{ surjective} \leftrightarrow F|_F \text{ surjective}
 \end{array}$$

Fig. 1.6 – Implications entre différentes propriétés sur les automates cellulaires 1D

$F|_P$ et $F|_F$ désignent la restriction de la fonction globale respectivement aux configurations périodiques et finies.

Théorème 1.1.6 (S. AMOROSO et Y. PATT, 1972 [6]). *L'injectivité et la surjectivité sont décidable pour les automates cellulaires 1D.*

Néanmoins, toutes les propriétés ne sont pas décidables. Un automate cellulaire $\mathcal{A} = (Q, f)$ est *nilpotent* si Ω_F est un singleton. En d'autre termes toutes les orbites de l'automate cellulaire convergent vers la même configuration uniforme. En regardant le comportement d'une configuration univers, on remarque que le temps de convergence est borné uniformément (*i.e.* il existe $n \in \mathbb{N}$, tel que $F^n(Q^{\mathbb{Z}}) = \{\omega q^\omega\}$). Bien que simple, ce comportement est néanmoins indécidable :

Théorème 1.1.7 (J. KARI, 1992 [30]). *La nilpotence des automates cellulaires est indécidable.*

La preuve de ce théorème repose principalement sur l'utilisation des diagrammes espace-temps en les couplant avec des propriétés connues sur les pavages. Elle permet d'apercevoir l'intérêt que présente une vision bidimensionnelle dans l'étude des automates cellulaires.

1.1.3 Nilpotence et trace

Une notion liée à la dynamique de l'automate cellulaire est de regarder l'ensemble des colonnes possibles des orbites. Formellement, la *trace* d'un automate cellulaire (Q, r, f) sur une configuration $c \in Q^{\mathbb{Z}}$ est la suite infinie $(F^i(c)(0))_{i \in \mathbb{N}}$. Cette notion a été formalisée et étudié par J. CERVELLE, E. FORMENTI et P. GUILLON dans [10].

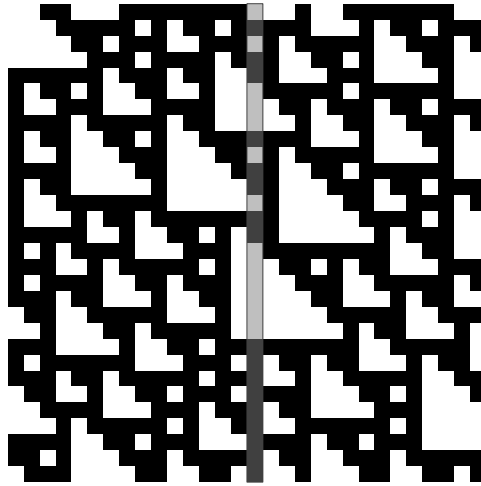


Fig. 1.7 – Exemple d'orbite de l'automate cellulaire 110 faisant apparaître la trace en grisé

Avec P. GUILLON, nous avons obtenu une caractérisation de la nilpotence

au travers de l'observation de l'ensemble des traces possibles. Une trace est *q-nilpotente* si elle appartient à l'ensemble Q^*q^ω .

Théorème 1.1.8 (P. GUILLON and G. R., 2008 [1]). *Un automate cellulaire (Q, f) est nilpotent si et seulement si, il existe un état $q \in Q$ tel que toutes ces traces sont q nilpotentes.*

Même si ce résultat semble très distinct des autres points présentés dans ce mémoire, sa preuve repose sur le point de vue bidimensionnel qui est celui adopté dans ce mémoire et il utilise des méthodes présentées dans la suite.

Démonstration. Le sens direct est évident. Soit (Q, r, δ) un automate cellulaire dont toutes les traces sont q -nilpotentes pour un état $q \in Q$. Par l'absurde, on suppose que l'automate n'est pas nilpotent pour construire une configuration dont la trace n'est pas q -nilpotente.

Comme toutes les traces de l'automate sont nilpotentes, on observe l'apparition des mots q^{2k+1} dans un intervalle de temps uniformément borné (cf. Fig. 1.8a). Formellement, pour tout $k \in \mathbb{N}$, il existe $J_k \in \mathbb{N}$ tel que pour toute configuration $c \in Q^{\mathbb{Z}}$ et pour toute position $i \in \mathbb{Z}$, $q^{2k+1} \in \{F^j(c)_{[i-k, i+k]} \mid 0 \leq j \leq J_k\}$ où $c|_I$ désigne la restriction d'une configuration finie $c \in Q^{\mathbb{Z}}$ à l'intervalle $I \subset \mathbb{Z}$. Si un tel k n'existe pas, on peut extraire des diagrammes espace-temps des bandes de largeur $2k + 1$ et de longueur arbitraire telles que, sur chaque ligne, il existe au moins un état qui ne soit pas q . Par compacité, on extrait une bande infinie puis une colonne qui contient une infinité d'états n'étant pas q , contredisant la nilpotence des trace.

À l'aide d'arguments du même type, on peut montrer qu'il existe un mot bloquant $b \in Q^*$. Un *mot bloquant* est un élément de $b \in Q^n$ tel que toutes les configurations $c \in Q^{\mathbb{Z}}$ vérifiant $c_{[-n/2, n/2]} = b$ ont la même trace (voir Fig. 1.8b). Comme cette trace est nilpotente, elle est de la forme wq^ω avec $w \in Q^J$ pour $J \in \mathbb{N}$.

Il est alors possible de construire des portions de diagrammes espace-temps avec une bordure de q et de les « empiler » à la façon de poupées gigognes. Cette opération prenant appui sur la caractérisation bidimensionnelle du diagramme espace-temps à l'aide de pavage qui sera un outils récurrent de ce mémoire.

Pour tout $K \in \mathbb{N}$, comme l'automate n'est pas nilpotent, il existe une configuration $c \in Q^{\mathbb{Z}}$, dont l'orbite contient un état différent de q après $k > 2K + J + J_{J+K} + 1$ étapes — i.e., telle que $F^k(c)(0) \neq q$. Comme la règle de transition est locale, ce point est entièrement déterminé par le mot $c_{[-k, k]}$. De par la propriété d'uniformité d'apparition des motifs de la forme $q^{2K+2J+1}$, il existe $j < J_{J+K}$ tel que $F^j(c)_{[-K-J, K+J]} = q^{2K+2J+1}$. On note $w_0 = F^j(c)_{[-k+j, k-j]}$ (cf. Fig. 1.8c).

On considère la configuration $c' = {}^\omega q b w_0 b q^\omega$. Comme la règle est locale, on a bien $F^{k-j}(c')(0) \neq q$. Comme b est bloquant, on sait que deux colonnes au dessus de ces mots sont fixées et contiennent uniquement des q à partir

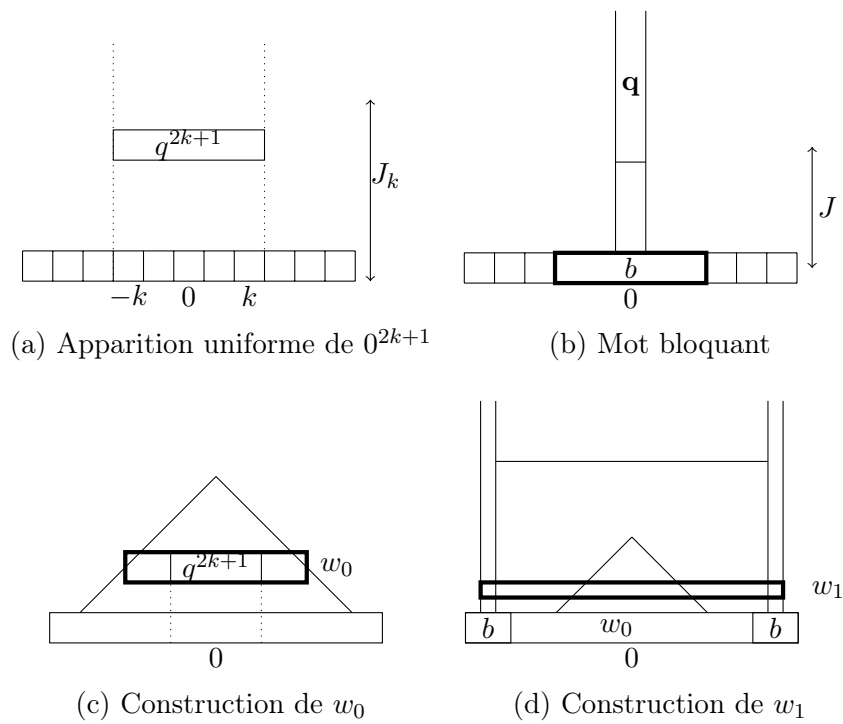


Fig. 1.8 – Équivalence entre nilpotence d'un automate et nilpotence de ces traces

de l'étape J . On pose $w_1 = F^J(c')_{\llbracket -k+j-|b|/2+J, k-j+|b|/2-J \rrbracket}$ (voir Fig. 1.8d). On considère finalement la configuration finie ${}^\omega q w_1 q {}^\omega$. Il est clair que le diagramme espace temps de cette configuration est remplie de q à l'exception d'une bande centrale. Comme cette bande est de largeur finie, la nilpotence des traces impose que cette configuration atteigne la configuration uniforme ${}^\omega q {}^\omega$ en un temps fini $t(k) \in \mathbb{N}$.

La portion n'étant pas dans l'état q du diagramme espace temps forme un pavé de largeur inférieure à k , de hauteur $t(k)$, contenant un vide triangulaire de taille $2K + 1$ en bas au centre et dont au moins un élément sur la colonne centrale est distinct de q . On construit alors, en « empilant » de telles configurations, une configuration dont la trace n'est pas nilpotente (voir Fig. 1.9). ■

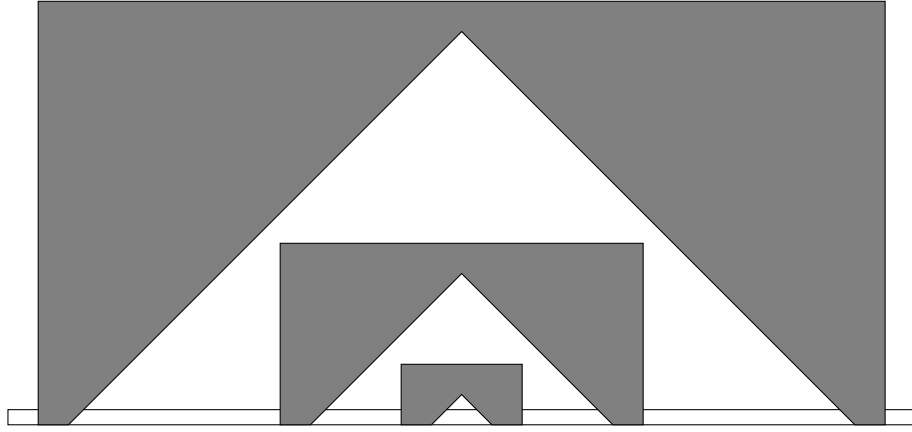


Fig. 1.9 – Empilement de motifs dans un diagramme espace-temps

Les zones blanches indiquent celles remplies avec l'état quiescent

1.2 Particules et collisions

1.2.1 Définitions

À l'aide d'une approche bidimensionnelle, nous proposons des définitions géométriques simples pour les objets récurrents aperçus dans la partie précédente. Ces définitions sont données de façon descriptive et se basent sur l'ensemble des colorations. Ces objets sont regroupés dans la figure 1.10.

Le premier élément, représenté dans la figure 1.10a est le fond qui se retrouve souvent dans de grandes portions de diagramme espace-temps et sert de « support » aux autres objets. Comme cet élément est de dimension 2, il est facilement définissable à l'aide des axes de périodicité.

Définition 1.2.1. *Un fond (Fig. 1.11a) est une coloration possédant deux périodes non colinéaires.*

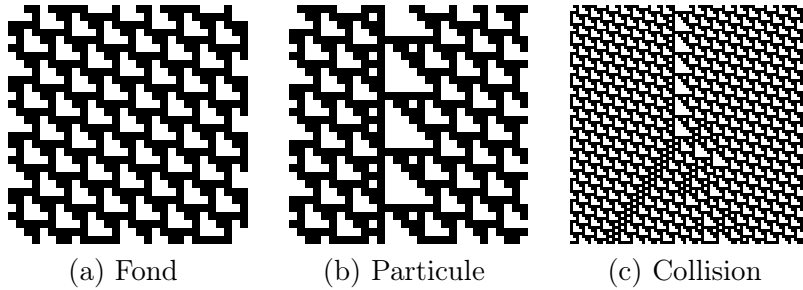


Fig. 1.10 – Exemple de fond, particule et collision de l'automate cellulaire 110

On notera en particulier que, comme la dimension de l'espace est deux, le fait d'avoir deux vecteurs de périodicité implique que la coloration est périodique suivant n'importe quelle direction.

Le deuxième élément est un élément de dimension 1 qui est présent entre deux fonds : la particule (voir la figure 1.10b). Pour le caractériser, il suffit de remarquer qu'en plus de l'axe de périodicité, la présence de deux fonds impose que suivant les autres axes, le mot obtenu est ultimement périodique.

Définition 1.2.2. Une particule (Fig. 1.11b) est une coloration $c \in Q^{\mathbb{Z}^2}$ caractérisée par deux vecteurs non colinéaires $u, v \in \mathbb{Z}^2$ vérifiant que u est un axe de périodicité de c et que pour tout $p \in \mathbb{Z}^2$, le mot $(c(p + iv))_{i \in \mathbb{Z}}$ est ultimement périodique.

De même que dans le cas précédent, La définition de particule induit que pour tout vecteur $w \in \mathbb{Z}^2$ et tout point $p \in \mathbb{Z}^2$, le mot extrait à partir de p et suivant w est ultimement périodique. Comme périodique implique ultimement périodique, un fond est une particule. Pour distinguer les « véritables » particules, une particule *propre* est une particule qui n'est pas un fond.

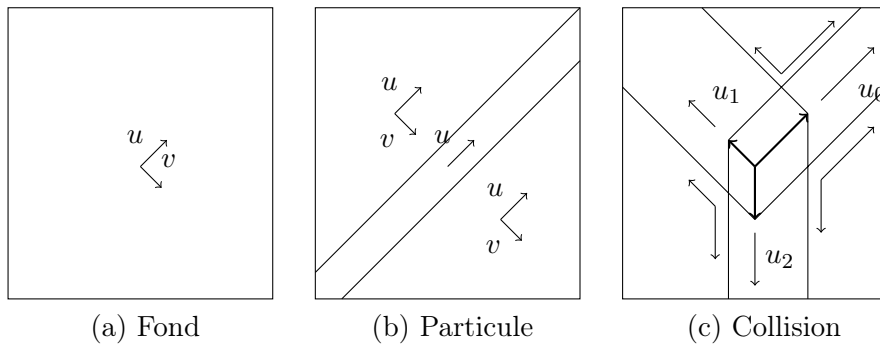


Fig. 1.11 – Représentations symboliques des fonds, particules et collisions

Le dernier élément, la collision (voir figure 1.10c), est obtenu lorsque deux ou plusieurs particules se rencontrent en un point et se détruisent ou génèrent

une ou plusieurs nouvelles particules. Sur le diagramme espace-temps, cet objet zéro dimensionnel se caractérise par une forme d'étoile. Pour le définir, il faut introduire la notion de *cône* $\triangleleft_v(u, u')$ (voir figure 1.12) entre deux vecteurs $u, u' \in \mathbb{Z}^2$ au point $v \in \mathbb{Z}^2$ comme l'ensemble des points contenus dans le cône issu de v et situé dans l'angle formé par les droites induites par u et u' .

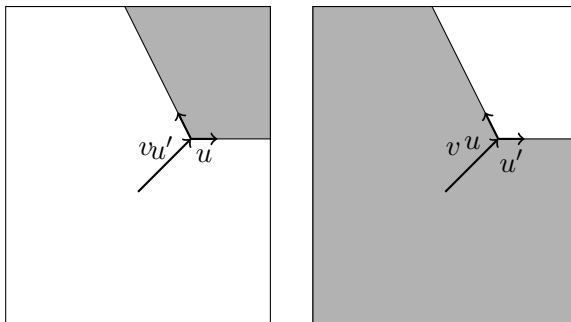


Fig. 1.12 – Exemples de cône $\triangleleft_v(u, u')$

Définition 1.2.3. Une collision (Fig. 1.11c) est une coloration $c \in Q^{\mathbb{Z}^2}$ caractérisée par une liste de vecteurs $(u_i)_{i \in \mathbb{Z}_m}$ dont deux vecteurs consécutifs ne sont pas colinéaires et tels que, pour tout $i \in \mathbb{Z}_m$, c est périodique de période u_i dans le cône $\triangleleft_{u_i}(u_{i-1}, u_{i+1})$, i.e. pour tout $p \in \triangleleft_{u_i}(u_{i-1}, u_{i+1})$, $c(p + u_i) = c(p)$.

Même si la définition fait intervenir de nouveaux objets géométriques, une particule est un cas particulier de collision : comme la particule possède un vecteur de périodicité, le défaut sur l'autre axe est borné et donc il existe un $k \in \mathbb{Z}^*$ tel que la liste $(u, kv, -u, -kv)$ caractérise la particule en tant que collision. Comme dans le cas des particules, une collision est *propre* si ce n'est pas une particule.

1.2.2 Propriétés

Lors de la définition de collision, l'utilisation de la figure 1.11c peut être trompeuse : cette figure possède, en effet, des propriétés particulières qui ne sont pas imposées par la définition. En particulier, sur cette figure, seuls deux cônes correspondant à des vecteurs successifs se superposent. Cette condition sera désignée sous le nom de *caractérisation normale*. Un premier résultat sur les collisions est que l'on peut toujours supposer l'existence d'une telle caractérisation.

Remarque 1.2.4. Toute collision possède une caractérisation normale.

Démonstration. Soit une collision dont la caractérisation ne possède pas cette propriété. On choisit un cône quelconque de vecteurs u_{k-1}, u_k et u_{k+1} . Soit

un ensemble maximal de cônes consécutifs contenant ce cône et vérifiant que seuls deux cônes successifs se chevauchent, les vecteurs concernés peuvent être choisis de la forme $u_{k-l}, \dots, u_{k+l'}$, comme dans l'exemple de la figure 1.13 ($k-l=4$ et $k+l'=6$). Il est possible de construire une caractérisation normale à l'aide de ces vecteurs.

Pour tout $j \in \llbracket -l+1, l'-1 \rrbracket$, la coloration est, par définition, u_j périodique dans $\triangleleft_{u_j}(u_{j-1}, u_{j+1})$. Le cône $\triangleleft_{u_{k-l}}(u_{k-l-1}, u_{k-l+1})$ (représenté dans l'exemple à l'aide d'un remplissage vertical) intersecte un autre cône que celui ancré en u_{k-l+1} . Comme les vecteurs sont partagés, entre deux cônes successifs, cet autre cône est celui ancré en $u_{k+l'-1}$ entre $u_{k+l'-2}$ et $u_{k+l'}$. On en déduit donc que le cône $\triangleleft_{u_{k-l}}(u_{k-l-1}, u_{k-l+1})$ est inclus dans $\triangleleft_{u_{k-l}}(u_{k-l'}, u_{k-l+1})$ (sur l'exemple, celui rempli à l'aide de lignes horizontales) et donc vérifie la propriété de périodicité. Il en va de même pour le cône $\triangleleft_{u_{k+l'}}(u_{k+l-1}, u_{k-l})$. ■

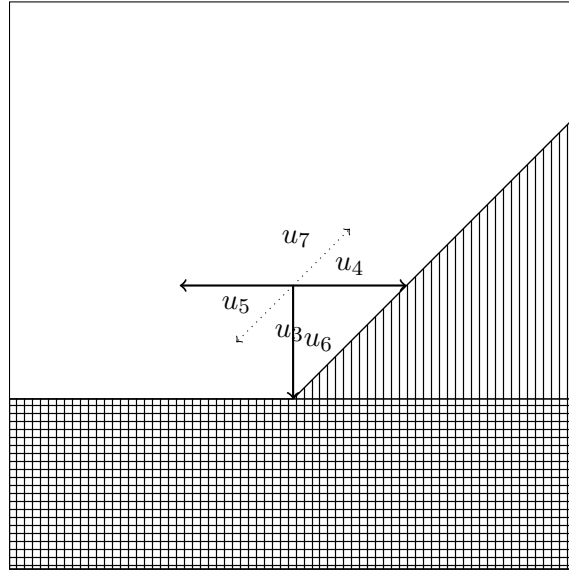


Fig. 1.13 – Exemple de caractérisation normale d'une collision

Cette caractérisation normale permet d'introduire une autre particularité des caractérisations de collision qui est l'ajout de vecteurs « fantômes ». Soit une caractérisation normale $(u_i)_{i \in \mathbb{Z}_m}$ d'une collision et v un vecteur alors il existe $j \in \mathbb{Z}_m$ et $k \in \mathbb{N}^+$ tel que $(u_0, \dots, u_j, kv, u_{j+1}, \dots, u_{m-1})$ est une caractérisation normale de la collision. Pour prouver cela, il suffit de remarquer que l'introduction d'un nouveau vecteur entraîne la réduction de taille de deux cônes et l'apparition de deux nouveaux (voir figure 1.14). Les deux nouveaux cônes étant inclus dans un fond qui est périodique selon toute direction par une remarque précédente, il vérifient bien la propriété de périodicité.

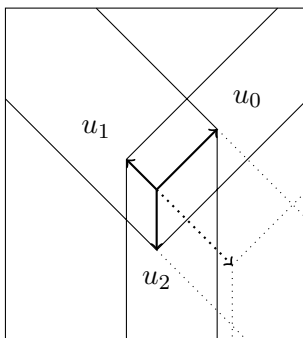


Fig. 1.14 – Introduction de particule fantôme

1.2.3 Généralité des définitions

Les définitions formelles de la partie précédentes introduisent un choix restrictif sur l'ensemble des objets considérés. Ce choix est orienté par des objets couramment observés dans les diagrammes espace-temps d'automates cellulaires qui rentrent *de facto* dans le cadre de nos définitions. La question qui se pose alors est celle de la portée des définitions vis-à-vis des autres objets présents dans les automates cellulaires.

Un premier objet — autre que fond, particule ou collision — que l'on rencontre dans la littérature est celui qui génère de façon périodique des particules : le *gun* (voir Fig. 1.15). Comme cet objet et les particules qu'il génère, est périodique, il rentre dans le cadre de la définition de particule. Néanmoins, cet objet fait apparaître une ambiguïté dans les définitions : une succession de particules régulièrement espacées peut-être vue comme un fond et le *gun* force à prendre cette vision. De ce fait, même si de telles structures restent dans le cadre des définitions, elles peuvent engendrer un changement de point de vue potentiellement néfaste à une bonne utilisation algorithmique. Sur ce point, la question de la possibilité future d'ajouter ces objets en tant qu'objet de première classe mérite d'être soulevée.

De nombreux objets sont également générés lors de constructions par les personnes étudiant les signaux. En particulier, il est possible de construire et considérer des signaux courbes (par exemple une parabole sur la figure 1.16). Ces constructions s'opposent à l'idée d'objets rectilignes reposant sur la périodicité. Néanmoins, ces constructions peuvent souvent être vues au travers de nos définitions comme étant composées de nombreuses particules et collisions interagissant entre elles mais il est impossible « d'effacer » les traits de constructions pour ne conserver qu'une particule courbe. Pour le moment, ces aspects ne semblent pas engendrer de réelles restrictions au niveau des constructions. Cependant, ils proposent une ouverture ultérieure possible au niveau des définitions.

Ces différentes remarques montrent que la portée des définitions propo-

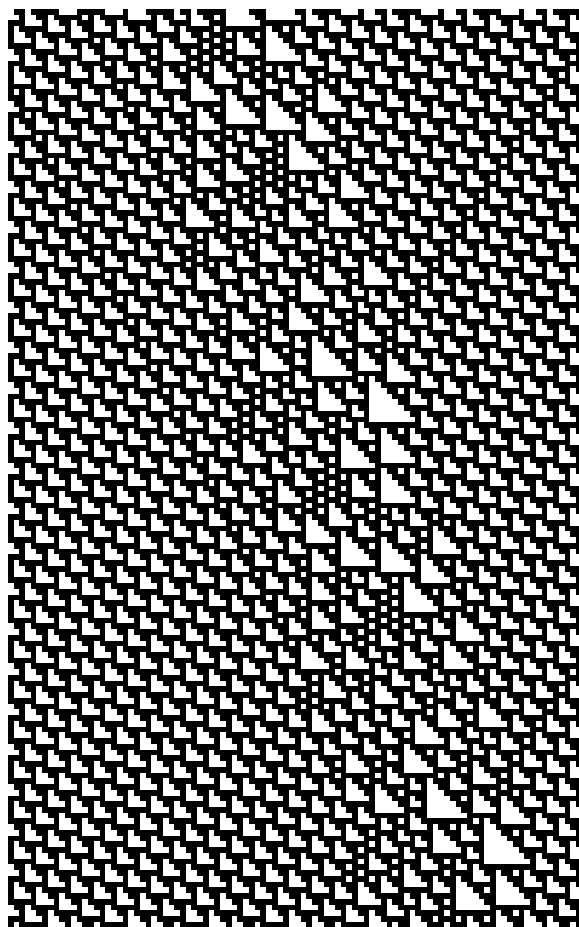


Fig. 1.15 – Exemple de « gun »

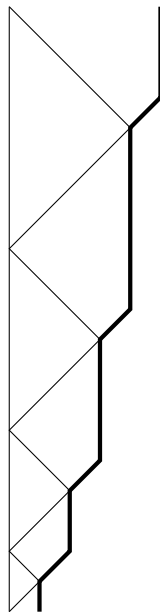


Fig. 1.16 – Construction d'un signal parabolique

sées est loin d'être négligeable et englobe de nombreux cas connus. Bien sûr, il existe d'autres objets échappant à ces définitions et il serait vain d'essayer d'englober tous les comportements possibles. En particulier, la plupart des phénomènes de synchronisation sortent du cadre de ce mémoire.

Pour en conclure avec la portée des définitions, une ouverture est de voir s'il existe des classes d'automates dans lesquels ces notions prennent une dimension particulière. Une piste prometteuse dans cette direction est l'étude de la classe des automates cellulaires réversibles, *i.e.* dont la fonction globale est inversible, pour lesquels toutes les orbites sont des diagrammes espace-temps.

1.3 Robustesse des définitions

Bien que reposant sur une intuition commune à de nombreuses personnes, les définitions de fond, particule et collision semblent faire intervenir une part importante d'arbitraire. Cette section est destinée à démontrer qu'il n'en est rien. En introduisant une notion d'automate à compteurs sur le plan, il est en effet possible de montrer que les fonds, particules et collisions sont des définitions robustes en les liant avec des coloriage « simples » du plan.

Intuitivement, il est possible d'interpréter les caractérisations obtenues de la façon suivante : le théorème 1.3.8 indique que les fonds sont les colorations dont l'information est finie tandis que le théorème 1.3.9 caractérise les particules comme les colorations dans lesquelles l'information est limi-

tée à une variable qui prend alors la forme d'une distance. Dans le cas des collisions, le théorème 1.3.11 montre qu'elles sont des colorations contenant une information paramétrée par deux variables alors que le théorème 1.3.12 montre que la réciproque est vraie si l'on exige que l'information permette de distinguer les points du plan. Intuitivement, il est possible de reformuler la condition précédente en demandant que l'information soit « visible » suivant les deux dimensions de la coloration. Ces résultats font l'objet d'une note soumise à TCS [4].

1.3.1 Automates de cartes

En dimension 1, la robustesse des objets simples peut être caractérisée de multiples façons différentes : machines finies [39], langages formels [11, 33], logique monadique du second ordre [8] . . . En dimension 2, ces approches ne menant plus à une notion commune [24], il faut en choisir une particulière. Pour notre cas, le choix effectué consiste à étendre l'approche par automate fini (ou à compteurs) en introduisant une extension pour une présentation de monoïde (le plan en étant une) : les automates de cartes.

On regarde alors le plan comme une présentation finie de monoïde. Une *présentation finie de monoïde* est une paire (G, R) où G est un alphabet fini et $R \subseteq G^* \times G^*$ est l'ensemble des *relateurs*. Le *monoïde* $\mathcal{G} = \langle G, R \rangle$ associé à la *présentation* est le plus grand monoïde qui respecte les relations, *i.e.*, pour tout $(u, v) \in R$, $u = v$ dans ce monoïde.

En particulier la ligne peut être vue comme la présentation de monoïde

$$(\mathbb{Z}, +) = \langle g, d \mid gd = dg = \epsilon \rangle \quad ,$$

et le plan comme

$$(\mathbb{Z}^2, +) = \langle n, s, e, o \mid ns = sn = eo = oe = \epsilon, ne = en \rangle \quad .$$

On introduit l'ensemble des *valeurs de test* $\Upsilon = \{0, +\}$ et l'ensemble des *actions* $\Phi = \{-, 0, +\}$. Le *test* $\tau : \mathbb{N} \rightarrow \Upsilon$ est défini par $\tau(i) = 0$ si et seulement si $i = 0$. Pour tout $k \in \mathbb{N}$, et tout $\phi \in \Phi^k$, la *modification* $\theta_\phi : \mathbb{N}^k \rightarrow \mathbb{N}^k$ est définie, pour tout $v \in \mathbb{N}^k$ et $i \in \llbracket 0, k-1 \rrbracket$, par :

$$\theta_\phi(v)(i) = \begin{cases} \max(0, v(i) - 1) & \text{si } \phi(i) = - \\ v(i) & \text{si } \phi(i) = 0 \\ v(i) + 1 & \text{si } \phi(i) = + \end{cases}$$

Un *automate* $\mathcal{A}$ à k compteurs sur un alphabet G est un quintuplet (G, k, Q, q_0, γ) où $k \in \mathbb{N}$ est le nombre de compteurs, Q est l'ensemble fini d'états, $q_0 \in Q$ est l'état initial et $\gamma : Q \times \Upsilon^k \times G \rightarrow Q \times \Phi^k$ la fonction de transition. Cette fonction peut être étendue en une *action* f de G sur $Q \times \mathbb{N}^k$ par $f(q, v, \epsilon) = (q, v)$ et $f(q, v, za) = (q'', \theta_\phi(v'))$ où $f(q, v, z) = (q', v')$ et

$\gamma(q', \tau(v'), a) = (q'', \phi)$ pour tout $q \in Q$, $v \in \mathbb{N}^k$, $z \in G^*$ et $a \in G$. Un *automate de carte à k compteurs sur une présentation de monoïde* $\mathcal{G} = \langle G, R \rangle$ (noté k -ACC) est un automate de carte à k compteurs sur l'alphabet G dont l'action est compatible avec la structure du monoïde. Plus formellement, pour tout $z \in G^*$, $z', z'' \in R$, $f(q_0, 0, zz') = f(q_0, 0, zz'')$.

Sous cette condition, on peut transférer l'action obtenue sur $\mathcal{G}$ en une *cartographie* $g : \mathcal{G} \rightarrow Q \times \mathbb{N}^k$, pour tout $z \in \mathcal{G}$, par $g(z) = f(q_0, 0, z)$. Pour un automate $\mathcal{A}$, la *hauteur de compteur minimale* (resp. *maximale*) est définie par $\min_p(z) = \min_{i \in \mathbb{Z}_k} v_i$ (resp. $\max_p(z) = \max_{i \in \mathbb{Z}_k} v_i$) où $g(z) = (q, v_i)$. Deux points distincts $z, z' \in \mathcal{G}$ sont *indistinguishables* par $\mathcal{A}$ si $g(z) = g(z')$. Comme les transitions ne dépendent que du fait que les compteurs soient vides ou non, on appelle *points discriminants* pour $\mathcal{A}$ les éléments $z \in \mathcal{G}$ vérifiant $\min_p(z) = 0$. Pour finir, une composante connexe $Z \subseteq \mathcal{G}$ est dite *indépendante* pour $\mathcal{A}$ si $\{\min_p(z) \mid z \in Z\}$ est un sous ensemble infini (et donc non borné) de $\mathbb{N}^+$. On étend également la notion de périodicité aux automates de la façon suivante : un automate de carte est *périodique* de période $z \in \mathcal{G} \setminus \{\epsilon\}$ si sa cartographie est périodique selon cette période, i.e., $g(z') = g(z'z)$ pour tout $z' \in \mathcal{G}$. Il est alors possible de caractériser la régularité d'un automate de carte à l'aide de sa cartographie :

Lemme 1.3.1. *Un k -ACC $\mathcal{A}$ sur un groupe $\mathcal{G}$ est périodique si et seulement s'il admet deux éléments indistinguishables par $\mathcal{A}$.*

Démonstration. Soit $\mathcal{A}$ un k -ACC sur un groupe $\mathcal{G}$. S'il est périodique de période $z \in \mathcal{G}$ alors ϵ et z sont indistinguishables par $\mathcal{A}$. Réciproquement, soit $z, z' \in \mathcal{G}$ deux points indistinguishables par $\mathcal{A}$ alors $z' - z$ est un vecteur de périodicité de $\mathcal{A}$. ■

Soit Σ un ensemble fini. Pour plonger la cartographie de $\mathcal{A}$ dans l'espace des colorations $\Sigma^{\mathbb{Z}^2}$, on introduit une fonction de *coloriage* $\phi : Q \rightarrow \Sigma$ que l'on étend sur $Q \times \mathbb{N}^k$ par, pour tout $(s, v) \in Q \times \mathbb{N}^k$, $\phi(s, v) = \phi(s)$. La *coloration engendrée* par $\mathcal{A}$ selon ϕ est la coloration $c \in \Sigma^{\mathcal{G}}$ définie par, pour tout $z \in \mathcal{G}$, $c(z) = \phi(g(z))$. On appelle $\mathcal{G}$ - k -*peinture* une coloration engendrée par un k -ACC sur $\mathcal{G}$.

Pour caractériser l'ensemble des $\mathcal{G}$ - k -peintures, on introduit la *translation* d'une coloration $c \in Q^{\mathcal{G}}$ par un vecteur $z \in \mathcal{G}$ comme étant la coloration $c_z \in Q^{\mathcal{G}}$ définie par $c_z(z') = c(zz')$ pour tout $z' \in \mathcal{G}$.

Lemme 1.3.2. *Pour tout groupe $\mathcal{G}$ et tout entier k , l'ensemble des $\mathcal{G}$ - k -peinture est stable par translation.*

Démonstration. Soit c une coloration engendrée par un k -ACC $(\mathcal{G}, k, Q, q_0, \gamma)$ selon ϕ . On considère un élément $z \in \mathcal{G}$ et on note $m_z = \max_c(z)$. Afin d'introduire une portion de l'information des compteurs dans les états, on introduit les fonctions $b : \mathbb{N} \rightarrow \llbracket 0, m_z \rrbracket$ et $t : \mathbb{N} \rightarrow \mathbb{N}$ définies par, pour tout $n \in \mathbb{N}$, $b(n) = \min(m_z, n)$ et $t(n) = \max(0, n - m_z)$. On étend de

façon canonique ces fonctions sur $\mathbb{N}^k$. On définit l'encodage $e : Q \times \mathbb{N}^k \rightarrow Q \times \llbracket 0, m_z \rrbracket^k \times \mathbb{N}^k$ par $e(q, v) = ((q, b(v)), t(v))$ pour tout $(q, v) \in Q \times \mathbb{N}^k$. Soit $\mathcal{A}'$ le k -ACC $(\mathcal{G}, k, Q \times \llbracket 0, m_z \rrbracket, q'_0, \gamma')$ construit de tel sorte que, sa cartographie g' vérifie $g'(z) = e(g(z))$ pour tout $z \in \mathcal{G}$ (en particulier, q'_0 est la première composante de $e(g(z))$). Le translaté c_z de c par z est la coloration engendrée par $\mathcal{A}'$ selon la fonction de coloriage $\phi' : Q \times \llbracket 0, m_z \rrbracket^k \rightarrow \mathbb{S}$ définie par $\phi'(q, t) = \phi(q)$ pour tout $(q, t) \in Q \times \llbracket 0, m_z \rrbracket^k$ ■

Une autre propriété importante des peintures est que, comme le comportement d'un automate de carte ne dépend pas de la hauteur de la pile si celle-ci est non vide, les colorations engendrées présentent de très fortes régularités :

Lemme 1.3.3. *Soit c une coloration engendrée par un k -ACC $\mathcal{A}$ sur un groupe $\mathcal{G}$. Soit $Z \subseteq \mathcal{G}$ une composante connexe indépendante pour $\mathcal{A}$. Dans ce cas, il existe une $\mathcal{G}$ -0-peinture c' vérifiant $c|_Z = c'|_Z$.*

Démonstration. Soit $\mathcal{A}$ un k -ACC $(\mathcal{G}, k, Q, q_0, \gamma)$ et $Z \subseteq \mathcal{G}$ une composante connexe indépendante. Soit un état de départ q'_0 présent dans Z avec des tailles de compteurs quelconque. Pour tout $q \in Q$, on note Z_q l'ensemble des points dont l'image par la cartographie contient l'état q (i.e. $Z_q = \{z \in Z \mid g(z) = (q, v), v \in \mathbb{N}^k\}$). Comme Q est fini et Z indépendante, il existe un q'_0 tel que $\{\min_p(z) \mid z \in Z_{q'_0}\}$ est infini. Considérons le 0-ACC $\mathcal{A}' = (\mathcal{G}, 0, Q, q'_0, \gamma)$ obtenu en « oubliant » les compteurs ; c'est-à-dire $\gamma'(q, a) = \gamma(q, +^k, a)$ pour tout $q \in Q$ et $a \in G$. Avant toute chose, il faut prouver que $\mathcal{A}'$ est bien un 0-ACC. Soit $z \in G^*$ et $(z', z'') \in R$. On note N la longueur maximale des chemins zz' et zz'' (formellement, $N = \max(|zz'|, |zz''|)$). Par construction, on peut choisir un $z_N \in Z_{q'_0}$ vérifiant $\min_c(z_N) > N$ ce qui implique l'absence de point discriminant sur le chemin de z_N à $z_N z z'$. On en déduit $\gamma'(q_0, z z') = q$ où $(q, n) = \gamma(g(z_N), z z')$. Il est en de même pour $z z''$. Il est alors possible de conclure en prenant n'importe quel élément $z_0 \in Z_{q'_0}$. Par l'indépendance de Z , $c'(z) = c(z_0 z)$ pour tout $z \in Z$. Il ne reste plus alors qu'à appliquer le lemme 1.3.2. ■

Les $\mathcal{G}$ - k -peintures définissent donc une hiérarchie croissante suivant k de colorations sur une présentation de monoïde $\mathcal{G}$.

1.3.2 Cas de $\mathbb{Z}$

En regardant $\mathbb{Z}$ comme le monoïde

$$(\mathbb{Z}, +) = \langle g, d \mid gd = \epsilon, dg = \epsilon \rangle \quad ,$$

muni du morphisme canonique $d = 1$, il est possible de montrer que les $\mathbb{Z}$ -0-peintures et les $\mathbb{Z}$ -1-peintures correspondent respectivement aux configurations périodique et ultimement périodiques.

Proposition 1.3.4. *Les $\mathbb{Z}$ -0-peintures sont exactement les configurations périodiques de $\mathbb{Z}$.*

Démonstration. Soit la cartographie $g : \mathbb{Z} \rightarrow Q$ d'un 0-ACC $\mathcal{A}$. Comme Q est fini, il existe deux éléments indistinguables par $\mathcal{A}$. D'après le lemme 1.3.1, l'automate est périodique.

Réciproquement, soit c une coloration périodique de $\mathbb{Z}$ de période $u \in \mathbb{Z}^+$. On considère l'automate de carte à 0 compteur $\mathcal{A} = (\mathbb{Z}, 0, Z_u, 0, \delta)$ avec $\delta(i, g) = (i - 1)$ et $\delta(i, d) = (i + 1)$. Ainsi, c est la configuration engendrée par $\mathcal{A}$ selon la fonction de coloriage $\phi : i \mapsto c(i)$. ■

Proposition 1.3.5. *Les $\mathbb{Z}$ -1-peintures sont exactement les configuration ultimement périodique de $\mathbb{Z}$.*

Démonstration. Soit la cartographie $g : \mathbb{Z} \rightarrow Q \times \mathbb{N}$ d'un 1-ACC $\mathcal{A}$. Comme les 0-ACC sont des 1-ACC, il est possible, sans perte de généralité, de se restreindre au cas g injectif, i.e., sans points indistinguables. Dans ce cas, il existe $k \in \mathbb{N}$ tel que $g^{-1}(Q \times \{0\}) \subseteq \llbracket -k, k \rrbracket$. Par le lemme 1.3.1, g est périodique sur $\llbracket -\infty, -k \rrbracket$ et $\llbracket k, +\infty \rrbracket$. Il s'en suit que c est ultimement périodique.

Réciproquement, soit c une configuration ultimement périodique de $\mathbb{Z}$ de période $u \in \mathbb{Z}^+$ et de défaut $k \in \mathbb{N}$. On considère le 1-ACC sur l'alphabet $\llbracket -k - u, k + u \rrbracket$ dont la cartographie vérifie, pour tout $i \in \mathbb{Z}$,

$$g(i) = \begin{cases} (i, 0) & \text{si } |i| \leq k + u \\ ((i - k)[u] + k, \lfloor \frac{i-k}{u} \rfloor) & \text{si } i > k + u \\ ((i + k)[u] - k - u, \lfloor \frac{k-i}{u} \rfloor) & \text{si } i < -k - u \end{cases}$$

La configuration engendrée par $\mathcal{A}$ selon le coloriage $\phi : i \mapsto c(i)$ est c . ■

À partir de deux compteurs, on peut difficilement espérer obtenir des colorations simples : Un automate fini à deux compteurs peut simuler n'importe quelle machine de Turing (voir M. MINSKY [39]). K. MORITA [41] a ensuite montré qu'il est suffisant de se restreindre aux automates réversibles. De ce fait, il n'est guère surprenant que n'importe quel langage récursivement énumérable puisse être encodé dans de telles machines et par analogie dans les $\mathbb{Z}$ -2-peintures.

Proposition 1.3.6. *Il existe des $\mathbb{Z}$ -2-peintures $\emptyset'$ -complètes.*

Idée de la preuve. Soit K un langage $\emptyset'$ -complet contenant 0. Soit $(n_i)_{i \in \mathbb{N}}$ une énumération calculable des éléments de K vérifiant $n_0 = 0$. Il existe une fonction calculable injective qui sur l'entrée n_i calcule n_{i+1} . En combinant la construction de K. MORITA [41] avec les techniques présentées dans [32], il est possible de construire un automate fini réversible à deux compteurs sur l'espace d'états $S_A \cup \{s_\alpha, s_\omega\}$ tel que, pour tout $i \in \mathbb{N}$, partant de la configuration $(s_\alpha, (n_i, 0))$ l'évolution de l'automate s'arrête dans la configuration

$(s_\omega, (n_{i+1}, 0))$. Il est également possible de construire un automate fini à deux compteurs réversible sur l'ensemble d'états $S_B \cup \{s_\alpha, s_\omega\}$ qui, pour tout $i \in \mathbb{N}$, partant de la configuration $(s_\omega, (n_i, 0))$ atteint et s'arrête dans la configuration $(s_\alpha, (n_i, 0))$ après exactement $2n_i$ étapes. L'union disjointe de ces deux automates est un 2-ACC $\mathcal{A}$ sur l'ensemble d'état $S = S_A \cup S_B \cup \{s_\alpha, s_\omega\}$ qui simule le fonctionnement des deux machines. La fonction de transition est définie par l'application de la transition de l'automate fini à compteur sur le générateur d et par son inverse sur le générateur g . Si on applique la fonction de coloriage $\phi : S \rightarrow \{0, 1\}$ vérifiant, pour tout $s \in S$, $\phi(s) = 1$ si et seulement si $s \in S_B$; alors la coloration c de $\mathcal{A}$ par ϕ contient le facteur $01^{2n}0$ si et seulement si $n \in K$. ■

1.3.3 Cas de $\mathbb{Z}^2$

En assimilant $\mathbb{Z}^2$ au monoïde

$$(\mathbb{Z}^2, +) = \langle n, s, e, o \mid ns = \epsilon, sn = \epsilon, eo = \epsilon, oe = \epsilon, ne = en \rangle \quad ,$$

munit du morphisme canonique $e = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ et $n = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, il est possible de faire mettre en relation $\mathbb{Z}^2$ - k -peintures et fond, particules et collisions. Les démonstrations de ces propriétés tirent partie des liens entre $\mathbb{Z}$ et $\mathbb{Z}^2$ qui transpirent sur les peintures :

Lemme 1.3.7. *Chaque ligne (resp. colonne) d'une $\mathbb{Z}^2$ - k -peinture est une $\mathbb{Z}$ - k -peinture.*

Démonstration. Soit une $\mathbb{Z}^2$ - k -peinture c , la restriction de c à $\{0\} \times \mathbb{Z}$ (resp. $\mathbb{Z} \times \{0\}$) est une $\mathbb{Z}$ - k -peinture. Le résultat peut s'étendre sur n'importe quelle colonne (resp. pile) par application du lemme 1.3.2. ■

En combinant ce lemme et les résultats obtenu dans la section précédente, il est alors possible de caractériser finement les peintures de $\mathbb{Z}^2$.

Théorème 1.3.8. *Les fonds correspondent aux $\mathbb{Z}^2$ -0-peintures.*

Démonstration. Soit $g : \mathbb{Z}^2 \rightarrow Q$ la cartographie d'un 0-ACC $\mathcal{A}$. Comme Q est fini, il existe deux éléments indistinguables par $\mathcal{A}$. De ce fait, la coloration n'est constituée que d'un nombre fini de lignes ou colonnes. En appliquant le lemme 1.3.7 et la proposition 1.3.4, on conclut que chacune est périodique.

Réciproquement, soit c une coloration de $\mathbb{Z}^2$ bipériodique. En toute généralité, on peut supposer que $\begin{pmatrix} m \\ 0 \end{pmatrix}$ et $\begin{pmatrix} 0 \\ n \end{pmatrix}$ sont des périodes. Soit le 0-ACC $\mathcal{A} = (\mathbb{Z}^2, 0, \mathbb{Z}_m \times \mathbb{Z}_n, 0, \delta)$ où $\delta(e)(x, y) = (x + 1, y)$, $\delta(n)(x, y) = (x, y + 1)$. c est la coloration engendrée par $\mathcal{A}'$ selon le coloriage $\phi : (x, y) \mapsto c(x)(y)$ pour tout $x, y \in \mathbb{Z}_m \times \mathbb{Z}_n$. ■

Théorème 1.3.9. *Les particules correspondent aux $\mathbb{Z}^2$ -1-peintures.*

Démonstration. Soit c une $\mathbb{Z}^2$ -1-peinture. Par l'absurde, si c n'est pas périodique, le lemme 1.3.7 permet d'en déduire que toutes les lignes de c sont des $\mathbb{Z}$ -1-peintures non périodiques. Le lemme 1.3.1 impose que chaque ligne contienne au moins un élément discriminant, ce qui contredit la finitude de $Q \times \{0\}$. On en déduit c est composé d'un nombre fini de lignes (ou colonnes) qui sont des $\mathbb{Z}$ -1-peintures. La proposition 1.3.5 permet de conclure que c est une particule.

Réciproquement, soit c une particule de période $u = \begin{pmatrix} x \\ y \end{pmatrix}$. Quitte à échanger les axes, on peut supposer $y > 0$. De par le lemme 1.3.7, c est composé d'un nombre fini de lignes ultimement périodiques. En toute généralité, toutes les lignes ont une période $u \in \mathbb{N}^+$ égale au défaut et vérifiant $u > x$. En appliquant la même méthode que dans la proposition 1.3.5 au bloc entier de lignes, on construit $\mathcal{A}$ le 1-ACC sur l'alphabet $\llbracket -2u, 2u \rrbracket \times \mathbb{Z}_y$ dont la cartographie vérifie $g(i, j) = \tilde{g}(i + \lfloor \frac{j}{y} \rfloor x, j[y])$ pour tout $i, j \in \mathbb{Z}$ où :

$$\tilde{g}(\tilde{i}, \tilde{i}) = \begin{cases} ((\tilde{i}, \tilde{i}), 0) & \text{si } |\tilde{i}| \leq 2u \\ ((\tilde{i}[u] + u, \tilde{i}), \lfloor \frac{\tilde{i}}{u} \rfloor - 1) & \text{si } \tilde{i} > 2u \\ ((\tilde{i}[u] - 2u, \tilde{i}), \lfloor \frac{-\tilde{i}}{u} \rfloor - 1) & \text{si } \tilde{i} < -2u \end{cases}$$

On remarque alors que la coloration engendré par $\mathcal{A}$ selon le coloriage $\phi : (i, j) \mapsto c(i)(j)$ est égale c . ■

Proposition 1.3.10. *Il existe une $\mathbb{Z}^2$ -2-peinture $\emptyset'$ -complète.*

Démonstration. Cette propriété découle directement de la proposition 1.3.6 car il est possible d'étendre de façon périodique n'importe quelle $\mathbb{Z}$ -2-peinture en une $\mathbb{Z}^2$ -2-peinture. Il suffit d'étendre la règle de transition par l'identité sur les générateurs n et s . ■

Ce dernier résultat semble réduire à néant les espoir de mettre en relation les collisions avec des peintures de $\mathbb{Z}^2$. Cependant, un regard sur les peintures complexes construites dans la proposition précédente permet d'en extraire une caractéristique commune : les peintures complexes présentés possède un vecteur de périodicité et ne sont donc pas de « véritables » objets bidimensionnels comme le sont les collisions. Cette observation permet d'obtenir une caractérisation fine des collisions.

Dans la suite, une *boule* de rayon $r \in \mathbb{N}^+$ et de centre $(x, y) \in \mathbb{Z}^2$ qui correspond à l'ensemble $\llbracket x - r, x + r \rrbracket \times \llbracket y - r, y + r \rrbracket$. Par soucis de concision, la mention du centre sera omise si celui-ci est $(0, 0)$.

Théorème 1.3.11. *Toute collision est une $\mathbb{Z}^2$ -2-peinture.*

Démonstration. Soit c une collision. La construction un 2-ACC engendrant cette coloration de fait séparément sur les quatre quarts du plan. On se place dans le cas du quart nord du plan. Pour tout $k \in \mathbb{N}$, le *mot nord de hauteur* k $w(k)$ est défini par $w(k) = c_{\llbracket -k, k \rrbracket \times \{k\}}$. Il existe un entier $K \in \mathbb{N}$ tel que

pour tout $k > K$, $w(k)$ est inclus dans un ensemble de cônes (*i.e.* évite la perturbation centrale). Comme le nombre de cônes est fini et que toute portion dans un cône est périodique, il existe un entier $l \in \mathbb{N}$ multiple de tous les vecteurs de périodicité intervenants dans les cônes rencontrés. On en déduit un entier n et $(2n + 1)l$ mots finis $a_{(l',n')} \in Q^*$ pour $l' \in \mathbb{Z}_l$ et $n' \in \llbracket 0, n \rrbracket$ et $b_{(l',n')} \in Q^*$ pour $l' \in \mathbb{Z}_l$ et $n' \in \llbracket 0, n - 1 \rrbracket$ tels que pour tout $l' \in \mathbb{Z}_l$ et $k \in \mathbb{N}$ vérifiant $kl > K$, on a la caractérisation suivante :

$$w(kl + l') = a_{(l',0)} b_{(l',0)}^k a_{(l',1)} \cdots a_{(l',n-1)} b_{(l',n-1)}^k a_{(l',n)} \quad .$$

Intuitivement, les mots a correspondent aux particules et les mots b correspondent aux fonds. La régularité d'une telle caractérisation tient au fait qu'il est toujours possible de prendre un multiple commun.

À partir de cette caractérisation et en utilisant des techniques similaires à celles employées dans la preuve du théorème 1.3.9, on construit un 2-ACC dont les états sont l'ensemble de toutes les lettres des mots $(2n + 1)l$ et dont la cartographie associe comme état à n'importe qu'elle élément du quart nord la lettre qui lui correspond dans la caractérisation. De plus, la cartographie peut être choisie telle que pour tout élément dans $w(kl + l')$, les compteurs soient $(0, k)$ (resp. $(k, 0)$) pour les lettres des mots $a_{(l',2i)}$ (resp. $a_{(l',2i+1)}$) et $(i, i - k)$ (resp. $(j - k, j)$) pour toutes les lettres de la j -ème répétition du mot $b_{(l',2i)}$ (resp. $b_{(l',2i+1)}$). Intuitivement, les lettres servent à coder dans quelle particule ou fond on se situe et les compteurs donnent la position dans cet élément : la somme des deux compteurs indique la hauteur du mot et dans un fond leur valeur sert à décompter le nombre de répétitions du fond avant de rencontrer la particule suivante. Comme les compteurs ne peuvent être incrémentés ou décrémentés que d'une unité au plus, il est nécessaire d'alterner l'utilisation du premier et de la deuxième compteur.

Une fois cette construction effectuée sur les quatre quarts du plan, il reste à combiner ces constructions. Il suffit alors de remarquer que les contacts entre les différentes constructions ont lieu uniquement sur les diagonales. Les conditions de recollement s'expriment alors sous la forme de l'égalité des quatre valeurs de l (il suffit de prendre un multiple commun) et sur la parité du nombre de particules dans chaque quart de plan (il suffit d'introduire des particules fantômes).

On conclut en remarquant qu'il ne reste plus que les points situés dans la boule de rayon K et que ces points sont en nombre fini. ■

Théorème 1.3.12. *Toute $\mathbb{Z}^2$ -2-peinture apériodique est une collision.*

Démonstration. Soit $\mathcal{A}$ un 2-ACC $(\mathbb{Z}^2, 2, Q, q_0, \gamma)$ apériodique. Dans ce cas, le lemme 1.3.1 induit que sa cartographie g est injective. L'idée intuitive de la preuve est de montrer que la coloration induite est structurée par les éléments discriminants, cette structuration donnant la perturbation ainsi que les particules.

Une première étape est de montrer que ces points ne peuvent pas être isolés. Plus spécifiquement, si une boule $\mathcal{B}$ de centre $c \in \mathbb{Z}^2$ et de rayon $r \in \mathbb{N}$ contient n points discriminants alors la boule $\mathcal{B}'$ de centre c et de rayon $r + |Q| + 1$ en contient au moins $n + 1$ point discriminants.

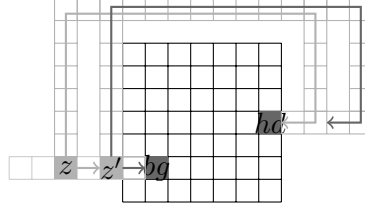


Fig. 1.17 – Chemins pour trouver un nouveau point discriminant

Par l'absurde, on suppose que $\mathcal{B}$ contient n points discriminants et que $\mathcal{B}'$ n'en contient aucun qui ne soit pas dans $\mathcal{B}$. On note par bg (resp. hg , hd) un point discriminant de $\mathcal{B}$ qui est le plus en bas à gauche (resp. en haut à gauche, en haut à droite) comme par exemple dans la figure 1.17. Sans perte de généralité, on peut supposer que les points bg et hd ont tous les deux le premier compteur vide. Parmi les points situés à gauche de bg ($\{bg - \binom{i}{0} \mid i \in \mathbb{N}\}$), la finitude de S implique l'existence de $z, z' \in \mathcal{B}' \setminus \mathcal{B}$ vérifiant $f(z) = (q, (a_0, a_1))$ et $f(z') = (q, (b_0, b_1))$ pour $q \in Q$ et tel que z soit à gauche de z' . Sous ces conditions, le chemin r^{i_0} de z' à bg ne rencontre aucun point discriminant avant bg . Le même chemin emprunté depuis z' vérifiant la même propriété, on en déduit $a_0 - (b_0 - 0) > 0$. Le même raisonnement appliqué au chemin allant de z' à hg dans $\mathcal{B}'$ en évitant $\mathcal{B}$ (voir Fig. 1.17) induit la contradiction $b_0 - (a_0 - 0) > 0$.

On peut alors montrer que ces points discriminants se répartissent entre une boule finie $\mathcal{B}$, et un ensemble fini de demi-droites issues de cette boule.

Soit $\mathcal{B}$ la boule contenant tous les points discriminants dont le compteur non nul est inférieur à $(|S| + 1)^3$. En itérant la propriété précédente, pour tout point discriminant $z \in \mathbb{Z}^2$, il existe une boule de rayon $(|S| + 1)^2$ centrée sur z contenant au moins $|S| + 1$ point discriminants. On en déduit un chemin de taille $(|S| + 1)^3$ contenant $|S| + 1$ point discriminants. Si deux points discriminants ont un compteur vide distinct, z est dans $\mathcal{B}$. Sinon, la finitude de Q impose l'existence de deux éléments z_a et z_b sur cette chaîne vérifiant $f(z_a) = (q, (a, 0))$ et $f(z_b) = (q, (b, 0))$ avec $q \in Q$ et $a, b \in \mathbb{N}^+$, $a < b$. Soit $z \in G^*$ la portion du chemin de z_a à z_b . Comme $f(z_a z^n) = (s, (a - n(b - a), 0))$ pour tout $n \in \mathbb{N}$ tel que $(a - n(b - a)) > (|S| + 1)^3$, on en déduit que z_a est sur une demi-droite issue d'un point de $\mathcal{B}$ et dont la pente est caractérisée par le point dans la boule et par la valeur $b - a$. Ces deux éléments étant en nombre fini, il en est de même pour les demi-droites.

De même que précédemment, on peut montrer que tout point $z \in \mathbb{Z}^2$ vérifiant $\min_c(z) < N$ a au moins un point discriminant à distance inférieure à

$N(|S|+1)$. Ce résultat nous permet de déduire que les composantes connexes entre deux demi-droites consécutives sont indépendantes.

Il ne reste alors plus qu'à ordonner les vecteurs des demi-droites par pente croissante en une liste finie $(u_i)_{i \in \mathbb{Z}_m}$. Quitte à prendre des multiples de u_i , les cônes $\angle_{u_i}(u_{i-1}, u_{i+1})$ évitent la boule centrale $\mathcal{B}$ et ne contiennent que des demi-droites de pente u_i . Dans ces cônes, tous les points (discriminants) dans les demi-droites sont u_i périodiques ; les autres points sont situés dans un sous-ensemble indépendant qui est bipériodique par application du lemme 1.3.1 et du théorème 1.3.8. On en conclut que la coloration est une collision. ■

Chapitre 2

Systèmes symboliques de particules et collisions

CE chapitre est consacré à l'étude des propriétés algorithmiques des particules et collisions. En se basant sur des définitions alternatives et syntaxiques de particule et collision, on introduit une opération syntaxique permettant de manipuler ces objets : le schéma de ligature. L'étude de cette opération amène à la description d'une procédure algorithmique permettant de caractériser les cas où cette opération correspond à une manipulation sur les colorations respectant les contraintes (théorème 2.3.1).

Sommaire

2.1	Particules et collisions syntaxiques	34
2.1.1	Géométrie discrète	34
2.1.2	Patrons	36
2.1.3	Patrons et décidabilité	41
2.2	Ligatures	43
2.2.1	Représentation symbolique	43
2.2.2	Schémas de ligature	44
2.2.3	Pondérations	46
2.3	Validité des schémas de ligature finis	48
2.3.1	Résolution	48
2.3.2	Un exemple complet	52
2.3.3	Expressivité des schémas de ligature finis	54

2.1 Particules et collisions syntaxiques

Cette section présente des définitions syntaxiques pour les particules et collision qui permettent d'obtenir des objets finis et manipulables. Ces définitions prennent appui sur des notions de géométrie discrète présentées dans le cadre des colorations.

2.1.1 Géométrie discrète

Dans toute cette section, on se place dans l'espace $\Sigma^{\mathbb{Z}^2}$. Une *forme* F est un sous-ensemble de $\mathbb{Z}^2$. Un *motif* M qui est une application partielle de $\mathbb{Z}^2$ dans Σ . On note $\text{Sup}(M)$ le *support* du motif. Un motif est *fini* si son support l'est. On remarquera que le support d'un motif est une forme. On note également $M|_F$ la *restriction* du motif M à la forme F .

Dans toute la suite, des variations de la version discrète du théorème de Jordan [34] seront implicitement utilisés. Deux points $\binom{x}{y}, \binom{x'}{y'} \in \mathbb{Z}^2$ sont *4-connectés* si $\binom{|x-x'|}{|y-y'|} \in \left\{ \binom{1}{0}, \binom{0}{1} \right\}$, *8-connectés* si $\binom{|x-x'|}{|y-y'|} \in \left\{ \binom{1}{0}, \binom{0}{1}, \binom{1}{1} \right\}$. Une forme F est *4-connexe*, *resp.* *8-connexe*, si pour toute paire de points $z, z' \in F$, il existe un chemin 4-connecté, *resp.* 8-connecté de points de F entre z et z' (voir figure 2.1). Le théorème de Jordan discret énonce que toute courbe fermée 4-connectée non vide sépare le plan en formes 8-connexes. Plus généralement, une *frontière* est une forme 4-connectée qui sépare le plan en n *bords* 8-connexes.

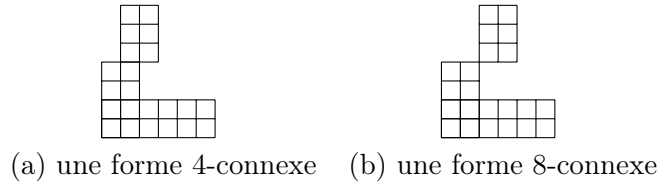


Fig. 2.1 – Connexité, frontières et bords

Le *translaté* d'un motif M par un vecteur $v \in \mathbb{Z}^2$ est le motif $v \cdot M$ défini, pour tout $z \in \text{Sup}(M)$, par $(v \cdot M)(z + v) = M(z)$. De même, pour deux motifs de supports disjoints M et M' , l'*union disjointe* de ces motifs comme étant le motif $M \oplus M'$ suivant :

$$(M \oplus M')(z) = \begin{cases} M(z) & \text{si } z \in \text{Sup}(M) \\ M'(z) & \text{si } z \in \text{Sup}(M') \end{cases}.$$

Dans le cadre des pavages, une *contrainte* est un couple (V, F) où V est une forme finie et F un sous ensemble de V^{Σ} qui correspond à l'ensemble des motifs autorisés. Un motif M *satisfait* la contrainte (V, F) si pour tout vecteur z , si $V \subseteq \text{Sup}(z \cdot M)$ alors $(z \cdot M)|_V \in F$. Un coloriage étant un cas particulier de motif, on peut parler d'un coloriage *satisfaisant une contrainte* ou

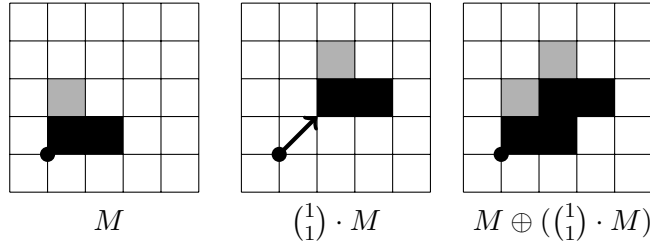


Fig. 2.2 – Motifs et opérations de bases

encore *valide*. Cette définition coïncide avec la notion de diagramme espace-temps valide si on considère la contrainte induite par l'automate cellulaire. Pour une forme F , on appelle *voisinage* de F sous la contrainte (V, F) la forme $\partial F = F \cup \{f + v \mid f \in F, v \in V\}$.

La notion de motif valide est préservée par translation mais pas par union disjointe. Le problème vient du fait que les contraintes impliquent de regarder sur un certain voisinage. Pour obtenir une méthode de construction préservant les contraintes, il est suffisant d'imposer un recouvrement partiel des différents motifs utilisés. On obtient alors un patchwork comme présenté dans la figure 2.3.

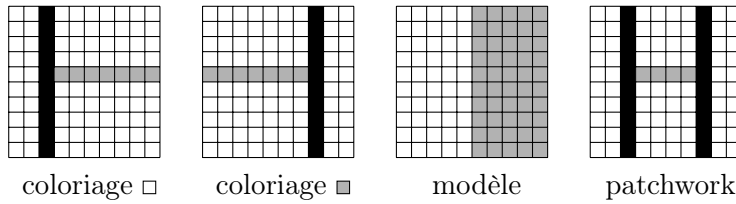


Fig. 2.3 – Patchwork

Définition 2.1.1. Soit $\mathcal{M}$ un ensemble de motifs satisfaisant la contrainte (V, F) . Un patchwork de $\mathcal{M}$ respectant (V, F) est une coloration c définie, pour tout $z \in \mathbb{Z}^2$ par $c(z) = \xi(z)(z)$ où $\xi : \mathbb{Z}^2 \rightarrow \mathcal{M}$ est le modèle du patchwork qui vérifie les conditions :

- pour tout $M \in \mathcal{M}$, $\partial \xi^{-1}(M) \subseteq \text{Sup}(M)$;
- pour tout $z \in \mathbb{Z}^2, v \in V$, $\xi(z)(z + v) = \xi(z + v)(z + v)$.

Lemme 2.1.2. Soit $\mathcal{M}$ un ensemble de motifs satisfaisant la contrainte (V, F) . Tout patchwork c de $\mathcal{M}$ respectant (V, F) est valide.

De ce fait, le notion de patchwork semble adaptée pour construire des colorations valides à partir de motifs valides.

Proposition 2.1.3. Soit c un ensemble de colorations valides. L'ensemble des patchworks sur c est un compact stable par application de l'opération de patchwork.

Démonstration. Soit c un ensemble de colorations valides et $(c_i)_{i \in \mathbb{N}}$ une suite de patchworks. Par compacité de l'espace des colorations, il existe une valeur d'adhérence c qui est une coloration. Soit une sous suite convergente $(c_{\sigma(i)})_{i \in \mathbb{N}}$. On considère le modèle obtenu de la manière suivante : pour chaque point $p \in \mathbb{Z}^2$, la valeur du modèle est celle d'une coloration $c_{\sigma(j)}$ qui coïncide avec c sur le voisinage de p . Le modèle obtenu est celui du patchwork c . ■

L'opération de patchwork définie, se pose alors la question de l'existence d'un ensemble simple de pavages valides dont la clôture par patchwork donne l'ensemble des pavages valides. De par l'indécidabilité de la pavabilité du plan, il n'existe pas de méthode pour construire un tel ensemble à partir des contraintes. Ce résultat conforte l'idée de se restreindre uniquement à un sous-ensemble des pavages valides en se fixant des limites sur l'ensemble dont on regarde la clôture.

2.1.2 Patrons

À l'aide des outils définis précédemment, il est possible de donner des définitions syntaxiques de fonds, particules et collisions. Ces définitions (appelées *patrons*) permettent d'obtenir une description finie et complète de ces objets. De plus, ces définitions coïncident avec celles données dans le chapitre précédemment. Dans toute cette partie, on suppose fixée la contrainte (V, F) .

Définition 2.1.4. Un patron de fond (voir Fig. 2.4a) est un triplet (M, u, v) où M est un motif fini et $u, v \in \mathbb{Z}^2$ deux vecteurs non colinéaires tels que $\mathfrak{F} = \bigoplus_{i,j \in \mathbb{Z}^2} (iu + jv) \cdot M$ est une coloration valide. $\mathfrak{F}$ est appelée coloration associée au patron de fond.

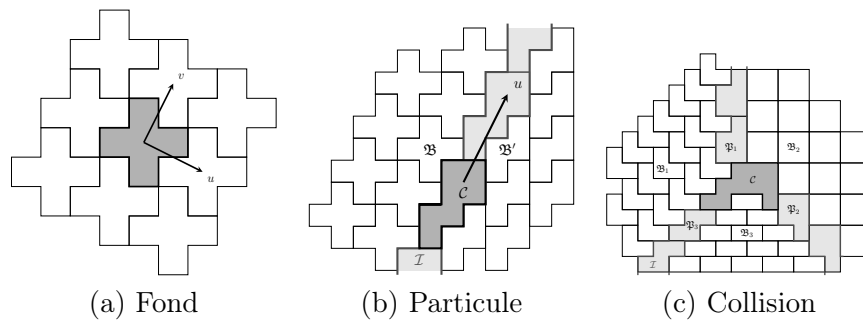


Fig. 2.4 – Patrons

Proposition 2.1.5.

- La coloration associée à un patron de fond est un fond ;
- Pour tout fond $\mathfrak{F}$ il existe un patron de fond dont il est la coloration associée.

Démonstration. Soit $\mathfrak{F}$ une coloration associée à un patron de fond (M, u, v) . Par construction, $\mathfrak{F}$ est périodique selon u et v .

Réciproquement, soit $\mathfrak{F}$ un fond caractérisé par u et v . On considère le quotient de $\mathbb{Z}^2$ par la relation d'équivalence par translation selon u et v (i.e., pour tout $x, y \in \mathbb{Z}^2, x \equiv y$ si et seulement si il existe $n, m \in \mathbb{Z}, x = y + nu + mv$). Comme cet ensemble est fini, on peut prendre une forme finie F qui contient un représentant de chaque classe de ce quotient. Dans ce cas, $(\mathfrak{F}|_F, u, v)$ est un patron de fond dont la coloration associée est $\mathfrak{F}$. ■

Un point remarquable est que la correspondance entre patron de fond et fond n'est pas injective. Le patron étant un objet syntaxique, il est alors possible de s'intéresser aux résultats de manipulation sur un fond.

Définition 2.1.6. *Le translaté d'un fond (M, u, v) par un vecteur $t \in \mathbb{Z}^2$ est le fond $(u \cdot M, u, v)$.*

La propriété que l'objet définit est bien un fond est évidente. Pour la définition de particule, un lemme préliminaire est nécessaire : ce lemme sert à montrer le résultat intuitif qu'une particule peut se décomposer à l'aide de fonds.

Lemme 2.1.7. *Une particule est composée d'une portion périodique de largeur finie séparant deux fonds.*

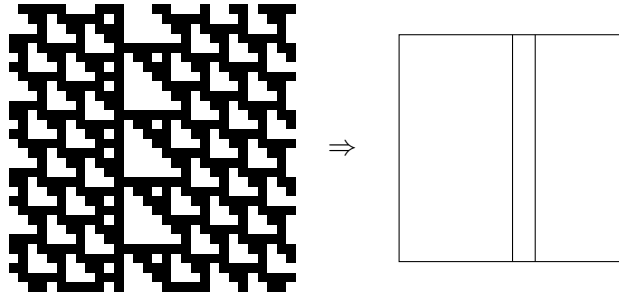


Fig. 2.5 – Décomposition d'une particule

Démonstration. Soit une particule caractérisée par le couple $(u, v) \in \mathbb{Z}^2$. Comme u est un vecteur de périodicité, il n'existe (à un shift près) qu'un nombre fini de mots extraits suivant v . Il en résulte donc que seule une portion de largeur bornée du diagramme n'est pas bipériodique. Quitte à faire grossir cette portion, elle sépare le plan en deux portions connexes. Dans chacune de ces portions, le coloriage est bipériodique. ■

Définition 2.1.8. *Un patron de particule (voir Fig. 2.4b) est un quadruplet $(M, u, \mathfrak{F}, \mathfrak{F}')$ où M est un motif fini, $u \in \mathbb{Z}^2$ un vecteur, $\mathfrak{F}$ et $\mathfrak{F}'$ deux patrons de fond tels que :*

1. le support de $\mathcal{I} = \bigoplus_{k \in \mathbb{Z}} ku \cdot M$ est une frontière avec deux bords (G et D);
2. $\mathfrak{P} = \mathfrak{F}|_G \oplus \mathcal{I} \oplus \mathfrak{F}'|_D$ est une coloration valide;
3. la fonction $\xi : z \mapsto \begin{cases} \mathfrak{F} & \text{si } z \in G \\ \mathfrak{P} & \text{si } z \in \text{Sup}(\mathcal{I}) \\ \mathfrak{F}' & \text{si } z \in D \end{cases}$ est le modèle d'un patchwork.

Ce patchwork (qui est égal à $\mathfrak{P}$) est appelé coloration associée au patron de particule.

Cette définition fait ressortir de façon explicite la portion $\mathcal{I}$ qui n'est pas bipériodique, appelée *perturbation* dans la suite. Comme pour le cas du fond, les patrons de particules correspondent aux particules.

Proposition 2.1.9.

- La coloration associée à un patron de particule est une particule;
- Pour toute particule $\mathfrak{P}$ il existe un patron de particule dont elle est la coloration associée.

Démonstration. Soit $\mathfrak{P}$ la coloration associée à un patron de particule $(M, u, \mathfrak{F}, \mathfrak{F}')$. Comme $\mathfrak{F}$ et $\mathfrak{F}'$ sont bipériodiques, il existe un entier $k \in \mathbb{N}^+$ tel que ku soit un vecteur de périodicité de ces deux fonds. ku est un vecteur de périodicité de $\mathfrak{P}$. De même, il existe un vecteur v , non colinéaire à u qui est à la fois une période de $\mathfrak{F}$ et de $\mathfrak{F}'$. Ce qui permet de conclure que $\mathfrak{P}$ est une particule caractérisée par u et v .

Réciproquement, soit $\mathfrak{P}$ une particule caractérisée par u et v . D'après le lemme 2.1.7, La particule se décompose en deux fonds $\mathfrak{F}$ et $\mathfrak{F}'$ et une bande u -périodique de largeur finie. Quitte à agrandir cette bande, elle forme une frontière avec deux bords et englobe une portion de chaque fond assurant que pour tout point à l'extérieur de cette bande, son voisinage ne contient que le fond correspondant. Comme cette bande est u -périodique et de largeur finie, elle peut-être générée par un motif fini M . La coloration obtenu est bien celle associée au patron de particule $(M, u, \mathfrak{F}, \mathfrak{F}')$. ■

Encore une fois, la correspondance entre patron et particule n'est pas injective.

Définition 2.1.10. Le translaté d'une particule $(M, u, \mathfrak{F}, \mathfrak{F}')$ par un vecteur $t \in \mathbb{Z}^2$ est la particule $(t \cdot M, u, t \cdot \mathfrak{F}, t \cdot \mathfrak{F}')$.

Comme dans le cas précédent, la définition de collision est mieux comprise à l'aide d'un lemme permettant d'obtenir la décomposition intuitive d'une collision en particules et fonds.

Lemme 2.1.11. Soit $\mathfrak{C}$ une collision munie d'une caractérisation normale $(u_i)_{i \in \mathbb{Z}_m}$. Cette collision peut se décomposer en m particules réparties dans

les cônes $\triangleleft_{u_i}(u_{i-1}, u_{i+1})$ pour tout $i \in \mathbb{Z}_m$ et une perturbation finie. De plus, deux particules consécutives ont un fond commun qui remplit l'intersection des deux cônes correspondants.

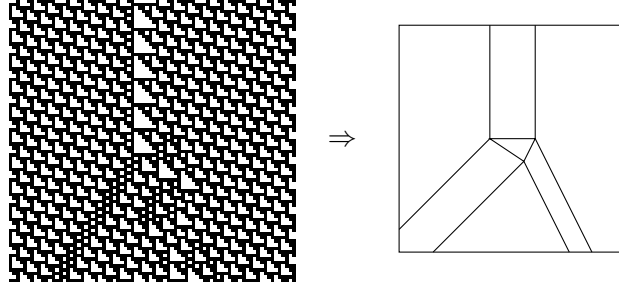


Fig. 2.6 – Décomposition d'une collision

Démonstration. Soit $\mathfrak{C}$ une collisions munie d'une représentation normale $(u_i)_{i \in \mathbb{Z}_m}$. On peut voir le plan comme la réunion des portions (non disjointes) $\triangleleft_{u_i}(u_{i-1}, u_{i+1})$ pour tout $i \in \mathbb{Z}_m$ et d'une portion finie. Pour tout $i \in \mathbb{Z}_m$, la portion $\triangleleft_{u_i}(u_{i-1}, u_{i+1})$ peut être divisée en trois sous portions : $\triangleleft_{u_i}(u_{i-1}, u_{i+1}) \cap \triangleleft_{u_{i+1}}(u_i, u_{i+2})$, $\triangleleft_{u_i}(u_{i-1}, u_{i+1}) \cap \triangleleft_{u_{i-1}}(u_{i-2}, u_i)$ et le reste. Sur les deux premières portions, le coloriage est bipériodique (selon u_i et u_{i+1} pour la première, u_i et u_{i-1} pour la seconde) alors que la dernière est une bande u_i périodique de largeur constante. De ce fait, chaque cône contient la restriction d'un particule. On peut noter que la décomposition en particules impose l'égalité d'un fond pour tout couple de particules associées successives. Il reste alors uniquement une portion finie qui n'appartient à aucun cône. ■

Définition 2.1.12. Un patron de collision (voir Fig. 2.4c) est un couple (M, L) où M est un motif fini et $L = (\mathfrak{P}_i)_{i \in \mathbb{Z}_m}$ une liste de patrons de particule $\mathfrak{P}_i = (M_i, u_i, \mathfrak{F}_i, \mathfrak{F}'_i)$ vérifiant :

1. pour tout $i \in \mathbb{Z}_m$, $\mathfrak{F}_i = \mathfrak{F}'_{i+1}$;
2. le support de $\mathcal{I} = M \oplus \bigoplus_{i \in \mathbb{Z}_m, k \in \mathbb{N}} ku_i \cdot M_i$ est une frontière avec m bords ;
3. pour tout $i \in \mathbb{Z}_m$, le support de $M \oplus \bigoplus_{k \in \mathbb{N}} (ku_i \cdot M_i \oplus ku_{i+1} \cdot M_{i+1})$ est une frontière avec deux bords (on note B_i celui situé à gauche de $\mathfrak{P}_i$) ;
4. $\mathfrak{C} = \mathcal{I} \oplus \bigoplus_{i \in \mathbb{Z}} \mathfrak{F}_i|_{B_i}$ est une coloration satisfaisant les contraintes ;

5. la fonction $\xi : z \mapsto \begin{cases} M & \text{si } z \in \text{Sup}(M) \\ \mathfrak{P}_i & \text{si } z \in \text{Sup}(\bigoplus_{k \in \mathbb{N}} ku_i \cdot M_i) \\ \mathfrak{F}_i & \text{si } z \in \text{Sup}(B_i) \end{cases}$ est le modèle d'un patchwork.

Ce patchwork est appelé coloration associée au patron de la collision.

Bien qu'elle semble plus complexe que dans les cas précédents, la définition de patron de collision peut être décomposée en plusieurs parties. La première condition requiert que deux particules consécutives aient un fond commun. Les deux conditions suivantes servent à assurer que les particules forment une étoile centrée sur la perturbation. La condition suivante demande que le résultat soit valide et la dernière condition assure l'existence de marges dans l'assemblage des particules. De la même façon que pour la particule, cette définition fait apparaître explicitement la portion non périodique $\mathcal{I}$, la *perturbation*.

Proposition 2.1.13.

- La coloration associée à un patron de collision est une collision ;
- Pour toute collision $\mathfrak{C}$ il existe un patron de collision dont elle est la coloration associée.

Démonstration. Soit $\mathfrak{C}$ la coloration associée à un patron de collision $(M, (M_i, u_i, \mathfrak{F}_i, \mathfrak{F}'_i)_{i \in \mathbb{Z}_m})$. On veut montrer que c'est une collision de caractérisation normale $(k_i u_i)_{i \in \mathbb{Z}_m}$ pour certains $k_i \in \mathbb{N}^+$. Pour $i \in \mathbb{Z}_m$, on considère le cône $\triangleleft_{u_i}(u_{i-1}, u_{i-1})$. Comme M est fini, il existe une constante $l_i \in \mathbb{N}^+$ telle que $\triangleleft_{l_i u_i}(u_{i-1}, u_{i-1}) \cap \mathcal{I}$ soit incluse dans $\bigoplus_{k \in \mathbb{N}} k \cdot M_i$. Comme le reste de ce cône est composé uniquement de fonds, il existe une constante l'_i telle que $\triangleleft_{l_i u_i}(u_{i-1}, u_{i-1})$ soit $l'_i u_i$ périodique. On prend alors $k_i = l_i l'_i$.

Réciproquement, soit $\mathfrak{C}$ une collision et $(u_i)_{i \in \mathbb{Z}_m}$ une caractérisation normale de cette collision. D'après le lemme 2.1.11, chaque cône $\triangleleft_{u_i}(u_{i-1}, u_{i-1})$ contient une particule de période u_i . Comme la forme de la perturbation de la particule sur le cône est une demi-droite, il existe un motif fini M_i telle que cette forme soit égale à $\text{Sup}(\bigoplus_{k \in \mathbb{N}} k \cdot M_i)$. De plus, on peut noter que l'ensemble des points n'appartenant à aucun cône est fini. De la même façon que pour la particule, quitte à agrandir certains motifs, on obtient un patron de collision. ■

On notera qu'une fois encore le patron d'une collision n'est pas unique.

Définition 2.1.14. Le translaté d'une collision $(M, (\mathfrak{P}_i)_{i \in \mathbb{Z}_m})$ par un vecteur $t \in \mathbb{Z}^2$ est la collision $(t \cdot M, (t \cdot \mathfrak{P}_i)_{i \in \mathbb{Z}_m})$.

À l'aide de ces définitions syntaxiques, il est maintenant possible de se fixer un ensemble de particules et collisions cohérent désigné sous le nom de système PaCo.

Définition 2.1.15. Un système PaCo est un couple (P, C) où P est un ensemble de patrons de particules propres et C un ensemble de patrons de collisions propres. De plus, pour toute collision $(M, (\mathfrak{P}_i)_{i \in \mathbb{Z}_m}) \in C$, pour tout $i \in \mathbb{Z}_m$, il existe un patron de particule $p \in P$ et un vecteur de déplacement $t \in \mathbb{Z}^2$ tel que $\mathfrak{P}_i = t \cdot p$.

Par un abus de notation, on note $\delta(c, p)$ ce vecteur bien qu'une telle notation ne soit pas injective. Cependant, le vecteur désigné sera toujours déterminé par le contexte. Comme on dispose d'un représentant canonique pour chaque particule, il est possible de définir une *orientation* de la particule à l'aide du vecteur de périodicité. Cette orientation permet de définir la notion de particule *entrante* ou *sortante* d'une collision. Dans le cas des automates cellulaires, l'orientation est choisie de façon à être dans le sens des temps croissants.

2.1.3 Patrons et décidabilité

À l'aide des patrons, il est possible de procéder à une étude des questions de décidabilité pour les fonds, particules et collisions et voir comment on peut extraire ces éléments.

Un premier point est que l'ensemble des patrons de fonds (resp. particules, collisions) est récursif.

Proposition 2.1.16. *L'ensemble des patrons de fond est récursif.*

Démonstration. Soit $u, v \in \mathbb{Z}^2$ deux vecteurs et M un motif fini. Pour savoir si (u, v, M) est un fond, il faut et il suffit d'effectuer les vérifications suivantes :

- vérifier que u et v ne sont pas colinéaires ;
- vérifier que le support de M pave bien le plan en le répétant suivant les vecteurs u et v (il suffit de vérifier que tous les points de M sont distincts modulo translation par les vecteurs u et v , et qu'il y a le bon nombre de points) ;
- vérifier que la coloration obtenue est valide (comme cette coloration est bipériodique, il suffit de vérifier une portion finie).

Toutes ces vérifications sont récursives. ■

Proposition 2.1.17. *L'ensemble des patron de particules est récursif.*

Démonstration. De même que pour les fonds, on se donne u un vecteur, M un motif fini et $\mathfrak{F}, \mathfrak{F}'$ deux fonds. Pour vérifier que le support de $\mathcal{I} = \bigoplus_{k \in \mathbb{Z}} ku \cdot M$ est une frontière avec deux bords, il suffit de vérifier localement du fait de la u -périodicité. Comme, de plus, les fonds induisent une coloration valide, il suffit de vérifier un portion finie des zones de contact pour s'assurer de la validité de la coloration et du patchwork. ■

Proposition 2.1.18. *L'ensemble des patrons de collisions est récursif.*

Démonstration. Le cas de la collision fonctionne exactement de la même façon que le cas de la particule, il suffit de vérifier localement autour de la perturbation pour s'assurer que les conditions sont respectées. ■

Les patrons sont donc des descriptions finies et calculables des fonds, particules et collisions. Il reste néanmoins le problème de l'existence d'un patron canonique pour un objet fixé. Pour résoudre ce problème, il suffit de s'intéresser à la décidabilité de savoir si deux patrons sont associées à un même objet.

Proposition 2.1.19. *Savoir si deux patrons de fonds (resp. particules, collisions) sont associés au même fond (resp. particule, collision) est décidable.*

Démonstration. La preuve de cette propriété reprend la méthodologie des trois preuves précédentes. En effet, il suffit de tester sur une portion finie pour vérifier les conditions des patrons. Cette portion est également suffisante dans le cas du test d'égalité de toute la coloration. ■

En combinant ces résultats, il est possible d'obtenir une proposition sur la calculabilité des objets définis.

Corollaire 2.1.20. *L'ensemble des fonds (resp. particule, collision) est un ensemble récursif.* ■

Les résultats précédents peuvent être étendus aux classes d'équivalence par translation utilisées dans ce mémoire.

Lemme 2.1.21. *Savoir si deux fonds (resp. particules, collisions) sont équivalents est décidable.*

Démonstration. Il suffit de remarquer qu'il n'existe qu'un nombre fini de translatés possibles pour le fond. Dans le cas des particules et des collisions, les perturbations nous restreignent également à tester un nombre fini de translations. ■

Corollaire 2.1.22. *Savoir si un couple (P, C) est un système PaCo est décidable.*

Ces résultats permettent de manipuler les systèmes PaCo et de vérifier si un tel objet est correct. Il est également possible d'exhiber une notion de représentant canonique en prenant le plus petit élément de la classe pour n'importe quel ordre total induit sur les descriptions finies. Cependant, cette notion n'est pas nécessaire dans le cadre de ce mémoire.

En regardant de plus près, il est possible de se rendre compte que les patrons contiennent beaucoup d'informations redondantes. On peut se poser la question du rôle joué par cette information du point de vue de la décidabilité. Sur ce point, il est possible de montrer qu'un changement de point de vue sur l'énumération des objets peut faire s'effondrer cette décidabilité :

Proposition 2.1.23. *Soit $\mathfrak{P}_i$ une liste de particules. Il est indécidable de savoir si il existe une collision faisant intervenir exactement ces particules.*

Idée de la preuve. On procède par réduction à partir de l'indécidabilité du problème de l'arrêt des machines de Turing. On utilise la transformation usuelle de machine de Turing en un automate cellulaire. On ajoute ensuite trois particules : deux de ces deux particules sont non parallèle et leur rencontre lance un calcul Turing sur l'entrée vide. La dernière particule est émise si la machine de Turing atteint l'état d'arrêt. On obtient alors bien que la collision formelle existe si et seulement si la machine de Turing s'arrête. ■

2.2 Ligatures

Les patrons permettent de voir les particules et collisions comme des objets syntaxiques. Cette partie introduit une méthode syntaxique d'assemblage de ces objets correspondant à l'assemblage utilisé usuellement.

2.2.1 Représentation symbolique

Pour donner l'intuition de l'opération de ligature, il est nécessaire d'introduire des représentations symboliques visuelles des objets. En effet, même si les patrons sont des représentations finies, ils peuvent être de très grande taille et ne sont donc pas adaptés pour servir de support à l'intuition. Pour avoir une représentation manipulable, l'idée est de s'appuyer sur les définitions formelles du chapitre précédent et, en particulier, sur les axes de périodicités. De ce point de vue, le fond est un objet bidimensionnel ; la particule est unidimensionnelle et la collision est ponctuelle.

Une représentation usuelle de ces trois objets est donc obtenue en prenant les éléments de géométrie ayant le même nombre de dimension et en ajoutant des couleurs pour les distinguer. Ainsi, les fond sont représentés par un remplissage uniforme du plan (figure 2.7), les particules sont représentées par des droites colorées séparant le plan en deux portions remplies avec la couleur des fonds correspondant. Finalement, les collisions sont représentées par un point coloré (représentant la perturbation) duquel sort un nombre fini de demi-droites (représentant les particules) et dont les portions de plan entre deux demi-droites consécutives sont remplies avec la couleur du fond correspondant.

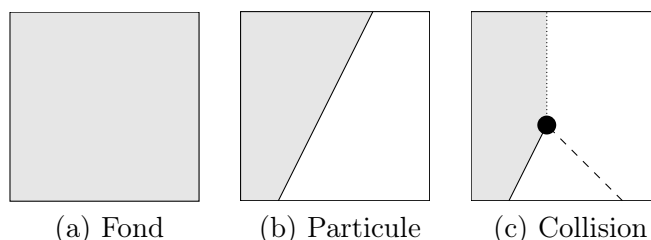


Fig. 2.7 – Représentation des fonds, particules et collisions

Ces représentations ont comme principale caractéristique d'être continues et, de ce fait, de laisser de côté les aspects discrets des objets. Ce choix permet d'introduire une méthode intuitive d'assemblage de collisions mais oblige par la suite à effectuer un travail supplémentaire pour s'assurer du respect des contraintes discrètes.

Pour finir, on remarquera que nos représentations ne font pas intervenir la position de l'origine sur la plan et qu'elles correspondent donc aux classes d'équivalence des objets par translation.

2.2.2 Schémas de ligature

Si on se fixe deux collisions ayant une particule en commun, une opération naturelle consiste à « relier » les deux collisions au moyen de cette particule comme présenté dans la figure 2.8. Cette opération sur les collisions permet de construire de nouveaux objets et est appelée *ligature*.

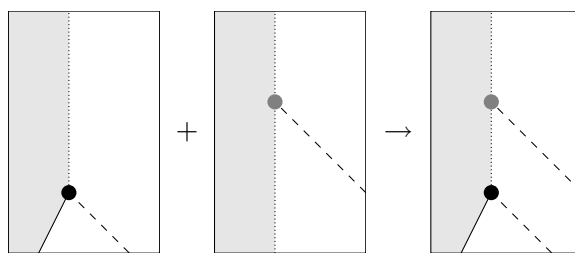


Fig. 2.8 – Ligature de deux collisions

Cette notion intuitive peut être étendue à un nombre quelconque de collisions dont certains paires sont liées au moyen de ligatures donnant naissance au *schéma de ligature*. Un moyen simple de décrire de tels assemblages est d'utiliser un graphe dont les sommets sont des collisions et les arêtes des particules (voir figure 2.9). Le problème qui se pose alors est la représentation des particules qui ne sont pas utilisées dans des ligatures et sont de ce fait uniquement liées à une collision. Pour ces particules, la solution retenue est d'introduire un sommet $\perp$ correspondant intuitivement à l'infini et de voir ces particules comme des arêtes entre ce sommet et la collision dont est issue la particule. Comme il est possible d'avoir plusieurs copies de la même particule ou collision dans le graphe, il est préférable de considérer un graphe anonyme et de munir les sommets et arêtes par les étiquettes des particules et collisions correspondantes.

Si un tel graphe permet bien intuitivement de représenter tous les schémas de ligature, la réciproque est fautive. La notion intuitive de schéma de ligature induit de nombreuses restrictions. Une première restriction est que la ligature de collision s'inscrit dans le plan. Cette condition peut être imposée de façon syntaxique en exigeant que le graphe soit un plongement planaire. Une seconde condition restrictive porte sur les étiquettes : les étiquettes cor-

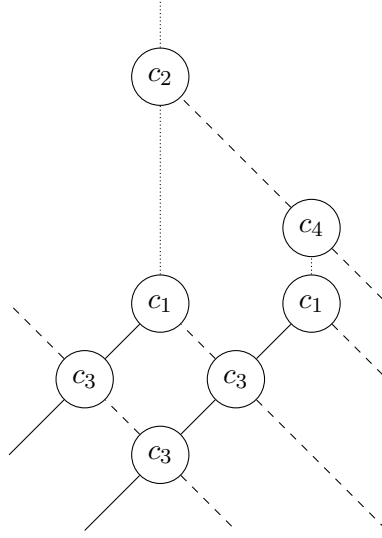


Fig. 2.9 – Exemple de schéma de ligature

respondant aux arêtes adjacentes à un sommet doivent des éléments de la collision correspondante. Comme le graphe est un plongement planaire, il est également possible de parler de l'ordre de ces arêtes qui doit bien sur respecté l'ordre des particules dans la collision.

Intuitivement, une contrainte imposant la pente des particule semble également requise. Nous faisons le choix de ne pas prendre en compte cette contrainte dans la définition formelle. Une première raison est qu'une telle contrainte est beaucoup plus complexe à exprimer et requiert l'introduction de la notion de position. La raison profonde est que cette contrainte est superflue dans le cadre de notre étude et est obtenue par effet de bord dans la suite. Néanmoins, lors de représentations de schéma de ligatures, il est courant d'essayer autant que faire se peut de respecter cette contrainte.

En combinant ces différents aspects, il est possible de donner une définition formelle de schéma de ligature pour un système PaCo (P, C) fixé. Soit $G = (V, E)$ un graphe et un élément $\perp \in V$ appelé *bord*. On note $\mathring{V}$ l'ensemble des *sommet intérieurs* $V \setminus \perp$ et $\mathring{E}$ l'ensemble des *arêtes intérieures* $E \cap \mathring{V}^2$.

Définition 2.2.1. Un schéma de ligature est un quintuplet $(V, \perp, E, \phi_V, \phi_E)$ où : V est un ensemble (potentiellement infini) de sommets, contenant un élément $\perp \in V$, $E \subseteq V^2 \setminus \{(\perp, \perp)\}$ est un ensemble d'arêtes et $\phi_V : \mathring{V} \rightarrow P$ et $\phi_E : E \rightarrow C$ sont des étiquettes. Ces éléments vérifient de plus que :

- (V, E) est un plongement planaire d'un graphe ;
- (V, E) restreint à $(\mathring{V}, \mathring{E})$ est connexe ;
- Pour tout élément $v \in \mathring{V}$, les arêtes adjacentes à v sont étiquetées par des particules équivalentes à celles entrant dans la composition de la collision $\phi_V(v)$. De plus, l'ordre d'apparition dans le plongement

planaire est respecté.

Par abus de langage, les arêtes et sommets seront par la suite assimilés à leurs étiquettes, le contexte permettant toujours de lever l'ambiguïté.

Comme le graphe est un plongement planaire, il est possible de parler de *face* comme représentée dans la figure 2.10. De plus une face sera dite *intérieure* si elle ne contient pas le sommet $\perp$, par opposition aux faces *extérieures* qui le contiennent. Dans tous les cas, la notion de *côté* d'une face est bien définie et l'ensemble des côtés d'une face F est noté $\partial(F)$. On remarquera pour finir que les côtés sont indexés par des particules.

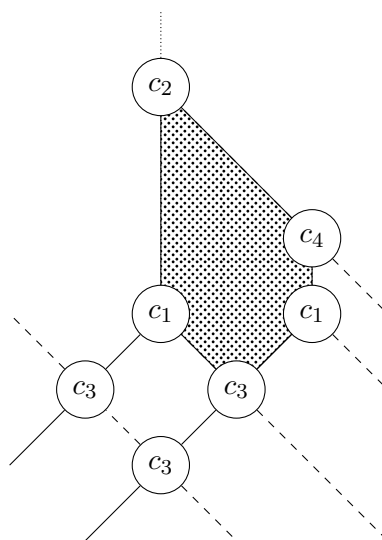


Fig. 2.10 – Faces d'un schéma de ligature

2.2.3 Pondérations

Les schémas de ligature étant défini de façon purement symbolique à partir de représentations symboliques, il est nécessaire de leur donner une interprétation en tant que coloration. Intuitivement, cette interprétation est obtenue en utilisant un patchwork des différentes collisions dont le schéma de ligature sert de modèle.

Cependant, pour mettre en place un tel modèle, il est nécessaire de disposer de positions explicites des collisions pour obtenir le modèle. Plutôt que de donner des positions absolues, nous faisons le choix de se munir de positions relatives, en munissant chaque particule du schéma de ligature d'un entier indiquant le nombre de répétitions d'une telle particule comme dans la figure 2.11.

Définition 2.2.2. Une pondération d'un schéma de ligature $G = (V, E)$ est une fonction $p : \overset{\circ}{E} \rightarrow \mathbb{N}^+$.

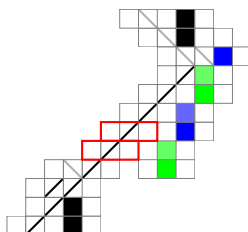


Fig. 2.11 – Exemple de ligature paramétrée par deux répétitions de la particule

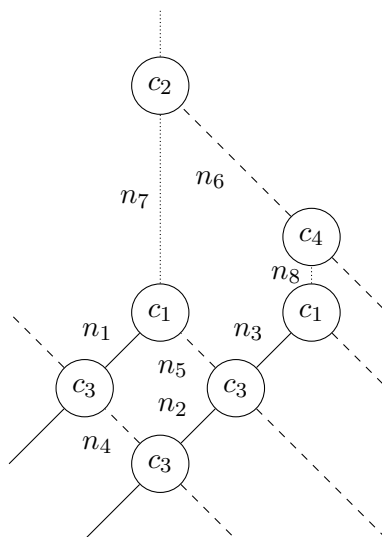


Fig. 2.12 – Schéma de ligature pondéré

Un schéma de ligature muni d'une pondération est appelé *schéma de ligature pondéré* (voir Fig. 2.12). À partir d'un tel schéma, il est possible d'essayer de construire la coloration correspondant à l'interprétation intuitive. Pour cela, une fois fixée une origine arbitraire, le modèle du patchwork est construit de sorte qu'il soit égal à chaque collision sur sa perturbation et qu'il en soit de même pour les restrictions des particules présentes dans le schéma. Pour finir, les portions du plan où le modèle n'est pas défini correspondent à l'intérieur des faces, auxquelles il est possible d'associer un unique fond.

Lorsqu'un tel modèle est bien défini et correspond effectivement à un patchwork, on dit que le schéma de ligature pondéré est *valide*. Pour un tel schéma, il existe bien une interprétation en terme de coloration. Pour un schéma de ligature fixé, la question qui se pose est de caractériser l'ensemble des pondérations aboutissant à une schéma pondéré valide. Cet ensemble est appelé *ensemble de validité* d'un schéma de ligature.

2.3 Validité des schémas de ligature finis

Dans le cas des schémas de ligature finis, il est possible de caractériser les ensembles de validité. Ce résultat constitue le cœur de ce chapitre puisqu'il justifie l'introduction de la construction syntaxique de schémas de ligature en fournissant une méthode pour en caractériser les colorations.

2.3.1 Résolution

Pour pouvoir formuler notre théorème, il est nécessaire d'introduire la notion d'ensemble semi-linéaire. Soit E un ensemble fini, un sous-ensemble $S \subseteq \mathbb{N}^E$ est *linéaire* s'il existe un élément $a \in \mathbb{N}^E$ et des vecteurs $u_i \in \mathbb{N}^E$ pour $0 \leq i < m$ tels que

$$S = \left\{ a + \sum_{0 \leq i < m} k_i u_i \mid k \in \mathbb{N}^m \right\} .$$

Un ensemble est *semi-linéaire* s'il est constitué d'une union finie d'ensembles linéaires. Les pondérations peuvent être vu comme des éléments de $\mathbb{N}^E$ et on peut obtenir le théorème suivant :

Théorème 2.3.1 (N. OLLINGER et G. R., 2008 [2]). *L'ensemble de validité d'un schéma de ligature fini est semi-linéaire. De plus, il existe un algorithme donnant explicitement le semi-linéaire à partir du schéma de ligature.*

Ce résultat est obtenu en montrant que les contraintes exprimant la validité d'un schéma de ligature pondéré peuvent être décrites à l'aide d'une formule dans l'arithmétique de Presburger. La forme des ensembles de validité

tenant à celle des solutions d'équations dans cette arithmétique décidable. Le reste de cette partie est consacrée à la construction de cette formule.

L'arithmétique de Presburger est l'ensemble des formules logiques sur les entiers munis de l'addition et de l'ordre. Pour des raison de concision, la répétition un nombre constant de fois de l'addition est représenté à l'aide du symbole multiplicatif.

Dans la suite, la construction de la formule traitera le cas général mais s'appuiera sur l'exemple de la figure 2.12 afin de faciliter la compréhension. La formule est obtenue comme une conjonction de nombreuses formules de bases exprimant que le modèle du patchwork lié à l'interprétation est bien défini.

On se fixe un schéma de ligature $(V, \perp, E, \phi_V, \phi_E)$. et une pondération $p : \mathring{E} \rightarrow \mathbb{N}$. Afin de pouvoir utiliser les positions des collisions, on introduit à l'aide de quantificateur existentiels $|\mathring{V}|$ nouvelles variables $\{p_v \mid v \in \mathring{V}\}$.

Il faut alors vérifier que ces positions sont bien définies (voir Fig. 2.13). On obtient alors, pour toute arête interne $e = (v, v') \in \mathring{E}$, l'équation

$$(\mathcal{E}_e) \quad p'_v + \delta(v', e) = u_e(p(e) - 1) + p_v + \delta(v, e)$$

où u_e est le vecteur de périodicité de e et $\delta(v, e)$ est vecteur de décalage entre v et e .

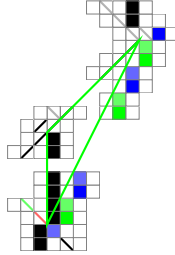


Fig. 2.13 – Contraintes sur les longueurs

À l'aide de ces positions, il est possible d'exprimer les contraintes sur le modèle du patchwork pour ses différents composants. Pour une collision $v = (M, L)$, l'ensemble des point de la perturbation est fini ($\mathcal{I}_v = \{p_v + a \mid a \in M\}$). Pour une particule $e = (v, v') = (M, u, \mathfrak{F}, \mathfrak{F}')$, l'ensemble des points de la perturbation dépend de la présence ou non du sommet $\perp$. Pour les particules internes, la perturbation consiste en l'union d'un nombre fini (fixé par la pondération) du motif M (i.e., $\bigcup_{0 \leq k \leq n_e - 1} \{p_v + \delta(v, e) + ku + a \mid a \in M\}$). Pour les particules externes, il s'agit d'une répétition infinie ($\bigcup_{k \in \mathbb{N}} \{p_v + \delta(v, e) + ku + a \mid a \in M\}$ ou $\bigcup_{k \in \mathbb{N}} \{p'_v + \delta(v', e) - ku + a \mid a \in M\}$). On note $\mathcal{I}_e = \{p_v + \delta(v, e) + ku + a \mid a \in M\}$ (ou $\{p'_v + \delta(v', e) - ku + a \mid a \in M\}$ dans le cas des particule sortantes de $\perp$)

Il reste alors à exprimer sous forme d'un nombre fini d'équations de Presburger que les éléments de ces ensembles sont disjoints en construisant

une formule pour chaque couple d'élément. Pour les collisions, il n'y a aucun problème puisque l'ensemble est fini. Pour les particules, l'ensemble est fini si on quantifie sur l'entier k . On obtient alors des équations de la forme suivantes :

- Cas de deux collisions v et v' :

$$(\mathcal{E}_{v,v'}) \quad \bigwedge_{i \in \mathcal{I}_v, j \in \mathcal{I}_{v'}} i \neq j \quad ;$$

- Cas d'une collision v et d'une particule interne e de vecteur de périodicité u_e et muni de la pondération $p(e)$:

$$(\mathcal{E}_{v,e}) \quad \bigwedge_{i \in \mathcal{I}_v, j \in \mathcal{I}_e} \forall k, 0 \leq k \leq (p(e) - 1) \Rightarrow i \neq j + ku_e \quad ;$$

- Cas de deux particules externe e de vecteur de périodicité u_e incidente à $\perp$ et d'une particule externe e' (de vecteur de périodicité $u_{e'}$) sortante de $\perp$:

$$(\mathcal{E}_{e,e'}) \quad \bigwedge_{i \in \mathcal{I}_e, j \in \mathcal{I}_{e'}} \forall k, k', i + ku_e \neq j - k'u_{e'}$$

Les autres cas se construisent tous exactement de la même façon. Pour le cas des fonds, il suffit de remarquer que notre première formule impose l'existence de zones correspondant à ces fonds et que l'union de ces zones et des supports de perturbation forme une partition du plan. De ce fait, il n'est besoin d'aucune condition supplémentaire.

Le fait le modèle du patchwork est bien défini s'exprime donc sous la forme de l'intersection de toutes les formules obtenus précédemment :

$$(\mathcal{E}) \quad \bigwedge_{e \in \dot{E}} \mathcal{E}_e \quad \bigwedge_{o, o' \in V \cup E | o \neq o'} \mathcal{E}_{o, o'} \quad .$$

Il reste maintenant à vérifier que ce modèle vérifie les conditions de patchwork. Parmi les deux conditions du patchwork, la première (sur l'ensemble de définition des éléments) est toujours vérifiée de par l'utilisation de pavages. Il est ensuite possible de montrer que la deuxième (qui impose l'existence d'une « bande ») commune est vérifiée (voir figure 2.14). Pour prouver cela, on considère d'abord un point l'image par le modèle est une collision. Si tous les points de son voisinage ont la même image, la condition de cohérence est vérifiée. Il en est de même si l'image des points est une particule ou un fond entrant dans la composition de la collision. Le cas le plus délicat est lorsque un des points du voisinage a pour image une autre collision ou particule (le cas du fond n'est pas possible car les particules sont des frontières). Dans ce cas, comme la particule ou la collision est un patchwork dont on se situe sur le bord d'une perturbation, la couleur de ce point est la même que celle du fond (ou de la particule) entrant dans la composition de l'élément et donc

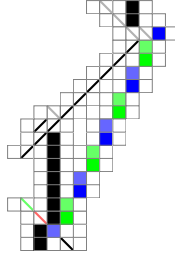


Fig. 2.14 – Contraintes de patchwork

est compatible avec celui de la collision de départ. Il en va de même pour le cas des particules et des fonds.

La conjonction de toutes les formules obtenues caractérise effectivement l'ensemble de validité d'un schéma de ligature. En remarquant que ces formules sont constructibles en espace logarithmique à partir du système PaCo, la décidabilité de l'arithmétique de Presburger et les résultats connus sur la forme des ensembles de solutions de formules [25] permettent de conclure. Même si la construction en elle-même reste valable dans le cas de schémas infinis, la formule obtenue devient infinie ne possède donc pas nécessairement de solutions régulières.

Bien que la formule précédente soit parfaitement correcte, elle présente un certain nombre de contraintes redondantes. Il est possible, à l'aide d'une construction alternative, d'obtenir une autre formule dans laquelle l'intuition derrière les sous-formules est plus simple et dont la résolution est de ce fait plus intuitive. Cette nouvelle vision repose sur l'utilisation des faces. Comme les faces sont l'ensemble des cycles élémentaires, pour vérifier la cohérence des valeurs de la pondération, il suffit de vérifier que pour chaque face, lorsqu'on effectue un tour, on revient au point de départ. On obtient alors, pour une face F caractérisée par une liste cyclique de particules et collisions $(v_i, e_i)_{i \in \mathbb{Z}_m}$, la formule :

$$(\mathcal{E}_F) \quad \sum_{i \in \mathbb{Z}_m} \delta(v_i, e_i) + u_{e_i} n_{e_i} - \delta(v_{i+1}, e_i) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad .$$

De plus, au vues des conditions syntaxiques des schémas de ligature sur l'ordre des particules dans les collisions, la validité de chaque face impliquée que toutes les faces sont disjointes. Cette propriété provient du fait que les perturbations forment des frontières. La formule générale peut alors s'écrire sous la forme d'une conjonction finie de formule dont chacune correspond à la validité d'une face :

$$(\mathcal{E}) = \bigwedge_{F \text{ face}} \mathcal{E}_F \wedge \mathcal{E}'_F \quad .$$

La formule $\mathcal{E}'_F$ étant obtenu en considérant la non superpositions des

perturbations pour une face :

$$(\mathcal{E}'_F) = \bigwedge_{o,o' \in (V \cup E) \cap F | o \neq o'} \mathcal{E}_{o,o'} \quad .$$

2.3.2 Un exemple complet

Afin d'illustrer la construction décrite précédemment, cette section présente un exemple complet de résolution d'un schéma de ligature. Pour cela, nous allons prendre un système PaCo constitué des quatre particules présentées dans la figure 2.15 et des sept collisions de la figure 2.16 ainsi que celles obtenues par rotation d'une de ces collision. Dans ces représentations, le motif fini composant l'objet est mis en valeur. Pour les particules, les vecteurs de périodicité sont indiqués et les particules composant les collisions sont reconnaissables sans ambiguïté. Pour obtenir des définitions syntaxes formelles, il ne manque que la position de l'origine que nous fixerons au centre des motifs (en prenant le point situé juste en dessous de ce centre pour le cas de la collision g). Cette méthode pour donner les patrons des particules et collisions présente l'avantage d'être visuelle et intuitive.

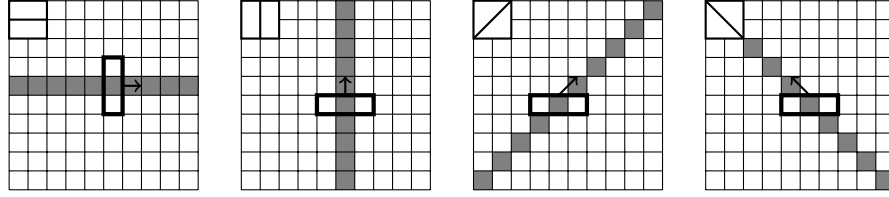


Fig. 2.15 – Particules utilisées dans la construction

On considère alors le schéma de ligature présenté dans la figure 2.17. Ce schéma de ligature comporte deux faces internes et 4 faces externes. Dans la suite, les arêtes sont notées à l'aide des sommets à leurs extrémités.

Dans la plupart des cas, la principale contrainte pour le calcul de l'ensemble de validité est lié à la formule de « retour au point de départ » qui rassemble les contraintes globales. Les formules obtenue dans cet exemple sont :

$$(\mathcal{E}_{F_1}) \quad \binom{2}{0} + \binom{1}{0} n_{a_1,h} + \binom{2}{0} + \binom{-2}{-2} - \binom{1}{1} n_{b,h} + \binom{-2}{-2} \\ + \binom{0}{2} + \binom{0}{1} n_{b,a_1} + \binom{0}{2} = \binom{0}{0}$$

$$(\mathcal{E}_{F_2}) \quad \binom{-2}{0} - \binom{1}{0} n_{a_2,b} + \binom{-2}{0} + \binom{2}{2} + \binom{1}{1} n_{b,h} + \binom{2}{2} \\ + \binom{0}{-2} - \binom{0}{1} n_{a_2,h} + \binom{0}{-2} = \binom{0}{0}$$

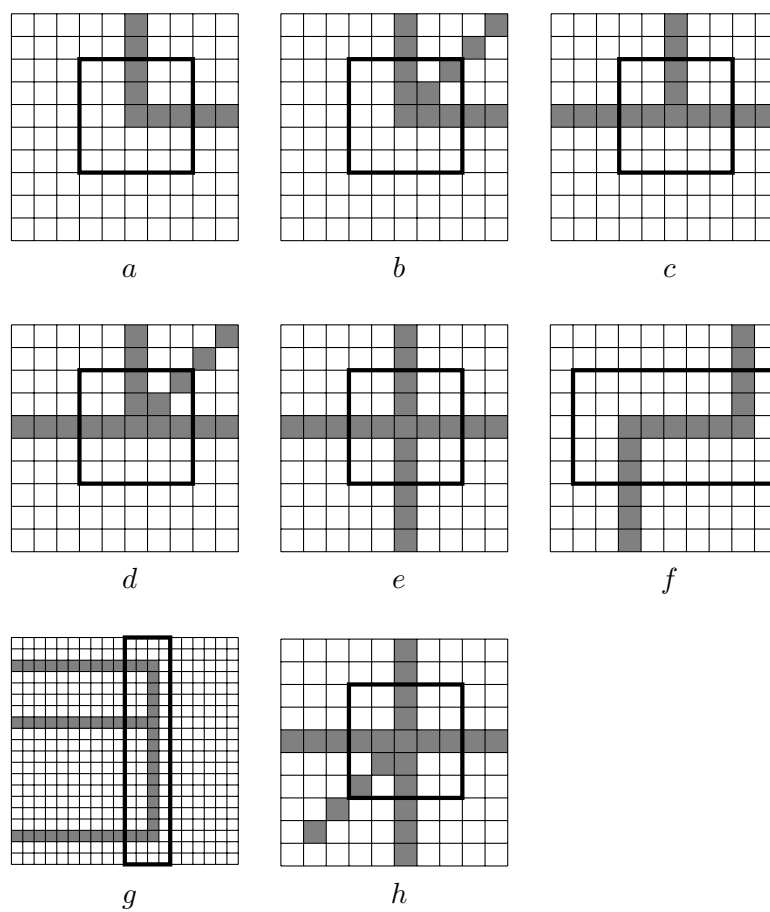


Fig. 2.16 – Collisions utilisées dans la construction

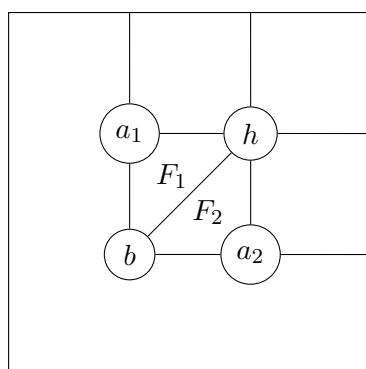


Fig. 2.17 – Exemple de schéma de ligature

La résolution de ces équation nous donne un l'ensemble de solution $\{n_{a_1,h}, n_{b,h}, n_{a_2,h}, n_{b,a_1} \mid n_{a_1,h} = n_{b,h} = n_{a_2,h} = n_{b,a_1}\}$.

Pour terminer la résolution, il suffit de vérifier que toutes les formules exprimant la non superposition des différents éléments sont vérifiées. Quoique simples à écrire, ces formules prennent un espace important et n'apportent aucune restriction supplémentaire. Dans un soucis de concision, nous avons choisi de ne pas les faire figurer ici.

2.3.3 Expressivité des schémas de ligature finis

Dans le cas des ligatures finies, la caractérisation permet de déduire un algorithme effectif pour calculer l'ensemble de validité en fonction du schéma de ligature. Cependant, bien que décidable, la complexité générale du problème de résolution d'une formule de Presburger est de complexité non élémentaire [20]. Dans l'optique de développer un outil spécifique pour les schémas de ligature, il est donc intéressant de se poser la question de la complexité de ce cas particulier. Dans cette section, une réponse partielle à cette question est apportée en montrant qu'il est possible d'encoder à l'aide d'un algorithme en temps polynomial¹ n'importe quelle formule existentielle dans l'arithmétique de Presburger sous forme d'un schémas de ligature.

Proposition 2.3.2. *Il existe une réduction polynomiale de la résolution de formules existentielles dans l'arithmétique de Presburger vers le calcul des ensemble de validité d'un schéma de ligatures fini.*

Démonstration. Comme usuellement, cette réduction repose sur l'utilisation de « gadgets » pour encoder les constructeurs. Le fonctionnement de la réduction peut-être décomposée en deux parties, une partie concerne la manipulation des formules booléennes et une autre des variables et comparaisons.

Pour cette construction, on considère le système PaCo de la section précédente (voir figures 2.15 et 2.16). Ces objets nous permettent de construire 5 gadgets (voir fig. 2.18) qui vont servir à composer notre schéma de ligature. On remarquera que chacun de ces gadget est en lui-même un schéma de ligature et que A est l'exemple utilisé précédemment dont l'ensemble de validité nous permet de l'interpréter comme un angle.

De la même façon, les gadgets B à E correspondent à des schéma de ligature et peuvent être interpréter de la façon suivante :

- le gadget B effectue une duplication ;
- le gadget C effectue un croisement ;
- le gadget D effectue une addition (ou une soustraction) ;
- le gadget E effectue une comparaison entre i et j si la pondération k vaut 1 et est toujours valide dans le cas $k = 0$.

¹On peut même montrer, comme dans la plupart des réduction, que cet algorithme est calculable en espace logarithmique

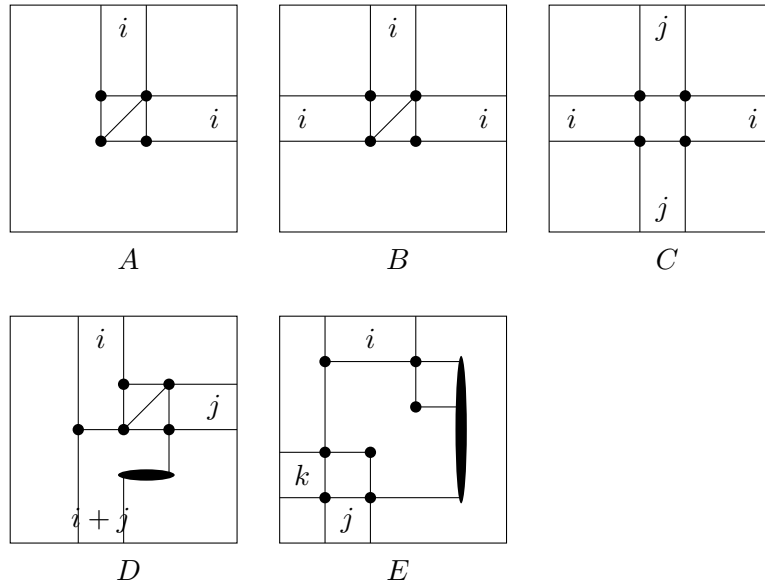


Fig. 2.18 – gadgets

Il est alors possible de construire un schéma de ligature associé à une formule de Presburger existentielle. Pour cela, la formule est d'abord réécrite à l'aide des lois de de Morgan et de l'antisymétrie de l'inégalité pour éliminer toutes les occurrences de la négation. La formule obtenue est alors encodée de la façon suivante :

- les variables existentielles sont assimilés à des pondérations et sont propagées ou dupliquées à l'aide des gadgets A , B ou C .
- les formules élémentaires sont obtenue en utilisant les gadgets D et E .
- la conjonction de deux formules élémentaires est obtenue en reliant les deux valeurs conditionnelles des sous formules à l'aide d'un gadget B ;
- le disjonction fait de même avec un connecteur D dont la sortie est limité à 1.

Cette construction vérifie que toutes les solutions de l'équation peuvent être étendue en une pondération valide et que la projection de toutes les pondérations valides sur les longueurs correspondant aux variables donne une solution de l'équation.

■

Chapitre 3

Extensions

A notion de schéma de ligature finie est donc une opération syntaxique permettant de construire de nouvelles colorations de façon simple. Un objectif est de construire à l'aide de cette méthode des colorations complexes encodant des comportements d'automates cellulaires. Afin de permettre de tels encodages, il est nécessaire d'introduire des extensions des schémas de ligature. Un premier axe est d'étendre l'ensemble de particules et collisions en introduisant de nouveaux objets pour enrichir l'espace. Une autre extension importante consiste à repousser la limite de finitude sur les schémas de ligature. Ce chapitre présente différentes pistes d'extensions utilisées dans le chapitre 4 sans avoir vocation à être exhaustif.

Sommaire

3.1	Familles régulières de signaux	58
3.1.1	Construction de signaux	58
3.1.2	Schémas de signaux	59
3.2	Schémas de ligature infinis	62
3.2.1	Représentation des schémas de ligature infinis . . .	62
3.2.2	Jeux de faces réguliers	64
3.2.3	Jeux de faces ordonnés	66

3.1 Familles régulières de signaux

Jusqu'à présent, le système PaCo était toujours donné. Dans le cas où ce système est fini, le corollaire 2.1.22 permet de vérifier la correction d'un tel élément. Néanmoins, cette finitude implique que l'encodage doit se faire dans un ensemble fini et exclus donc l'encodage des entiers par exemple. Pour contourner ce problème, cette section présente des constructions d'ensembles infinis réguliers de particules et de collisions engendrées par un système PaCo fini et assurant que le résultat est bien un système PaCo. De tels systèmes offrent un espace plus important pour encoder les informations.

3.1.1 Construction de signaux

À l'aide d'un nombre fini de particules, la figure 3.1 présente deux techniques pour encoder une infinité d'informations : soit à l'aide du nombre de répétitions de particules, soit au travers de l'espacement entre deux particules consécutives. Il est évidemment possible de mélanger ces deux approches. Les objets ainsi créés peuvent être décrits à l'aide d'une succession de couples particule, vecteur, le vecteur indiquant l'espacement à la particule suivante. Une telle présentation fait implicitement l'assertion que les particules utilisées sont toutes parallèles. Comme cette notion est construite à l'aide de « morceaux » de particules, il n'est pas surprenant de voir apparaître encore une fois la notion de patchwork.

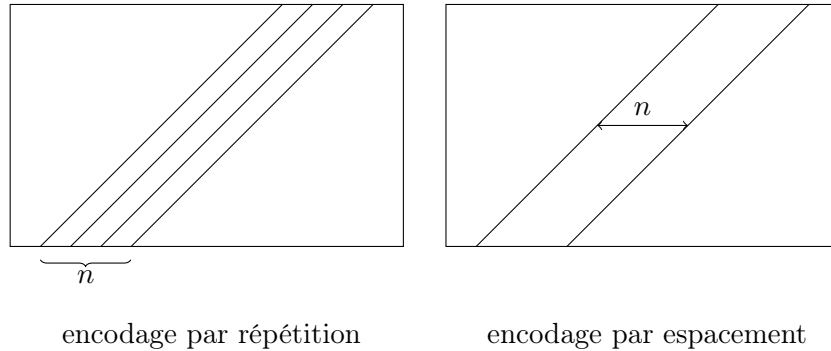


Fig. 3.1 – Encodage d'informations complexes à l'aide de particules

Définition 3.1.1. Soit $\mathfrak{P}_p$ un ensemble de particules de même vecteur de périodicité u . Un signal sur $\mathfrak{P}_p$ est une liste finie de couples $(\mathfrak{P}_i, z_i)_{i \in \llbracket 0, M \rrbracket} \in \mathfrak{P}_p \times \mathbb{Z}^2$ telle que :

- pour tout $i \in \llbracket 1, m \rrbracket$, l'union des supports des perturbations $\mathcal{I}_{i-1}$ et $\mathcal{I}_i$ forment une frontière à trois bords. On note P_i le bord à droite de $\mathcal{I}_{i-1}$ et à gauche de $\mathcal{I}_i$ et $\mathfrak{F}_i$ le fond associé ;
- si on note P_0 (resp. P_{m+1}) le bord à gauche (resp. à droite) du support

de $\mathcal{I}_0$ (resp. $\mathcal{I}_m$) et $\mathfrak{F}_0$ (resp. $\mathfrak{F}_{m+1}$) le fond associé, alors la fonction :

$$\xi z \mapsto \begin{cases} \mathcal{I}_i & \text{si } z \in \text{Sup}(\mathcal{I}_i) \\ \mathfrak{F}_i & \text{si } z \in P_i \end{cases}$$

est le modèle d'un patchwork.

En regardant la coloration obtenue, l'unicité du vecteur de périodicité impose que celle-ci est périodique. Cette constatation permet de remarquer que les signaux peuvent être ajoutés au système PaCo.

Lemme 3.1.2. *Un signal est un particule.* ■

Les signaux fournissent donc une méthode pour étendre l'ensemble des particules d'un système PaCo. Cette extension est utile pour faciliter l'encodage de l'information dans les particules mais n'augmente pas pour autant la puissance d'expression d'un système PaCo, ce qui est fait avec les signaux pouvant l'être à l'aide des particules les composant.

Lors de la présentation des signaux, nous avons fait le choix de prendre des particules parallèles ; il est également possible de construire de nouvelles particules à l'aide d'un ensemble de particules non parallèles et de collisions dont le comportement serait périodique comme dans la figure 3.2. Une telle extension est une piste de développement possible et peut-être mise en relation en particulier avec les « guns ».

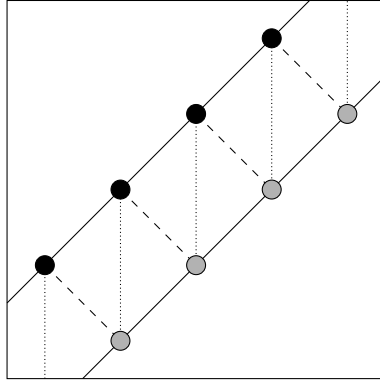


Fig. 3.2 – Construire des signaux avec des particules non parallèles

3.1.2 Schémas de signaux

À l'aide des signaux, il est possible d'enrichir l'ensemble de particules d'un système PaCo. Cet enrichissement est inutile s'il ne permet pas d'enrichir également l'ensemble des collisions. Dans cette section, nous allons voir qu'on peut enrichir l'ensemble des collisions en se basant sur les signaux à l'aide de *schémas de signaux*. Ces nouvelles collisions sont construites à

partir des collisions données pour les particules existantes. La méthode principale de construction de telles collisions repose sur l'utilisation de schémas de ligature.

Proposition 3.1.3. *Toute pondération valide d'un schéma de ligature fini est une collision.*

Démonstration. La preuve de ce résultat repose sur l'utilisation de l'aspect syntaxique des collisions.

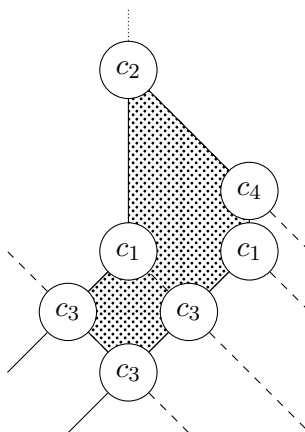


Fig. 3.3 – Lien entre schéma de ligature pondéré et collision

Soit un schéma de ligature muni d'une pondération valide. On considère le patchwork associé à cette pondération (*cf.* Fig. 3.3). On peut remarquer que l'union des supports des faces internes (perturbations des collisions, perturbation des particules internes et fond associés aux faces) a un support fini. De plus, il reste alors uniquement des particules sortantes ou entrantes qui ne se croisent pas, séparées par des fonds. On obtient bien une nouvelle collision dont la perturbation est constituée de l'ensemble des faces internes et dont la liste des particules est composée des particules externes. ■

Les schémas de ligature fournissent donc une méthode pour augmenter l'ensemble des collisions. Une piste est d'essayer de générer en une seule fois des ensembles de collisions en tirant parti de la régularité de certains ensembles de signaux.

Une cas simple est lorsque le paramétrage des signaux porte uniquement sur l'espacement entre les différentes particules. Dans ce cas, un seul schéma de ligature fini peut correspondre à un ensemble de collisions faisant intervenir ces signaux. Le théorème 2.3.1 donne directement un ensemble régulier de collisions possibles.

Le cas des paramétrage sur le nombre de particules dans les signaux est plus complexe. Dans ce cas, en effet, on se retrouve avec une famille de schémas de ligature finis correspondant à des collisions entre les signaux

dont on souhaite tirer parti de la régularité. Cette régularité peut s'exprimer sous de nombreuses et diverses formes. Dans ce mémoire, différentes pistes sont proposées : leur choix correspond au souhait de présenter un ensemble de méthodes génériques utilisables dans un large spectre de cas. Ce choix laisse un large choix de perspectives dans l'étude de la génération de telles collisions.

Un premier exemple de régularité dans de telles familles de schémas de ligature se produit lors de l'étude de signaux paramétrés par une répétition dans le cas où les particules supplémentaires n'influent pas sur les collision produites. Un exemple fréquent de telle phénomène est le croisement de signaux. Dans ce cas simple, on parle de famille croissante de schéma de ligature.

Définition 3.1.4. *Une famille de schémas de ligature $(V_i, E_i, \phi_{V_i}, \phi_{E_i})_{i \in \mathbb{N}}$ est croissante si elle forme une chaîne croissante pour l'inclusion (i.e., pour tout $i \in \mathbb{N}$, $V_i \subset V_{i+1}$, $E_i \subset E_{i+1}$ et $\phi_{V_{i+1}}$ (resp. $\phi_{E_{i+1}}$) coïncide avec ϕ_{V_i} (resp. ϕ_{E_i}) sur V_i (resp. E_i).*

Lemme 3.1.5. *Soit $(V_i, E_i, \phi_{V_i}, \phi_{E_i})_{i \in \mathbb{N}}$ une famille de schémas de ligature croissante et p_i la famille des pondération valides associées à chaque schéma, alors pour tout $i, j \in \mathbb{N}$, si $i < j$ p_j restreinte à V_i est une pondération valide du schéma de ligature $(V_i, E_i, \phi_{V_i}, \phi_{E_i})$.*

Ce lemme incite à chercher les pondérations valides de telles familles par récurrence. De plus, sous certaines conditions, il est même possible de trouver une pondération $p : E = \cup_{i \in \mathbb{N}} E_i \rightarrow \mathbb{N}$ dont la restriction à tout schéma de ligature de la famille est une pondération valide. Ce cas se produit fréquemment lorsque les faces présentes dans les familles de schémas de ligature sont des parallélogrammes. Dans de tels cas, on se contente d'exhiber un ensemble de pondération valide et non de calculer l'ensemble de validité de chacun des schémas de ligature de la famille.

L'utilisation de familles infinies de schémas de ligature peut également faire apparaître des suites de faces ayant un nombre croissant de côtés (voir figure 3.4). Dans de tel cas, il est souvent déraisonnable d'essayer de calculer l'ensemble de validité des schémas de ligature mais il est possible d'en extraire facilement un ensemble de pondérations dont on peut prouver la validité.

Pour conclure, un moyen simple d'étendre les systèmes PaCo est d'introduire des signaux puis des collisions basées sur ces signaux à l'aide de pondérations valides de schémas de ligature. Les pistes présentées dans ce mémoire forment un point de départ montrant qu'une telle approche est intéressante mais elles montrent très vite leur limites. Une perspective sur ces extensions consiste à obtenir de bonnes caractérisations de sous-ensembles de signaux ou de famille de schémas de ligature permettant de construire algorithmiquement les collisions associées.

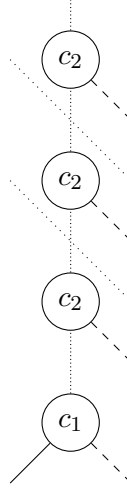


Fig. 3.4 – Exemple de famille de faces possédant un nombre arbitraire de côtés

3.2 Schémas de ligature infinis

Une limitation principale de notre approche à l'aide de schémas de ligature est que les résultats permettant de lier ces constructions aux coloriage sont limités aux schémas de ligature finies. Cette partie s'intéresse aux problèmes rencontrés dans le cadre des schémas de ligature infinis et propose quelques extensions pour manipuler de tels objets.

3.2.1 Représentation des schémas de ligature infinis

Un des premier problème des schémas de ligature infinis est la représentation syntaxique de tels objets. Il est évidemment illusoire de vouloir décrire n'importe quel schéma de ligature ou n'importe quel ensemble de schémas. L'idée de cette section est de considérer des ensembles de schémas de ligature infinis donnés à l'aide de conditions restrictives et d'utiliser ces conditions pour prouver l'existence de pondérations valides. Cette approche restrictive se justifie par l'indécidabilité de l'existence d'un schéma de ligature admettant une pondération valide.

Proposition 3.2.1. *Étant donné un système PaCo, il est indécidable de savoir si il existe un schéma de ligature infini avec une pondération valide.*

Idée de la preuve. La preuve se fait par réduction du problème de la pavabilité du plan à l'aide d'un jeu de tuile de Wang.

On considère une particule horizontale par couleur horizontale. On fait de même avec les couleurs verticales. On ajoute 4 particules diagonales (respectivement bg, bd, hg, hb).

Pour chaque tuile, on construit une collision qui transforme les particules codant les couleurs des cotés gauche et bas de la tuile plus les particules diagonales bg et bd en les couleurs codant les cotés droit et haut (plus les particules hg et hb) comme dans la figure 3.5. Pour finir, on ajoute une collision $\mathfrak{C} : hd + hg \vdash bd + bg$.

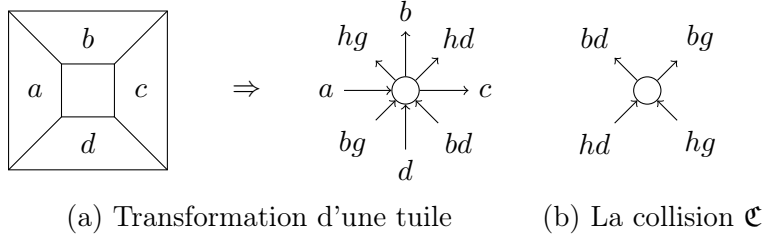


Fig. 3.5 – Transformation d'un jeu de tuiles de Wang en système PaCo

On vérifie alors facilement que tout schéma infini possédant une pondération valide est une grille et que l'on peut associer un pavage valide à tout schéma de ligature infini et réciproquement. ■

Dans le cas de schémas de ligature infinis, il existe également des problèmes liés aux propagations de contraintes : l'ajout d'une particule peut avoir des répercussions arbitrairement loin. Une idée pour éviter cette difficulté est de considérer des contraintes locales et regarder les schémas satisfaisant ces contraintes (un peu comme pour les pavages). En se référant à la fin du chapitre précédent, on remarque que les conditions sur la validité des pondérations peut s'exprimer comme la réunion de contraintes indépendantes sur les faces. De ce fait, les contraintes sont posées sur l'assemblage des différentes faces (voir fig 3.6).

Définition 3.2.2. *Un jeu de faces est un couple $(C, \mathbb{F})$ où C est un ensemble fini de couleurs et $\mathbb{F}$ un ensemble de face dont tous les cotés sont munis d'une couleur.*

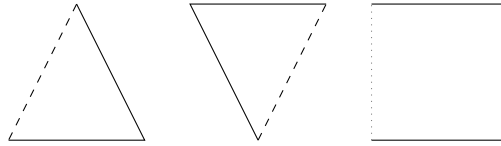


Fig. 3.6 – Exemple de jeu de face

Définition 3.2.3. *Un schéma de ligature appartient à un jeu de faces $(C, \mathbb{F})$ s'il existe une injection de l'ensemble des faces du schéma dans $\mathbb{F}$ tel que les particules liant deux faces adjacentes dans le schéma aient la même couleur.*

Si on regarde attentivement, la contrainte posée est plus forte que la seule contrainte de couleur. En combinant cette contrainte avec celles du schéma de ligature, on remarque que la contrainte est orientée : chaque face est située d'un côté différent de la particule les liant. De ce fait, les contraintes d'un jeu de faces sont plus proches de contraintes géométriques (qui sont orientées) que de réelles contraintes de coloration (à la Wang)

Au vu de la définition, il est indécidable de savoir si un schéma de ligature est compatible avec un jeu de faces donné car il faut « deviner » l'injection. On prouve aisément ce résultat à partir de l'indécidabilité de la pavabilité du plan.

Les restrictions engendrées par les contraintes sur les faces vont nous permettre d'exhiber des pondérations valides pour n'importe quelle portion finie d'un élément d'un jeu de faces fixé. Par la suite, l'objectif n'est pas de caractériser complètement l'ensemble des schémas de ligature infinis d'un jeu admettant des pondération valide mais de donner des critères simples pour prouver l'existence de pondération valide d'ensembles de schémas de ligature infinis d'un jeu. Dans cette optique, deux principales pistes sont proposées : les jeux de faces réguliers et les jeux de faces ordonnés.

3.2.2 Jeux de faces réguliers

Dans un premier temps, nous allons nous intéresser au cas des jeux de faces pour lesquels il existe une pondération valide dans laquelle toutes les faces colorées de la même façon ont la même pondération. Pour donner une telle pondération, il suffit de donner une fonction de pondération de l'ensemble des faces colorées et de vérifier localement que la pondération est correcte.

Comme les jeux de faces sont donnés à l'aide d'un ensemble fini de faces, il est possible de décrire finiment une pondération d'un schéma de ligature infini de ce jeu en donnant pour chaque face, la valeur de la pondérations sur les particules de ses faces.

Définition 3.2.4. *Un jeu de faces finiment pondéré est un jeu $(C, \mathbb{F})$ qui possède une pondération valide p de toutes les faces colorées de $\mathbb{F}$ compatible avec les couleurs (c'est à dire que, pour les particules identiques de la même couleurs, la pondération est la même).*

De par le fait que les schémas de ligature appartiennent au jeu, on obtient que tous les schémas de ligature d'un tel jeu possèdent une pondération valide.

Proposition 3.2.5. *Tout schéma de ligature d'un jeu finiment pondéré admet une pondération valide.*

Démonstration. Pour prouver ce résultat, il suffit de montrer qu'il est possible de déduire de la fonction de pondération des faces une pondération

induite valide du schéma de ligature infini. Un premier point est de remarquer que cette pondération est bien définie pour tout schéma de ligature appartenant au jeu puisque les valeurs prises par la pondération sur les particules sont les mêmes quelque soit la face adjacente considérée.

Ensuite, il ne reste plus qu'à observer que la pondération obtenue satisfait bien la formule (infinie) de Presburger obtenue à la fin du chapitre précédent puisqu'elle satisfait toutes ses sous formules. ■

Bien que très simple, les jeux finiment pondérés sont suffisants pour certaines constructions de système PaCo de complexité maximale. En particulier, nous donnerons un exemple complet d'utilisation de cette méthode dans la section 4.3.

Cette notion peut-être dans un premier temps étendu en ne considérant non plus une seule pondération possible par face mais un ensemble (régulier) de pondérations possibles et en se donnant une méthode permettant de sélectionner la bonne pondération. Dans ce mémoire, la méthode choisie consiste en un automate à compteur compatible avec le graphe des faces. Cette méthode présente un parallèle avec la notion d'automate de cartes présenté dans le chapitre 1. On se fixe un jeu $(C, \mathbb{F})$. Pour chaque élément $F \in \mathbb{F}$, on se donne une suite $(S_i)_{i \in \mathbb{N}}$ de pondérations valides de la face (pour tout $i \in \mathbb{N}$, $S_i \in \mathbb{N}^{\sqsupset(F)}$). Un *automate de pondération* est la donnée, pour tout élément $F \in \mathbb{F}$, d'une fonction $f_F : \sqsupset(F) \times \mathbb{F} \times \mathbb{N} \rightarrow \mathbb{N}$. Intuitivement, la fonction peut être interprétée de la façon suivante : si on utilise la pondération i d'une face F dans un schéma et que cette face est adjacente à une face F' par l'intermédiaire d'une particule p , alors on utilise la pondération $f_F(p, F', i)$ de la face F' . Comme pour l'automate de carte, il est possible de définir la notion de *compatibilité* avec un jeu de face en demandant que la valeur de la fonction ne dépende pas du chemin pris. Cette condition pouvant être vérifiée en se restreignant aux cycles élémentaires qui correspondent dans ce cas, aux faces qu'il est possible de mettre autour d'une collision fixée.

Définition 3.2.6. *Un jeu de carte est déterministe s'il possède un automate de pondération compatible.*

À l'aide de cet automate de face, il est possible de déduire une pondération des éléments du jeu, tant que la valeur de l'automate est définie. Ce problème de définition est inhérent au fait que les ensembles de pondérations valides des faces sont des semi-linéaires et ne possèdent donc pas de chaînes infinies décroissantes. Pour contourner ce problème, un schéma de ligature est *compatible* avec l'automate à partir du point (F, S_i) où F est une face colorée du schéma et S_i une pondération valide de cette face si la fonction de transition locale est partout définie sur le schéma en partant de F munie de la pondération i .

Proposition 3.2.7. *Tout schéma de ligature d'un jeu déterministe compatible avec l'automate à partir d'un point (F, S_i) admet une pondération*

valide.

Démonstration. La preuve de ce résultat est exactement la même que dans le cas finiment pondéré. La pondération obtenue en utilisant l'automate de carte est bien définie et vérifie la formule de Presburger infinie caractérisant les pondérations valides. ■

On peut noter que si un schéma de ligature est compatible avec un automate de pondération à partir d'un point (F, S_i) alors pour toute face de l'automate F' , il existe une pondération valide de cette face S'_j telle que le schéma est compatible avec l'automate en partant de (F', S'_j) (il suffit de prendre pour S'_j la pondération associée à F' en partant de (F, S_i)).

Nous allons maintenant montrer un exemple complet d'utilisation de cette méthode pour construire un automate cellulaire capable de calculer la suite de Syracuse. On considère l'automate cellulaire et son système PaCo donné sur la figure 3.7. Cet automate étant construit de façon *ad-hoc*, il n'est pas étonnant de trouver une table de transition partielle. Le système PaCo comprend 8 particules et 8 collisions. Dans la suite, on considère les 8 faces présentées pour montrer que cet automate peut, sous certaines conditions, calculer la suite de Syracuse.

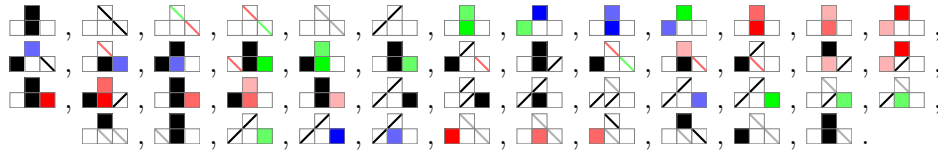
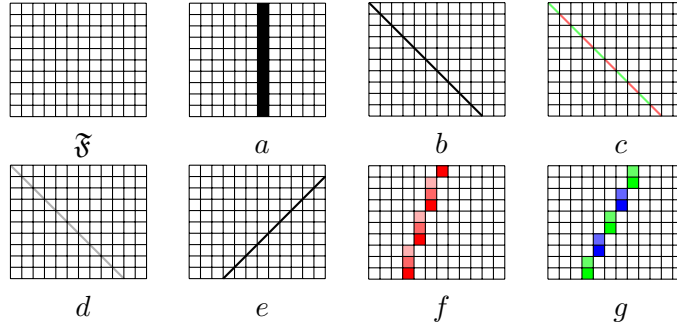
Avant de passer aux schémas de ligature finis, nous pouvons regarder deux schémas de ligature finis qui « correspondent » à une étape de calcul de la suite de Syracuse selon que l'entrée est paire ou impaire (voir figure 3.8).

Il est facile de voir qu'il suffit « d'empiler » les schémas de ligature finis obtenus précédemment pour obtenir le calcul de la suite. En regardant les différentes faces, on remarque que dans la plupart des cas, une valeur de pondération suffit à déterminer entièrement les autres valeurs (cette propriété tient à la forme triangulaire des faces). La seule exception étant la face F_4 mais cette propriété devient vraie si on se restreint à l'ensemble de faces utilisées pour la construction. De ce fait, l'automate de face est simple puisqu'il associe toujours l'unique pondération possible.

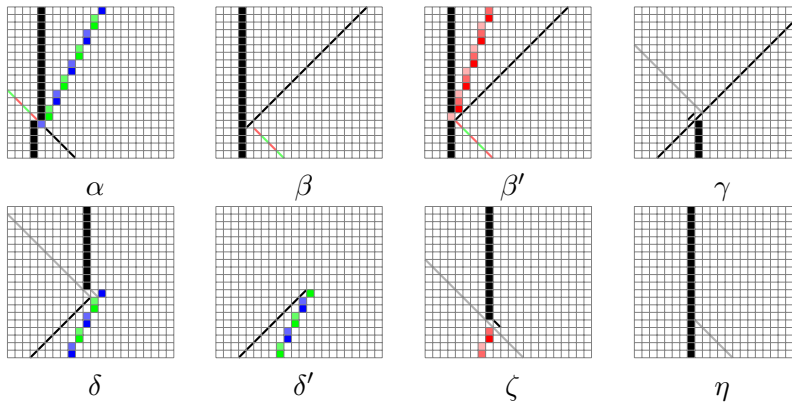
Pour finir, il suffit de vérifier (ce qui est trivial ici) que l'automate de pondération est compatible avec le jeu de faces. Il ne reste plus alors qu'à regarder dans quels cas cet automate est bien défini. Si on parcourt dans le sens du calcul, on observe que l'automate est bien défini tant que le choix de l'étape (pair ou impaire) est correcte et que la valeur encodée est plus grande qu'une borne fixée (par des contraintes de non recouvrement de collisions). On conclut alors que cette automate permet bien d'effectuer des calculs de Syracuse en utilisant le déterminisme des automates cellulaires.

3.2.3 Jeux de faces ordonnés

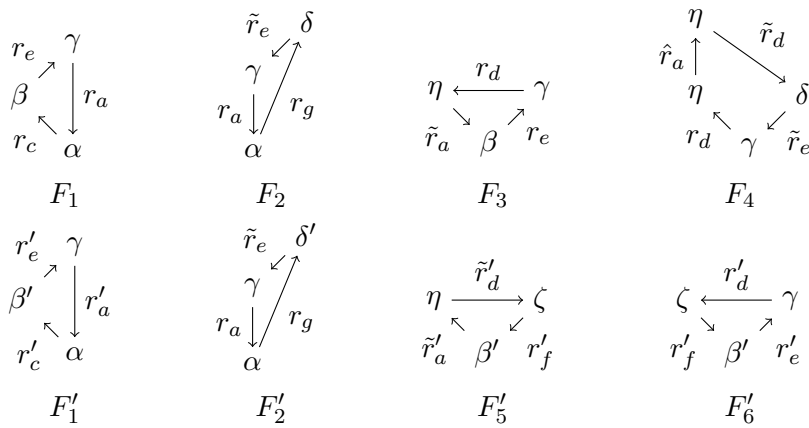
Les différentes méthodes présentées précédemment permettent dans certains cas de donner de façon finie une pondération valide pour un ensemble

(a) Ensembles des transition dont le résultat est différent de $\square$ 

(b) Fond et particule



(c) Collisions



(d) Faces

Fig. 3.7 – Système PaCo de l'automate calculant Syracuse

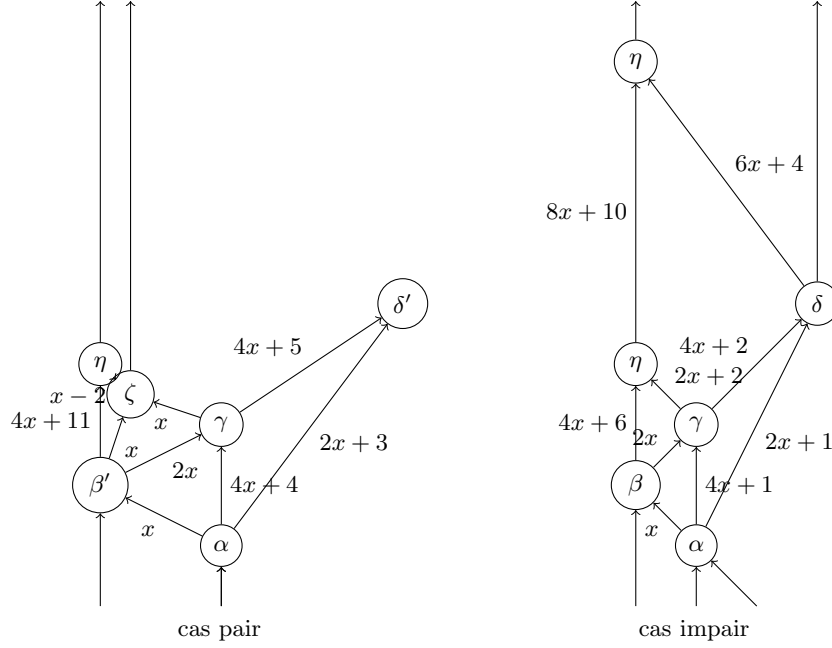


Fig. 3.8 – Schémas de ligature finis pour l'automate calculant Syracuse

de schémas de ligature infinis caractérisé par un jeu de faces. En regardant attentivement, une limitation de ces approches est qu'elle imposent de « fixer » la pondération intégrale de chaque face au moment où on la traite. Cette approche revient intuitivement à se limiter à un linéaire avec un seul générateur pour l'ensemble des pondérations valides de chaque face. Un exemple intuitif consiste à regarder une face rectangulaire. Dans un telle face, il semble préférable de fixer indépendamment la longueur et la largeur. Cette section développe une méthode permettant de répondre à ces attentes.

L'idée intuitive de la méthode consiste à obtenir un ordre partiel sur les arêtes du schéma de ligature tel que l'on puisse construire de façon progressive une pondération suivant cet ordre. Pour cela, les faces colorées du jeu seront munies d'un ordre partiel sur leur particules. Cet ordre est construit de telle sorte que, quelque soit la valeur des éléments minimaux choisis parmi un ensemble fixé, il est possible de fixer des valeurs aux autres éléments de telle sorte que la face soit valide.

Définition 3.2.8. Soit F une face, une orientation élémentaire de F est un couple (E^-, O) où

- $E^- \subseteq \sqsupset(F)$ est l'ensemble des d'arêtes entrantes ;
- $O \subset (E^-, E^+)$ où $E^+ == \sqsupset(Fa) \setminus E^-$ est l'ordre sur les arêtes.

E^+ est appelé ensemble des arêtes *sortantes*. L'ordre induit sur les arêtes est défini par $e < e'$ si $(e, e') \in O$. Une telle orientation est représentée à l'aide flèche pour indiquer l'ordre comme dans la figure 3.9.

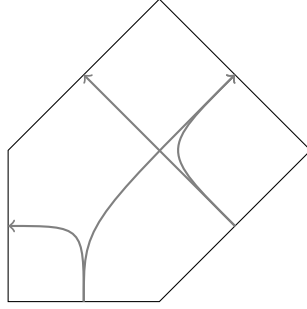


Fig. 3.9 – Exemple d'orientation élémentaire d'une face

À l'aide de cet ordre, il reste à déterminer comment obtenir une pondération valide. Pour cela, on munit chaque arête entrante d'un semi-linéaire indiquant l'ensemble des pondérations possibles et on munit chaque arête sortant d'une fonction linéaire, dépendant uniquement des arêtes inférieures à celle-ci, qui nous donne la valeur de la pondération de cet arête. Il reste alors à exiger que les pondérations obtenues soient valides.

Définition 3.2.9. Une orientation d'une face F est une orientation élémentaire (E^-, O) munie :

- pour tout $e \in E^-$ d'un ensemble semi-linéaire L_e ;
- pour tout $e \in E^+$ d'une fonction linéaire P_e dont les variables sont comprises dans l'ensemble $\{e' | e' < e\}$.

Pour tout $l = (l_e)_{e \in E^-} \in \mathbb{N}^{E^-}$, tel que pour tout $e \in E^-$, $l_e \in L_e$, la pondération $(l, (P'_e(l))_{e' \in E^+})$ est valide.

L'ensemble des valeurs prises par $(P'_e(l))$ est appelé *ensemble de sortie* de l'orientation pour l'arête e' . L'ensemble des pondérations de la forme $(l, (P'_e(l))_{e' \in E^+})$ est appelé ensemble des pondération *considérées* par l'orientation. Comme on travaille avec des ensembles semi-linéaires et des fonctions linéaires, l'ensemble obtenu est semi-linéaire.

Il est possible de caractériser l'ensemble des pondérations considérées dans une orientations à l'aide des générateurs présents dans le semi-linéaire. Pour cela, on introduit la notion d'ensemble *simple* de pondération valide qui va caractériser les degrés de liberté des pondérations. Intuitivement, un tel ensemble sera un ensemble semi-linéaire tel que tout générateur possède un un côté sur lequel il est le seul à avoir une projection non nulle.

Définition 3.2.10. Un ensemble semi-linéaire $S \subset \mathbb{N}^m$ est simple s'il est de la forme $S = \cup_{0 \leq i < k} \left\{ p_i + \sum_{j \in \mathbb{Z}_n} k_i u_i \mid k \in \mathbb{N}^n \right\}$ pour $k, n \in \mathbb{N}$, $p_i, u_i \in \mathbb{N}^m$.

Proposition 3.2.11.

- Les pondérations considérées par une orientation forment un ensemble semi-linéaire simple ;

- Pour tout ensemble semi-linéaire simple de pondérations valides, il existe une orientation dont cet ensemble correspond à l'ensemble des pondérations considérées.

Démonstration. On se fixe une face F et une orientation $(P_e)_{e \in E^+}, (L_e)_{e \in E^-}$. On remarque alors que l'ensemble des pondérations considéré possède $|E^-|$ générateurs qui ne dépendent que de l'élément e de E^- considéré et de l'ensemble des éléments de E^+ supérieurs à e . On en conclut que l'ensemble des pondérations considérées est simple.

Réciproquement, si on se fixe un ensemble simple de pondérations valides, alors on peut choisir pour E^- l'ensemble des arêtes qui caractérisent un générateur. On remarque alors qu'une fois fixées ces valeurs, il n'existe qu'une seule pondération valide dans l'ensemble. On peut donc associer une fonction $F_e : \mathbb{N}^{E^-} \rightarrow \mathbb{N}$ à toutes les arêtes de E^+ . Il n'est pas difficile de voir que cette fonction est linéaire. On conclut en choisissant pour tout $e \in E^-$ le semi-linéaire correspondant aux valeurs possibles des pondérations pour ces arêtes. ■

Il est alors possible d'étendre l'orientation à un jeu de face. Dans ce cas, les contraintes du jeu induisent des contraintes syntaxiques sur les orientations. En particulier, lorsque deux faces peuvent être liées par une arête, il est raisonnable d'imposer que cette arête ne soit pas une arête de sortie simultanément pour les deux faces. De plus, lorsque cette arête est à la fois une arête de sortie, on demande que l'ensemble de sortie soit inclus dans l'ensemble d'entrée. Ces deux conditions assurent une *compatibilité* des orientations avec le jeu.

Définition 3.2.12. *Un jeu de faces ordonné est la donnée d'un jeu $(C, \mathbb{F})$ munie d'une orientation des éléments de $\mathbb{F}$ vérifiant que pour toute arête reliant deux faces, cette arête est soit entrante dans les deux orientations, soit entrante dans l'une et sortante dans l'autre. Dans le premier cas, on exige que les semi-linéaires associées soit non disjoints, dans l'autre cas, on exige que l'ensemble de sortie soit inclus dans le linéaire d'entrée.*

On notera que cette propriété peut être vérifiée de façon finie. On peut également noter que si la définition de jeu ordonné impose de choisir une unique orientation pour chaque face, il est néanmoins possible de « dupliquer » un élément de $\mathbb{F}$ pour obtenir une face avec deux orientations différentes.

On s'intéresse maintenant au cas de schémas de ligature compatibles avec un jeu ordonné. Intuitivement, l'ordre du jeu va nous servir à construire la pondération valide du schéma de ligature. Pour cela, il suffit que le schéma de ligature ne possède pas de cycle ni de chaîne infinie décroissante partant d'une arête.

Définition 3.2.13. *Un schéma de ligature bien ordonné d'un jeu ordonné est un schéma de ligature du jeu dont l'ordre induit est un bon ordre.*

Proposition 3.2.14. *Tout schéma de ligature bien ordonné admet une pondération valide.*

Démonstration. On fixe un schéma de ligature bien ordonné. Il est possible de définir une pondération valide de la façon suivante : pour toutes les arêtes qui ne sont pas sortantes, on attribue une valeur quelconque dans l'intersection des deux ensembles linéaires associées à cette arête (une telle valeur existe bien par définition d'un jeu de faces ordonné). Pour les arêtes sortantes, on fixe la valeur à l'aide de la fonction associée dans la pondération. La propriété de bon ordre nous garantit qu'on affecte une pondération à toute arête après un nombre fini d'étapes. Comme la pondération est valide pour toutes les faces, on obtient bien une pondération valide du schéma de ligature. ■

Un exemple complet d'utilisation de cette méthode est présenté dans la section 4.2.

Malgré des problèmes d'indécidabilité, il est possible de décrire des schémas de ligature infinis et surtout de prouver l'existence de pondérations valides pour de tels ensembles de schémas. Les méthodes présentées dans ce mémoire proposent des pistes pour de telles approches mais laissent un grand nombre de perspectives dans l'étude de ces objets, que ce soit dans la description des schémas de ligature infinis, des méthodes pour prouver l'existence de pondération valide ou la recherche de méthodes pour, plus généralement, étudier l'ensemble de validité de ces schémas.

Chapitre 4

Applications aux automates cellulaires universels

CE chapitre est dédié à l'application des systèmes PaCo à l'étude et la construction d'automates cellulaires complexes. Il est en effet possible de montrer que des liens forts existent entre les systèmes PaCo et la notion de simulation entre automates cellulaires. Ce chapitre présente également deux utilisations des systèmes PaCo pour prouver l'universalité Turing d'un automate cellulaire (Thm. 4.2.1) et construire un petit automate cellulaire intrinsèquement universel (Thm. 4.3.1)

Sommaire

4.1	Universalités	74
4.1.1	Simulation et groupage	74
4.1.2	Liens avec les systèmes PaCo	76
4.2	Universalité Turing de l'automate cellulaire	78
4.2.1	Simulation et système PaCo	79
4.2.2	Jeu de faces associé aux calculs	86
4.3	Un intrinsèque universel à 4 états	89
4.3.1	Construction du système PaCo	89
4.3.2	Jeu de faces finiment pondérées	92

4.1 Universalités

Cette section présente un ordre sur la complexité des comportements présentés par les automates cellulaires : le groupage. Cet ordre est spécifique aux automates cellulaires et présente des liens intéressants avec les systèmes de particules et collisions.

4.1.1 Simulation et groupage

Les automates cellulaires ont ceci de particulier qu'ils agissent sur un ensemble de mots infinis. De ce fait, leur espace d'application diffère sensiblement du cadre habituel (celui des mots finis) et il est donc peu évident de comparer les automates cellulaires aux autres modèles de calcul (comme les machines de Turing). De plus, contrairement à une machine de Turing, un automate cellulaire ne possède pas de condition d'arrêt. La notion de simulation entre ces deux systèmes doit donc prendre en compte ces deux points.

Définition 4.1.1. *Une machine de Turing est un sextuplet $(\Sigma, Q, B, q_0, q_f, \delta)$ où :*

- Σ est un alphabet fini ;
- Q est un ensemble fini d'états ;
- $B \in Q$ est le blanc ;
- $q_0, q_f \in Q$ sont les états initial et final ;
- $\delta : Q \times \Sigma \rightarrow Q \times \Sigma \times \{\leftarrow, \rightarrow\}$ est la fonction de transition.

Une machine de Turing agit sur un ruban biinfini contenant un mot B -fini sur l'alphabet Σ muni d'une unique tête, dans un état $q \in Q$, pointant sur une case du ruban. La dynamique est définie de la façon suivante : soit une configuration dont la tête est dans l'état q et lit la lettre σ telle que $(q', \sigma', d) = \delta(q, \sigma)$. Le successeur de la configuration est obtenu en remplaçant la lettre sous la tête par σ' , en changeant l'état de la tête en q' et en déplaçant la tête vers la gauche (resp. vers la droite) si $d = \leftarrow$ (resp. $d = \rightarrow$). On initialise la machine à l'aide d'un mot fini $w \in \Sigma^*$ en prenant un ruban rempli avec l'état B sur lequel on place le mot w ; la tête est placée sur la première lettre de ce mot dans l'état initial q_0 . On exécute alors des transitions successives de la machine jusqu'à ce que la tête soit dans l'état final q_f ; on dit alors que la machine *s'arrête*. Dans ce mémoire, les machines de Turing sont uniquement vues comme des reconnaisseurs de langages et la notion de résultat du calcul n'est pas introduite.

Pour définir la simulation d'une machine de Turing par un automate cellulaire, il faut d'abord encoder les configurations initiales de la machine de Turing (qui sont des mots finis) en des configurations initiales de l'automate cellulaire. Une approche possible est de se restreindre aux configurations finies de l'automate cellulaire qu'il est possible d'obtenir à l'aide d'une fonction

d'encodage en espace logarithmique. Dans ce mémoire, le cadre utilisé est celui plus générique dans lequel l'ensemble d'arrivée de l'encodage est celui des configurations ultimement périodiques. La condition d'arrêt est donnée par une portion finie de configuration (ou un ensemble fini de portions finies de configuration) et l'automate « s'arrête » si la configuration courante contient une des ces portions finies. Ces choix donnent une notion de simulation similaire à celle définie par B. DURAND et Zs. RÓKA [18].

Définition 4.1.2. *Un automate cellulaire $\mathcal{A} = (Q, r, f)$ simule une machine de Turing sur l'alphabet Σ s'il existe un mot fini $h \in Q^*$ et une fonction en espace logarithmique $f : \Sigma^* \rightarrow (Q^*)^3$ telle que la machine s'arrête sur l'entrée $w \in \Sigma^*$ si et seulement si le mot h apparaît dans l'orbite de l'automate cellulaire sur l'entrée ${}^\omega \text{gcd}^w$ ou $(g, c, d) = f(w)$.*

Définition 4.1.3. *Un automate cellulaire $\mathcal{A}$ est Turing universel s'il peut simuler n'importe quelle machine de Turing.*

Cette définition caractérise la complexité de l'automate cellulaire au sens du calcul Turing. Il existe également une autre caractérisation de la complexité des automates cellulaires à l'aide d'un ordre basé sur l'ensemble des diagrammes espace-temps de l'automate cellulaire. Une première approche a été proposée par I. RAPAPORT à la fin des années 1990 [45]. Cette approche a ensuite été développée par N. OLLINGER [43] pour donner naissance à la notion de *simulation intrinsèque* telle que nous la connaissons actuellement. Cette notion repose principalement sur l'idée d'inclusion des diagrammes espace-temps. Intuitivement, un automate cellulaire $\mathcal{A}$ simule un autre automate cellulaire $\mathcal{A}'$ si l'ensemble des diagrammes espace temps de l'automate $\mathcal{A}'$ peut se « retrouver » dans ceux de l'automate $\mathcal{A}$. Cette notion repose sur une notion d'encodage entre automates cellulaires et une notion de transformation géométrique appelée groupage.

Définition 4.1.4. *Un automate $\mathcal{A}' = (Q', r, f')$ s'encode dans un automate $\mathcal{A} = (Q, r, f)$ (noté $\mathcal{A}' \sqsubseteq \mathcal{A}$) s'il existe une fonction surjective $e : Q \rightarrow Q'$ telle que pour tout $c \in Q'^r$, $f(\tilde{e}^{-1}(c)) \subset e^{-1}(f'(c))$ où $\tilde{e} : Q^r \rightarrow Q'^r$ est l'extension usuelle de e .*

Intuitivement, il semble raisonnable de considérer qu'un automate est moins complexe que l'automate dans lequel il s'encode. On pourra noter que savoir si un automate cellulaire s'encode dans un autre est *NP*-complet. Dans le cas particulier où la fonction e est injective, on parle de *sous-automate*.

Définition 4.1.5. *Le groupé $n \times m$ avec décalage s d'un automate $\mathcal{A} = (Q, r, f)$ est l'automate $\mathcal{A}^{(n,m,s)} = (Q^n, r', f')$ vérifiant pour tout $c \in Q^{\mathbb{Z}}$, $p \in \mathbb{Z}$, $F'((c_{\llbracket kn, (k+1)n-1 \rrbracket})_{k \in \mathbb{N}})(p) = F(c)_{\llbracket kn+s, (k+1)n+s-1 \rrbracket}^m$.*

Le groupé d'un automate cellulaire consiste à regrouper son diagramme espace-temps en prenant des rectangles de taille $m \times n$. Intuitivement, un automate et son groupé présentent la même complexité. Savoir si un automate

est un groupe d'un autre est dans la même classe de complexité que l'isomorphisme de graphes. La notion de simulation entre automates cellulaires s'obtient alors en couplant la notion d'encodage avec celle de groupage.

Définition 4.1.6. *Un automate $\mathcal{A}'$ est simulé par un automate $\mathcal{A}$ (noté $\mathcal{A}' \preceq \mathcal{A}$) s'il existe $m, n, t, m', n', t' \in \mathbb{N}$ tels que $\mathcal{A}'^{(m', n', t')} \sqsubseteq \mathcal{A}^{(m, n, t)}$.*

Cette relation de simulation étant une relation transitive, il est possible d'obtenir un ordre partiel en considérant les classes d'équivalence définies par cette relation. On obtient alors un ordre sur l'ensemble des automates cellulaires qui « mesure » la complexité de tels automates. En étudiant le graphe de l'ordre ainsi obtenu, on remarque qu'il existe une classe maximale pour cette relation [43]. Les éléments de cette classe sont appelés automates cellulaires intrinsèquement universels.

Définition 4.1.7. *Un automate cellulaire est intrinsèquement universel s'il simule n'importe quel automate cellulaire.*

Cette notion de simulation présente l'avantage de caractériser la complexité intrinsèque des automates cellulaires. Il est possible de montrer que de nombreuses sous-classes connues d'automates cellulaires (surjectif, injectifs ...) forment des idéaux ou des filtres de l'ordre [43, 48]. En particulier, il est possible de montrer que la notion d'universalité intrinsèque est plus fine que celle d'universalité Turing.

Proposition 4.1.8 (Ollinger, 2002 [43]). *Il existe un automate cellulaire Turing universel dont la distance à l'ensemble des automates cellulaires intrinsèquement universels est infinie.*

Adapté à l'étude de la complexité des automates cellulaires, la notion de simulation permet de caractériser finement l'étendue de leur comportement. Une question ouverte et structurante à ce sujet est de caractériser la robustesse de la notion d'universalité intrinsèque. En effet, s'il est possible d'obtenir une notion plus fine ou plus large de simulation en jouant sur les contraintes imposées dans l'encodage ou les transformations géométriques possibles dans le groupage, la classe des automates cellulaires intrinsèquement universels semble rester la même. Prouver un tel résultat affirmerait encore plus la validité de cette notion.

4.1.2 Liens avec les systèmes PaCo

La notion de simulation caractérisant la complexité des automates cellulaires par l'intermédiaire des diagrammes espace-temps, il n'est pas étonnant que ces notions soient intimement liées aux systèmes de particules et collisions. Cette section présente quelques résultats de base sur ces liens et offre des pistes pour étudier plus en détails ces liens.

Proposition 4.1.9. *Soient deux automates cellulaires $\mathcal{A}$ et $\mathcal{A}'$ tels que $\mathcal{A}'$ est un sous-automate de $\mathcal{A}$ alors tout système PaCo de $\mathcal{A}'$ est isomorphe à un système PaCo de $\mathcal{A}$.*

Démonstration. Pour montrer ce résultat, il suffit de remarquer que la fonction d'encodage $e : Q \rightarrow Q'$ étant injective, elle induit une injection de $Q'^{\mathbb{Z}^2}$ dans $Q^{\mathbb{Z}^2}$. Cette injection préservant la périodicité, le résultat est immédiat. ■

Dans le cas plus général d'encodage, le résultat ne tient plus. Il se produit alors un phénomène d'apparition de déchets qui présente une perspective intéressante pour la suite.

Proposition 4.1.10. *Soit $\mathcal{A}$ un automate, $m, n, p \in \mathbb{N}$. Tout système PaCo de $\mathcal{A}^{(m,n,p)}$ est isomorphe à un système PaCo de $\mathcal{A}$*

Démonstration. Ce résultat s'obtient directement en utilisant l'injection canonique de $(Q^{m \times n})^{\mathbb{Z}^2}$ dans $Q^{\mathbb{Z}^2}$ puis en remarquant que cette injection reste bien définie modulo translation. ■

Dans la définition d'universalité, la relation de groupage est utilisée pour quotienter les classes. Il est donc sensé de s'intéresser à la réciproque du résultat précédent. Intuitivement, le groupage $\langle m, n, p \rangle$ permet d'introduire dans les diagrammes espace-temps la position à l'origine modulo (m, n) . De ce fait, on fait passer les conditions de validité concernant ce modulo au niveau syntaxique. Ceci implique que le système PaCo du groupé est « plus restrictif » que celui de l'automate de départ. Formellement cette condition se traduit par le fait que la transformation de $Q^{\mathbb{Z}^2}$ dans $(Q^{m \times n})^{\mathbb{Z}^2}$ ne préserve pas les classes d'équivalence par translation. Néanmoins, ce phénomène n'a pas d'influence sur les pondérations valides de schémas de ligature. Une perspective possible sur cet axe est de caractériser formellement le liens entre ces systèmes PaCo. Ces liens sont également visibles sur les automates cellulaires intrinsèquement universels.

Proposition 4.1.11. *Il existe un automate cellulaire universel $\mathcal{A}$ muni d'un système PaCo fini (P, C) tel que toutes les simulations correspondent à des schémas de ligatures infinis du système PaCo.*

Idée de la preuve. Pour prouver ce résultat, il suffit de prendre un automate cellulaire intrinsèquement universel de le simuler par un autre automate (intrinsèquement universel) dans lequel les états sont encodés par des particules et les transitions par des collisions. Pour un tel automate, le groupage n'a pas d'influence sur les particules et collisions utilisées pour la simulation. Comme l'automate intrinsèque considéré simule cet automate, la simulation peut se faire à l'aide de schémas de ligature de particules et collisions de cet automate qui sont isomorphes à un système PaCo de l'automate. ■

En plus de montrer ce résultat, il est possible d'utiliser la preuve précédente pour montrer que les pondérations valides des schémas de ligatures utilisées dans la construction ne font intervenir qu'un nombre fini de pondérations par face et donc peuvent être prouvées valides en utilisant un jeu de faces régulier. Intuitivement, il semble également possible de montrer que lors d'une simulation à l'aide d'un système PaCo, les particules sont conservées au cours du calcul. La formalisation de ce résultat demandant l'introduction de nouvelles notions qui ne sont pas encore mures, il est laissé à l'heure actuelle en tant que conjecture.

En essayant de comparer les systèmes de particules et collisions avec l'universalité Turing, on obtient des résultats négatifs montrant, à quelque part, que ces notions sont disjointes :

Proposition 4.1.12. *Il existe un automate cellulaire Turing universel qui ne possède pas de particule (et par conséquent pas de collision).*

Idée de la preuve. Ce résultat tire son origine de l'existence d'un automate Turing universel dans lequel l'information se propage à une vitesse maximale sous linéaire. ■

Les résultats précédents permettent de montrer que les systèmes de particules et collisions sont des éléments de la complexité des automates cellulaires qui sont distincts de la complexité pour le calcul Turing. Il reste de nombreuses perspectives dans l'étude des liens entre systèmes de particules et collisions et cette section peut se voir comme la prémisse d'une telle étude.

4.2 Universalité Turing de l'automate cellulaire 110

Dans cette section, les systèmes PaCo sont utilisés pour montrer que la règle 110 est intrinsèquement universelle. Cette conjecture avait été énoncée par S. WOLFRAM dans son livre *A new kind of science* [53]. Ce résultat a ensuite été prouvé par M. COOK [14]. La preuve de cette propriété repose sur l'utilisation de particules et collisions, présentées dans le livre de S. WOLFRAM, mais l'absence d'outils pour manipuler ces objets rend la preuve de M. COOK complexe de par le besoin d'une étude combinatoire longue et poussée. Cette section présente une nouvelle version de la preuve reprenant la construction originelle mais utilisant le formalisme de particules et collisions ce qui permet de la rendre beaucoup plus intuitive et de se concentrer sur les points clefs de la simulation. Cette nouvelle preuve a été publiée et présentée lors de la conférence *JAC 2008* [5].

Théorème 4.2.1 (M. COOK, 2004[14]). *La règle 110 est Turing universelle.*

4.2.1 Simulation et système PaCo

L'objectif de cette partie est de présenter comment simuler une machine de Turing à l'aide de l'automate cellulaire 110. Cet automate présente un vaste ensemble de particules et collisions et de nombreux travaux ont été fait pour extraire et analyser les différents objets apparaissant dans l'évolution de cet automate cellulaire. Il n'est donc pas complètement étonnant qu'un tel automate soit capable de simuler n'importe quelle machine de Turing.

Pour prouver que l'universalité Turing, plutôt que de simuler directement une machine de Turing, le choix est d'introduire un système de calcul simple et plus adapté qui nous servira d'intermédiaire : les *systèmes de Post cycliques*. Ces systèmes sont basés sur les système de Post introduits par E. POST dans les années 1940 [44]. Ces systèmes sont définis par une liste cyclique de mots sur l'alphabet binaire $\{0, 1\}$.

Définition 4.2.2. *Un système de Post cyclique est une liste cyclique $(w_i)_{i \in \mathbb{Z}_m}$ où $w_i \in \{0, 1\}^*$ pour tout $i \in \mathbb{Z}_m$.*

Une configuration d'un système de Post cyclique se compose d'un mot fini w' et d'un index $j \in \mathbb{Z}_m$. La dynamique est obtenue de la façon suivante : on regarde la première lettre w'_0 du mot w' . On obtient alors une nouvelle configuration en retirant la première lettre de w' et en ajoutant le $j^{\text{ième}}$ mot de la liste cyclique si $w_0 = 1$. On prend pour index $j + 1$ dans tous les cas. Si le mot obtenu est vide, le calcul s'arrête.

J. COCKE et M. MINSKY [13] ont montré que les systèmes de Post sont capable de simuler n'importe quelle machine de Turing. Dans la suite, deux propriétés supplémentaires concernant cette réduction sont utilisées : il est possible de se restreindre aux système de Post cyclique dont tous les mots ont des tailles multiples de six et pour toute machine de Turing, il existe une constante $K \in \mathbb{N}$ telle que pour toute simulation de cette machine par le système de Post cyclique, on lit au plus K fois la lettre 0 consécutivement. Ces deux restrictions ne sont pas difficiles à obtenir mais elles demandent de regarder en détails le fonctionnement de la simulation des machines de Turing. Si le lecteur est intéressé sur ce point, il peut se référer à [5] qui présente en détails comment obtenir ces restrictions.

La construction est présentée à l'aide du le système PaCo utilisé. Afin de faciliter la lecture et l'intuition, les objets sont définis par des extraits de diagramme espace-temps dans lesquels le motif fini est mis en valeur. Cette représentation présente l'avantage d'être à la fois complète et aussi intuitive.

Le système utilise un unique fond présenté dans la figure 4.1. On peut noter que quoique complexe, ce fond apparaît la plupart du temps si on regarde une évolution de la règle 110 partant d'une configuration aléatoire uniforme.

Dans ce fond, on distingue 18 particules présentés dans la figures 4.2. Pour permettre une meilleure intuition dans la suite, les symboles des collisions

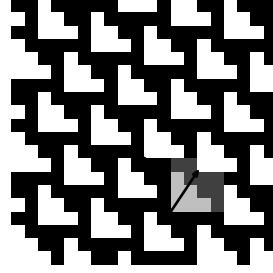


Fig. 4.1 – système PaCo de 110 : fond

comportent une flèche indiquant la direction de leur axe de périodicité. La construction fait également intervenir 23 collisions qui sont représentées dans les figures 4.3, 4.4 et 4.5. La légende permet d'indiquer les particules présentes dans la collision.

Ces éléments servent de base à des signaux encodant les différents éléments du système de Post cyclique. Cet encodage repose sur l'espacement entre les particules qui est pour le moment implicite.

La liste cyclique de mots est encodée en la dépliant pour obtenir une suite de mots infinis et en indiquant par un signal le début de la liste et la séparation entre deux mots. Ceci est fait à l'aide de 4 signaux : pour le début de la liste, le signal $\overleftarrow{S} = \overleftarrow{i} \overleftarrow{d_1 d_4} \overleftarrow{i}^4 \overleftarrow{i_2}$; pour la séparation de mots, le signal $\overleftarrow{D} = \overleftarrow{d_5 d_2 d_1 d_4} \overleftarrow{i}^4 \overleftarrow{i}$ et pour les lettres, les signaux $\overleftarrow{F}_x = (\overleftarrow{i} \overleftarrow{i_2})^4$ (l'encodage de la lettre se faisant dans l'espacement).

Les lettres situées dans la file sont encodées par deux signaux : un signal temporaire $\overleftarrow{W}_x = \overleftarrow{i}^4$ qui sert à placer la lettre puis le signal $\overleftarrow{C}_x = \overleftarrow{c}^4$. Lors de l'analyse de la lettre, deux signaux indiquent si l'on doit effacer (signal $\overrightarrow{B} = \overrightarrow{b_2}$) ou copier (signal $\overrightarrow{P} = \overrightarrow{p}$) le mot de la liste.

Il reste à introduire un signal d'horloge $\overrightarrow{T} = \overrightarrow{s}^4$ et d'un signal contenant des déchet $\overleftarrow{J} = \overleftarrow{i}^2$ ainsi que des versions légèrement altérées de certains signaux : $\tilde{F}_x = (\overleftarrow{i} \overleftarrow{i_2})^3 (\overleftarrow{i} \overleftarrow{i})$, $\tilde{S} = \overleftarrow{i} \overleftarrow{d_1 d_4} \overleftarrow{i}^4 \overleftarrow{i}$, $\tilde{D} = \overleftarrow{d_5 d_2 d_1 d_4} \overleftarrow{i}^4 \overleftarrow{i}$, $\tilde{P} = \overrightarrow{a_3 a_7}$ and $\tilde{B} = \overrightarrow{b_1}$.

L'ensemble des collisions peut être enrichi pour prendre en compte les signaux construits. La liste des collisions ainsi construite est donnée dans la figure 4.6. Pour déterminer ces collisions, il suffit de donner le schéma de ligature fini correspondant. Ces schémas de ligatures sont donnés dans la figure 4.7 à l'exception des schémas de ligatures faisant intervenir des signaux altérés car ils sont similaires au cas non altéré : $\tilde{\mathfrak{C}}$ est exactement comme $\mathfrak{C}$, $\tilde{\mathfrak{S}}_0$ diffère juste par l'utilisation de la collision g'_8 au lieu de f' (il en va de même pour $\tilde{\mathfrak{M}}_0$). De la même façon, $\tilde{\mathfrak{S}}_1$ et $\tilde{\mathfrak{M}}_1$ sont obtenus en remplaçant la collision d' situé en haut par w'_1 . Finalement, $\mathfrak{E}_0$, $\tilde{\mathfrak{E}}_0$, $\mathfrak{E}_1$ et $\tilde{\mathfrak{E}}_1$ sont composés d'une seule collision.

Avec ces données, il est possible de trouver des valeurs pour les espace-

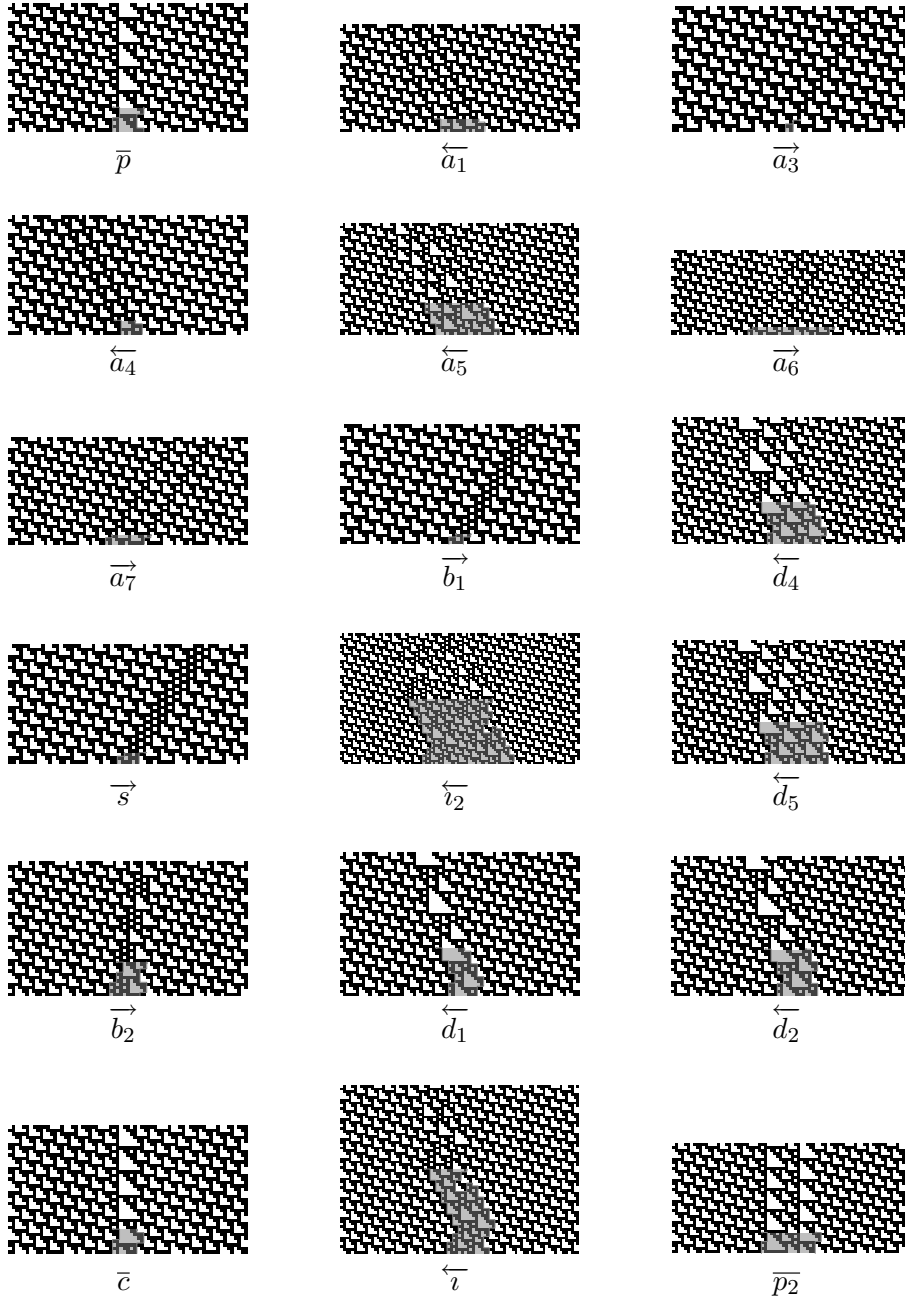


Fig. 4.2 – Particules utilisées dans la construction

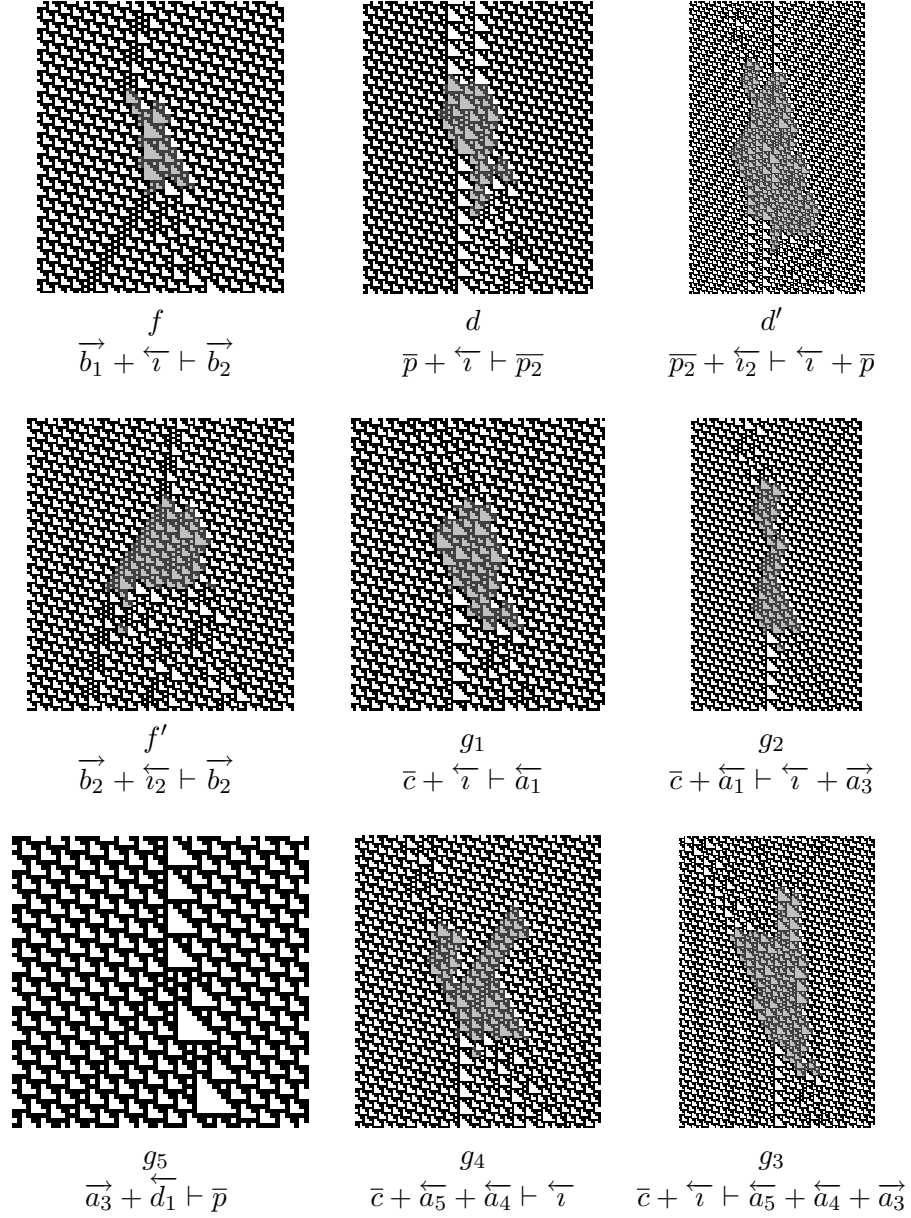


Fig. 4.3 – Ensemble de collisions de base

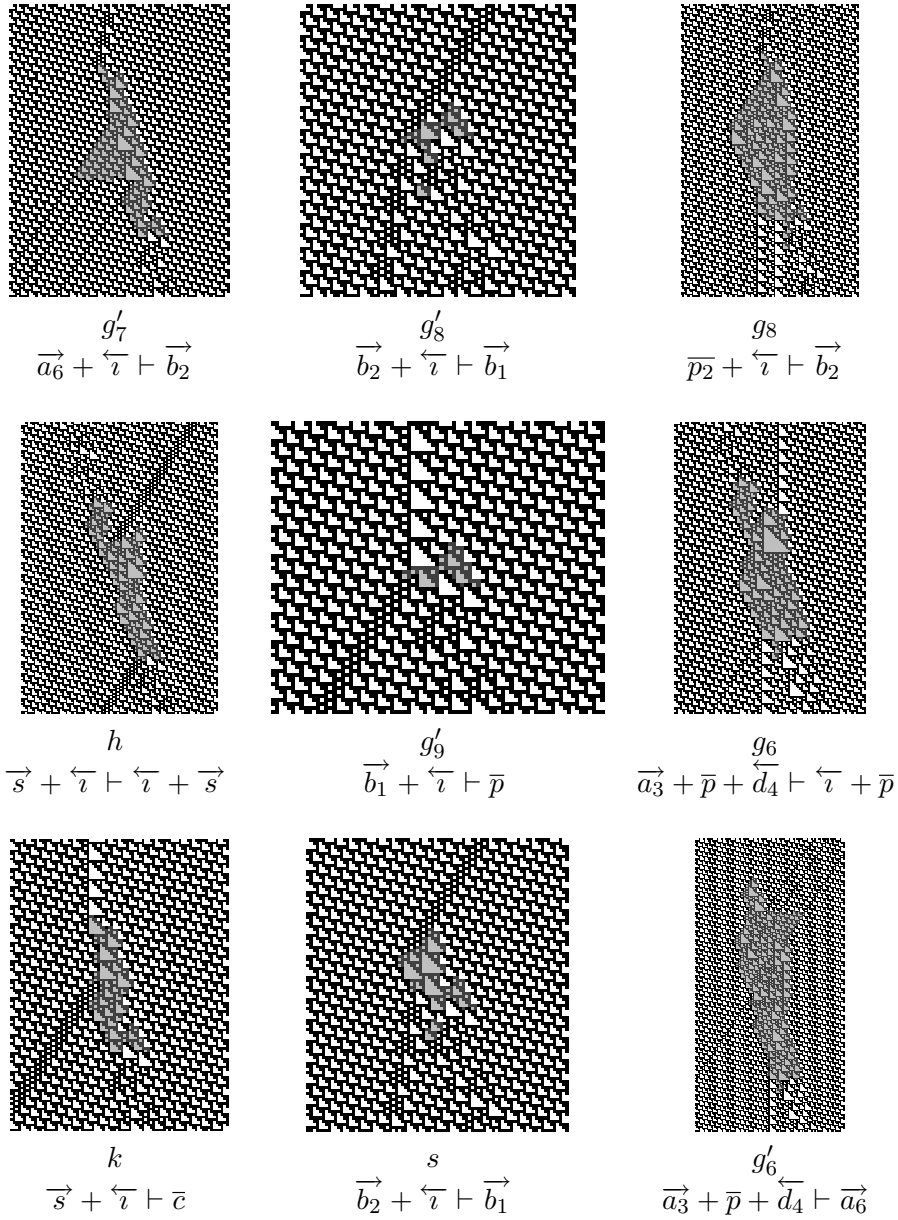


Fig. 4.4 – Ensemble de collisions de base (suite)

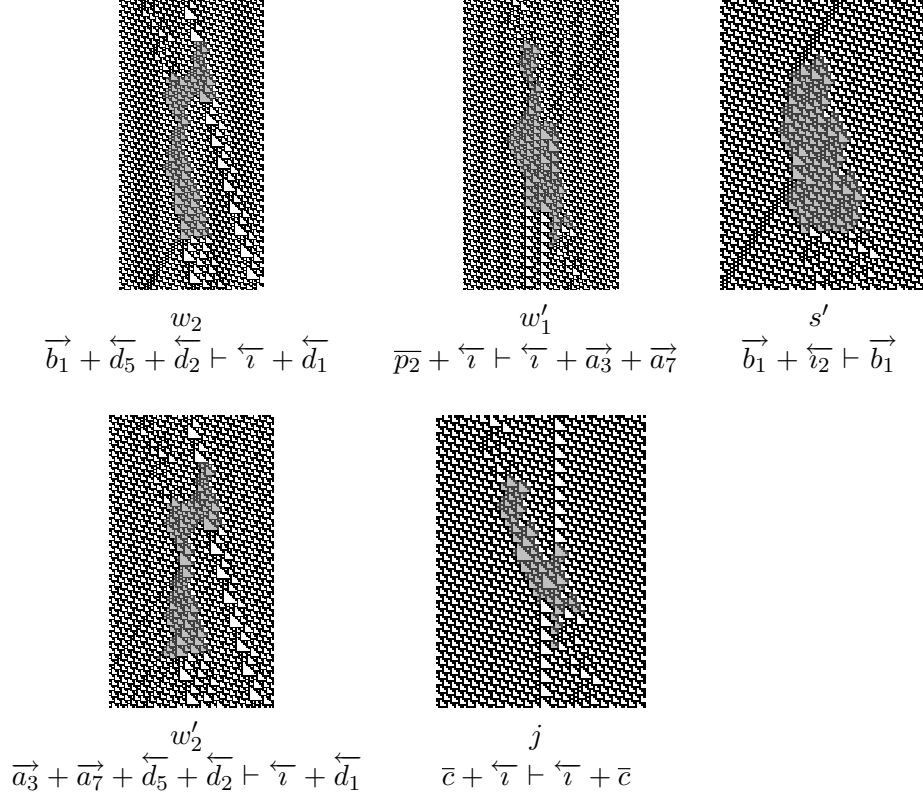


Fig. 4.5 – Ensemble de collisions de base (fin)

$\mathfrak{R}$	$\vec{T} + \overleftarrow{W}_x \vdash \overleftarrow{C}_x$	
$\mathfrak{C}'$	$\vec{T} + \overleftarrow{J} \vdash \overleftarrow{J} + \vec{T}$	
$\mathfrak{C}$	$\overleftarrow{C}_x + \overleftarrow{W}_y \vdash \overleftarrow{W}_y + \overleftarrow{C}_x$	$\tilde{\mathfrak{C}}$ $\overleftarrow{C}_x + \overleftarrow{J} \vdash \overleftarrow{J} + \overleftarrow{C}_x$
$\mathfrak{S}_0$	$\overleftarrow{C}_0 + \overleftarrow{S} \vdash \overleftarrow{J} + \overleftarrow{B}$	$\tilde{\mathfrak{S}}_0$ $\overleftarrow{C}_0 + \tilde{S} \vdash \overleftarrow{J} + \tilde{B}$
$\mathfrak{S}_1$	$\overleftarrow{C}_1 + \overleftarrow{S} \vdash \overleftarrow{J} + \overleftarrow{P}$	$\tilde{\mathfrak{S}}_1$ $\overleftarrow{C}_1 + \tilde{S} \vdash \overleftarrow{J} + \tilde{P}$
$\mathfrak{M}_0$	$\overleftarrow{B} + \overleftarrow{F}_x \vdash \overleftarrow{B}$	$\tilde{\mathfrak{M}}_0$ $\overleftarrow{B} + \tilde{F}_x \vdash \tilde{B}$
$\mathfrak{M}_1$	$\overleftarrow{P} + \overleftarrow{F}_x \vdash \overleftarrow{W}_x + \overleftarrow{P}$	$\tilde{\mathfrak{M}}_1$ $\overleftarrow{P} + \tilde{F}_x \vdash \overleftarrow{W}_x + \tilde{P}$
$\mathfrak{E}_0$	$\tilde{B} + \overleftarrow{D} \vdash \overleftarrow{S}$	$\tilde{\mathfrak{E}}_0$ $\tilde{B} + \tilde{D} \vdash \tilde{S}$
$\mathfrak{E}_1$	$\tilde{P} + \overleftarrow{D} \vdash \overleftarrow{S}$	$\tilde{\mathfrak{E}}_1$ $\tilde{P} + \tilde{D} \vdash \tilde{S}$

Fig. 4.6 – Collisions faisant intervenir des signaux

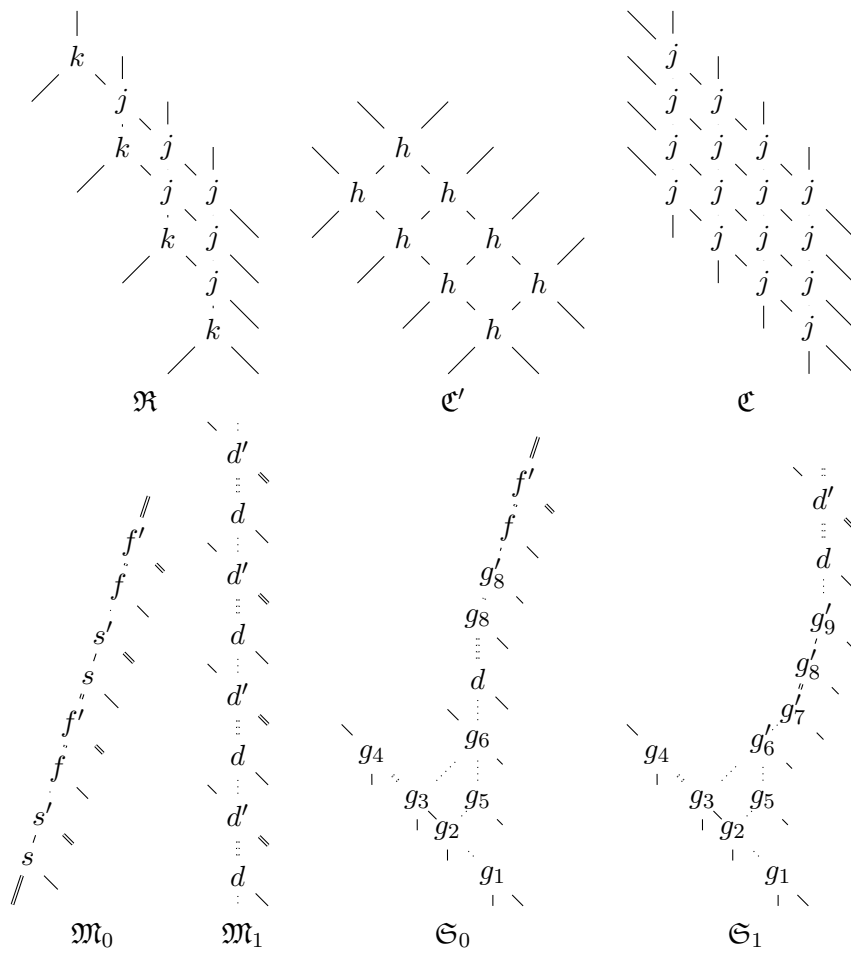


Fig. 4.7 – Constructions des collisions associées aux signaux

ment des signaux qui sont compatibles avec toutes les collisions souhaitées. Il suffit de résoudre tous les schémas de ligature fini et de remarquer que l'intersection des solutions est non vide. Une fois construit ce système, il ne reste plus qu'à l'utiliser pour effectuer la simulation du système de Post cyclique.

4.2.2 Jeu de faces associé aux calculs

Cette section décrit le fonctionnement de la simulation et montre, à l'aide d'un jeu de faces orientée, que le système symbolique peut réellement être implémenté dans l'automate cellulaire.

Le fonctionnement de la simulation est résumé dans la figure 4.8. Pour encoder une configuration du système de Post cyclique, on procède de la façon suivante : au centre, on encode le contenu de la file de la droite vers la gauche à l'aide de signaux $\overrightarrow{C}_x$. À droite, on place la liste cyclique dépliée à l'aide de l'alphabet $\overleftarrow{F}_x$. Les mots sont séparés par des signaux $\overleftarrow{D}$ et on trouve un signal $\overleftarrow{S}$ avant le premier mot. Pour chaque mot, on remplace la dernière lettre par une version altérée du signal (si le mot est vide, on mets une version altérée du délimiteur précédent). Sur la gauche, on mets une infinité de signaux d'horloge $\overrightarrow{T}$. On peut noter que la configuration ainsi obtenue est ultimement périodique.

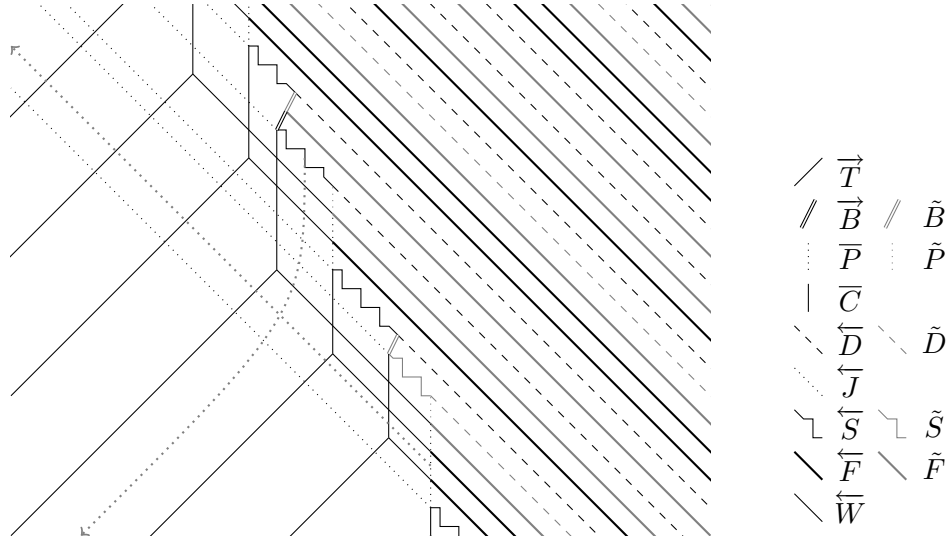


Fig. 4.8 – Fonctionnement symbolique de la simulation et orientation

Le fonctionnement dynamique de la simulation est le suivant : la première collision est celle entre la première lettre de la file et le signal de début de mot. Cette collision produit efface les deux signaux entrant et produit un signal d'effacement $\overrightarrow{B}$ (collision $\mathfrak{S}_0$) ou un signal de copie $\overrightarrow{P}$ (collision

$\mathfrak{S}_1$) selon la lettre d'entrée. Ce signal rencontre ensuite toutes les lettres du premier mot de la liste et les efface ($\mathfrak{M}_0$) ou les transforme en lettre de la pile ($\mathfrak{M}_1$) toujours selon la première lettre du mot de la file. Lorsque le signal de copie ou d'effacement rencontre la dernière lettre du mot, la version altérée du signal encodant cette lettre altère également ce signal ($\tilde{\mathfrak{M}}_0$ or $\tilde{\mathfrak{M}}_1$); le signal résultant rencontre finalement le séparateur de mot qu'il transforme en nouveau début de la liste ($\mathfrak{E}_0$ or $\mathfrak{E}_1$). Si la première lettre de la file était 1, les lettres générées traversent toute la file ($\mathfrak{C}$) avant d'être ajouté à la fin de la queue lors de la rencontre avec un top d'horloge $\vec{T}$ (collision $\mathfrak{R}$). On peut également noter que certaines de ces collisions génèrent des signaux de déchets $\overleftarrow{J}$ qui s'évacuent par la gauche sans perturber la simulation. Après ces étapes, le système est de nouveau prêt pour une nouvelle transition.

Une fois construit des schémas de ligature, il reste à prouver que ces schémas correspondent bien à des diagrammes espace-temps valide de l'automate cellulaire. Ceci est obtenu en utilisant le jeu de face orienté tel que présenté dans la figure 4.8

Dans ce jeu de face orienté, on peut remarquer la présence d'une famille infinie de faces ayant un nombre arbitrairement grand de côté (voir figure 4.9). Lorsque l'on étudie l'ensemble de validité de ces faces, on remarque qu'il existe une pondération valide que dans le cas où le nombre de signaux $\overleftarrow{F}_x$ incidents est un multiple de 6. Cette contrainte explique la restriction portée sur le système de Post cyclique simulé.

En remarquant que les schémas de ligature correspondant aux calculs valide du système PaCo associées sont bien ordonné, on en déduit l'existence d'une pondération valide pour tous les calculs en appliquant la proposition 3.2.14.

Ce résultat n'est néanmoins pas suffisant pour obtenir la simulation. Il manque encore un point important qui consiste à montrer que l'encodage initial des valeurs ne dépend pas du calcul effectué. Pour cela, il suffit de montrer que les pondérations valides des faces externes ne dépendent pas des autres faces. Ici, il faut distinguer deux cas : le cas des espacement entre les mots encodant la liste cyclique du système de Post est simple car tous les côtés intervenant dans les faces concernés sont des élément minimaux de notre jeu de face et peuvent de ce fait, être choisis arbitrairement parmi les valeurs du semi-linéaire associé. Il suffit de vérifier que ces ensembles ne sont pas disjoints. Le cas des signaux d'horloge est un peu plus complexe : leur taille dépend d'une arête entrante ainsi que du nombre de symboles effacés depuis la précédente copie. Les arêtes entrantes sont toutes distinctes et n'ont pour le moment aucune contrainte. Pour obtenir une pondération des faces externes concernées de façon uniforme, il faut supposer le nombre d'effacements successifs possible borné (ce qui est la deuxième condition posée sur système de Post cyclique). Avec cette condition, il est possible de choisir la valeur de l'arête entrante de façon à compenser le décalage introduit

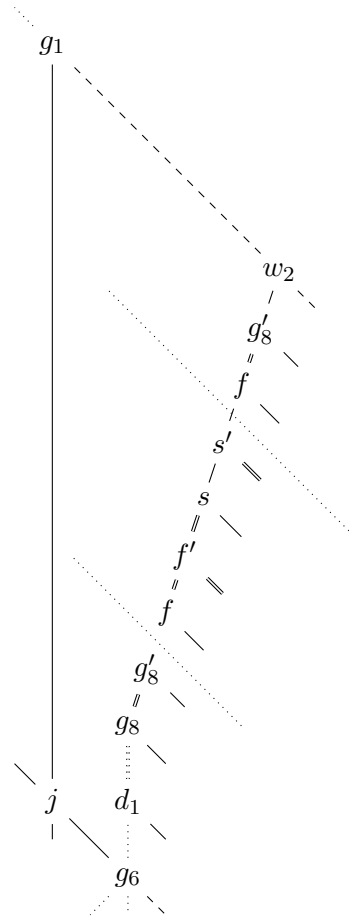


Fig. 4.9 – Famille infinie de face utilisée dans le construction de la règle 110

par les effacements.

Cette dernière étape nous permet donc de conclure à l'universalité Turing de la règle 110. L'utilisation des PaCo permet de suivre de très près l'intuition et de reléguer au second plan les problèmes combinatoires dans sa mise en pratique. Une fois ce résultat obtenu, la question de l'universalité intrinsèque d'une telle règle se pose. Sur ce point, la conjecture sur la conservation des particules amène à penser que la système PaCo utilisé ne peut l'être pour prouver l'universalité intrinsèque. La question de savoir si cette règle est intrinsèquement universelle est toujours ouverte.

4.3 Un intrinsèque universel à 4 états

Dans la section précédente, les systèmes PaCo ont servis, partant d'un automate avec un comportement riche, de montrer que celui-ci est capable de calculs Turing. Dans cette partie, cette approche est utilisée pour construire de petits automates complexes. Plus particulièrement, cette section est dédiée à construire un automate cellulaire intrinsèquement universel à quatre états qui est à ce jour, le plus petit automate cellulaire intrinsèquement universel connu. Ce résultat fait l'objet d'une soumission à CSP 2008 [3]. Dans ce mémoire, la construction de l'automate est donnée en partant du système PaCo pour aller vers la simulation dynamique ; cette construction nous permet d'affirmer le résultat suivant :

Théorème 4.3.1 (G. R.). *Il existe un automate cellulaire intrinsèquement universel de rayon 1 et avec 4 états.*

4.3.1 Construction du système PaCo

Comme la plupart des automates qui sont construit explicitement, la règle de transition de cet automate à 4 états peut être divisée en différents cas qui servent chacun à une fonction bien précise. Les transitions nécessaires sont donc introduites au fur et à mesure des particules et collisions construites.

Contrairement aux autres constructions usuelles de petits automates, cet automate cellulaire n'utilise pas un fond principal uniforme mais un fond en damier (voir Fig. 4.10). Ce fond particulier utilise deux états au lieu d'un mais nous permet facilement de construire de nombreuses particules avec les états restants : en effet, pour chaque état, il est possible de construire deux particules avec des directions différentes selon la position relative par rapport au damier (voir Fig. 4.10). De plus, on remarque que le fond possède deux « calages » possibles. En utilisant cette propriété, il est possible de créer une particule verticale tirant profit d'un défaut d'alignement entre deux fonds.

Cette construction permet de vérifier plusieurs propriétés très intéressantes au niveau des particules. En particulier, Comme les particules sont très fines (leur vecteur de périodicité est $(1, 1)$, $(0, 1)$ ou $(-1, 1)$), il n'existe

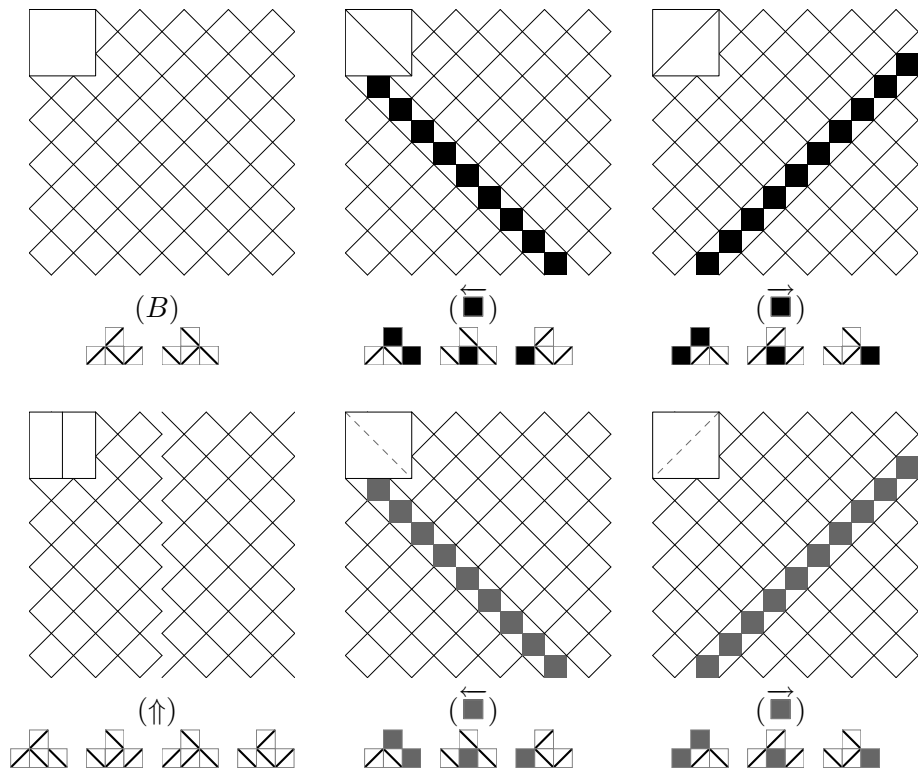


Fig. 4.10 – Le fond et les particules principales de l'automate

qu'une seule possibilité de rencontre entre deux particules. Ceci entraîne un « déterminisme » des collisions.

Comme la particule verticale est la seule à induire un décalage du fond, elle ne peut être ni créée, ni détruite dans les collisions. De ce fait, les collisions entre deux particules obliques sont susceptibles de donner naissance uniquement à d'autres particules obliques. Sous ces hypothèses, on remarque que les transitions utilisées pour définir ces collisions sont disjointes et qu'il est donc possible de définir ces collisions indépendamment.

Il est alors possible de définir trois de ces collisions entre particules diagonales ainsi que deux collisions faisant intervenir la particule verticale et une particule oblique. Ces collisions sont présentées dans la figure 4.11.

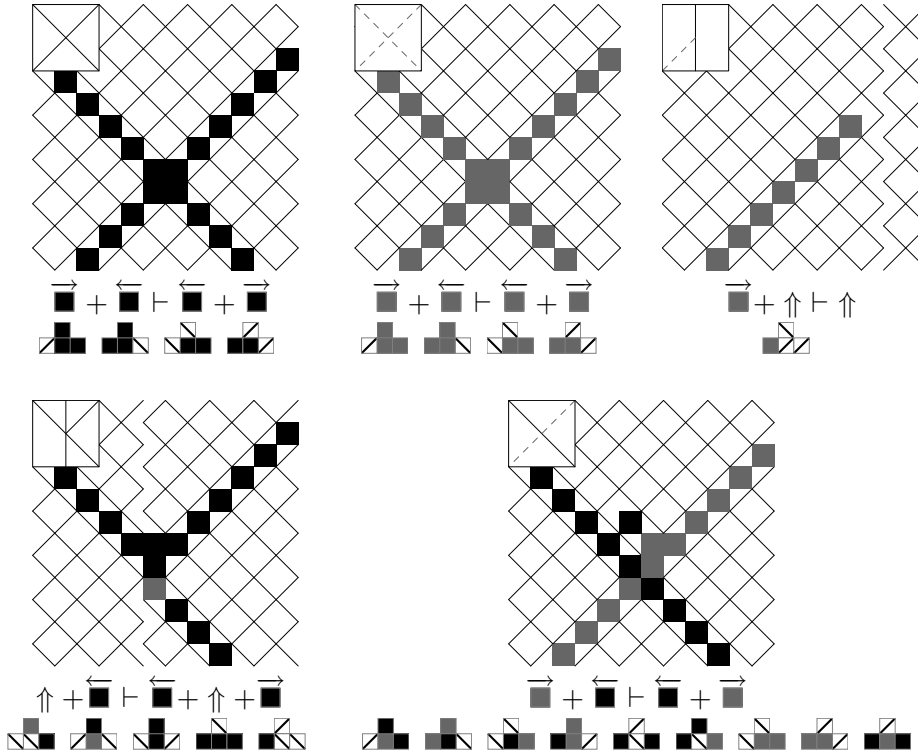


Fig. 4.11 – Collisions de base

La construction de cet automate se heurte alors à la dualité fond, particule : intuitivement, on souhaite utiliser des signaux encodant par la nombre de répétitions de particules régulièrement espacés. De tels objets peuvent soit être vu comme n répétitions de particules ou alors comme un fond situé entre deux particules d'espacement n . Ici les deux types seront utilisés pour les signaux l et m décrits dans la figure 4.12. Pour le cas de l , l'espacement entre deux particules étant trop faible, la limite de nos définitions de signaux est atteinte et oblige à utiliser la vision de fond. Cette vision a pour avantage d'éviter des schéma de signaux avec des familles infinies de face mais au prix

d'introduction de plus de particules et collisions.

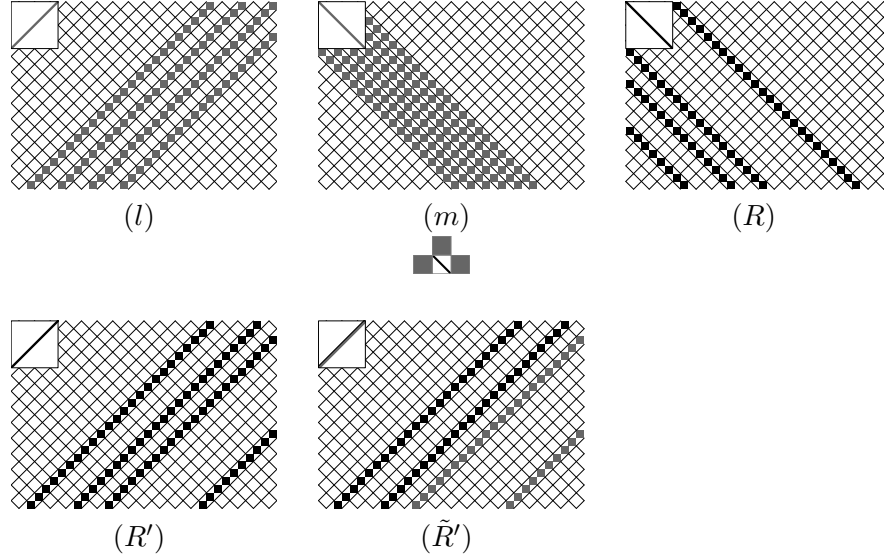


Fig. 4.12 – Signaux

À l'aide de schéma de signaux, il est possible de construire un ensemble de collisions qui va nous servir dans la suite pour la simulation. Cet ensemble comporte 7 collisions qui sont présentés dans la figure 4.13.

Une fois ce système PaCo construit, nous allons montrer comment l'utiliser pour la simulation de n'importe quel automate cellulaire. Avant cela, nous pouvons récapituler la règle de l'automate cellulaire construit en donnant explicitement sa table de transition sur la figure 4.14. On remarquera qu'il y a dans cette table de transition des transitions non définies. Ces transitions ne sont pas utilisées dans la simulation et peuvent donc être choisie de façon arbitraire.

4.3.2 Jeu de faces finiment pondérées

Pour montrer que l'automate construit est intrinsèquement universel, il suffit de prouver qu'il est capable de simuler n'importe quel automate cellulaire de rayon 1 ne prenant pas en compte son dernier argument. Il est connu que tout automate cellulaire peut être simulé par un tel automate avec cette restriction. On se fixe donc un ensemble d'état S et une fonction de transition $\delta : S^2 \rightarrow S$.

Cette simulation passe par la construction de schémas de ligatures capable de simuler une cellule d'un automate cellulaire. Une valeur $N \in \mathbb{N}$ étant fixée, on encode, dans le signal R , N valeurs $(r_i)_{i \in \mathbb{N}_m}$. On construit alors le schéma de ligature présenté dans la figure 4.15. On peut regarder la dynamique d'un tel schéma en considérant comme entrée les trois signaux R , l et m .

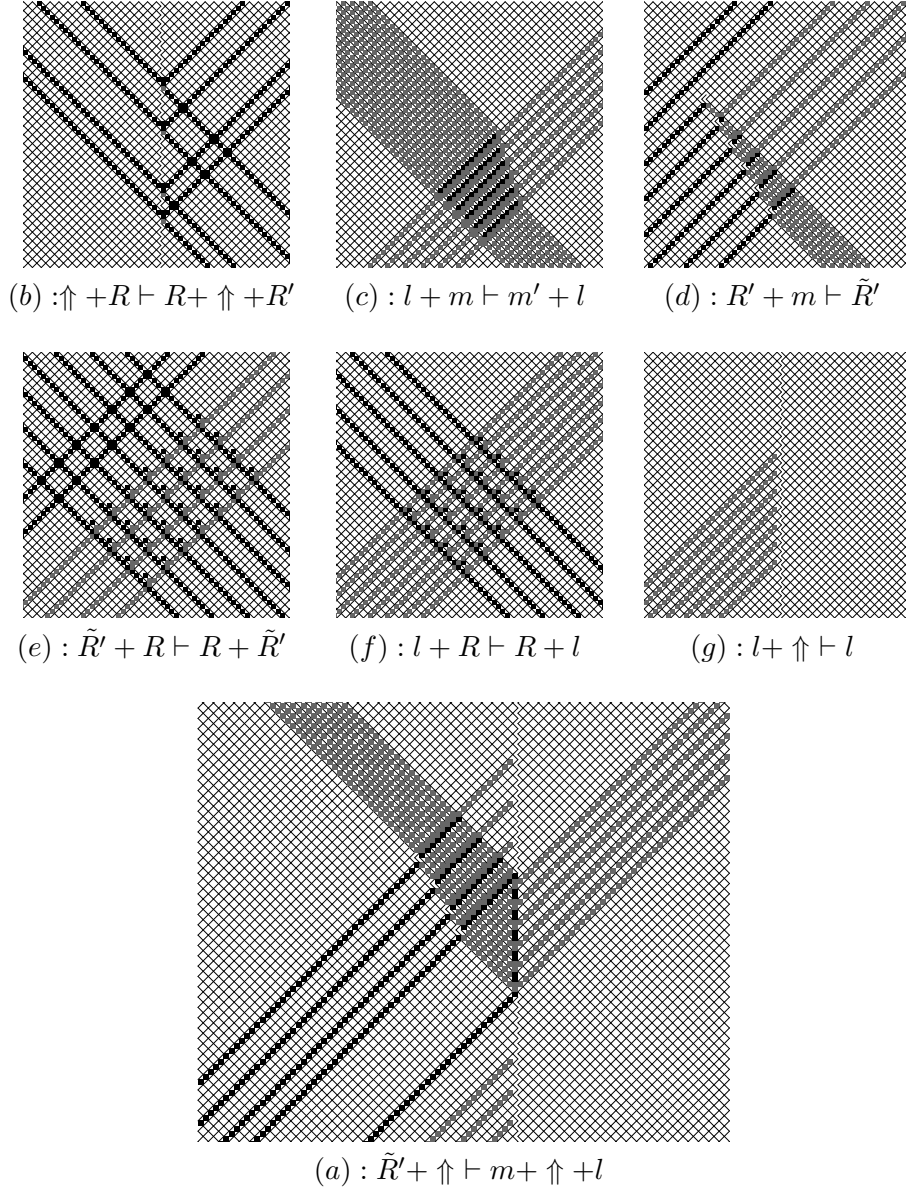
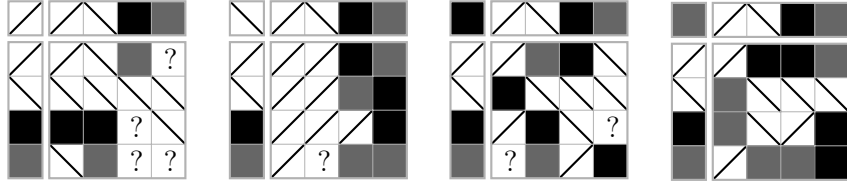


Fig. 4.13 – Collisions entre signaux



L'intersection de la ligne i et de la colonne k du tableau j représente $f(i, j, k)$

Fig. 4.14 – Table de transition

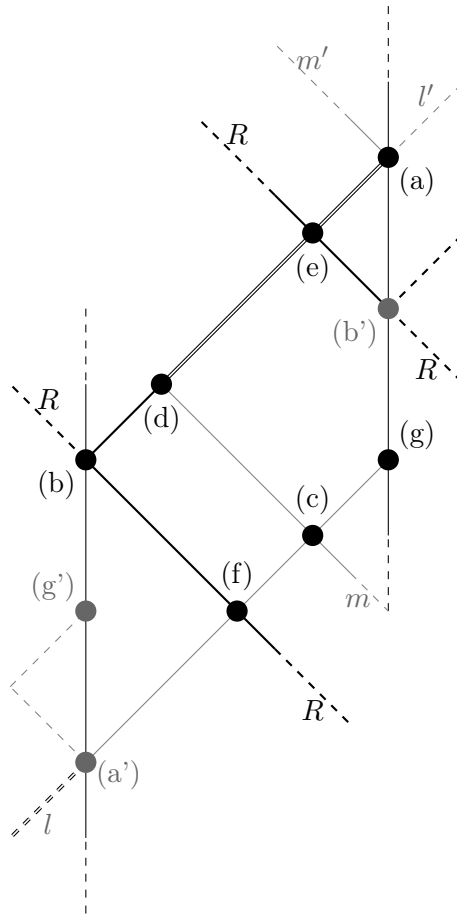


Fig. 4.15 – Une cellule de base

La dynamique est alors la suivante :

- le signal l croise le signal R dans la collision f ;
- le signal l rencontre le signal m donnant naissance à un nouveau signal de type m encodant la somme des deux et un signal de type l qui sera un déchet (collision c) ;
- le signal de déchet est effacé par la particule $\uparrow$ lors de la collision g ;
- De son côté, le signal R est dupliqué lors de la rencontre avec la particule $\uparrow$ produisant un nouveau signal R' (collision b) ;
- Ce nouveau signal rencontre le signal encodant la somme des deux entrées dans la collision d , il en résulte une version altérée $\tilde{R}'$ dans laquelle un nombre d'intervalles égal à la valeur encodée sont transformés ;
- Ce signal altéré traverse le signal R intact (collision e) et rencontre finalement la particule $\uparrow$ (collision a). Dans cette collision, les intervalles altérés sont effacés, le premier intervalle intact génère deux signaux l' et m' dont la valeur dépend de la taille du dit intervalle. Les intervalles supplémentaires sont effacés mais entraînent néanmoins une modification de la valeur du signal m' .

Lors de l'analyse de la cellule, on remarque que les signaux R ne sont jamais modifiés. Vu sous cet angle, la cellule prend en entrée deux signaux l et m et donne en sortie deux autres signaux l' et m' . Il reste alors à vérifier l'existence d'éléments dans l'ensemble de validité de ces schémas. À R fixé, il est possible de construire un jeu de face finiment pondéré englobant toutes les cellules pour dont les valeurs de signaux l et m vérifient que $m + l < N$. Ce jeu permettant d'assurer l'existence d'une pondération valide commune.

L'idée intuitive est alors de montrer comment encoder la règle dans le signal R pour obtenir la simulation de la fonction de transition f de l'automate cellulaire simulé. Cette idée va se heurter à un problème lié au fait que les valeurs émises l' et m' sont différentes d'une valeur dépendant du calcul effectué. Pour contourner, il est nécessaire d'introduire encore une étape supplémentaire de raisonnement. Cette fois, les cellules sont groupées par deux (voir figure 4.16) et l'encode porte uniquement sur des bits spécifiques de chaque élément. Le détail de cet encodage est présenté dans la figure 4.17. Dans cet encodage, les $\perp$ indiquent des portions non codantes qui peuvent être choisies arbitrairement. L'idée est qu'il est alors possible de choisir $R(l + m)$ pour encoder complètement la règle : les portions données dans la figure servent à fixer les portions codantes dans l' , les $\bullet$ peuvent être choisis pour fixer les valeurs de m' .

À l'aide de cet encodage, on montre donc que l'automate cellulaire construit est bien intrinsèquement universel. On peut néanmoins noter que l'encodage ne nous permet pas de conclure à l'universalité intrinsèque pour la notion de sous-automate stricte. Il semble exister une méthode permettant d'obtenir une simulation pour cette condition supplémentaire mais sa mise en place étant longue et complexe et n'apportant que peu au résultat final, nous ne l'avons pas mise en pratique. La question de savoir si ces deux

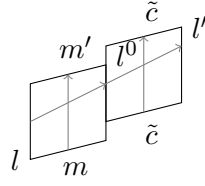


Fig. 4.16 – Groupement de cellules

	tête (	bits				) pied	
N	1111	(11	11	11	11)	111	
l	0101	($\perp\perp$	s_i0	$\perp\perp$	00)	$\perp\perp\perp$	
m	0000	($\perp\perp$	00	$\perp\perp$	s'_i0)	$\perp\perp\perp$	
$\tilde{c}$	1000	(00	00	00	00)	000	
$l + m$	01xx	($\perp\perp$	$s_i\perp$	$\perp\perp$	$s'_i\perp$)	$\perp\perp\perp$	où $xx = 01$ ou 10
$R(l + m)$	10 ●●	($\bar{t}_i1$	●●	11	●●)	100	
m'	0000	($\perp\perp$	00	$\perp\perp$	t_i0)	$\perp\perp\perp$	
l^0	010 $\perp$	($\perp\bar{t}_i$	$1\perp$	$\perp1$	$1\perp$)	$\perp10$	
$N - l^0 + 1$	101 $\perp$	($\perp t_i$	$0\perp$	$\perp0$	$0\perp$)	$\perp10$	
l'	0101	($\perp\perp$	t_i0	$\perp\perp$	00)	$\perp\perp\perp$	

Fig. 4.17 – Encodage des valeurs pour une transition $\delta(s, s') = t$

notions d'universalité intrinsèque coïncident reste néanmoins une question importante pour la compréhension du calcul.

Perspectives

MOTIVÉ par l'observation des phénomènes d'auto-organisation dans les automates cellulaires, ce mémoire propose des définitions de fond, particule et collision reposant sur une approche bidimensionnelle. Ces objets correspondent à ceux utilisés dans les constructions d'automates cellulaires avec des comportements complexes, tout particulièrement dans le cadre du calcul par signaux. Cette dualité s'exprime par la présence de deux façon alternatives de définir ces objets : une façon simple et géométrique et une façon algorithmique donnant des représentations finies et récursivement énumérables. De plus, il est possible de montrer que ces notions coïncident avec des colorations simples du plan. Ces colorations simples sont définies en introduisant la notion d'automate de carte à compteur qui correspond à une extension possible en dimension deux de la notion de langage régulier tel que défini à l'aide d'automates finis à compteurs. Ces résultats donnent une justification supplémentaire à l'étude et l'utilisation de ces notions.

Une fois ces notions définies, il est possible de leur associer une représentation symbolique continue basée sur la vision utilisée dans le calcul par signaux. Cette représentation permet d'introduire une opération syntaxique d'assemblage des particules et collisions afin de construire de nouveaux objets : les schémas de ligature. Cette opération correspond à l'idée intuitive utilisée dans de nombreuses constructions d'automates cellulaires. Il est possible d'associer une coloration à tout schéma de ligature. Le principal résultat de ce mémoire énonce que dans le cas des schémas de ligature finis, il est possible de déterminer algorithmiquement si la coloration associée au schéma de ligature existe réellement. Ce résultat implique qu'il est désormais possible de se concentrer sur l'assemblage syntaxique des particules, collisions et de se reposer sur une vérification algorithmique pour savoir si cette construction se transpose effectivement au niveau des colorations. Ceci fait des schémas de ligature un outils pour construire et manipuler des colorations complexes.

Pour faciliter les constructions à l'aide de particules et collisions, il est possible d'introduire une méthode permettant, à partir d'un ensemble initial de particules et d'en construire de nouvelles et ainsi d'enrichir l'ensemble des éléments à disposition pour faire des schémas de ligature. Il est également

possible de dépasser la restriction sur la finitude des schémas de ligature en introduisant des techniques permettant non plus de déterminer si les colorations associées sont valides mais néanmoins de garantir l'existence de telles colorations. En combinant ces méthodes, il est alors possible de construire des comportements complexes d'automates cellulaires en se concentrant sur l'aspect symbolique de haut niveau et sans perdre de temps à analyser des contraintes locales. De ce fait, il est possible de voir les particules et collisions comme un outil pour comprendre et manipuler le calcul dans les automates cellulaires. Cette intuition est renforcée par les liens existant entre les systèmes de particules et collisions et la notion de simulation entre automate cellulaires.

En résumé, ce mémoire présente un ensemble cohérent de définitions et méthodes permettant de manipuler les particules et collisions pour les constructions sur automates cellulaires. Nous présentons ici quelques discussions sur des choix arbitraires (ou pas) effectués au cours de ce mémoire et sur les différentes pistes possibles pour étendre ce travail.

Automates de cartes

La validité des définitions de fonds, particules et collisions sont appuyées par leur relation avec les $\mathbb{Z}^2$ - k -peintures. Cependant, dans ce résultat, il est possible de se poser la question du choix de cette notion de peinture. Même si cette notion est intuitive et étend bien les notions équivalents connues en dimension 1, il existe d'autres extensions possibles. Sur ce point, il est possible de modifier par exemple la définition d'automate de carte et, en particulier, la condition de cohérence de l'automate. Dans ce mémoire, nous avons fait le choix d'exiger que l'état interne de l'automate soit cohérent sur tous les chemins partant de l'origine. Au vu de l'utilisation faite de ces automates, il aurait aussi été possible de demander la condition, à priori moins restrictive, que la coloration associée soit bien définie — cette définition permet à l'état interne de ne pas être unique du moment que toutes les projections par la fonction de coloriage mènent à la même couleur. De la même façon, il est possible de souhaiter se débarrasser de la référence à l'origine. Pour cela, il suffit d'étendre la contrainte de compatibilité de l'état interne à partir de n'importe quel état et suivant n'importe quel chemin. Une telle définition est plus restrictive que celle utilisée dans ce mémoire. À partir de ce moment, se pose la question de savoir si les peintures définies dépendent ou non de la contrainte de compatibilité choisie. Il nous semble que, au moins pour le cas sans et à un compteur, ces différentes définitions amènent à la même notion de peinture. Ce résultat reste néanmoins une conjecture à prouver et apportera un appui supplémentaire à l'idée que les fonds, particules et collisions sont vraiment des éléments essentiels dans les automates cellulaires. Et même dans le cas où cette conjecture ne serait pas vraie, il serait très intéressant de voir à quoi correspondent, au niveau des automates cellulaires,

les nouvelles colorations simples obtenues.

Extension des concepts

Dans tout ce mémoire, le choix a été fait de se restreindre à des automates cellulaires de dimension 1 (*i.e.*, agissant sur une ligne de cellule). Un axe d'extension usuel de résultat sur ces automates est de regarder s'il est possible de généraliser les concepts pour des automates travaillant en dimension supérieure (sur le plan, dans l'espace, ...). Du point de vue des colorations, cela revient à considérer des colorations en dimensions 3 ou plus. Un premier problème pour effectuer un travail similaire dans ce cadre est que les notions de particules et collisions sont moins évidentes à formaliser. En particulier, si on se concentre sur le cas de la dimension 3, on peut avoir 4 objets : l'espace, le plan, la droite et le point. S'il ne fait aucun doute que l'espace correspond au fond, les autres cas ne sont pas aussi évidents. Une approche pourrait être d'assimiler les particules aux droites et les collisions aux points. Cette approche permet alors d'obtenir des résultats similaires à ceux obtenus dans cette thèse mais font complètement l'impasse sur l'existence des plans. De plus, dans un tel cas, on remarque facilement que les particules ne forment plus des frontières et que donc il ne peut y avoir qu'un unique fond dans la construction. De la même façon, il est possible de construire d'autres extensions de particules et collisions en dimensions supérieures mais à chaque fois, le résultat obtenu semble être très restrictif par rapport à l'expressivité des automates cellulaires. Néanmoins, une extension de ces notions en utilisant les peintures est peut-être la piste la plus prometteuse. À première vue, cette piste risque également d'aboutir à une impasse mais pourrait permettre de mieux comprendre les limitations de ces objets en dimension supérieure.

Sans aller aussi loin qu'augmenter la dimension, il reste des éléments présents et reconnus dans les comportements d'automates cellulaires que notre système peine à modéliser et utiliser. En particulier, on peut citer le cas des guns. De ce point de vue, il semble intéressant d'étudier des pistes pour incorporer ces éléments en tant qu'objets de base et faciliter leur manipulation. Un autre moyen pour « traiter » ces objets est de s'intéresser à des classes particulières d'automates dans lesquelles ces objets ne sont pas présents. Dans de tels cas, on espère que le formalisme de particules et collisions peut réussir à caractériser le plus possible le comportement de l'automate ou, au moins, prendre un relief particulier. En particulier, le cas des automates réversibles semblent être une perspective très importante : ces automates présentent en effet des propriétés de régularité — comme par exemple le fait que l'ensemble limite contient toutes les configurations et que toute configuration périodique en espace appartient à un cycle — qui ont des répercussions sur les fonds, particules et collisions. Une étude plus poussée des propriétés des particules et collisions dans ces systèmes semble permettre d'obtenir des résultats intéressants d'autant plus que les automates cellulaires réversibles modélisent

une certaine réalité physique. Ce raisonnement est également valable pour la classe des automates cellulaires conservateurs qui bien qu'orthogonale à la classe précédente semble présenter également des propriétés syntaxiques fortes au niveau des particules et collisions.

Automatisation

Même si ce mémoire présente un ensemble d'outils pour manipuler des particules et collisions, il n'existe pas encore de programme mettant en pratique ces résultats et permettant d'automatiser la manipulation de ces objets. La réalisation d'un tel programme est évidemment une perspective à court terme très importante de cette thèse. Pour permettre une bonne manipulation des particules et collisions, ce programme a besoin de plusieurs fonctionnalités. Une première fonctionnalité importante est la capacité d'extraire, de manière automatique ou semi-automatique, des particules et collisions. En effet, les particules et collisions pouvant être de grande taille, il est impossible de mener une étude d'un large panel de celles-ci sans l'aide d'un outils d'extraction. Le fait que ces objets soient récursivement énumérables est évidemment un atout important pour l'existence d'une telle méthode. Une fois ces objets extraits, on souhaite bien sûr les manipuler de façon abstraite pour construire des schémas de ligature et que le programme soit capable de résoudre directement les schémas de ligature finis à l'aide de la réduction présentée dans le chapitre 2 et d'un algorithme pour résoudre les formules de Presburger.

Une fois un tel programme mis en place, on peut espérer que la manipulation de ces particules et collisions par l'intermédiaire des schémas de ligature permettra de mieux les comprendre et, de ce fait, de développer des méthodes plus générales ou d'obtenir des résultats de caractérisations pour le cas des schémas de ligature infinis. Bien évidemment, de telles méthodes ont complètement leur place comme fonctionnalités du programme. À terme, l'espoir est qu'un tel programme nous permette de manipuler les particules et collisions de façon simple et intuitive.

Auto-organisation

Pour conclure ce mémoire, il est important d'évoquer une question sur les particules et collisions qui se trouve être à l'origine de notre étude : peut-on expliquer l'apparition des particules et collisions dans les orbites des automates cellulaires ? Ce mémoire n'aborde jamais cette question de manière frontale mais vise à améliorer la compréhension de ces objets au travers de définitions formelle et d'outils pour les manipuler. À l'aide des définitions introduites dans ce mémoire, il semble désormais possible d'énoncer (et même de prouver) des résultats formels sur l'apparition de ces objets. Un exemple de tel résultat serait de prouver que pour certains automates cellulaires par-

ticuliers, avec probabilité 1, l'orbite d'une configuration aléatoire correspond, au bout d'un temps fini, à un schéma de ligature. Ce type de résultat est très intéressant car il permet de faire un lien entre l'observation et l'aspect théorique de l'étude des particules et collisions.

En ce qui concerne la compréhension des particules et collisions en elles-mêmes, ce mémoire apporte des pistes de réponse et montre surtout qu'il est possible d'obtenir des résultats liant intuition et aspects formels. Bien sûr, il reste encore un très long chemin à parcourir avant de comprendre ces notions qui sont très fortement liées à la notion de calcul parallèle.

Bibliographie

Références personnelles

- [1] Pierre GUILLON et Gaétan RICHARD. *Nilpotency and limit sets of cellular automata*. Dans *MFCS* (édité par Edward OCHMAŃSKI et Jerzy TYSZKIEWICZ), v. 5162 de *Lecture Notes in Computer Science*, p. 375 – 386. Springer, **2008**.
- [2] Nicolas OLLINGER et Gaétan RICHARD. *Collisions and their catenations : Ultimately periodic tilings of the plane*. Dans *Fifth Ifip International Conference On Theoretical Computer Science – Tcs 2008*, v. 273, p. 229 – 240. Springer Boston, **2008**.
- [3] Nicolas OLLINGER et Gaétan RICHARD. *A particular cellular automaton*. Dans *The Complexity of Simple Programs 2008*. **2008**. À paraître.
- [4] Nicolas OLLINGER et Gaétan RICHARD. *Automata on the plane vs particles and collisions*, **2009**. Accepté avec révisions à TCS.
- [5] Gaétan RICHARD. *Rule 110 : universality and catenations*. Dans *Proceedings of the First Symposium on Cellular Automata "Journées Automates Cellulaires"* (édité par Bruno DURAND), p. 141 – 160. **2008**.

Références

- [6] Serafino AMOROSO et Yale N. PATT. *Decision procedures for surjectivity and injectivity of parallel maps for tessellation structures*. *Journal of Computer and System Sciences*, v. 6(5) p. 448 – 464, **1972**.
- [7] François BLANCHARD, Petr KÚRKA et Alejandro MAASS. *Topological and measure-theoretic properties of one-dimensional cellular automata*. *Physica D : Nonlinear Phenomena*, v. 103(1 - 4) p. 86–89, **1997**.
- [8] Julius R. BÜCHI. *On a decision method in restricted second order arithmetic*. Dans *Proceedings of the International Congress on Logic, Methodology, and Philosophy of Science*, p. 1 – 11. Stanford University Press, **1962**.
- [9] Arthur W. BURKS, éditeur. *Essays on Cellular Automata*. University of Illinois Press, Urbana, **1970**.

- [10] Julien CERVELLE, Enrico FORMENTI et Pierre GUILLON. *Sofic trace subshift of a cellular automaton*. Dans *CiE* (édité par S. Barry COOPER, Benedikt LÖWE et Andrea SORBI), v. 4497 de *Lecture Notes in Computer Science*, p. 152 – 161. Springer, **2007**.
- [11] Noam CHOMSKY. *Three models for the description of language*. IRE Transactions on Information Theory, v. 2(3) p. 113 – 124, **1956**.
- [12] Alonzo CHURCH. *An unsolvable problem of elementary number theory*. American Journal of Mathematics, v. 2(58) p. 345 – 363, **1936**.
- [13] John COCKE et Marvin MINSKY. *Universality of tag systems with $P = 2$* . Journal of the ACM, v. 11(1) p. 15 – 20, **1964**.
- [14] Matthew COOK. *Universality in elementary cellular automata*. Complex Systems, v. 15(1) p. 1 – 40, **2004**.
- [15] Karel ČULÍK II, Jan K. PACHL et Sheng YU. *On the limit sets of cellular automata*. SIAM Journal on Computing, v. 18(4) p. 831 – 842, **1989**.
- [16] Karel ČULÍK II et Sheng YU. *Undecidability of CA classification schemes*. Complex systems, v. 2(2) p. 177 – 190, **1988**.
- [17] Jean-Christophe DUBACQ, Bruno DURAND et Enrico FORMENTI. *Kolmogorov complexity and cellular automata classification*. Theoretical Computer Science, v. 259(1-2) p. 271 – 285, **2001**.
- [18] Bruno DURAND et Zsuzsanna RÓKA. *The game of life : universality revisited*. Dans *Cellular Automata : a parallel model* (édité par Marianne DELORME et Jacques MAZOYER), p. 51 – 76. Kluwer, **1999**.
- [19] Nazim FATÈS. *Robustesse de la dynamique des systèmes discrets : le cas de l'asynchronisme dans les automates cellulaires*. Thèse de doctorat, Laboratoire de l'Informatique de Parallélisme, **2004**.
- [20] Michael J. FISCHER et Michael O. RABIN. *Super-exponential complexity of Presburger arithmetic*. Rapport technique, Massachusetts Institute of Technology, **1974**.
- [21] Patrick C. FISCHER. *Generation of primes by a one-dimensional real-time iterative array*. Journal of the ACM, v. 12(3) p. 388 – 394, **1965**.
- [22] Niloy GANGULY, Biplab K. SIKDAR, Andreas DEUTSCH, Geoffrey CANRIGHT et P. Pal CHAUDHURI. *A survey on cellular automata*. Rapport technique, Centre for High Performance Computing, Dresden University of Technology, **2003**.
- [23] Martin GARDNER. *Mathematical games : The fantastic combinations of John Conway's new solitaire game life*. Scientific American, v. 223 p. 120 – 123, **1970**.
- [24] Dora GIAMMARRESI et Antonio RESTIVO. *Two-dimensional languages*. Dans *Handbook of Formal Languages* (édité par Arto SALOMAA et Grzegorz ROZENBERG), v. 3, p. 215 – 267. Springer-Verlag, **1997**.

-
- [25] Seymour GINSBURG et Edwin H. SPANIER. *Semigroups, Presburger formulas, and languages*. Pacific Journal of Mathematics, v. 16 p. 285 – 296, **1966**.
 - [26] Gustav A. HEDLUND. *Endomorphisms and automorphisms of the shift dynamical system*. Mathematical Systems Theory, v. 3 p. 320 – 375, **1969**.
 - [27] Wim HORDIJK, Cosma R. SHALIZI et James P. CRUTCHFIELD. *Upper bound on the products of particle interactions in cellular automata*. Physica D : Nonlinear Phenomena, v. 154(3-4) p. 240 – 258, **2001**.
 - [28] Oscar H. IBARRA et Tao JIANG. *Relating the power of cellular arrays to their closure properties*. Theor. Comput. Sci., v. 57 p. 225 – 238, **1988**.
 - [29] Jarkko KARI. *Decision Problems Concerning Cellular Automata*. Thèse de doctorat, University of Turku, **1990**.
 - [30] Jarkko KARI. *The nilpotency problem of one-dimensional cellular automata*. SIAM Journal on Computing, v. 21(3) p. 571 – 586, **1992**.
 - [31] Jarkko KARI. *Theory of cellular automata : a survey*. Theoretical Computer Science, v. 334(1-3) p. 3 – 33, **2005**.
 - [32] Jarkko KARI et Nicolas OLLINGER. *Periodicity and immortality in reversible computing*. Dans *MFCS* (édité par Edward OCHMAŃSKI et Jerzy TYSZKIEWICZ), v. 5162 de *Lecture Notes in Computer Science*, p. 419 – 430. Springer, **2008**.
 - [33] Stephen C. KLEENE. *Representation of events in nerve nets and finite automata*. Dans *Automata Studies* (édité par Claude SHANNON et John MCCARTHY). Princeton University Press, **1956**.
 - [34] T. Yung KONG et Azriel ROSENFELD. *Digital topology : introduction and survey*. Computer Vision, Graphics, and Image Processing, v. 48(3) p. 357 – 393, **1989**.
 - [35] Ivan KOREC. *Real-time generation of primes by a one-dimensional cellular automaton with 9-states*. Dans *MCU (2)* (édité par Maurice MARGENSTERN), p. 101–116. IUT Metz, **1998**.
 - [36] Petr KŮRKA. *Topological dynamics of cellular automata*. Dans *Codes, Systems, and Graphical Models* (édité par Brian MARCUS et Joachim ROSENTHAL), v. 123 de *The IMA Volumes in Mathematics and its Applications*, p. 447 – 386. Springer-Verlag, **2001**.
 - [37] Bruno MARTIN II. *Automates cellulaires, information et chaos*. Thèse de doctorat, École Normale Supérieure de Lyon, **2001**.
 - [38] Jacques MAZOYER et Véronique TERRIER. *Signals in one-dimensional cellular automata*. Theoretical Computer Science, v. 217(1) p. 53 – 80, **1999**.
 - [39] Marvin MINSKY. *Computation : Finite and Infinite Machines*. Prentice Hall, Englewoods Cliffs, **1967**.

- [40] Edward F. MOORE. *Machine models of self-reproduction*. Dans *Proceedings of Symposia in Applied Mathematics*, v. 14, p. 17 – 33, **1962**.
- [41] Kenichi MORITA. *Universality of a reversible two-counter machine*. Theoretical Computer Science, v. 168(2) p. 303 – 320, **1996**.
- [42] John MYHILL. *The converse of Moore's garden-of-eden theorem*. Dans *Proceedings of the American Mathematical Society*, v. 14, p. 658 – 686, **1963**.
- [43] Nicolas OLLINGER. *Automates cellulaires : structures*. Thèse de doctorat, École Normale Supérieure de Lyon, **2002**.
- [44] Emil POST. *Formal reductions of the general combinatorial decision problem*. American Journal of Mathematics, v. 65(2) p. 197 – 215, **1943**.
- [45] Ivan RAPAPORT. *Inducing an order on CA by a grouping operation*. Thèse de doctorat, École Normale Supérieure de Lyon, **1998**.
- [46] Alvy R. SMITH III. *Cellular automata complexity trade-offs*. Information and Control, v. 18(5) p. 466 – 482, **1971**.
- [47] Véronique TERRIER. *Décidabilité en arithmétiques faibles. Temps réel sur automates cellulaires*. Thèse de doctorat, Université Claude Bernard Lyon 1, **1991**.
- [48] Guillaume THEYSSIER. *Automates cellulaires : un modèle de complexités*. Thèse de doctorat, École Normale Supérieure de Lyon, **2005**.
- [49] Alan M. TURING. *On computable numbers, with an application to the Entscheidungsproblem*. Proceedings of the London Mathematical Society, v. 2(42) p. 230 – 265, **1936**.
- [50] John VON NEUMANN. *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign, IL, USA, **1966**.
- [51] John WATROUS. *On one-dimensional quantum cellular automata*. Dans *36th Annual Symposium on Foundations of Computer Science*, p. 528 – 537. Society Press, **1995**.
- [52] Stephen WOLFRAM. *Universality and complexity in cellular automata*. Physica D : Nonlinear Phenomena, v. 10 p. 1 – 35, **1984**.
- [53] Stephen WOLFRAM. *A new kind of science*. Wolfram Media Inc., Champaign, Illinois, United States, **2002**.
- [54] Konrad ZUSE. *Rechnender Raum*. Elektronische Datenverarbeitung, v. 8 p. 336 – 344, **1967**.

Index

- arête
 - intérieure, 45
- automate cellulaire, 6
 - 110, 6, 78
 - élémentaire, 6
 - injectif, 11, 13
 - nilpotent, 13, 14
 - surjectif, 11, 13
- automate de carte à compteurs, 24
 - périodique, 24
- automate de pondération, 65
- bord, 34
- caractérisation normale, 18
- cartographie, 24
- classification, 9
- collision, 18, 28, 29, 39, 40, 42, 60
 - patron, 39, 41
 - perturbation, 40
 - propre, 18
 - translaté, 40
- coloration, 7
 - bipériodique, 8
 - engendrée, 24
 - périodique, 8
- coloriage, 24
 - valide, 35
- composante indépendante, 24
- cône, 18
- configuration, 6
 - finie, 7
 - périodique, 6, 26
 - ultimement périodique, 6, 26
- diagramme espace-temps, 8
- encodage, 75
- ensemble de validité, 48
- ensemble limite, 8
- état, 6
 - quiescent, 7
- face, 46
 - côté, 46
 - intérieure, 46
 - orientation, 69
 - orientation élémentaire, 68
- fond, 16, 27, 36, 42
 - patron, 36, 41
 - translaté, 37
- forme, 34
- frontière, 34
- groupé, 75
- gun, 20, 59
- jeu de faces, 63
 - déterministe, 65
 - finiment pondéré, 64
 - ordonné, 70
- ligature, 44
- machine de Turing, 74
- mot bloquant, 14
- motif, 34
 - translaté, 34
- orbite, 6
- particule, 17, 27, 37, 38, 42, 59
 - fantôme, 19
 - orientation, 41

- patron, 37, 41
- perturbation, 38
- propre, 17
- translaté, 38
- patchwork, 35
- peinture, 24
 - $\mathbb{Z}$ -0, 26
 - $\mathbb{Z}$ -1, 26
 - $\mathbb{Z}$ -2, 26
 - $\mathbb{Z}^2$ -0, 27
 - $\mathbb{Z}^2$ -1, 27
 - $\mathbb{Z}^2$ -2, 28, 29
- points
 - discriminants, 24
 - indistinguishables, 24
- pondération, 46
- schéma de ligature, 44, 45, 48
 - bien ordonné, 70
 - croissant, 61
 - pondéré, 48
 - valide, 48
- schéma de signaux, 59
- shift, 8
- signal, 58, 59
- signaux, 11, 20
- simulation
 - intrinsèque, 76
 - Turing, 75
- sommet
 - intérieur, 45
- sous-automate, 75
- support, 34
- système dynamique discret, 8
- système PaCo, 40, 42, 62
- temps réel, 11
- trace, 13
 - nilpotente, 14
- transition
 - fonction globale, 6, 9
 - règle locale, 6
- union disjointe, 34
- universalité
 - intrinsèque, 76, 89
 - Turing, 75, 78
- vecteur, 8
 - colinéaires, 8

Résumé

Cette thèse a pour objet l'étude des systèmes de particules et collisions dans les automates cellulaires. En se basant sur des observations expérimentales, nous proposons des définitions formelles de ces objets et montrons qu'ils peuvent être mis en relation avec des coloriage réguliers du plan. À l'aide d'une représentation sous forme syntaxique de ces objets, nous introduisons une opération syntaxique d'assemblage : les schémas de ligature. Cette opération peut être interprétée en termes de coloriage et correspond à une opération intuitive utilisée dans l'étude algorithmique des automates cellulaires. Nous prouvons que, dans le cas d'assemblages finis, le lien entre l'opération syntaxique et l'interprétation peut être complètement caractérisé de façon algorithmique. Nous explorons ensuite des pistes d'extension de ces systèmes facilitant l'encodage et permettant de dépasser le cas fini. Enfin, nous étudions les applications de tels systèmes en lien avec l'universalité dans les automates cellulaires. En particulier, nous donnons une nouvelle preuve de l'universalité de l'automate cellulaire 110 et présentons la construction d'un automate cellulaire intrinsèquement universel de rayon 1 et à 4 états.

Mots clefs : Automates cellulaires, particules et collisions, pavages, algorithmique, universalités.

Abstract

The main goal of this thesis is to study systems of particles and collisions in cellular automata. Starting from experimental observations, we give formal definitions of these objects and show how they relate to regular colorings of the plane. Using a symbolic representation of those objects, we introduce a syntactical operation on them: catenation schemes. This operation is linked to an informal operation usually used in algorithmics on cellular automata through a coloring interpretation. We show that, in the case of finite catenation schemes, this link can be completely characterised in an algorithmic way. Then we explore possibilities of extensions of this result to ease encoding or overcome the finite limitation. At last, we study applications of these systems to study universality in cellular automata. In particular, we give a new proof of universality of rule 110 and give the construction of an intrinsically universal cellular automaton with radius 1 and only 4 states.

Keywords: Cellular automata, particles and collisions, tilings, algorithmic, universalities.