

UNIVERSITÉ DE LIÈGE



DÉPARTEMENT DE PHYSIQUE

Complexité Algorithmique des Automates Cellulaires

Capet Arthur



Mémoire présenté en vue de l'obtention du grade de licencié en Sciences Physiques
Année académique 2005-2006

Je tiens à remercier tous les enseignants qui, au delà de leur devoir professionnel, ont tenu à me transmettre une passion. Parmi eux je voudrais remercier tout particulièrement André Hautot qui, depuis notre première candidature, s'est efforcé, par le biais de séminaires improvisés, de questions troublantes et de bonnes blagues, d'agrandir le champ de notre questionnement scientifique.

Je remercie aussi mes compagnons de discussions, Marie, Max, Antoine, Aurore, P-X, Christophe, pour ne citer que les plus proches, pour toutes les heures d'envolées lyriques passées ensemble qui ont fait de ces années d'études non pas un tâche mais une quête.

Enfin, je tiens à remercier ma famille et mes parents, à qui je dois tout.

Table des matières

Introduction	3
I Concepts essentiels de la théorie de l'information	4
1 Contexte	5
2 Rappel : l'entropie en physique classique	7
2.1 L'entropie en thermodynamique	7
2.2 L'entropie en physique statistique	8
3 Théorie de l'information	10
3.1 L'entropie en théorie de l'information	10
3.1.1 Entropie de Shannon	10
3.1.2 Information manquante	16
3.1.3 Entropie algorithmique de Kolmogorov	17
3.2 Mesure déterministe de la complexité.	19
3.2.1 La Complexité de production de Lempel & Ziv	19
3.2.2 La T-complexité de Titchener	20
3.3 Lien entre la complexité de Titchener et l'entropie de Shannon : la T-entropie . . .	21
3.3.1 Dépendance de la T-entropie en N	22
3.3.2 Mesures sur des suites aléatoires	23
3.3.3 De la largeur de la distribution	24
3.3.4 Le cas des suites biaisées	25
3.3.5 Discussion	26
II Les automates cellulaires	28
4 Familiarisation aux Automates Cellulaires	30
4.1 Définition et notations	30
4.2 Classification	32
4.2.1 Les quatre classes de Wolfram	32
4.2.2 Tentatives de classification effective	37

5	Investigations personnelles	42
5.1	Dépendances aux conditions initiales	42
5.1.1	Exposants de Liapounov	42
5.1.2	Interprétation des exposants de Lyapounov dans le cadre de la théorie de l'information	43
5.1.3	Mesure Λ de la dépendance aux C.I. des A.C.	44
5.1.4	Résultats et commentaires	44
5.2	Evolution de la T-Entropie	49
5.2.1	Observations	49
5.2.2	Conclusions et comparaisons avec des études antérieures	51
5.3	Explorations dans l'espace des phases	53
5.4	Règles inversibles	58
6	Discussion sur la notion de complexité	65
	Conclusion	66
	Appendices	68
A	Démonstration de l'expression de l'entropie de Shannon	69
B	Illustration des règles utilisées	71
C	Analyse DFA	74
C.1	Introduction	74
C.2	Méthode de Hurst	74
C.2.1	Méthode	74
C.2.2	Résultats obtenus et commentaires	75
C.3	Méthode DFA	76
C.3.1	Résultats obtenus et commentaires	77
C.4	Conclusion	77
	Bibliographie	82

Introduction

Ce mémoire consiste en une discussion sur les automates cellulaires et sur l'apport de ces derniers en physique fondamentale. Nous situerons d'abord le contexte général en rappelant quelques résultats essentiels de la théorie de l'information. Le fil conducteur suivi lors de ces rappels théoriques aura pour issue l'établissement d'une mesure de l'information applicable aux suites finies. Une fois cette mesure établie, nous l'appliquerons à quelques cas solubles analytiquement afin d'en apprécier le comportement.

Dans la deuxième partie, après avoir présenté les automates cellulaires, nous nous efforcerons, par le biais de résultats trouvés dans la littérature et d'analyses personnelles, de mettre en valeur certaines propriétés intéressantes de ces derniers. Ces investigations s'inscriront dans le cadre général du problème de la classification des automates cellulaires et de l'établissement de moyens effectifs permettant de juger de manière automatique des propriétés d'un automate donné.

Cette partie s'achèvera sur une question ouverte concernant le cas particulier des automates inversibles. Ce cas théorique ouvre en effet une discussion sur l'évolution globale de l'entropie et fait apparaître une confrontation directe entre les deux approches de la théorie de l'information : l'entropie statistique et l'entropie algorithmique.

Première partie

**Concepts essentiels de la théorie
de l'information**

Chapitre 1

Contexte

Le premier mot qui se forme dans la tête d'un scientifique lorsqu'on lui parle d'organisation est le mot "Entropie". Ce mot est utilisé dans de nombreux domaines, et a été introduit indépendamment sous plusieurs définitions. Au départ beaucoup n'y ont vu qu'une source de confusion, mais avec le temps il s'est avéré des liens étroits entre ces diverses définitions au point qu'il est presque unanimement admis de nos jours que les entropies thermodynamiques, algorithmiques, et informationnelles ne sont que plusieurs expressions d'une seule et même grandeur. Lorsque Shannon introduit, en 1948, la notion du contenu informationnel d'une suite de chiffres, c'était dans un souci pratique de transmission de message. Cette étude est à situer dans le contexte des élans de l'ère de la communication et de l'engouement des mathématiciens de l'époque pour les notions de transmission de messages et de cryptographie. Il nomma déjà ce contenu informationnel *entropie*, mais il fallut attendre Kolmogorov dans les années 60' pour faire le lien suivant entre la théorie de l'information et la thermodynamique.

A une certaine précision donnée, tout système physique peut être décrit par une suite de chiffres ou de symboles.

On pourrait, par exemple, préciser une à une les positions et les vitesses de tous les atomes du système dans une suite univoquement décodable. Evidemment, dans beaucoup de cas - pensons notamment à celui d'un cristal - une telle description serait une perte de temps, car il existe des manières beaucoup plus courtes de décrire le système sans perdre d'information à son sujet. C'est ici que se situe la charnière : à système simple, description simple et à système complexe description complexe. Le contenu informationnel d'une suite décrivant un système doit donc être relatif à l'entropie de ce système. Et il est amusant de remarquer à quel point cette idée est proche de l'image classique qu'on se fait de l'entropie. "*C'est une mesure de l'ordre régnant dans le système.*", entend-on souvent. Or qu'est ce que l'ordre d'un système, sinon une structure dans la description qu'on peut en faire, permettant de raccourcir celle-ci ?

Plusieurs auteurs s'appliquèrent, dans les années qui suivirent, à mettre en évidence les concordances entre ce nouveau point de vue qu'est la théorie de l'information et les théories préexistantes. Citons entre autres Solomonoff, Chaitin, Sinai, Feynman, Bennet et Jaynes.

Dans cette première partie nous commencerons par définir quelques notions essentielles de la théorie de l'information pour aboutir à une mesure du contenu informationnel applicable sur une

suite finie. Mais commençons tout d'abord par rappeler à quoi correspond ce mot, *entropie*, et comment il a été défini avant l'arrivée de la théorie de l'information.

Chapitre 2

Rappel : l'entropie en physique classique

Le premier principe stipule que toute transformation observable dans la nature doit s'effectuer en conservant l'énergie totale. Le deuxième principe, qui traite de l'entropie, appose une deuxième contrainte en imposant à ces transformations une *direction d'évolution*.

La plupart des équations de la physique classique sont déterministes et réversibles, c'est à dire qu'elle sont symétriques pour la transformation $t \mapsto -t$. Or l'expérience du quotidien nous enseigne que l'énorme majorité des phénomènes macroscopiques sont complètement irréversibles. Par exemple, on imaginerait mal la chaleur d'une pièce converger spontanément vers un coin de celle-ci en laissant le reste à moindre température. La diffusion dans toute la pièce d'une quantité de chaleur initialement localisée dans un coin, irait, par contre, plus dans le sens de notre intuition, de sorte qu'étant donné l'observation d'une répartition uniforme de la température, il n'est pas possible de savoir quelle était sa répartition initiale. Le concept d'entropie permet de rendre compte de cette irréversibilité.

2.1 L'entropie en thermodynamique

L'entropie fut d'abord introduite dans le cadre de la thermodynamique, et peut y être formulée comme suit : Soit un système physique passant d'un état d'équilibre A à un autre B, au contact d'une source de température T_{source} , avec laquelle une quantité de chaleur Q est échangée au cours de la transformation. L'entropie, définie à une constante additive près, est telle que sa variation lors de cette transformation respecte la relation :

$$\Delta S_{AB} \equiv S(B) - S(A) \geq \int_A^B \frac{\delta Q}{T_{source}}, \quad (2.1)$$

soit la quantité totale de chaleur échangée au cours de la transformation divisée par la température du thermostat. L'unité utilisée est le Joule/Kelvin. L'égalité ci-dessus a lieu dans le cas d'une transformation réversible. Pour les cas réels (irréversibles), l'équation 2.1 donne lieu à une inégalité qui fixe les limites des transformations que l'on peut espérer observer. En particulier, pour un système isolé on a $\delta Q = 0$ et donc $S(B) > S(A)$, l'entropie ne peut qu'augmenter. C'est ce *principe*, com-

munément admis¹, qui impose cette "direction" aux transformations observables : "Un système isolé évolue presque toujours en augmentant son entropie". L'énoncé exact, qui résiste aux problèmes posés occasionnellement par les fluctuations et permet d'éviter les contradictions, est celui de Kelvin, mais il illustre moins bien les conséquences que nous voulions mettre ici en valeur.

En particulier, dans le cas de N molécules d'un gaz parfait, classique et monoatomique contenues dans un volume V à une température T , on peut écrire

$$S(N, U, V) = N k \ln \left[\gamma_1 \left(\frac{U}{N} \right)^{3/2} \left(\frac{V}{N} \right) \right] \quad (2.2)$$

où $k = 1.38 \cdot 10^{-23} J/K$ est la constante de Boltzmann et γ_1 une constante appropriée.

2.2 L'entropie en physique statistique

Les méthodes de la physique statistique donnent une nouvelle interprétation de cette fameuse entropie. Boltzmann lie l'entropie thermodynamique d'un système au nombre de configurations microscopiques possibles correspondant à l'état macroscopique observé.

Soit un système isolé de N particules en mouvement. Ce système est défini par la donnée des $6N$ coordonnées généralisées, par exemple les positions q_i et impulsions p_i des N molécules dans un repère cartésien. Une configuration microscopique précise du système correspond à un point de l'espace des phases à $6N$ dimensions. Mais il est clair que plusieurs points de cet espace de phase donnent les mêmes caractéristiques macroscopiques, telle l'énergie interne U ou toute autre variable d'état. On peut lier le macroscopique au microscopique en écrivant l'hamiltonien du système

$$H = \sum_{i=1}^{3N} \frac{p_i^2}{2m} + V(\{q_i\}) \quad (2.3)$$

qui est la somme des énergies du système. Pour une énergie interne U fixée l'équation

$$H(q_1, \dots, q_{3N}, p_1, \dots, p_{3N}) = U \quad (2.4)$$

définit une hypersurface dans l'espace des phases. Si cette énergie est connue avec une certaine imprécision ΔU , la relation $U - \Delta U/2 \leq H \leq U + \Delta U/2$ définit alors une région $\Gamma(N, U, V)$ de l'espace de phase, accessible au système. Dès lors que l'on accepte un volume élémentaire fini, imposé par les imprécisions inhérentes à chaque coordonnée q_i , p_i ² on peut se poser la question d'une partition de l'espace de phase en un nombre fini Ω de configurations possibles du système.

$$\Omega(N, U, V) = \int_{\Gamma(N, U, V)} \frac{dq_1 \dots dq_{3N} dp_1 \dots dp_{3N}}{h^{3N} N!} \quad (2.5)$$

Où le terme $N!$ est introduit pour tenir compte de l'indiscernabilité des molécules.

Le principe de Boltzmann s'écrit alors

$$S(N, U, V) = k \ln \Omega(N, U, V), \quad (2.6)$$

¹Bien qu'il ne s'agisse pas là d'un énoncé rigoureux.

²Rappelons que le principe d'incertitude d'Heisenberg fixe une limite inférieure naturelle à ce volume élémentaire en stipulant que $\prod_{i=1}^{3N} \Delta p_i \Delta q_i \geq \frac{h}{4\pi}$

et l'entropie S correspond au logarithme naturel du nombre de configurations microscopiques compatibles avec les propriétés macroscopiques observées.

Prenons l'exemple d'un gaz dans une boîte : S'il est connu que le gaz est réparti de façon homogène dans toute la boîte (concentration, pression et température homogènes dans toute la boîte), il existe de très nombreuses façons d'arranger les molécules du gaz dans la boîte afin d'obtenir cette homogénéité à grande échelle. Si on impose maintenant que toutes les molécules soient confinées dans un coin de la boîte, ces molécules ayant une certaine occupation spatiale, il est évident qu'il existe moins de configurations microscopiques correspondantes. Le principe de Boltzman stipule donc que le système tendra à évoluer vers la première situation. Il est assez intuitif que le système évolue naturellement vers l'état macroscopique correspondant au plus grand nombre de situations microscopiques. En fait, ces nombres étant gigantesques, c'est ce qu'on observerait même si les états microscopiques étaient tirés au hasard.

On peut chercher une correspondance entre ces deux formes de l'entropie[1].

Posons $V/N = (\Delta q)^3$. Ceci représente le volume moyen disponible par molécule et Δq peut être interprété comme la précision sur les coordonnées spatiales. Pour l'énergie moyenne par molécule U/N , on a,

$$U/N = \frac{1}{2m}(\langle p_x^2 \rangle + \langle p_y^2 \rangle + \langle p_z^2 \rangle) = \frac{3}{2m} \langle p_x^2 \rangle \equiv \frac{3}{2m}(\Delta p)^2 \quad (2.7)$$

où Δp , racine carrée des fluctuations quadratiques d'une composante de l'impulsion donne l'ordre de variation de celles-ci autour de la valeur moyenne $\langle p_{x,y,z} \rangle = 0$. On a alors, en réécrivant 2.2 et en choisissant γ_2 et γ de manière appropriée,

$$S(N, U, V) = N k \ln[\gamma_2(\Delta x)^3(\Delta p)^3] = k \ln \left[\frac{(\Delta x)(\Delta p)}{\gamma} \right]^{3N} \quad (2.8)$$

qui est un cas particulier de 2.6 avec 2.5.

Chapitre 3

Théorie de l'information

3.1 L'entropie en théorie de l'information

My greatest concern was what to call it. I thought of calling it 'information', but the word was overly used, so I decided to call it 'uncertainty'. When I discussed it with John von Neumann, he had a better idea. Von Neumann told me, 'You should call it entropy, for two reasons. In the first place your uncertainty function has been used in statistical mechanics under that name, so it already has a name. In the second place, and more important, nobody knows what entropy really is, so in a debate you will always have the advantage.

C.E. Shannon [32]

En 1948, Claude Elwood Shannon (1916-2001) étudiait la transmission optimale de messages. En cherchant à formaliser les modes de communication, il fut amené à introduire la notion de contenu informationnel, censée représenter la quantité d'information réellement contenue dans un message. Sous le coup d'un trait d'humour dû à John Von Neumann, il décida de nommer *entropie d'une source* son aptitude à générer de l'information. Ce n'est que plus tard que Kolmogorov vit l'interprétation physique que l'on pouvait en faire et le parallélisme puissant qui reliait cette quantité d'information à l'entropie physique des systèmes dynamiques.

3.1.1 Entropie de Shannon

Un message émis par une source est une suite de symboles pouvant prendre une valeur parmi M possibles ($\{a_1, \dots, a_M\}$ est l'alphabet de la source). Le message est habituellement transmis par un canal de transmission comportant un certain nombre m d'états physiques différents. Nous choisirons ce canal binaire, c'est à dire qu'il ne peut transmettre que des 0 et des 1.

La question que se posait Shannon est la suivante :

"Comment transmettre un message de la source (codage M -aire), en un message binaire, et ce, le plus économiquement possible ?"

C'est à dire en le moins de symboles possible (rappelons nous que Shannon parlait de bande passante sur une ligne télégraphique). Evidemment cela doit se faire en conservant une réversibilité du codage afin que le message originel puisse être rétabli.

Voyons comment formaliser la chose.

La source est caractérisée par une série de lois internes qui donnent (ou non) un sens aux messages émis. Ces lois lient la probabilité d'occurrence d'un symbole aux derniers symboles émis. Soit $p_i = p[a_i]$ les probabilités d'occurrence du symbole a_i , ou de l'événement i . Si ces probabilités sont indépendantes du passé de la source, la source est entièrement caractérisée par la donnée des p_i et décrit un processus stochastique, biaisé ou non selon que les différents événements sont équiprobables ou pas.

Considérons, pour se faire une idée, un exemple de deux sources différentes émettant toutes deux en binaire. On note la suite des symboles émis,

$s_1 = 0000000000000100000000000100001000000000000000000001000000000$
 $s_2 = 01100111110001001100101001000001101001110010011110101011110101101$

Il saute aux yeux que la suite s_1 sera plus facile à transmettre à un éventuel ami éloigné, en tout cas on devrait pouvoir le faire d'une façon plus courte qu'en énumérant tous les symboles un à un. Il suffirait, par exemple, de coder d'une manière univoque le nombre de fois qu'il faut recopier le symbole "0" entre deux symboles "1" consécutifs ou de citer les places où se trouvent les "1". Cette *facilité* est bien une conséquence des lois qui régissent le comportement de la source, définies, rappelons-nous, par les probabilités d'occurrence des symboles. Si l'on nous donnait le chiffre suivant de la suite s_1 , on sent aussi qu'il serait moins intéressant de recevoir un 0 que de recevoir un 1. Cette jonction entre la probabilité d'occurrence d'un événement et l'information qu'apporte cet événement est au centre de nos préoccupations.

Shannon[2] voulait une quantité qui puisse mesurer l'information apportée par l'occurrence d'un événement. Cette quantité H dépend des probabilités d'occurrence de l'événement, supposées connues. Pour ce faire, $H(\{p_i\})$ doit respecter les conditions suivantes :

1. H doit être continue sur les p_i .
2. Si les événements i sont équiprobables, soit $p_i = p_j \forall (i, j) \in \{1, \dots, M\} \times \{1, \dots, M\}$, H doit être une fonction monotone croissante en M . L'occurrence d'un événement parmi quatre possibles apporte effectivement plus d'information que celle d'un événement parmi deux possibles.
3. Si on sépare un choix en plusieurs sous-choix, le H initial devrait être la somme pondérée des H propres à chacun des "sous-choix". Ceci est illustré sur la figure 3.1. À gauche nous avons trois possibilités avec $p_1 = \frac{1}{2}$, $p_2 = \frac{1}{3}$ et $p_3 = \frac{1}{6}$; à droite nous avons d'abord le premier choix avec les probabilités $(\frac{1}{2}, \frac{1}{2})$; puis le second avec les probabilités $(\frac{2}{3}, \frac{1}{3})$. Les probabilités étant au final identiques pour les deux situations, l'entropie devrait être la même : On devrait avoir

$$H\left(\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\right) = H\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{1}{2}H\left(\frac{1}{3}, \frac{2}{3}\right) \quad (3.1)$$

où le coefficient $\frac{1}{2}$ est inséré puisque le second choix ne doit être fait qu'une fois sur deux.

Shannon démontre[2] (voir appendice 1 pour la démonstration) que la seule fonction qui satisfait ces trois conditions est de la forme suivante.

$$H = -K \sum_i p_i \ln p_i \text{ symbole/symbole, } i \in \{1 \dots M\} | p_i \neq 0 \quad (3.2)$$

où la somme ne court que sur les événements ayant lieu au moins une fois dans la suite, afin d'éviter le cas "0 ln 0". K est une constante positive, choisie égale à l'unité pour des raisons qui s'illustreront dans ce qui suit.

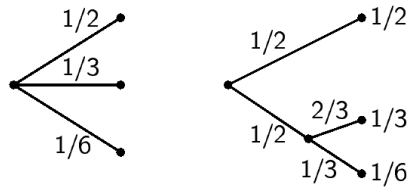


Fig. 3.1 – Illustration d’une manière équivalente d’interpréter un choix, en le séparant en sous-choix.

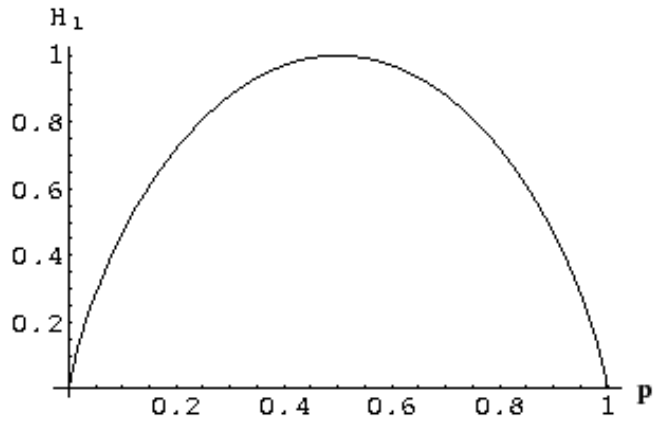


Fig. 3.2 – La fonction $H = -p \log_2 p + (1 - p) \log_2 (1 - p)$, soit l’entropie de Shannon pour une suite de bits, en fonction de la probabilité p d’avoir un 1.

H , l’entropie statistique ou entropie de Shannon, s’exprime en symbole/symboles soit en bit/symbole si le logarithme naturel dans la formule ci-dessus est remplacé par le logarithme en base deux. Elle représente l’incertitude moyenne sur l’occurrence d’un événement avant qu’il ne se produise, et en même temps l’information qu’il apporte quand il se produit.

Examinons le cas où la source peut générer deux symboles : le ”1” avec la probabilité p , et le ”0” avec la probabilité $1 - p$. Le graphique suivant (Fig. 3.2) donne la valeur de H en fonction de p .

$H(p)$ est positif partout sauf en 0 et en 1 où elle est nulle. En effet, l’information apportée par l’occurrence d’un 0 est nulle si elle vient d’une source qui ne produit que des 0. Le nombre de bits nécessaires pour coder un message issu d’une telle source est en effet quasi nul puisqu’il suffit d’écrire ”écrit 0 N fois”. On peut constater aussi que l’information par symbole est maximum lorsque les occurrences sont équiprobables, l’incertitude étant alors à son maximum.

Un biais dans les probabilités permet de comprimer la suite. En effet si les différents symboles n’apparaissent pas avec la même probabilité, on peut utiliser ce biais pour comprimer de manière effective¹ la suite. Dans le meilleur des cas la suite comportera exactement $|s_{opt}| = H_{Shannon}(s)|s|$ symboles, d’où l’unité en symbole/symboles et le choix de fixer la constante K à l’unité. Un exemple d’un tel algorithme de compression effective est l’algorithme de Huffman dont un exemple est

¹L’adjectif effectif sera souvent utilisé dans ce mémoire. Il signifie que le problème est soluble de manière générale par un algorithme prédéfini.

donné plus bas.

Jusqu'à présent nous n'avons considéré que des suites d'événements non-corrélés. Si l'occurrence d'un symbole dépend du passé de la source, on dit que la suite est corrélée et l'étude de son entropie doit tenir compte de ce fait.

Dans la suite $s = 01010101010101010101010101010101$, les "0" et les "1" sont équiprobables, ce qui selon notre définition signifie que l'entropie est maximum. Il semble pourtant aisé de prolonger la suite sans se tromper (et sans effectuer de calcul!!). Dans le cas des suites corrélées, il nous faut considérer non plus les probabilités d'occurrence de chaque symbole pris indépendamment, mais bien des paires, triplets, etc... On voit directement dans l'exemple ci-dessus que si l'on calcule H pour l'alphabet composé des quatre symboles {"00", "01", "11", "10"}, on aura un résultat nul (toutes les probabilités étant nulles sauf une, $p["01"]$, unitaire).

L'entropie statistique des suites corrélées se calcule en considérant les symboles par blocs de k caractères, de plus en plus longs et en traitant ces blocs comme des symboles indépendants d'un alphabet plus vaste qui en comporterait M^k :

$$H_k = -\frac{1}{k} \sum_{i=1}^{M^k} p_i \log_2 p_i \text{ bit/symbole} \quad (3.3)$$

Où p_i représente cette fois la probabilité d'occurrence de chacun des M^k blocs de k caractères. La théorie précédente est toujours d'application mais l'entropie statistique dépend de k. Notons que si l'alphabet est biaisé sans qu'il y ait corrélation entre symboles successifs, les probabilités d'occurrence des symboles de l'alphabet étendu M^k sont directement héritées des $p_i : p[a_i a_j] = p[a_i].p[a_j]$. On vérifie facilement que l'on a alors, $H_k = H$, pour tout k (c'est une conséquence directe de 3.1).

Une suite pour laquelle toutes les occurrences de singletons sont équiprobables, mais aussi celles des différentes paires, triplets, etc... est dite normale au sens de Borel. Ces suites possèdent une entropie maximum et sont statistiquement incompressibles. On attend d'une suite intrinsèquement aléatoire qu'elle soit normale au sens de Borel. La suite s_{opt} obtenue après la compression statistique optimale d'une suite biaisée est aussi normale puisqu'elle ne peut pas être compressée plus avant.

Pour le cas général d'une suite infiniment longue, on obtient la limite de compression effective, appelée limite étendue de Shannon :

$$H_\infty = -\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^{M^k} p_i \log_2 p_i \text{ bits/symboles} \quad (3.4)$$

qui représente donc le nombre minimal de bits d'encodage par symbole de la suite à compresser. Si la suite présente des corrélations de portée k, on aura $H_\infty = H_k \leq H_j$, pour tout $j \leq k$. L'adjonction de lois liant les symboles successifs entre eux restreint le choix inhérent à chaque symbole, et diminue ainsi l'entropie.

Exemple de compression effective : l'algorithme de Huffman.

L'algorithme de Huffman [8] constitue un exemple de compression effective d'une suite biaisée. Il consiste à établir un code préfixe qui exploite au mieux le biais entre les probabilités des différents

001, 100 → 00011, 010 → 00010, 001 → 00011, 000 → 00000} et la suite

$s_3 = 000110111000111010111101000010100101101010111000110010111101110101111111111111$
 $10111111110010100010110010011010010011001101101000011000000100100100100111011101100$
 $101111010011.$

Soit une compression de 0.725 bit/bit. En considérant des ordres de plus en plus grands on se rapprochera ainsi de la limite de compression correspondant à l'entropie de Shannon. Il est évident que l'algorithme est aussi plus performant pour des suites plus longues.

Applications.

Shannon lui même ne mesure pas l'ampleur de cette définition, amenée dans un but principalement pragmatique de transfert de message. Partant d'assumptions logiques représentant les comportements attendus pour une grandeur appelée "quantité d'information", il en dérive la forme mathématique compatible et prend pour garanties les correspondances avec la physique.

[...]The real justification of these definitions, however, will reside in their implications.

Shannon, 1948,[2]

Néanmoins il ne parait pas absurde de penser qu'il puisse exister une grandeur définie pour chaque système, au même titre que l'énergie, qui quantifie l'information contenue dans le système.

Il est important de remarquer que la définition de Shannon s'applique non pas à une suite particulière mais à une source capable d'émettre un ensemble de suites. En particulier, les p_i sont supposés connus, ou tout du moins identifiables. Shannon ne dit rien de l'information portée par une suite particulière. Au contraire, il faudrait comprendre ici que l'information portée par une suite particulière dépend de la probabilité d'occurrence d'autres suites émises par la même source. Il va de soi qu'une quantité propre à un système, une variable d'état comme celle que nous recherchons ne peut dépendre d'autres occurrences potentielles. En particulier, elle doit pouvoir être déterminée à la seule analyse d'une suite décrivant un système particulier.

Mais comment appliquer la théorie de Shannon à une suite particulière? Une première réponse serait de compter les n_i occurrences du mot i de longueur k dans la suite de longueur N , et de définir les fréquences d'apparitions $n_i / \sum_i n_i$ qui tendrait vers p_i si N tend vers l'infini. Il s'avère, lors d'expériences numériques la suite est fatalement finie et l'entropie statistique calculée s'en trouvera systématiquement biaisée. La figure 3.4, montre les mesures de H_k en fonction de k , pour différents types de suites de 2000 caractères. On y observe une divergence plutôt que la convergence attendue. En particulier, la suite utilisée pour le graphique 3.4.c devrait présenter un haut niveau d'organisation (cf. Partie II), qui n'est pas détecté ici. La divergence est due à la taille limitée de N et à l'écart, croissant avec k , entre les probabilités réelles et les fréquences observées. On pourrait donc s'en contenter si on utilisait des suites extrêmement longues, mais la croissance du temps de calcul en fonction de N pour les expériences menées dans la suite exige qu'on se limite à des suites de taille raisonnable.

Certes, les comportements observés ne sont pas identiques pour toutes les suites et on pourrait en tirer quelques indications sur leur entropie respective. Malgré cela, cette mesure ne convient pas pour ce travail, en ce qu'elle exige un choix arbitraire de l'ordre k considéré, ce qui, comme nous l'avons vu, est directement lié à la portée des corrélations identifiables.

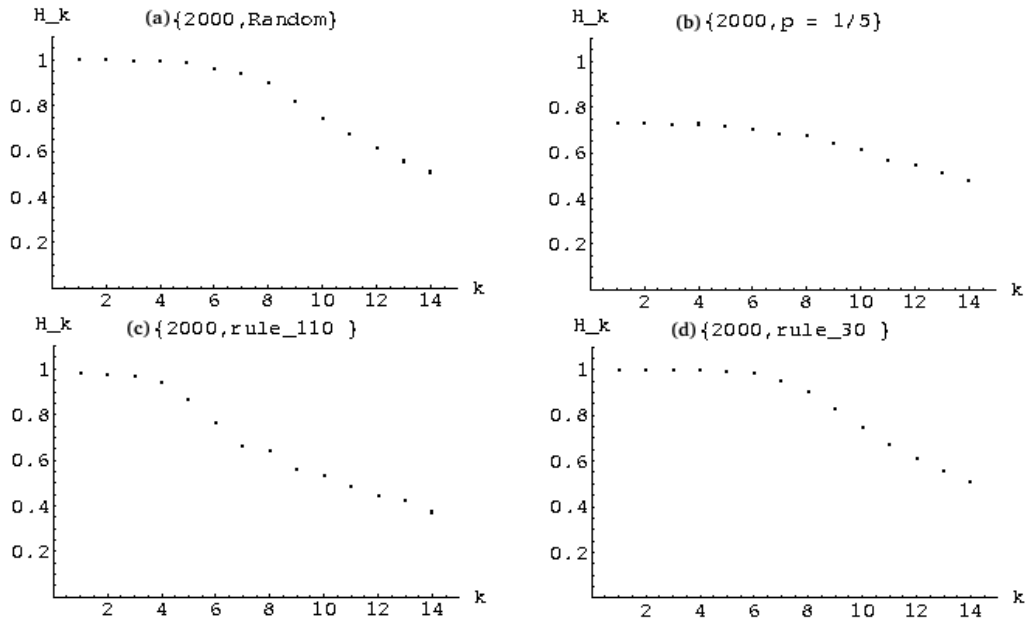


Fig. 3.4 – Mesure de l’entropie de Shannon H_k en fonction de l’ordre k pour différents types de suites de longueurs $N=2000$. (a) Suite aléatoire. (b) suite biaisée, $p = 1/5$. (c) Suite obtenue après 500 pas de l’automate 110 (cf. Partie II.), cette suite devrait présenter un haut niveau d’organisation. (d) Suite obtenue après 500 pas de l’automate 30, cette suite présente une chaoticité topologique (pseudo-aléatoire).

Cherry [14], propose une réinterprétation de la théorie de Shannon qui semble bien mieux ouvrir la voie de la mesure déterministe² et effective de l’information d’une suite particulière : celle de l’information manquante.

3.1.2 Information manquante

Il est intéressant de considérer une autre approche de la même quantité. Imaginons une source délivrant un message en respectant un schéma donné des probabilités d’occurrence des différents symboles et un observateur au courant de ce schéma. Nous avons vu que H quantifiait l’importance du choix que cet observateur aurait à faire pour deviner le dernier caractère émis par la source, en admettant qu’il suive, pour ce faire, une stratégie optimale dans ses questions, selon le schéma mentionné. On peut montrer que cela équivaut au *nombre de questions binaires* à poser pour identifier le caractère en question. Remarquons l’analogie suivante : étant donné un système pouvant se trouver dans Ω états équiprobables, il faudrait $\log_2 \Omega$ questions pour le localiser exactement. Ceci correspond à l’entropie de Boltzmann 2.6 si l’on accepte que 1 bits/symboles correspond à $k \ln 2$ Joules par Kelvins³.

Cette alternative à la définition de Shannon permet de passer des inaccessibles probabilités, dont l’estimation dépend fortement de N , à un nombre de questions ou d’étapes nécessaires à la reconstruction d’une suite, grandeur énumérable et calculable. Cette nouvelle façon d’aborder la question mènera à la définition de la T-Complexité, introduite par Titchener pour appréhender la complexité d’une suite particulière finie. Cet algorithme étant celui utilisé dans la suite de ce

²L’adjectif déterministe est utilisé ici dans le sens où cette mesure ne fait pas intervenir de probabilités.

³Cette valeur de conversion est obtenue dans plusieurs exemples liant l’entropie informationnelle à l’entropie thermodynamique[8].

travail, nous le présenterons plus complètement dans la section 3.2.

3.1.3 Entropie algorithmique de Kolmogorov

Nous discernons à ce stade les suites normales, dont la définition est donnée plus haut, les suites biaisées, pour lesquelles les événements ne sont ni corrélés ni équiprobables, et les suites corrélées pour lesquelles la probabilité d'occurrence d'un événement dépend des événements passés. Les premières sont statistiquement incompressibles, les deuxièmes le sont jusqu'à la limite H_1 , et pour les troisièmes, il faut monter dans les valeurs de k pour trouver la compression optimale.

Mais il nous reste un cas à traiter : il existe en effet des suites qui ont l'apparence de suites aléatoires (elle sont normales au sens de Borel) mais qui découlent en réalité d'un programme très court. Un exemple direct réside dans la suite des décimales de $\pi - 3$ (en binaire)

$s = 001001000011111101101010100010001000010110100011000010001101001100010011 \dots$

L'entropie statistique de cette suite vaut un bit/symbole, et pourtant on pourrait la compresser de manière très efficace en écrivant un des nombreux algorithmes capables de générer les décimales de pi.

C'est le point de vue de Kolmogorov. Simultanément avec Solomonoff et Chaitin, bien que de manière indépendante, il définit [3] l'entropie algorithmique $k(s)$ d'une suite s , comme la longueur du plus court programme permettant de restituer la suite au complet et de s'arrêter ensuite. Cette définition semble dépendre de la machine utilisée pour lire le programme en question. Mais Kolmogorov [3] a montré que l'on peut exiger que le programme doit être exécutable par une machine universelle (capable d'émuler toutes les autres), ramenant de la sorte la définition à une constante additive près⁴.

$$k_u(s) \leq k_M(s) + c_M \tag{3.5}$$

Où c_M est la longueur du programme d'émulation de la machine M par la machine universelle U .

Dans la mesure où les processus physiques sont souvent descriptibles par des lois générales, cette définition semble être propice à définir une entropie physique. Notamment, elle reste applicable dans un cadre élargi, qui déborde de celui de la physique des systèmes désordonnés, pour lesquels K coïncide avec H . Cette assumption se révèle être d'une portée fondamentale, et ce principalement pour la raison qui suit : $k(s)$ s'avère être une grandeur incalculable ! Etant donné une suite s , il n'existe aucun algorithme capable de trouver le plus court programme capable de générer s et de s'arrêter ensuite. Bien sûr dans quelques cas simples, on peut voir que tel programme est indubitablement le plus court mais on ne peut définir une procédure effective qui trouvera ce programme pour n'importe quelle suite en certifiant que c'est là le plus court : il s'agit d'un problème indécidable. Cette indécidabilité est directement issue du problème de l'arrêt des machines de Turing. On pourrait imaginer, par exemple de considérer toutes les suites plus courtes que celle à compresser et de les donner à une machine de Turing universelle pour voir si, par hasard, ce ne serait pas justement là le plus court programme en question. Mais il faut que le programme s'arrête de lui même, et s'il ne s'arrête pas après un certain nombre de pas, il est impossible de savoir si c'est parce qu'il ne s'arrêtera jamais ou parce que l'on a pas effectué assez de pas. On ne

⁴En théorie algorithmique de l'information on considère toujours des suites de longueur N tendant vers l'infini, de sorte que les termes ne dépendant pas de N sont négligeable.

peut donc pas juger effectivement l'issue de ces programmes, d'où l'indécidabilité. Loin d'être une tare de la théorie algorithmique de l'information, cette "incalculabilité" est un de ses résultats les plus importants. En effet, c'est par le biais de cette entropie algorithmique que l'indécidabilité fait son apparition en physique. Ce fait est d'une portée philosophique considérable, mais malgré que le sujet soit passionnant, nous considérons qu'il sort du cadre de notre travail et nous ne l'aborderons pas ici plus avant.

Il importe de remarquer que contrairement à l'entropie de Shannon, qui s'intéresse à l'information qu'une source donnée est capable de produire, l'entropie de Kolmogorov concerne une suite individuelle. En d'autres termes, H caractérise les messages émis par une source donnée, $k(s)$ caractérise un message donné.

Si on note K le nombre de bits de ce programme optimum par symbole de la suite initiale, soit $K(s) = \frac{k(s)}{|s|}$, K s'exprime en bit/symbole. Il découle immédiatement que pour une suite s donnée on aura toujours $K(s) \leq H(s)$ puisque on peut écrire comme programme

print [s_{opt}]

où s_{opt} représente la suite comprimée optimalement du point de vue statistique.

Toutefois, *L'immense majorité des suites ne sont pas structurées par un programme court.* Il en résulte que l'approche statistique est exacte, sauf pour un ensemble dénombrable, et par conséquent négligeable, de suites.

Ce résultat est basé sur le fait que l'infinité des suites de bits infiniment longues est non-dénombrable, tandis que celle des programmes provoquant l'arrêt des machines de Turing est dénombrable. On peut, en effet, mettre en bijection l'ensemble des suites de bits de longueurs infinies avec l'ensemble, non-dénombrable, des réels compris entre 0 et 1. Imaginons, par exemple, de partitionner $[0,1[$ en deux parties et d'en prendre la première moitié si le premier bit est un 0, ou la seconde si c'est un 1. On partitionne ensuite cette moitié en deux et on en prend une des deux selon la valeur du deuxième bit, et ainsi de suite. La dénombrabilité des programmes provoquant l'arrêt des machines de Turing provient principalement du fait que les programmes provoquant l'arrêt de la machine doivent être de longueur finies. Suivant la même méthode, on peut donc mettre en bijection l'ensemble des programmes avec l'ensemble, dénombrable, des rationnels.

Type de suite	Définition	Contenu Informationnel
Normale	Equiprobabilité des singletons, paires, triplets, etc...	$H_\infty = H_k = 1$ <i>symbole/symbole</i> , $\forall k$.
Biaisée	Evénements successifs non-corrélés mais probabilités d'événements différents non-égales.	$H_\infty = H_k = H_1$, $\forall k$.
Correlée	L'occurrence d'un événement dépend des événements précédents.	$H_\infty = H_k < H_j$, $\forall j < k$, pour des corrélations de portée k .
Structurée	Issues d'un programme court, peuvent être d'apparence normale.	$K < H_\infty$ mais incalculable. La meilleure approximation calculable est H_∞ .

TAB. 3.1 – Tableau récapitulatif des différents types de suites et des moyens de calculer leur contenu informationnel.

3.2 Mesure déterministe de la complexité.

Concrètement, comment peut-on mesurer la complexité d'une suite finie ?

Comme on l'a vu, il est vain de vouloir mesurer K , qui nous est inaccessible. Mais ce n'est pas grave, puisque nous avons vu aussi que l'information de Shannon était pratiquement toujours équivalente. On peut donc se contenter de cette dernière. Malheureusement, la définition de l'entropie de Shannon ne permet pas cette mesure déterministe souhaitée, puisqu'elle dépend d'inaccessibles probabilités et est par conséquent entâchée d'erreurs pour des suites finies. Bien que ces erreurs puissent être estimées[5], nous nous trouvons devant un choix arbitraire quand il s'agit de choisir l'ordre k auquel on évalue cette entropie.

L'équivalence, évoquée plus haut, de l'information manquante permet de relier l'information contenue dans une suite et l'effort à fournir pour reconstituer celle-ci. Nous allons voir dans cette section comment cette alternative d'interprétation permet de définir une quantité d'information effectivement mesurable sur des suites finies.

3.2.1 La Complexité de production de Lempel & Ziv

Lempel et Ziv, motivés par la recherche d'un algorithme de compression effective⁵, ont cherché à mesurer l'effort à fournir pour reconstruire une suite à partir de son "vocabulaire" [15].

Leur méthode consiste à établir un vocabulaire à partir de la suite de la manière suivante. Nous désignerons par \mathcal{A}^* l'ensemble de toutes les suites finies formées à partir de l'alphabet \mathcal{A} et par $\mathcal{A}^+ = \mathcal{A}^*/\Lambda$ son homologue excluant la suite vide Λ . Le sous-ensemble \mathcal{A}^n denotera l'ensemble des suites de longueur n . Soit une suite $x(n) \in \mathcal{A}^n$, Lempel et Ziv mesureront sa complexité de production C_s comme l'effort minimum requis pour reconstruire x à partir de \mathcal{A} . Ils utilisent pour ce faire un algorithme RLPC⁶, et mesurent cet effort comme étant le nombre de pas RLPC à effectuer pour reconstituer la suite. Pratiquement, on lit la suite de gauche à droite en la décomposant en sous-suites $p_i \in \mathcal{A}^+$ non-vues précédemment. Chacun de ces nouveaux mots p_i lus est ajouté dans un vocabulaire \mathcal{V} . Partant de \mathcal{V} initialement vide, on y ajoutera au fur et à mesure les mots p_i

⁵Lequel algorithme est utilisé dans la méthode ZIP de compression de fichiers.

⁶Recursive Linear Pattern Copying Algorithm

obtenus en suffixant les mots déjà présents dans l'alphabet pour en obtenir des nouveaux. Le premier mot inscrit sera automatiquement un symbole de l'alphabet utilisé.

Considérons l'exemple $x = 01000111011110111$, où nous avons choisi $\mathcal{A} = \{0, 1\}$. Le premier symbole est le 0, puis vient le 1. Les deux premiers mots ajoutés au vocabulaire seront donc $\mathcal{V} = \{0, 1\}$. Le symbole suivant est un 0, mais il est déjà présent dans \mathcal{V} . On annexe donc le symbole suivant pour former le mot suivant et on obtient $\mathcal{V} = \{0, 1, 00\}$. On continue ensuite de la même manière jusqu'à obtenir $\mathcal{V} = \{0, 1, 00, 01, 11, 011, 110, 111\}$. \mathcal{V} est donc bien un compte-rendu de l'histoire de la production de x et la taille de son vocabulaire, ici $\#\mathcal{V} = 8$, est une mesure de la complexité de production $C_s(x(n))$. $\#\mathcal{V}$ sera, en effet, plus petit pour les suites présentant une structure répétitive que pour les suites complexes. En particulier, pour une source aléatoire et pour $n \rightarrow \infty$, $C_s(x(n))$ est borné par une fonction de l'entropie de Shannon, mais aucune relation générale entre ces deux valeurs n'a été démontrée.

3.2.2 La T-complexité de Titchener

La T-Complexité C_T d'une suite est définie ([7],[6]) d'une manière similaire à $C_s(x(n))$. Les différences résident en ceci que les mots sont, cette fois, créés à partir des mots déjà présents dans le vocabulaire⁷ et que la participation de chaque mot à C_T est pondérée par le nombre de fois que ce mot se répète.

La suite $x(n) \in \mathcal{A}^n$ est parcourue pour être décomposée en mots $p_i \in \mathcal{A}^+$ auxquels sont associés des exposants $k_i \in \mathbb{N}^+$, avec $i = (1, 2, \dots, q)$, où $q \in \mathbb{N}^+$ respecte

$$x = p_q^{k_q} p_{q-1}^{k_{q-1}} \dots p_i^{k_i} \dots p_1^{k_1} a_0, \quad a_0 \in \mathcal{A}^+ \quad (3.6)$$

Dans cette expression, mot p_i est contraint d'être composé des mots précédents comme suit :

$$p_i = p_{i-1}^{m_{i,i-1}} p_{i-2}^{m_{i,i-2}} \dots p_j^{m_{i,j}} \dots p_1^{m_{i,1}} a_i, \quad a_i \in \mathcal{A} \text{ et } 0 \leq m_{i,j} \leq k_i \quad (3.7)$$

La complexité C_T est finalement définie par la somme :

$$C_T = \sum_{i=1}^q \log_2(k_i + 1) \quad (3.8)$$

Cette sommation permet de pondérer la participation d'un terme en fonction du nombre de fois qu'il se répète de manière bien plus efficace que dans l'algorithme L&Z. Deux mots ne se répétant pas participeront à C_T à concurrence de $2 \log_2(1 + 1) = 2$ taugs⁸. Un même mot se répétant deux fois participera, lui, à concurrence de $\log_2(1 + 2) = 1.585$ taugs.

Concrètement l'algorithme s'effectue en plusieurs pas que nous illustrons par un exemple.

Soit $\mathcal{A} = \{0, 1\}$ et $x = 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1$ que l'on considère comme une suite de mots de longueurs 1.

1. Le dernier symbole, $a_0 = 1$ est laissé de côté pour une raison qui sera bientôt évidente.
2. On commence par sélectionner l'avant-dernier symbole qui sera notre premier mot $p_1 = 0$.
3. En parcourant la suite *de droite à gauche*, on constate qu'il n'apparaît consécutivement qu'une seule fois, on a donc $k_1 = 1$.

⁷on utilise ici un "Recursive Hierarchical Pattern Algorithm" (RHPC)

⁸l'unité utilisée est expliquée plus loin.

4. On parcourt maintenant la suite de *de gauche à droite* en regroupant chaque occurrence de p_1 avec le mot suivant. On obtient $x = 1, 1, 01, 1, 01, 1, 00, 00, 1, 01, 1, 1, 01$, où l'on peut voir que le dernier symbole a_0 devait bien être laissé de côté pour pouvoir être groupé avec l'avant dernier.
5. On recommence ensuite les étapes 2 à 4. On identifie l'avant-dernier mot : $p_2 = 1$. Il apparaît maintenant deux fois à la suite : $k_2 = 2$. Cette fois, puisque $k_2 = 2$, on regroupera chaque occurrence de p_2 avec le mot suivant *ou* avec les mots suivants, jusqu'à deux fois, si p_2 se répète. On obtient $x = 1101, 101, 100, 00, 101, 1101$.
6. Les étapes suivantes sont obtenues en prenant l'avant dernier mot pour p_i , en comptant de droite à gauche les k_i occurrences, et en regroupant de gauche à droite les k_i occurrences de p_i avec leur voisins immédiats. Elles donneront les résultats affichés ci-dessous (nous reprenons aussi les premières étapes pour permettre une vue d'ensemble) :

$$\begin{array}{llll}
a_0 = 1 & & \rightarrow x = 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1. \\
p_1 = 0 & k_1 = 1 & \rightarrow x = 1, 1, 01, 1, 01, 1, 00, 00, 1, 01, 1, 1, 01. \\
p_2 = 1 & k_2 = 2 & \rightarrow x = 1101, 101, 100, 00, 101, 1101 \\
p_3 = 101 & k_3 = 1 & \rightarrow x = 1101, 101100, 00, 1011101. \\
p_4 = 00 & k_4 = 1 & \rightarrow x = 1101, 101100, 001011101. \\
p_5 = 101100 & k_5 = 1 & \rightarrow x = 1101, 101100001011101. \\
p_6 = 1011 & k_6 = 1 & \rightarrow x = 1101101100001011101.
\end{array}$$

On poursuit ainsi jusqu'à ce que la suite entière soit réduite à un mot, ce qui marque la fin de l'algorithme. On peut maintenant calculer la $C_T = \sum_{i=1}^6 \log_2(k_i + 1) = 6.58496$ taugs. L'unité utilisée, le *taugs*, correspond au nombre de pas de T-augmentation effectués. En effet pour alléger le texte nous n'avons introduit ici que le côté pratique de la méthode sans parler de la théorie sous-jacente[6] qui fait intervenir les notions de T-augmentation de codes. Il est cependant à remarquer que chacune des étapes de regroupement correspond à un décodage de la suite en un code préfixe, ce code étant à chaque fois *augmenté* jusqu'à contenir la suite elle-même comme un de ses mots les plus longs.

Il est amusant de remarquer qu'en fait la mesure réellement effectuée est celle de l'effort nécessaire pour construire le vocabulaire à partir de la suite plutôt que l'inverse, mais cela est évidemment équivalent.

On peut dériver analytiquement, une borne supérieure [6], et la borne inférieure est définie en considérant une suite répétant le même caractère.

$$\log_2 n \leq C_T(x(n)) \leq li(n \ln(\#\mathcal{A})) \quad (3.9)$$

Où la fonction $li(z)$ est la fonction logarithmique intégrale qui peut s'écrire $li(z) = \int_{\mu}^z \frac{1}{\log u} du$, avec $\mu = 1.451369$ si z est plus grand que μ [33].

3.3 Lien entre la complexité de Titchener et l'entropie de Shannon : la T-entropie

La T-entropie est une fonction linéarisée de la T-complexité définie par ce qui suit. Comme exprimé plus haut, on a [6] pour une suite $x(n)$ de longueur n :

$$\log_2 n \leq C_T(x(n)) \leq li(n \ln(\#\mathcal{A})) \quad (3.10)$$

On pourrait imaginer un coefficient H_T tel que ,

$$C_T(x(n)) = li(n H_T \ln(\#\mathcal{A})) = li(I_T) \quad (3.11)$$

En effectuant quelques transformations, les inégalités ci-dessus deviennent successivement :

$$\log_2 n \leq li(n H_T \ln(\#\mathcal{A})) \leq li(n \ln(\#\mathcal{A})) \quad (3.12)$$

$$li^{-1}(\log_2 n) \leq n H_T \ln(\#\mathcal{A}) \leq n \ln(\#\mathcal{A}) \quad (3.13)$$

$$\frac{li^{-1}(\log_2 n)}{n} \leq H_T \ln(\#\mathcal{A}) \leq \ln(\#\mathcal{A}) \quad (3.14)$$

$$(3.15)$$

Pour $n \rightarrow \infty$, le terme de gauche tend vers 0, on a donc $0 \leq H_T \leq 1$, dont l'unité est le symbole/symbole, ici bit/bit.

Titchener, à partir de cette forme et de résultats empiriques [7] conclut une correspondance entre H_T et H l'entropie de Shannon définie plus haut, ceci bien qu'il n'existe aucune relation analytique directe et générale entre ces deux grandeurs. Nous mesurons donc la quantité $H_T(x(n)) = \frac{li^{-1}(C_T(x(n)))}{n \ln(2)}$ bit/bit.

Avant de pouvoir interpréter les résultats issus de cette mesure, il nous faut tester son comportement dans certains cas connus afin de pouvoir en juger la pertinence.

3.3.1 Dépendance de la T-entropie en N.

Dans un premier temps nous examinons la dépendance de H_T avec N , la longueur de la suite considérée. Nous avons utilisé pour ce faire les chiffres binaires des nombres π , et e .

Pourquoi ces suites? Bien que cela n'ait jamais été ni démontré ni infirmé, ces suites sont communément[11] admises comme étant normales. Or c'est bien le comportement de la mesure sur une suite normale qui nous intéresse ici.

Pour chacune de ces deux suites, nous avons donc calculé la T-entropie pour des parties de plus en plus longues. Les résultats obtenus sont montrés sur la figure 3.5 pour e . Le même comportement est observé pour les chiffres binaires de π et pour une suite aléatoire construite à partir de la fonction `Random[]` de Mathematica™⁹. Une dépendance en N est visible, comme on s'y attendait, mais il y a plus surprenant : la valeur vers laquelle on semble converger est bien en dessous de 1 ! La relation entre H_T et H_S semble en effet devoir passer par un facteur de renormalisation[20]. Rappelons-nous que la définition de l'entropie de Shannon est définie à un facteur K près (cf. Appendice 1). Certes, la valeur adoptée par Shannon, $K=1$, est justifiée par la correspondance entre l'entropie en bit/symbole propre à une suite biaisée et le nombre de bits minimum nécessaire pour la décrire(cf. Compression de Huffman, section 3.1.1). Dans le cas de la méthode de Titchener, il n'y pas de lien direct avec une méthode de compression permettant de justifier ce facteur de renormalisation.

⁹Comme on peut s'y attendre, ceci montre que cette mesure ne détecte pas la structure algorithmique des suites propres à π et e

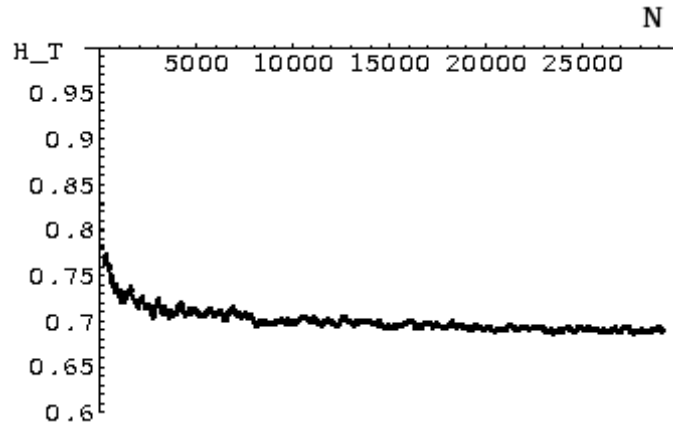


Fig. 3.5 – H_T calculée en fonction de N , sur la suite des chiffres binaires de e . La courbe ci-dessus à été obtenue via un lissage par moyenne sur 20 points voisins (à partir d'un point tout les $N = N + 20$).

La méthode utilisée ici[20] consiste à mesurer H_T pour des suites aléatoires d'une longueur N fixée et à en déduire le facteur de normalisation correspondant, en imposant que la valeur obtenue sur ces suites corresponde à 1 bit/bit.

Il n'y a donc plus qu'à fixer un N pour les expériences à venir et à identifier le facteur de renormalisation correspondant.

3.3.2 Mesures sur des suites aléatoires

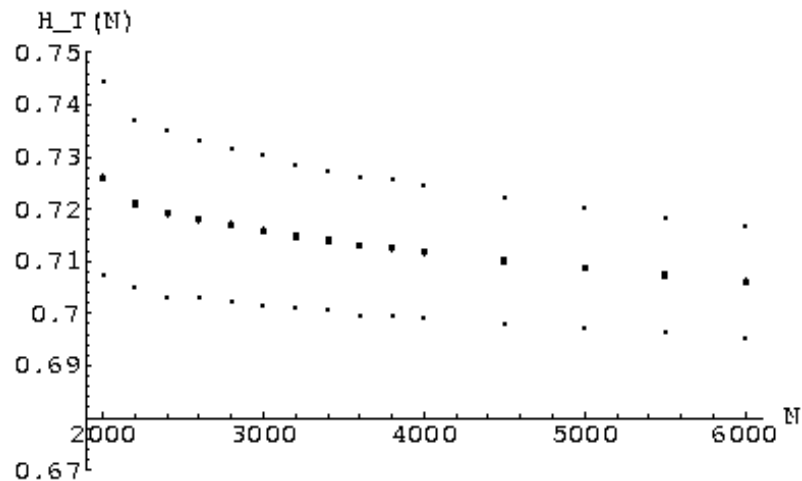


Fig. 3.6 – Valeur moyenne $\mu(N)$ et écart-type correspondant de H_T mesurés pour différentes valeurs de N . Les moyennes ont été effectuées sur 5000 suites de longueur N , générées aléatoirement.

Nous avons donc mesuré H_T , pour différentes valeurs de N , en considérant dans chaque cas 5000 suites générées aléatoirement par Mathematica™. Pour chaque N les valeurs obtenues pour les 5000 suites se répartissent en une distribution gaussienne. La figure 3.6 montre la valeur moyenne $\mu(N)$ de la T-entropie et l'écart-type correspondant pour différentes valeurs de N . Nous retiendrons donc les valeurs $\mu(N)$ comme étant les facteurs de renormalisation pour des suites de longueur N et noterons $H_{TN}(N) = \frac{li^{-1}(C_T(x(N)))}{N \ln(2) \mu(N)}$, la-T-entropie normalisée.

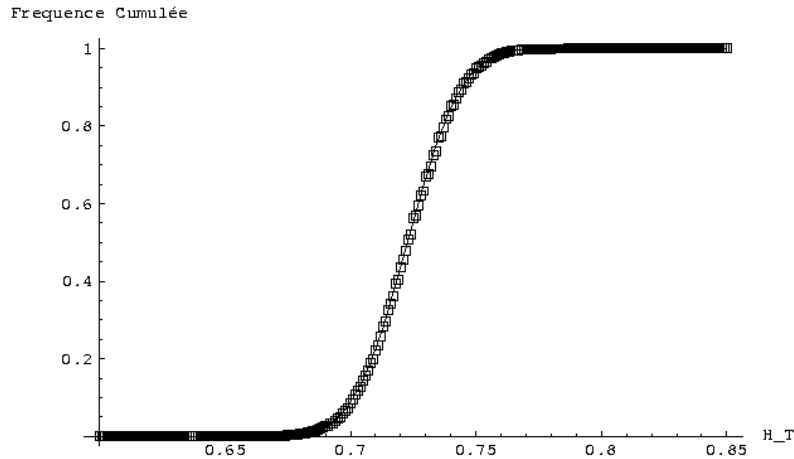


Fig. 3.7 – Distribution obtenue pour 10^4 suites aléatoires, $N = 2000$. Afin de se défaire du choix arbitraire des classes, on montre ici les fréquences cumulées. La ligne continue montre le comportement théorique d’une distribution normale ($\mu = 0.722897$, $\sigma = 0.0168699$). Les petits carrés correspondent aux résultats obtenus.

La figure 3.7 montre les fréquences cumulées de la distribution de H_T dans le cas particulier $N=2000$, pour 10000 suites aléatoires. L’axe vertical représente donc la proportion de ces 10000 suites pour lesquelles la T-entropie est plus petite que la valeur en abscisse. La ligne continue représente le cas théorique d’une distribution normale $N(\mu, \sigma)$ avec $\mu(2000) = 0.722897$ et $\sigma = 0.0168699$, qui correspondent aux valeurs de la moyenne et de la déviation standard calculées pour cette série de mesures. Dans la suite $H_T N$ désignera la T-entropie normalisée pour des suites de 2000 caractères. Nous préciserons explicitement $H_T N(N)$ lorsque $N \neq 2000$, auquel cas l’entropie aura donc été normalisée selon un autre facteur (cf. fig. 3.6).

3.3.3 De la largeur de la distribution

Cette distribution gaussienne amène cependant quelques soucis de cohérence théorique. En effet, une fois renormalisée, la mesure donne bien un information de 1 bit/bit *en moyenne*. Mais cela signifie que, ponctuellement, il existe des suites, situées à droite de la distribution, pour lesquelles on mesure une entropie plus grande qu’1 bit/bit. Or il est évident qu’un bit ne peut transporter plus qu’un bit d’information. En particulier, les suites que nous considérons dans la deuxième partie et pour lesquelles nous avons recours à cette mesure représentent les états d’un automate aux conditions aux bords cycliques. Les résultats obtenus en mesurant H_{TN} sur les équivalents cycliques d’une suite doivent donc être considérés comme équivalents. Lors du calcul de H_T pour les 2000 équivalents cycliques d’une suite aléatoire, nous obtenons une distribution identique à celle présentée plus haut. Nous assumons cette largeur dans la distribution comme étant un artefact dû à la finitude des suites analysées et nous la considérerons par la suite comme une erreur caractéristique sur la mesure de H_{TN} .

Notons cependant que la déviation standard diminue avec N (cf. figure 3.8). Ceci est en accord avec la théorie de Shannon, impliquant une valeur unique pour les suites aléatoires infiniment longues.

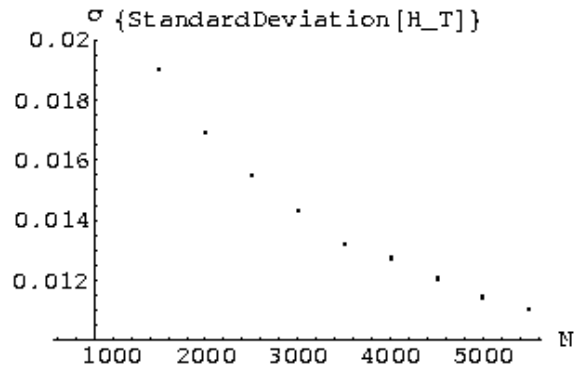


Fig. 3.8 – Déviation standard sur les mesures de H_T en fonction de N . Ces mesures correspondent à celles de la figure 3.6.

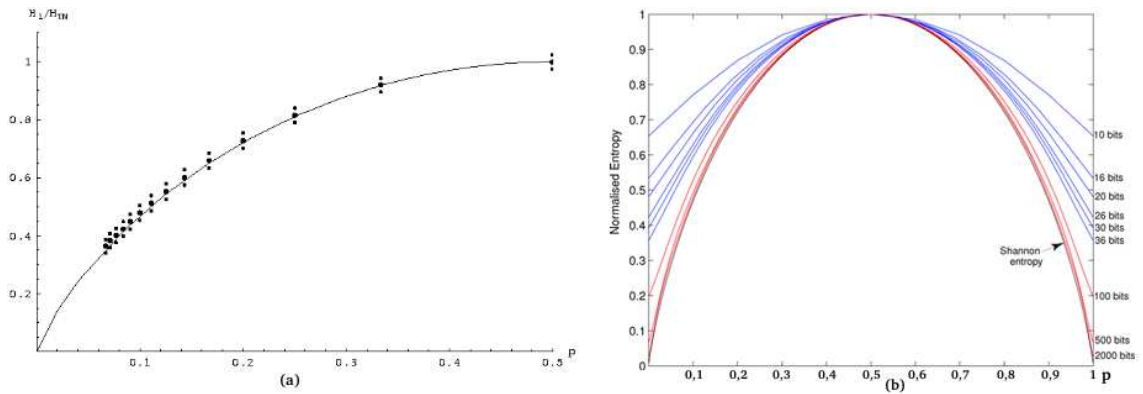


Fig. 3.9 – (a) Comparaison des mesures de H_{TN} obtenues sur des suites biaisées de 2000 bits, en fonction du biais p . La ligne continue représente la prédiction théorique de Shannon, et chaque point correspond à une moyenne sur 1000 mesures, accompagné des valeur $\mu \pm \sigma$. (b) Résultats obtenus par M. Titchener [20].

3.3.4 Le cas des suites biaisées

Au delà de la renormalisation pour des suites normales, il nous faut savoir si la correction à effectuer est la même pour les suites biaisées. C'est précisément ce caractère non-normal qu'il nous plairait de mettre en évidence. Pour ce faire, nous avons généré grâce à Mathematica™ des suites de plus en plus biaisées, avec p la probabilité de 1 variant sur $\{1/2, 1/3, \dots, 1/14, 1/15\}$, et calculé à nouveau les valeurs de H_{TN} correspondantes. Les points affichés sur la figure 3.9.a correspondent aux valeurs moyennes obtenues sur 1000 suites pour chaque valeur de p . La légère surestimation occurant pour les petites valeurs de p est en bon accord avec les prédictions de Titchener[20] (cf. Fig. 3.9.b). La distribution reste gaussienne pour chaque valeur de p et l'écart-type reste sensiblement égal.

En vue de mesurer la sensibilité de la T-entropie à des corrélations d'un ordre supérieur, nous avons imaginé d'effectuer le même type d'expérience mais de façon à biaiser, cette fois, les probabilités à l'ordre deux, en laissant les occurrences des singletons équiprobables. Pour pouvoir en faire une représentation graphique, nous avons ramené ce biais à une seule quantité p . Les suites ont été générées en générant des paires de bits selon l'arbre de probabilité présenté sur la figure 3.10.

On tire d'abord un 0 ou un 1, avec un probabilité $1/2$. Ensuite, selon le deuxième tirage de

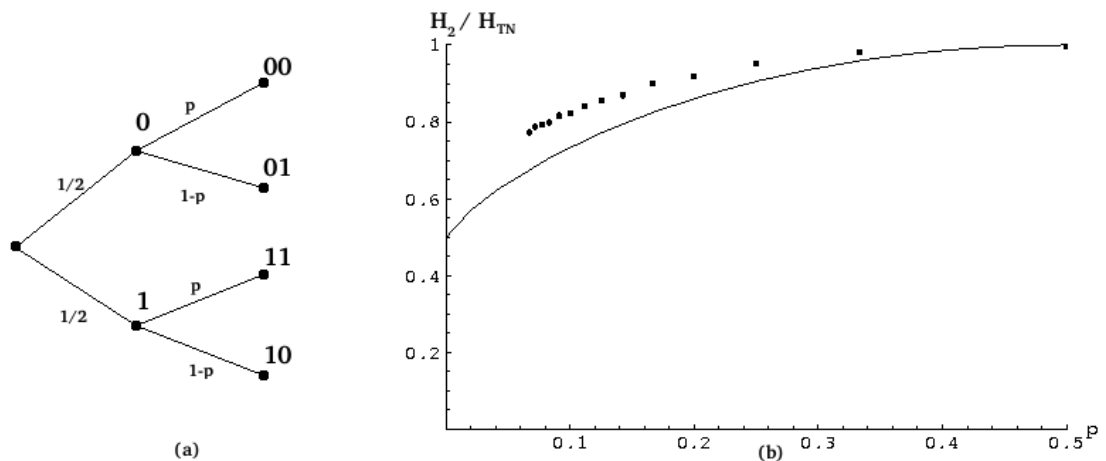


Fig. 3.10 – (a) Arbre de probabilité utilisé pour la génération de suites biaisées aux second ordre. (b) Correspondance entre la valeur mesurée H_{TN} et la valeur théorique donnée par la fonction de Shannon H_2 .

probabilité p , la paire de bits ajoutée à la suite sera, ce symbole répété deux fois ou, ce symbole suivi de son opposé. De cette manière les occurrences des singletons sont gardées équiprobables et le biais, introduit au second ordre, ne dépend que d'une quantité p . Les probabilités associées aux quatre issues sont donc $\{p[00] \rightarrow \frac{p}{2}; p[01] \rightarrow \frac{1-p}{2}; p[11] \rightarrow \frac{p}{2}; p[10] \rightarrow \frac{1-p}{2}\}$. Le calcul du second ordre de l'entropie de Shannon donne $H_2 = -\frac{1}{2} [2 \cdot \frac{p}{2} \log(\frac{p}{2}) + 2 \cdot \frac{1-p}{2} \log(\frac{1-p}{2})]$. C'est la fonction représentée sur la figure 3.10.b. L'écart à la théorie s'y manifeste beaucoup plus rapidement, au fur et à mesure que l'on s'écarte de l'équiprobabilité. On peut expliquer ce fait de la façon suivante : les symboles que nous considérons ici ne sont plus des singulets mais des paires. Or une suite de longueur N , ne contient que $N/2$ paires. Puisque, comme nous l'avons vu, l'écart entre la mesure et la théorie dépend de N (cf. fig. 3.9.b), il est compréhensible que cet écart mesuré pour des corrélations d'un ordre supérieur y soit plus sensible. Néanmoins, l'écart observé ici est plus grand que celui observé par Titchener pour des suites de 1000 caractères (cf. fig. 3.9.b). Ceci semble montrer que si la mesure H_{TN} est sensible aux biais des probabilités, cette sensibilité décroît en fonction de l'ordre k des corrélations. Cette sensibilité s'accroît toutefois lorsque l'on considère des suites de plus en plus longues.

3.3.5 Discussion

La procédure suivie dans cette première partie est cohérente avec la théorie de l'information, en ce sens qu'elle suit le même cheminement. Ici, on pose que les suites aléatoires achèvent une entropie de 1 bit/bit, et on est donc amené à effectuer une normalisation. La même procédure est suivie en théorie de l'information, où l'on définit une suite aléatoire par son caractère incompressible, d'où la norme de Shannon, $K=1$.

Mais cette théorie de l'information est basée sur la considération de suites infinies. En particulier, pour ces suites infinies, la constante de l'information contenue dans la table de décodage, nécessaire à la décompression, est négligeable par rapport à l'information contenue dans la suite. Plusieurs considérations mènent à se poser des questions sur l'applicabilité de la théorie de l'information aux suites finies.

Par exemple, il existe beaucoup plus de suites contenant 532 bits d'information que de suites de 532 bits (pour les 2^{532} suites, il existe une infinité de tables de décodage possibles¹⁰).

La distribution gaussienne peut être comprise de la manière suivante. Imaginons l'ensemble des suites de longueur N . De par leur caractère fini, ces suites seront nécessairement biaisées, d'une manière ou d'une autre. Choisissons une table de décodage particulière et appliquons-la à toutes ces suites. Certaines de ces suites en seront compressées mais d'autres, pour lesquelles la table n'est pas optimale, en verront leur longueur augmentée. Ceci est nécessaire, car il n'y pas assez de suites plus courtes que N pour y trouver des suites représentant de manière univoque toutes les suites de longueur N . On comprend donc que certaines suites semblent porter une information en dessous de la moyenne et d'autres en dessus. La justification de la forme gaussienne de cette distribution trouve son explication dans la théorie de l'information, mais nécessite des résultats non-évoqués ici. Nous ne nous y attarderons donc pas.

Mark Titchener voit les choses un peu différemment[20]. Pour lui, il est étonnant qu'une suite générée aléatoirement possède justement le maximum d'entropie possible, ou, en d'autres termes, qu'elle réalise systématiquement la compression maximum de l'information qu'elle contient. A l'instar d'un tas de sel créé en versant du sel dans une salière (de manière aléatoire), qui n'occupe pas le taux de compression maximum et qu'il est possible de comprimer ultérieurement en tapotant sur la salière du bout de son doigt (un algorithme peu coûteux et très efficace), une suite aléatoire finie aurait, pour lui, une entropie non-maximum.

Pour Titchener, l'incompressibilité des suites aléatoires n'est pas due au fait qu'elles contiennent l'information maximum de 1 bit/bit, mais tout simplement parce qu'il n'existe pas assez de suites courtes pour les décrire toutes uniquement. C'est là, toujours selon lui, la raison pour laquelle un ensemble de suites générées aléatoirement verra une distribution gaussienne de l'entropie. La suite typique, correspondant au top de la distribution, ne contient qu'une information située entre le minimum et le maximum possibles pour des suites de cette longueur, tandis que d'autres contiennent une information supérieure ou inférieure.

Il ne s'agit pas ici de refaire la théorie de l'information, et la procédure suivie sauve la mise en assumant la largeur de la distribution comme due à une erreur caractéristique. Néanmoins, nous tenions à mettre ces questions en évidence.

Nous reparlerons de la T-Entropie dans la section 5.2 lorsque nous l'utiliserons pour mesurer la variation de l'entropie d'une suite variant sous l'action d'un automate cellulaire, mais, auparavant, il convient de préciser ce que sont exactement ces fameux automates.

¹⁰Nous parlons ici de tables du style de celles obtenues par l'algorithme de Huffman.

Deuxième partie

Les automates cellulaires

Dans cette deuxième partie, nous utiliserons les outils de la théorie de l'information pour mettre en valeur certains comportements intéressants des automates cellulaires.

Dans un premier temps nous décrirons succinctement les automates cellulaires (A.C.) par quelques définitions et notations. Nous illustrerons ensuite les propriétés principales de ces étonnants systèmes, en en discutant les possibilités de classification. Enfin, nous présenterons quelques analyses personnelles plus précises effectuées sur ces derniers. Nous y caractériserons les automates par leur dépendance aux conditions initiales (C.I.), par l'étude des variations de la T-entropie d'une suite sous l'action d'un automate et par la considération de la dynamique des A.C. dans l'espace de phase qui leur est associé. Ce dernier point nous amènera à considérer les notions d'automates injectifs, surjectifs et inversibles et à discuter de la compatibilité des observations faites avec la généralisation des comportements globaux sous forme d'un principe unique, du type du second principe.

Chapitre 4

Familiarisation aux Automates Cellulaires

4.1 Définition et notations

Un automate cellulaire (A.C.) est un système discret dans l'espace et dans le temps. Généralement il consiste en un ensemble de cellules pouvant présenter un nombre fini d'états et évoluant simultanément par pas temporels discrets. A chaque pas, des lois d'évolution assignent à chaque cellule un nouvel état, en fonction de l'état des cellules comprises dans un certain voisinage autour de la cellule en question. On peut imaginer des automates de toutes dimensionalités et de tout types.

Formellement un automate \mathcal{A} est défini par

- sa dimension d et sa topologie (géométrie du réseau, conditions aux bords et taille N dans le cas d'un tableau fini, ...)
- le nombre S d'états que peuvent présenter ses cellules. \mathcal{S} est l'ensemble fini des états accessibles à une cellule, $\mathcal{S} = \{s_1, s_2, \dots, s_S\}$.
- son vecteur de voisinage $V = \{v_1, v_2, \dots, v_n\}$ ¹.
- sa règle d'évolution locale, $f : \mathcal{S}^n \rightarrow \mathcal{S}$.

La règle f peut être soit probabiliste et faire intervenir une composante aléatoire, soit déterministe. De même, elle peut être soit détaillée précisément et être définie dans chaque configuration précise de V , soit totalistique et ne dépendre que du nombre de cellules présentant tel état dans le voisinage sans tenir compte de leurs positions.

On appelle configuration une application $c : \{1 \dots N^d\} \rightarrow \mathcal{S}$ qui associe un état à chaque cellule du tableau et on notera C l'ensemble de toutes les configurations, avec $\#C = S^{N^d}$.

On notera F la fonction de transition globale, $F : C \rightarrow C$, qui consiste en l'action simultanée de la règle locale f sur toutes les cellules du tableau. Généralement le voisinage est adjacent et on définit r , la portée d'interaction. On a donc, à une dimension, $n = 2r + 1$.

¹On peut introduire la notion d'histoire et prendre en compte l'état qu'arborait la cellule plusieurs pas temporels auparavant, mais cela est équivalent à un système du premier ordre (pas de passé) comportant des états supplémentaires

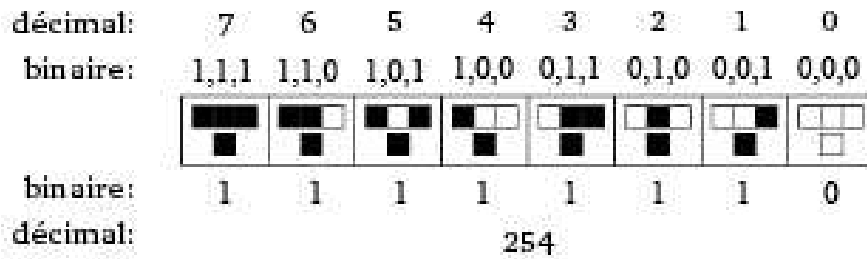


Fig. 4.1 – Détail de la représentation de Wolfram pour la règle 254.



Fig. 4.2 – Représentation de l'évolution pour la règle 254.

Les investigations réalisées dans le cadre de ce travail concernent principalement l'étude des automates les plus simples. Nous ne perdons pas, cependant, en généralité en restreignant ainsi le champ de notre étude, car ceux-ci présentent déjà tous les types de comportements fondamentalement dignes d'intérêt. Ces automates, appelés automates élémentaires, sont donc des automates à une dimension, composés de cellules pouvant présenter 2 états différents et interagissant avec leurs premiers voisins de manière déterministe, soit $d = 1, r = 1, k = 2$.

Le voisinage V contient donc trois cellules : celle qui va évoluer, et ses deux voisines directes. Puisque chacune de ces cellules peut présenter 2 états, le voisinage peut donc présenter $2^3 = 8$ formes différentes. Il suffira, pour définir complètement f , de préciser l'état que doit prendre la cellule dans chacune des 8 configurations possibles de V . Ceci a conduit S. Wolfram [12] à définir pour les règles une représentation simple que nous utiliserons tout au long de ce travail. L'exemple illustré sur la figure 4.1 permet d'en comprendre rapidement le mécanisme. Le principe est basé sur l'unicité de la correspondance entre nombre décimaux et nombres binaires. On lit en décimal le nombre formé par les états (binaires) que prend la cellule dans les 8 configurations possibles, mises dans l'ordre lexicographique.

Etant donné que pour chacune des 8 configurations de V on peut assigner deux états différents, il existe $2^8 = 256$ automates élémentaires. Parmi ceux-ci beaucoup sont équivalents par inversion noir-blanc, ou symétrie gauche-droite [30]. Au total, il reste 88 règles fondamentalement différentes que nous appellerons automates fondamentaux.

L'étude des automates à une dimension offre le confortable avantage de pouvoir visualiser directement l'évolution sur papier. Il suffit pour ce faire de disposer l'une en dessous de l'autre les lignes correspondantes aux étapes successives (cf Fig4.2).

Sur cette image on a présenté assez peu de pas pour que la zone noire ne touche pas les bords.

Tout se passe comme si le tableau était infini mais il est évident que pour un automate fini il est nécessaire de préciser l'état des voisins des cellules situés aux extrémités. Dans le présent travail nous avons systématiquement considéré des conditions aux bords cycliques. C'est à dire que le voisin de droite de la cellule situé à l'extrême droite du tableau sera celle située à l'extrême gauche, et inversement. Tout se passe donc comme si l'automate évoluait sur la paroi d'un cylindre que nous aurions déroulé pour obtenir les figures présentées dans ce travail.

L'automate n 254 présenté dans les exemples ci-dessus n'est pas passionnant : son évolution est univoque. Heureusement ce n'est pas le cas pour tous les automates, sans quoi ce mémoire aurait peu d'intérêt. Les automates, en effet, montrent des comportements très divers que l'on peut rassembler sous quatre types caractéristiques. Ces quatre classes, qui ont donné naissance à la classification de Wolfram, sont détaillées dans la section suivante.

4.2 Classification

4.2.1 Les quatre classes de Wolfram

Lors de ses études sur les automates cellulaires S. Wolfram a mis en évidence quatre types de comportements caractéristiques, sur base desquels il a construit une classification phénoménologique [16]. Cette classification est loin d'être complète car elle ne donne ni critères précis, ni moyens effectifs, applicables directement à un automate donné, de spécifier son appartenance à une classe ou une autre. Néanmoins, cette classification est commode pour la discussion en ce sens qu'elle se base sur des phénomènes observables et se rapproche de l'intuition que nous avons des systèmes dynamiques. Nous présenterons donc chacune de ces classes avec quelques exemples, avant de discuter des tentatives de classifications effectives actuellement en cours.

La classification de Wolfram est basée sur l'observation heuristique de l'évolution de l'automate :

- Classe 1 : Evolution vers un état homogène.
- Classe 2 : Evolution vers un ensemble de figures stables ou periodiques dans l'espace et dans le temps.
- Classe 3 : Evolution vers un comportement "apériodique" ou "chaotique".
- Classe 4 : Permet le support de structures complexes, localisées et persistantes.

Première classe : l' Attracteur Unique

La règle 254, présentée comme exemple plus haut, est typiquement de classe 1. C'est le cas d'un attracteur unique : quelles que soient les conditions initiales, l'automate évolue univoquement vers un état homogène. Dans ce travail nous utiliserons la règle 160 comme exemple de cette classe, dont le comportement est illustré sur la figure 4.3



Fig. 4.3 – Représentation graphique de la règle 160 et comportement typique sur des C.I. aléatoires.

Deuxième Classe : Comportement Périodique

On observe pour les automates de cette classe la répétition de motifs périodiques simples. Leur cas sera discuté plus amplement dans la partie sur la dépendance aux conditions initiales. L'exemple typique considéré sera la règle 108 (cf. Fig. 4.4).

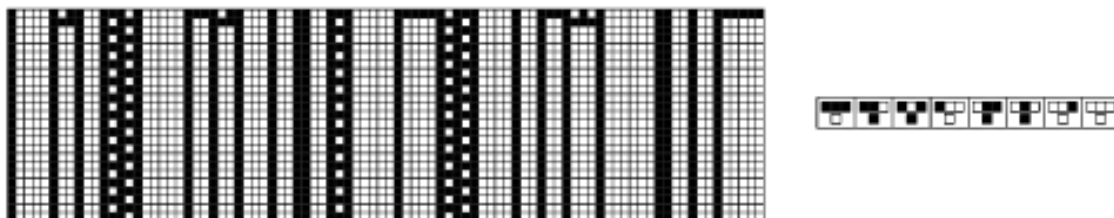


Fig. 4.4 – Comportement typique de la règle 108 sur des C.I. aléatoires et représentation graphique de la règle.

Troisième Classe : l'Apparition du Chaos

C'est à partir d'ici que les choses deviennent intéressantes. Cette classe concerne les automates qui présentent un comportement d'apparence chaotique. Lorsqu'on nous parlons ici de *comportement chaotique* nous signifions un comportement ne présentant pas de structures dans l'espace ni dans le temps. Nous utiliserons aussi le terme de chaos topologique.

Il existe trois façons pour un système dynamique d'exhiber un comportement chaotique.

- Soit la règle d'évolution est aléatoire : peut importe l'état antérieur du système, le nouvel état du système est choisi de manière tout à fait aléatoire. La suite obtenue est alors vraiment aléatoire, il n'existe aucune structure sous-jacente. Par rapport aux systèmes dynamiques continus, c'est l'équivalent d'un bruit stochastique dû à l'environnement.
- Soit la règle est déterministe et simple et le comportement chaotique n'apparaît que pour des conditions initiales aléatoires. Le comportement chaotique n'est alors dû qu'à la retranscription des conditions initiales mise en avant par la règle. On peut comparer ceux-ci aux systèmes amplificateurs, du type de la transformation du boulanger illustrée dans la section 5.1, si l'on considère les décimales, inconnues, situées en deçà de la précision accessible comme source de ce caractère aléatoire. Pour ceux-ci, en effet, le caractère chaotique est dû à la sensibilité aux conditions initiales et à l'impossibilité de définir celles-ci avec une précision infinie.
- Enfin il existe des mécanismes qui, isolés de toute influence extérieure et lancés sur des conditions initiales simples et définies vont présenter un comportement chaotique². Cette fois la génération du chaos est intrinsèque à la règle utilisée et non pas une conséquence de l'environnement ou des conditions initiales.

²Pour rappel, il s'agit bien d'apparence chaotique, les règles étant déterministes, plusieurs expériences numériques avec les mêmes règles sur les mêmes C.I. donneront évidemment le même résultat.

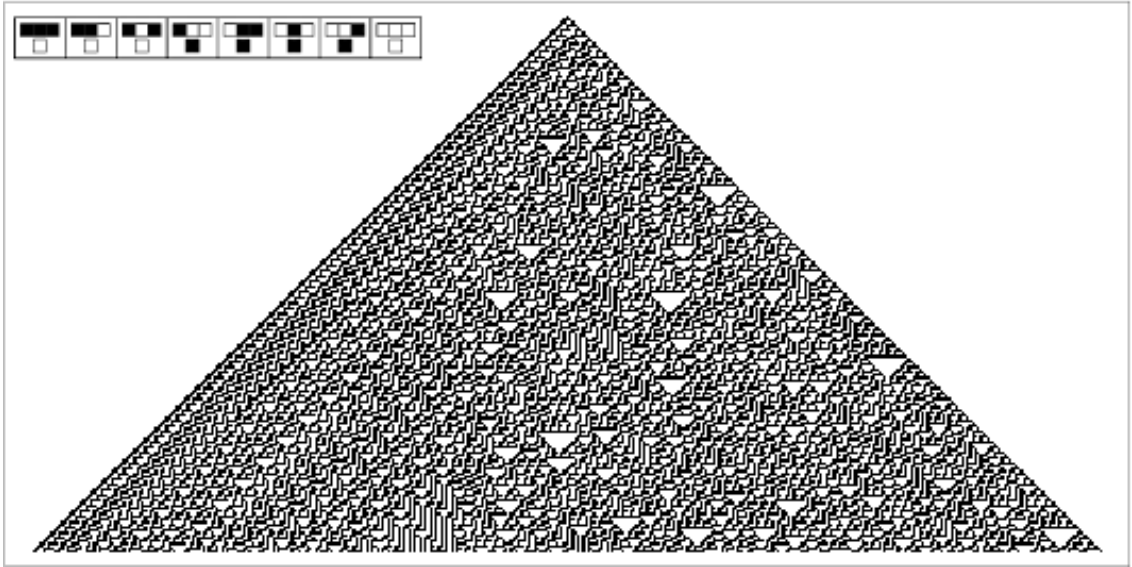


Fig. 4.5 – Comportement typique de la règle 30 sur des condition initiales élémentaires. La partie droite de l'évolution exhibe un comportement d'apparence chaotique.

Les mêmes distinctions existent au sein des A.C.. La première catégorie est, bien entendu, exclue par le déterminisme des règles utilisées. Par contre, il importe de différencier les automates dont le comportement chaotique est visible même lorsqu'ils sont lancés sur des conditions initiales élémentaires³, comme la règle 30(cf. Fig. 4.5) ou la règle 45, de ceux qui ne font qu'entretenir le caractère aléatoire des C.I. implémentées.

Tel est le cas, en particulier, des règles qui ont la propriété d'être additives, soit $F(u \oplus v) = F(u) \oplus F(v)$ pour deux configurations u et v , où la définition de l'opération \oplus varie d'un cas à l'autre. Pour les règles 90 et 150, par exemple, on a $u \oplus v \equiv \text{Modulo}[u + v, 2] = \text{XOR}[u, v]$ (cf Fig. 4.6). Il est évident que de telles règles mènent à des comportements chaotiques si elles sont lancées sur des conditions initiales aléatoires, puisqu'à chaque pas le caractère aléatoire des conditions initiales est propagé dans le système. Mais il ne faut pas voir là un comportement intrinsèquement chaotique de ces règles.

Quatrième Classe : l'Apparition de Structures Complexes.

"Study of cellular automata has given rise to the "Edge of Chaos" hypothesis (EOC). In its basic form this is the hypothesis that in the space of dynamical systems of a given type, there will generically exist regions in which systems with simple behavior are likely to be found, and other regions in which systems with chaotic behavior are to be found. Near the boundary of these regions more interesting behavior, neither simple nor chaotic, may be expected."

C. Langton, 1995 [13]

Cette classe est évidemment la plus intrigante. Elle est définie par l'apparition de structures localisées persistentes (cf exemple Fig. 4.7) qui se déplacent selon différentes vitesses, en restant

³Par "conditions initiales élémentaires" nous entendons : une cellule noire dans une mer de cellules blanches.

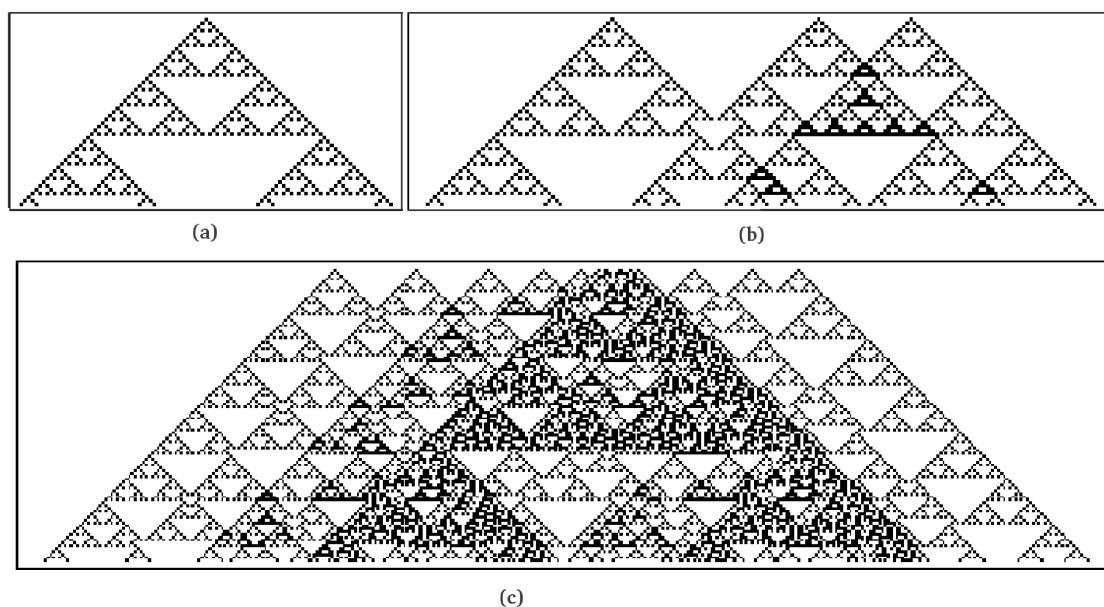


Fig. 4.6 – Illustration du comportement chaotique des règles additives, lorsque elles sont initialisées sur des configurations aléatoires.

identiques à elles-mêmes avec, éventuellement, une certaine périodicité.

L'apparition spontanée de ces structures mouvantes est d'une portée théorique fondamentale. Nous les appellerons désormais des "glisseurs" en référence aux nombreuses études réalisées sur le célèbre automate 2D totalistique "The Game of Life" de J. Conway [17].

Premièrement, la définition d'interactions entre objets microscopiques (cellules) mène à l'apparition d'objets macroscopiques (glisseurs), eux-mêmes constitués de ces objets microscopiques et possédant des propriétés nouvelles. On ne peut être qu'interpellé par le fait que ce phénomène d'émergence de propriétés nouvelles rappelle les explications actuelles concernant l'émergence de la vie : l'idée qu'une entité présente des capacités supérieures et semblant indépendantes, ou tout au moins fortement découplés, de celles de ses constituants. Evidemment l'analogie est ténue, et en particulier les cellules sont ici fixes dans l'espace. Mais ce qui importe, d'un point de vue fondamental, c'est l'apparition d'entités macroscopiques et ce découplage apparent des comportements aux niveaux microscopique et macroscopique. Ces glisseurs interagissent effectivement entre eux de manière complexe, parfois s'annihilant totalement, parfois donnant naissance à de nouveaux glisseurs, parfois continuant leur chemin comme si de rien n'était. Un inventaire complet des glisseurs pouvant apparaître dans la règle 110 ainsi que des différentes collisions a été effectué par l'équipe de S. Wolfram [12], mais nous ne nous attarderons pas sur le sujet.

Deuxièmement, et d'une manière plus formelle, l'existence de ces glisseurs rend ces automates capables d'effectuer des calculs. On pourrait imaginer, effectivement, de coder d'une certaine façon un problème dans les conditions initiales et d'en lire la solution après un certain nombre de pas. Exactement de la même façon qu'on code les données initiales d'un problème et la méthode de résolution dans les transistors d'un PC, et qu'on peut retrouver la solution à ce problème en allant

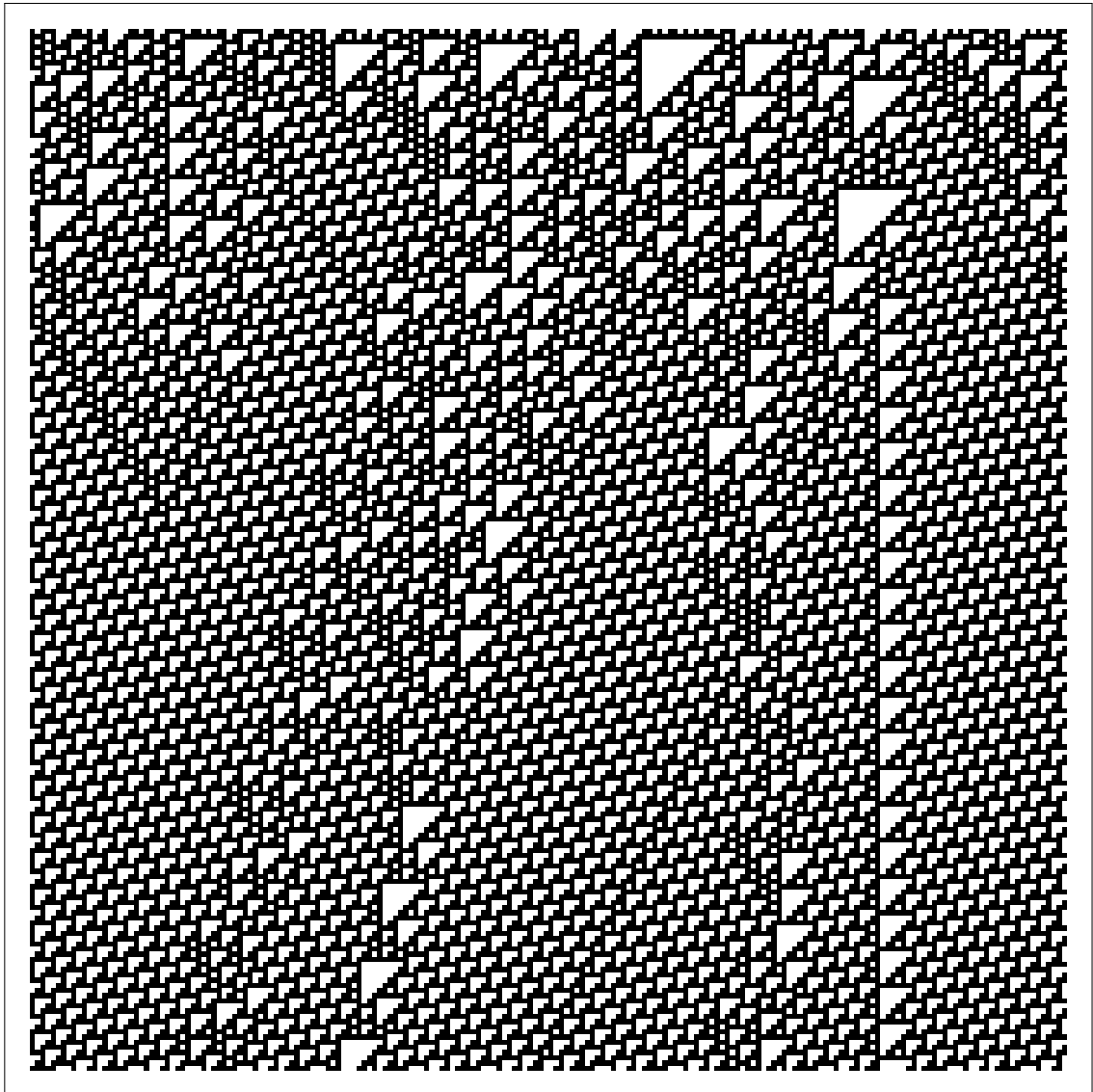


Fig. 4.7 – Comportement typique de la règle 110 sur des condition initiales aléatoires.

chercher dans les bons transistors après les avoir laissé évoluer selon la dynamique propre à ce PC. La notion de capacité de calcul sera souvent utilisée dans ce travail. Elle se rapporte à l'idée qu'un système donné est capable d'effectuer certains calculs, dans les limites de ses capacités⁴. Un PC est un système physique d'une grande capacité de calcul. En fait, il est carrément universel. En deux mots, l'universalité est la capacité pour un système donné, de pouvoir effectuer tous les calculs imaginables. Elle est basée sur la capacité de ce système à émuler n'importe quel autre système. L'universalité est donc en quelque sorte une borne supérieure à cette capacité de calcul.

On a dans l'idée qu'un système qui possède une grande capacité de calcul, doit présenter quelques complications dans sa construction physique (cf. l'exemple du PC). En fait cette idée est fautive et il existe des systèmes très simples qui présentent une grande capacité de calcul. Les complications de construction ne sont là que pour rendre plus simple l'utilisation de ce système par l'homme. En particulier, un automate de la règle 110 est à ce jour le système le plus simple dont l'universalité ait été prouvée⁵. Nous reparlerons des notions de capacité de calcul dans la section 6. Les seuls automates élémentaires soutenant des structures complexes sont ceux de règles 110 et 54, ainsi, cela va de soi, que leur équivalents symétriques.

Plutôt qu'une quatrième classe à part, de nombreux auteurs considèrent cette classe comme la frontière entre la deuxième et la troisième⁶. Ne fut-ce que visuellement, on peut comprendre qu'il s'agit là de l'évolution, apparemment non-prévisible (classe 3), de structures sur un fond continu (classe 2).

En supplément aux exemples donnés ci-dessus, l'appendice B reprend toutes les règles utilisées dans ce travail. Nous y détaillons leur comportement sur divers types de conditions initiales.

Répartitions des classes

Les répartitions en classes de quelques types simples d'automates totalistiques sont données sur la figure 4.8, issue de [12]. On voit que plus le nombre d'éléments intervenant dans le type de règle est grand, plus la classe 3 est représentée. Comme nous allons le voir dans la section 5.1, le comportement en classe 3 est imputable à une grande portée d'interaction, aussi suffit-il qu'un seul des éléments de $f(V)$ permette ce type d'interaction, pour que le comportement en classe 3 prenne le dessus.

4.2.2 Tentatives de classification effective

La recherche d'une procédure effective, permettant de classifier automatiquement une règle donnée, constitue, encore aujourd'hui, un des sujets principaux des études théoriques sur les A.C.. Pourquoi une classification ?

Premièrement, les automates offrent l'avantage de pouvoir englober d'un même regard, d'une part, les interactions locales, et d'autre part, les structures et mécanismes émergents. Ce regard simultané sur le microscopique et le macroscopique permet de sonder plus profondément

⁴Ne demandez pas à un robot manipulant un boulier de vous factoriser un nombre premier.

⁵La preuve est effectuée en établissant la capacité de la règle 110 à émuler un système préalablement établi universel.

⁶Des études de la question ont été menées [12] par Christopher Langton[13], Wentian Li et Norman Packard[29] en 1986 pour les CA ordinaire, puis par Hyman Hartman en 1985 pour les CA probabilistes et enfin par Hugues Chaté et Paul Manneville en 1990 pour les CA continus.

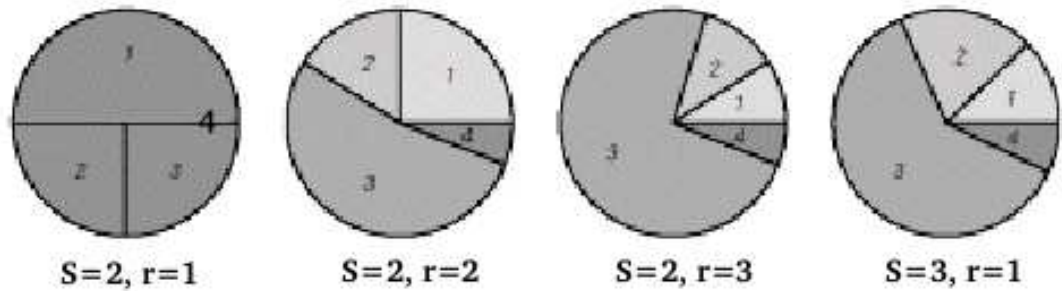


Fig. 4.8 – Fréquences de classe pour différent type d'automates. Issu de [12].

les mécanismes d'auto-organisation, et la découverte d'un critère précis menant à l'émergence de la complexité serait assurément une découverte des plus intéressantes.

Deuxièmement, les automates fournissent une voie intéressante et simple de modéliser des systèmes physiques grâce à cette propriété d'évolution parallèle. Mais le plus souvent, il est impossible de prédire le comportement qu'exhibera un automate. Ce fait est dû à l'irréductibilité analytique de la plupart des A.C.. Il n'existe pas, sauf pour les classes 1 et 2 et dans quelques autres cas rares, de raccourci analytique permettant de décrire le comportement d'un A.C. sans en calculer tout les pas. Etant donné un comportement attendu, on pourrait essayer toutes les règles jusqu'à tomber sur celle qui convient, mais c'est oublier le nombre astronomique de règles possibles. Pour des automates 1D à voisinage adjacent le nombre de règle croît comme S^{2r+1} , pour un tel automate à 3 états et de portée d'interaction de 2, il existe $9 \cdot 10^{115}$ règles différentes. L'intérêt d'une classification permettant de trier préalablement les règles dignes d'intérêt devient dès lors évident.

Enfin, il est important de définir un critère précis de sélection, pour ne pas entâcher nos observations de l'arbitraire inhérent au regard humain. Une règle pourrait montrer des propriétés très intéressantes de transport de l'information tout en ayant une apparence chaotique à nos yeux.

Un des grands obstacles à la mise au point d'une classification effective basée sur le comportement des CA (classification phénoménologique) est due à cette irréductibilité analytique. En effet, pour la plupart des problèmes posés, cette irréductibilité mène très vite à l'indécidabilité. Expliquons-nous par un exemple : imaginons que l'on définisse l'appartenance à la classe 1 par le fait de la convergence vers un état homogène. Certes certains A.C. convergent très vite, mais le problème se situe toujours à la frontière entre deux classes où la situation est plus ambiguë. Certains systèmes prendront un certain temps à converger, et étant donné l'irréductibilité, la seule façon d'apprécier leur convergence est de calculer toutes les étapes. Or, après un nombre fini de pas, fait inévitable dans l'expérimentation, on ne peut décider arbitrairement si l'automate n'a pas encore convergé parce qu'il ne convergera jamais ou parce que l'on a pas effectué assez de pas. Ceci est un exemple qui illustre le problème. Il existe, bien sûr, de nombreux cas pour lesquels la classification est directe, mais les cas indécidables subsistent toujours.

Afin de contourner ce problème, inhérent aux analyses phénoménologiques, certains auteurs ont tenté de mettre au point des classifications basées sur les tables des A.C.. Les tables des A.C. sont les mots de longueur S^{2r+1} dans l'alphabet S qui sont les images de toutes les configurations possibles de V , mises dans l'ordre lexicographique. C'est exactement la même chose que dans la

représentation de Wolfram, avant de passer en décimal.

Le but recherché ici est de paramétrer l'espace des règles des A.C. afin d'identifier des zones de cet espace ou les règles présentent tel ou tel comportement. Plus exactement, ce sont surtout les frontières entre ces zones que l'on cherche à cibler avec précision, afin de justifier ou d'infirmer l'hypothèse EOC ("Edges of Chaos") citée plus haut. Cette hypothèse, s'inspirant de résultats similaires obtenus dans le cadre de l'étude des systèmes dynamiques continus, consiste à supposer l'existence d'une frontière dans l'espace des paramètres entre des zones menant à un comportement chaotique et d'autres menant à un comportement régulier, et suppose que, sur cette frontière, des comportements plus "complexes" sont observables. L'hypothèse EOC reste encore largement controversée. Notamment parce que l'existence d'une telle frontière précise n'a encore jamais été analytiquement prouvée.

Nous citons ici trois exemples de ce type de recherche.

Paramètre λ de Langton

Un état stable s , pour un automate donné, est un état tel que $f(s, \dots, s) = s$. A titre d'exemple, l'état blanc est stable pour les règles 22, 30, 110, ... , l'état noir l'est pour les règles 150, 160, 254... On note $\mathcal{P}(s)$ la proportion de configurations de V qui mènent à s , soit le nombre d'états s dans la table de l'automate considéré divisé par la taille de cette table. Le paramètre λ_s de Chris Langton[13] est alors défini par $\lambda_s = 1 - \mathcal{P}(s)$.

Ce paramètre peut servir pour générer des règles montrant un comportement donné. On commence par fixer λ_s , et donc $\mathcal{P}(s)$. Puis, on parcourt la table en assignant aux différentes configurations, d'abord, autant d'états s qu'indiqué par la proportion $\mathcal{P}(s)$, puis, les autres états, de manière équitable. Si $\lambda_s = 0$, toutes les configurations mèneront à l'état stable, et on aura donc une règle de classe 1. Si $\lambda_s = 1$, aucune des configurations ne mènera vers l'état stable et on se retrouvera avec une règle de classe 3. Entre ces deux valeurs extrêmes on observe des comportements différents, allant du type classe 1 au type classe 4.

Ce paramètre a été assez largement contesté pour différentes raisons (voir notamment [24], [18]). Premièrement, il dépend fort de l'état s considéré, et en particulier, n'est applicable que sur les automates admettant un état stable. Deuxièmement, ce paramètre est assez grossier, et, pour un λ_s fixé, on peut trouver plusieurs règles menant à des comportements complètement différents.

Une grosse partie de la discussion sur ce sujet est basée sur une mauvaise définition de concepts tels que capacité de calcul, complexité des règles, etc... Ceux-ci seront rediscutés dans le chapitre 6.

Etude du champ moyen

Une approche similaire, quoique plus complète et permettant plus de précision dans le paramétrage de l'espace de phase, est rendue possible grâce à l'étude du champ moyen [13].

La théorie du champ moyen est une étude basée sur les mesures de probabilités d'occurrence de blocs de cellules. Numériquement, ces mesures de probabilités sont effectuées en comptant les occurrences sur des suites suffisamment longues.

Le concept de départ est le suivant : pour l'automate $A = (d, s, V, f)$, la probabilité d'avoir un

bloc b au temps $t + 1$ est la somme des probabilités de blocs B au temps t , tel que $f(B) = b$.

$$p_b^{t+1} = \sum_{B|f(B)=b} p_B^t \quad (4.1)$$

Considérons le cas d'un automate linéaire de voisinage adjacent de portée r à deux états, soit $d = 1$, $\mathcal{S} = \{0, 1\}$, $V = \{-r, \dots, r\}$. Si b est le bloc de longueur unitaire $\{1\}$, la somme s'effectue sur les blocs de longueurs $2r + 1$ qui, sous l'action de f , ont pour résultat 1. Notons $\#0(B)$ (resp. $\#1(B)$) le nombre de 0 (resp. 1) contenus dans le bloc B . Si nous acceptons pour le moment que la probabilité pour une cellule d'être dans un état donné ne dépend pas de l'état des autres cellules, alors la probabilité d'un bloc est le produit des probabilités des cellules constituant ce bloc, soit $p_B = p_1^{\#1(B)}(1 - p_1)^{\#0(B)}$. En utilisant l'équation 4.1 on obtient :

$$p_1^{t+1} = \sum_{B|f(B)=1} (p_1^t)^{\#1(B)}(1 - p_1^t)^{\#0(B)} \quad (4.2)$$

Cette équation du champ moyen définit une dynamique dans l'espace réel, pour p_1 , qui simule l'action d'un automate sur des mesures de probabilités décorréelées. Tout se passe comme si l'automate subissait l'action d'un bruit supprimant les corrélations entre cellules à chaque pas.

Voyons maintenant comment il est possible de se servir de cette équation pour paramétrer l'espace des règles. Tout d'abord, il faut remarquer que seuls les blocs menant à 1 participent à la somme ci-dessus. Ensuite, que si deux blocs B et B' sont tels que $\#1(B) = \#1(B')$, ils participent à la somme de manière exactement équivalente. En notant a_i le nombre de blocs contribuant à la somme et contenant i cellules dans l'état 1, l'équation 4.2 peut être réécrite sous la forme

$$p_1^{t+1} = \sum_{i=0}^{2r+1} a_i (p_1^t)^i (1 - p_1^t)^{2r+1-i} = g(\{(a_0, \dots, a_{2r+1}\}, p_1^t). \quad (4.3)$$

Dans l'approximation du champ moyen, tout automate présentant le même vecteur \vec{a} présentera le même comportement, et on peut donc caractériser une règle par son vecteur \vec{a} , soit la situer dans un espace à $2r + 1$ dimensions.

Evidemment, l'hypothèse concernant la décorrélation entre cellules voisines est très lourde. Mais il existe [25] des généralisations à des ordres supérieurs où l'on considère les probabilités de blocs de taille $n > 1$, en définissant ces dernières à partir des équations de Markov qui assurent que le principe d'inférence d'entropie maximum soit respecté lorsqu'on passe des probabilités de blocs de taille 1 à des blocs de taille supérieure. Chaque ordre supérieur de cette généralisation permet de prendre en compte des corrélations à distance plus grande, et agit comme un raffinement du paramétrage de l'espace des règles.

Cette étude donne des résultats assez convaincants, dans le sens où une légère variation dans l'espace des paramètres donne lieu à des légers changements de comportement.

Paramètre κ , ou complexité de Kolomogoroff des tables des A.C.

Une autre approche, plus mathématique, est proposée dans [18]. Les auteurs y définissent l'ensemble des paramètres de classifications en stipulant certaines conditions que ceux-ci doivent respecter : quantifier la régularité dans les tables, être effectivement calculables et respecter certaines conditions de normalisation. Ils prouvent, ensuite, l'existence d'un paramètre "optimal". Ils

définissent pour une table x , $\kappa = \frac{K(x|l(x))+1}{l(x)}$ où $l(x)$ est la longueur de la table et $K(x|y)$ est la complexité algorithmique de x sachant y , soit pour une machine de Turing ϕ , $K(x|y) = \min\{l(p) | \phi < p, y \geq x\}$. La définition présentée à la section 3.1.3 correspond au cas particulier où y est le mot vide. Ils montrent donc que ce paramètre κ respecte les critères de paramètre optimal... à ceci près qu'il est incalculable ! Ils proposent donc d'avoir recours à la mesure de la compressibilité des tables comme approximation. N'ayant trouvé dans la littérature aucun résultat montrant l'intérêt d'une telle approche nous n'approfondirons pas le sujet plus avant, et en particulier, nous ne citerons pas ici la définition du caractère "optimal" sus-mentionné.

Chapitre 5

Investigations personnelles

5.1 Dépendances aux conditions initiales

Nous commençons cette section en présentant l'intérêt de l'étude des dépendances aux C.I. par un exemple de dynamique continue. Nous y serons amenés à définir la notion d'exposants de Lyapounov pour les systèmes continus. Par la suite nous établirons la définition d'un coefficient Λ similaire aux exposants de Lyapounov, mais applicables aux systèmes discrets et plus particulièrement aux A.C.. Enfin nous présenterons les résultats obtenus en discutant de l'interprétation que l'on peut en faire par rapport au système de classes de Wolfram.

5.1.1 Exposants de Liapounov

Il est connu, depuis les travaux de Henri Poincaré sur le problème à trois corps, que certains systèmes dynamiques montrent une forte dépendance aux conditions initiales. Cette sensibilité a pour résultat que l'issue d'une expérience basée sur ce genre de dynamique est, en pratique, imprédictible. C'est le cas, par exemple, des lancers de dés ou des jeux de roulette. La dynamique est telle qu'une variation infinitésimale dans les conditions initiales va changer du tout au tout l'issue de l'expérience.

L'exemple suivant est appelé "la transformation du boulanger". On considère une pâte uniforme, piquée çà et là de raisins. A chaque étape on étire la pâte jusqu'à deux fois sa longueur initiale, on la coupe en deux et on replace les deux parties l'une au dessus de l'autre avant de les écraser. Ainsi, au final, la pâte est dans le même état qu'au début, et seules ont changé les positions des raisins dans la pâte. La figure 5.1 montre quelques premiers pas d'évolution de ce système.

On voit que, quelle que soit la distance séparant deux raisins, cette distance sera doublée à l'étape suivante (si les deux raisins sont initialement dans la même moitié). Faisons deux fois la même expérience, avec les raisins placés tous aux mêmes endroits sauf un, placé 1 micron plus à gauche dans la seconde expérience. Après un pas les deux configurations différeront de 2 microns, après deux pas de 4, et ainsi de suite à chaque pas, si bien qu'après t pas la différence aura grandi d'un facteur 2^t . Cette croissance exponentielle des perturbations est typique des systèmes chaotiques. Pour quantifier la dépendance aux conditions initiales, Liapounov a introduit l'exposant λ tel que, si $\delta(0)$ est la distance séparant deux conditions initiales infiniment proches, la distances

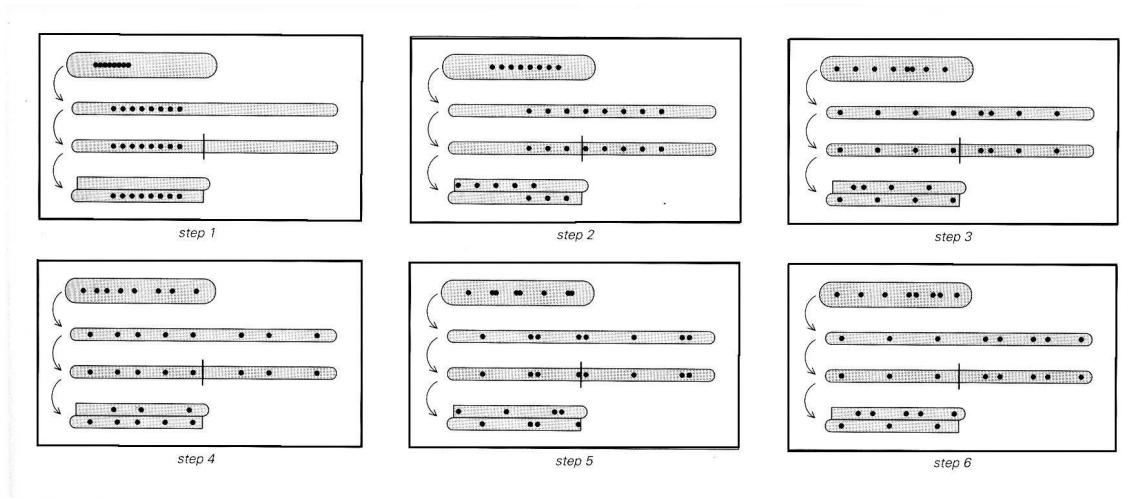


Fig. 5.1 – Quelques pas de la transformation du boulanger. Issue de [12]

séparant les deux trajectoires en un temps ultérieur t est $\delta(t) \sim \delta(0) \exp(\lambda t)$. Pour l'exemple cité on a que λ est égal à $\ln 2$.

De manière générale les exposants de Liapounov sont définis comme suit, pour un système évoluant dans un espace des phases de dimension d : considérons une boule de dimension d et de rayon ϵ , centrée sur $X(0)$. Suivant la loi d'évolution ¹, la boule sera déformée en un ellipsoïde d -dimensionnel de demi-axes $l_1(t) \geq l_2(t) \geq \dots \geq l_d(t)$. Les exposants sont alors définis par

$$\lambda_i = \lim_{t \rightarrow \infty} \lim_{\epsilon \rightarrow 0} \ln \frac{l_i(t)}{\epsilon} \quad (5.1)$$

Par convention on numérote les coefficients λ_i du plus grand au plus petit. Si on a $\lambda_i \leq 0, \forall i \in \{0, \dots, d\}$, les deux trajectoires resteront proches et le système sera dit Lyapounov-stable. Par contre si $\exists i$ tq $\lambda_i > 0$, les trajectoires divergeront et le système sera dit Lyapounov-instable.

5.1.2 Interprétation des exposants de Lyapounov dans le cadre de la théorie de l'information

Imaginons maintenant la même expérience effectuée avec un seul raisin. A chaque étape -on en effectue N - on note la position X du raisin, en notant 0 si $X(t) \leq L/2$, et 1 si $X(t) \geq L/2$, où L est la longueur de la pâte (au repos). On a donc maintenant une suite s de 0 et de 1, de longueur N , dont on aimerait bien calculer l'entropie. Commençons par celle de Shannon. En principe, on la calcule en analysant une suite infiniment longue. Concrètement, on peut l'évaluer numériquement, en estimant les probabilités $P(W_m)$ pour des séquences W_m de longueur m , à partir de la fréquence d'apparition de W_m parmi les N bits de la suite s . On calcule

$$H = \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{W_m} P(W_m) \log_2 \frac{1}{P(W_m)} \text{ bit/bit}, \quad (5.2)$$

qui est une constante.

¹ $\frac{dX(t)}{dt} = f(X(t))$ (resp. $X(t) = g(X(t-1))$) pour un système à temps continu (resp. discret)

Qu'en est-il de $K(s)$? Comprenons bien, il s'agit de la longueur du plus court programme capable de générer s . Ce programme doit contenir d'abord un terme de longueur constante contenant le mécanisme d'évolution, un terme en $\log_2 N$ pour écrire, en binaire, le nombre N d'éléments de la suite à générer, et enfin un terme précisant les conditions initiales. C'est ici qu'est le hic. Car pour pouvoir déterminer le N_{eme} bits, on doit avoir une précision $\delta(T) \leq \frac{1}{2}$, donc, une connaissance des conditions initiales $\delta(0)$ telle que $\delta(0) \exp(\lambda T)$ reste plus petit que $\frac{1}{2}$, soit $\delta(0) \leq \frac{1}{2} \exp(-\lambda T)$. Ceci nécessite la donnée d'un nombre de bits égal à ²

$$\log_2\left(\frac{L}{\delta(0)}\right) = \log_2(2 L \exp(\lambda T)) = \log_2(2L) + \log_2(\exp(\lambda T)) = C^{ste} + \frac{1}{\ln 2} \lambda T. \quad (5.3)$$

On voit donc que l'entropie algorithmique de systèmes chaotiques va augmenter avec le nombre de pas, jusqu'à rattrapper l'entropie de Shannon qui, elle, reste constante. On pourra dire alors que le phénomène a défini un processus stochastique puisque la suite des résultats est incompressible au-delà de la limite de Shannon. Rappelons que Shannon démontre qu'une suite de longueur N peut se compresser en une suite de longueur $\sim T \frac{H}{\ln 2}$. Si on supprime les termes constants et le terme en $\log_2 T$ de l'expression de la longueur du programme de compression optimal discuté plus haut, on a

$$N_T \sim T \frac{H}{\ln 2} \sim \frac{1}{\ln 2} \lambda T \Rightarrow \lambda \sim H \quad (5.4)$$

Cette coïncidence n'est pas due au hasard : on constate que, dans une application unidimensionnelle, λ prend une signification entropique, dans la mesure où il exprime le nombre de bits par unité de temps nécessaire à la détermination de la trajectoire[1].

5.1.3 Mesure Λ de la dépendance aux C.I. des A.C.

Il est évident que la définition, présentée plus haut, des exposants de Lyapounov ne peut être utilisée telle quelle pour étudier les A.C.. Cette difficulté vient du caractère discret de l'espace utilisé ici. En effet, bien qu'il soit possible de considérer l'espace de phase des A.C. (cf. Section 5.3), soit l'ensemble C de toutes les configurations, la notion de distance entre deux points de cet espace est loin d'être immédiate. Wolfram[16] et Grassberger[19] définissent pour ce faire $d(c_1, c_2)$, la distance entre deux configurations c_1 et c_2 , comme étant le nombre de cellules dont l'état diffère entre c_1 et c_2 . On pourrait utiliser cette définition et, partant de là, calculer l'écart $\delta(t) = d(c(c_1, t), c(c_2, t))$, où $c(c_1, t)$ représente la configuration obtenue après t pas à partir de la configuration c_1 . Mais, vu que nos A.C. ne comportent que deux états, une cellule a vite fait de changer deux fois d'état et de se retrouver comme si elle était restée inchangée. Nous avons donc plutôt considéré ici $\delta(t)$ comme étant la largeur de la perturbation, soit $1 +$ la distance entre le premier et le dernier bits perturbés. Pour différents automates, nous avons mesuré l'écart $\delta(t)$ entre deux configurations, obtenues en partant de deux configurations initiales différant d'un seul bit, soit $\delta(0) = 1$. Nous considérons que les perturbations croissent comme, $\delta(t) = \delta(0) + 2\Lambda t$, où Λ peut être vu comme la vitesse de propagation du bord de la perturbation. La mesure effectuée est donc $\Lambda = \frac{(\delta(t) - \delta(0))}{2t}$.

5.1.4 Résultats et commentaires

Les études ont été effectuées avec $N = 8.10^4$ et pour t allant jusque 4.10^4 . Ceci afin d'éviter que les deux fronts ne se rejoignent sous l'effet des conditions aux bords périodiques. Les valeurs

²on partitionne L en $\frac{L}{\delta(0)}$ parties, puis il faut préciser quelle est la partie occupée par son "numéro".

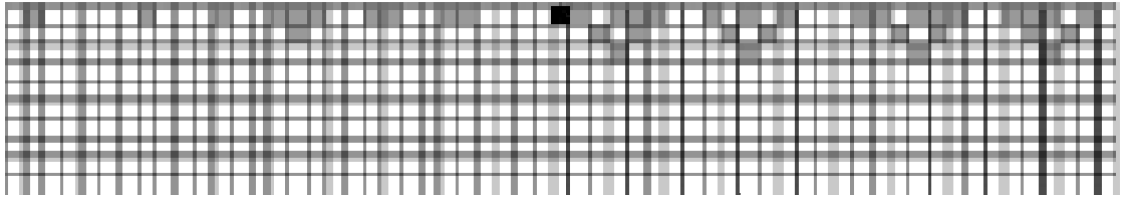


Fig. 5.2 – Comportement typique d’une perturbation introduite dans les conditions initiales de l’automate de règle 160.

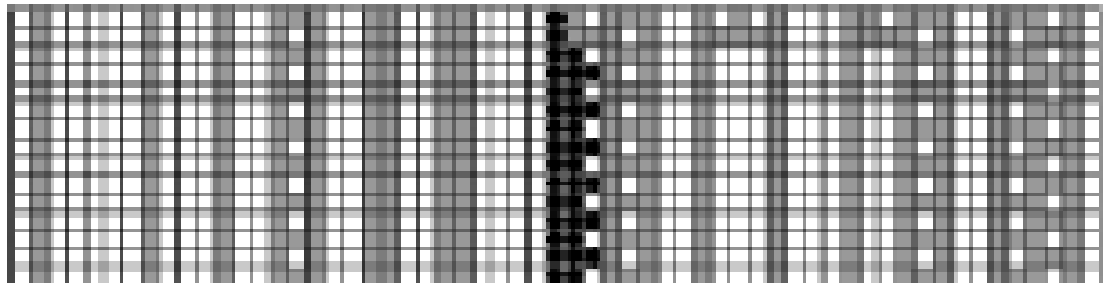


Fig. 5.3 – Comportement typique d’une perturbation introduite dans les conditions initiales de l’automate de règle 108. La perturbation reste localisée.

données pour Λ sont obtenues en moyennant les valeurs calculées pour chaque pas. Les exemples donnés sont les comportements typiques observés sur plusieurs simulations.

Classe 1

Pour les automates de classe 1, on comprend vite que toute perturbation va régresser, pour finir, comme le reste des cellules, dans l’état stable. Nous illustrons ce fait sur la figure 5.2. Cette récession correspond à un Λ négatif, calculable en prenant en compte un $\delta(0) > 1$ et en prenant soin d’arrêter le calcul une fois que la perturbation a totalement disparu. $\Lambda \leq 0$ traduit donc bien la notion de stabilité.

Classe 2

Comme on peut le voir sur la figure 5.3, une perturbation introduite dans un système de classe 2 restera localisée. Cela correspond à vitesse nulle et on mesure effectivement $\Lambda = 0$.

Il est une interprétation intéressante à faire de ce résultat. En effet, le fait que les perturbations restent localisées revient à dire que les interactions entre cellules distantes sont annulées. Tout se passe comme si on avait affaire à des systèmes de taille N très petite mis l’un à côté de l’autre, et n’interagissant pas entre eux. Or, le nombre de configurations que peut prendre un système discret de taille limitée est fini, et puisqu’on considère ici des évolutions déterministes, un système de taille limitée retombera nécessairement dans un comportement périodique. De fait, il retournera forcément sur une configuration déjà visitée, éventuellement après avoir visité tous les cas possibles, et suivra dès lors la même évolution. C’est le cas pour tous les systèmes de taille limitée. Mais, le nombre de configurations, 2^N dans notre cas, variant très vite avec la taille N , les périodicités peuvent vite devenir énormes pour des grands systèmes. Dans le cas de très petits systèmes, comme

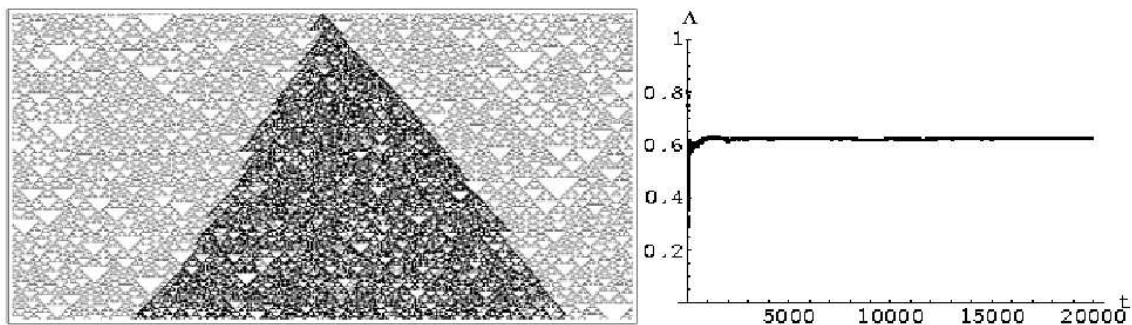


Fig. 5.4 – Comportement typique d’une perturbation introduite dans les conditions initiales de l’automate de règle 30. La perturbation se propage de manière uniforme et Λ converge rapidement.

c’est le cas ici, on observe des périodicités très courtes, d’où le comportement en classe 2.

Classe 3

Nous présentons dans la figure 5.4 le comportement typique d’une perturbation pour des automates de classe 3 intrinsèquement générateurs de chaos topologique (cf. section 4.2.1), ainsi que la convergence du calcul de Λ .

Le même comportement est observé pour les règles 22 et 45, du même type. A savoir, une convergence rapide de Λ vers une valeur positive. Les résultats obtenus sont affichés dans le tableau 5.1. Les valeurs précises $\Lambda = 1$ obtenues pour les règles 90 et 150 sont directement imputables à leur propriété d’additivité explicitée à la section 4.2.1. En effet, puisque ces règles ne font que superposer les structures dues à chaque point singulier dans les conditions initiales, l’aspect de la perturbation, soit la structure spatio-temporelle des bits ayant changé d’état entre les deux cas issus de C.I. différents par un point, aura la même forme que la structure émanant d’un point noir isolé (C.I. élémentaires).

Classe 4

La figure 5.5 montre le comportement d’une perturbation pour la règle 110, ainsi que la convergence de Λ pour les règles 110 et 54. La règle 54 est une règle de classe 4, mais qui ne montre apparemment pas autant de variété dans les glisseurs qu’elle supporte que la règle 110. Malgré cette apparence plus simple de la règle 54 lorsqu’on en observe le comportement typique (voir Appendice B), la vue du mode de propagation d’une perturbation révèle d’importantes structures sous-jacentes qui n’apparaissent pas spontanément à l’œil nu.

Les valeurs calculées pour Λ sont affichées dans le tableau 5.1. Ici l’exposant est encore positif, mais la convergence est beaucoup moins rapide. La convergence est aussi systématiquement plus rapide pour la règle 54 que pour la règle 110. La variabilité de Λ est à imputer au fait que la perturbation se propage par le biais de glisseurs et suit donc une évolution saccadée. Il ressort de ces observations que le temps de convergence semble être une bonne mesure de la complexité.

Dans ses tentatives de paramétrer l’espace des règles, Langton[13] utilisait le temps de convergence des études en champs moyens pour juger de la complexité des règles. D’une manière semblable, nous avons voulu apprécier ici le temps de convergence moyen du coefficient Λ , en examinant le nombre de pas nécessaire pour que les 2000 dernières valeurs calculées présentent une déviation

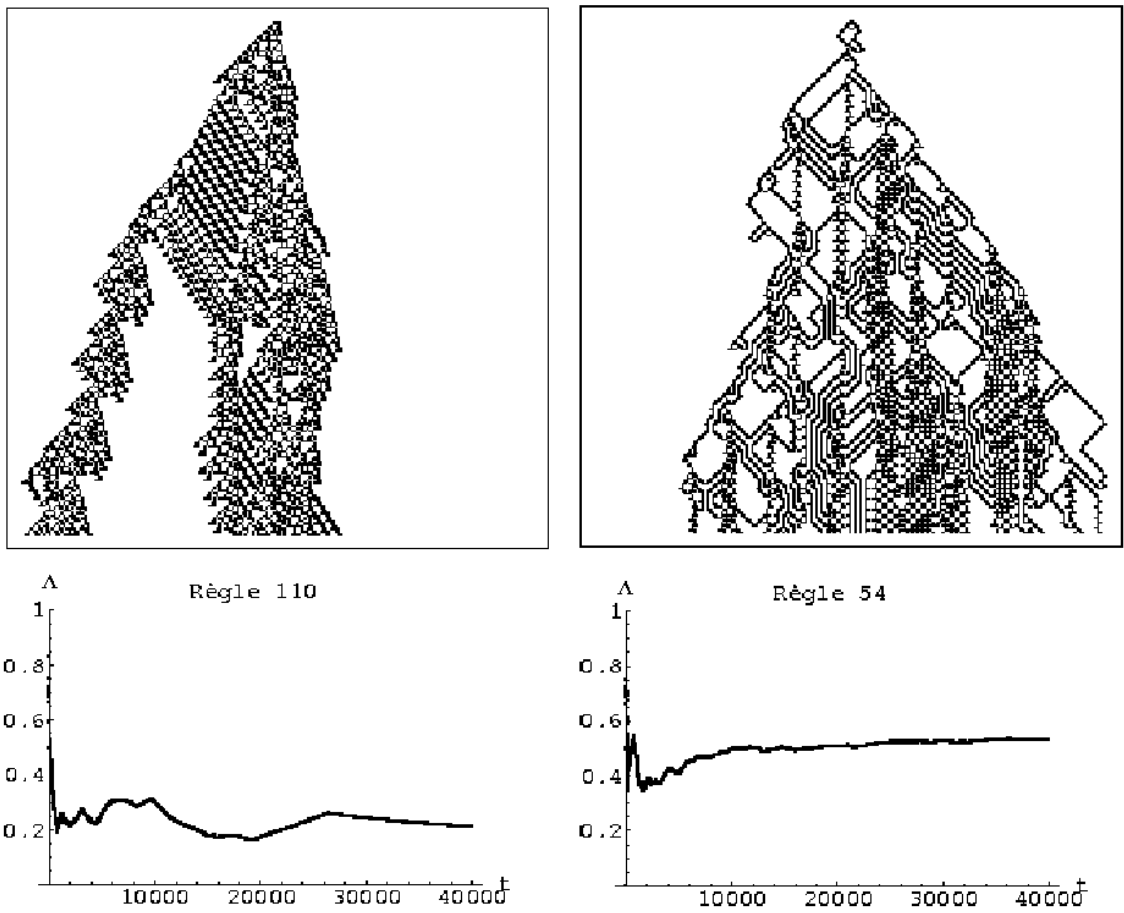


Fig. 5.5 – Comportement typique d’une perturbation pour les règles 110 et 54 et convergence de Λ sont montrés. Les perturbation se propagent par le biais de glisseurs et Λ converge lentement.

standard plus petite que 0,001. Les résultats moyens obtenus sur 10 expériences identiques pour les règles 22,30,45,54 et 110, sont affichés dans la table 5.1.

règle	Valeur moyenne	Ecart-type	Temps de convergence
22	$\lambda = 0.77$	± 0.007	7591 ± 1834
30	$\lambda = 0.619$	± 0.006	7024 ± 795
45	$\lambda = 0.583$	± 0.01	9181 ± 1026
54	$\lambda = 0.5$	± 0.04	21573 ± 4329
110	$\lambda = 0.23$	± 0.04	24819 ± 10206
90	$\lambda = 1$	± 0	/
150	$\lambda = 1$	± 0	/

TAB. 5.1 – Valeurs de Λ calculées pour différentes règles de classe 3 et 4. Les valeurs sont obtenues en moyennant sur $4 \cdot 10^4$ pas temporels. Les temps de convergences sont évalués en exigeant que la déviation standard soit plus petite que 0.001 sur les 2000 mesures précédant le temps t en question.

5.2 Evolution de la T-Entropie

Nous utilisons dans cette section la T-Entropie, définie à la section 3.3, pour apprécier l'évolution de l'entropie de suites sous l'action de règles d'automates élémentaires. Concrètement, voici comment nous procédons. Nous commençons par fixer N à 2000 bits, et choisissons des C.I. particulières, dont nous calculons la T-entropie. Ensuite, nous faisons évoluer cette ligne par le biais d'une règle d'A.C., nous calculons la T-Entropie de la ligne ainsi obtenue, et ainsi de suite sur 10000 pas. Vu la trivialité des classes 1 et 2, nous ne considérons ici que des règles de classe 3 et 4. Les résultats concernant les classes 1 et 2 ne sont que succinctement décrits.

5.2.1 Observations

Afin d'aller plus loin dans l'analyse de l'évolution de $H_{TN}(t)$, et en particulier, d'en sonder la nature des oscillations, nous avons appliqué au signal $H_{TN}(t)$ une méthode de l'analyse fractale, appelée méthode DFA. Ces analyses ayant été faites dans le cadre d'un travail ultérieur, nous annexons celui-ci dans lequel sont explicitées en détail les méthodes et procédures suivies (cf. Annexe C). Nous n'utiliserons, dans le présent travail, que les résultats issus de ces analyses. Brièvement, cette analyse permet de caractériser la nature des corrélations dans le signal en fonction de leur portée. Un exposant α_H y est identifié qui caractérise le type des oscillations dans le signal. Cet exposant permet de préciser s'il s'agit d'oscillations erratiques, de type mouvement brownien, d'oscillations corrélées, anti-corrélées ou de tendance générale. Le choix de placer en annexe les détails de cette méthode a été fait principalement pour des raisons de continuité dans la lecture.

Classe 1

Il est évident, dans le cas des A.C. de classe 1, que H_{TN} chutera directement à 0, lorsque la ligne aura atteint un état homogène, et y restera stable.

Classe 2

Pour les automates périodiques, H_{TN} convergera rapidement vers une valeur fixe, dépendant des condition initiales, et soit y restera fixe, soit oscillera régulièrement, signature d'un attracteur fixe ou cyclique.

Classe 3

Nous considérons successivement plusieurs cas, différant par le type de C.I., puis par le genre de classe 3 auquel nous avons affaire (intrinsèque ou additive). Commençons par des C.I. aléatoires. Nous laissons donc évoluer le système pendant 10^4 pas temporels, en prenant soin de calculer H_{TN} pour chaque pas. Pour les règles 22, 30, 45, 90 et 150, le même comportement est observé : H_{TN} reste localisé autour de la valeur 1, avec des oscillations erratiques. Si on étudie la distribution de $H_{TN}(t)$, on constate que la distribution obtenue est exactement la même que dans le cas où l'on avait pris des suites aléatoires. Des suites corrélées entre elles par des règles intrinsèquement génératrices de chaos topologique ont donc la même distribution entropique que des suites sélectionnées de manière aléatoire. Jusqu'à maintenant le caractère chaotique attribué à ces règles n'était basé que sur des observations visuelles. Voici à présent un résultat concret qui semble bien confirmer nos dires. Une étude DFA sur le signal confirme encore ces résultats, en caractérisant, pour les deux cas "évolution de classe 3" et "tirages aléatoires", les oscillations de $H_{TN}(t)$ par un exposant $\alpha_H \sim 1/2$. Cette

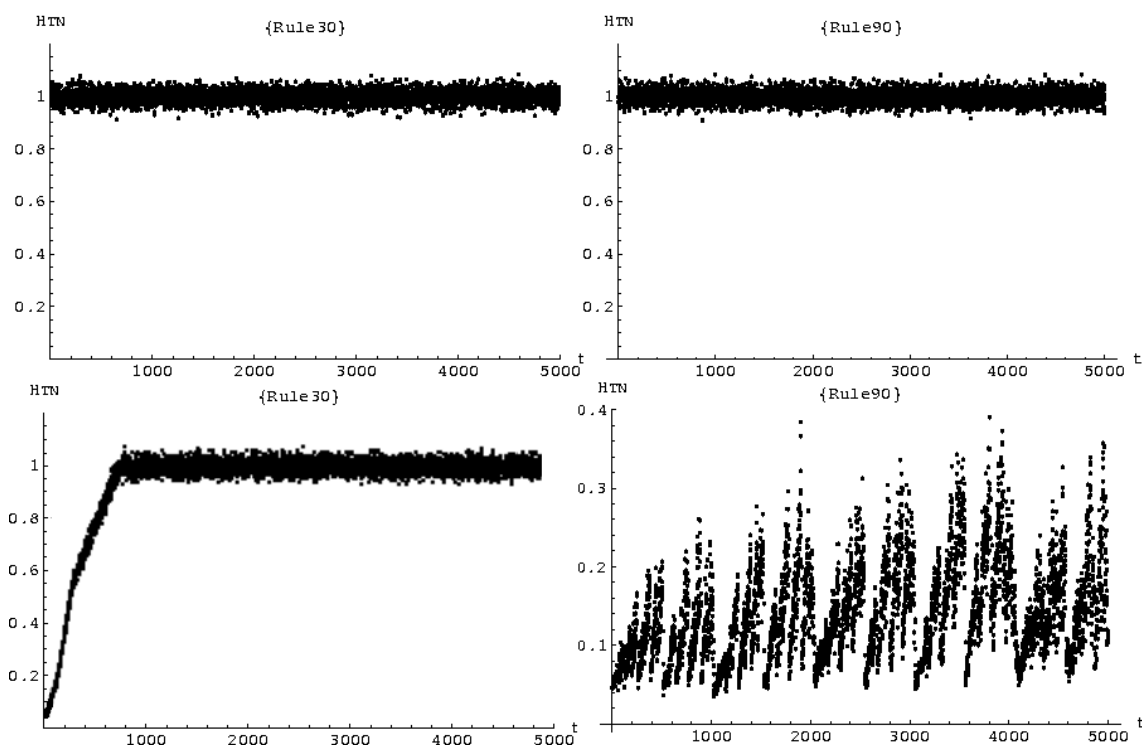


Fig. 5.6 – $H_{TN}(t)$ pour les règles 30 et 90, dans le cas de C.I. aléatoires et biaisées.

valeur correspond à la signature d'un bruit du type mouvement brownien, inhérent à l'erreur sur la mesure de H_{TN} (voir section 3.3.3) et caractéristique de la distribution gaussienne observée.

Considérons à présent des C.I. biaisées, avec $p = 1/500$. Nous avons, maintenant, une suite initialement composée d'environ quatre 1 dans une mer de 0. La figure 5.6 montre l'évolution de H_{TN} avec le temps t , pour les règles 30 et 90. Il apparaît que, cette fois, on peut faire la distinction entre les règles intrinsèquement génératrices de chaos topologique et les règles dont le chaos apparent n'est qu'une conséquence de leur additivité. Les premières évoluent directement vers le régime précisé plus haut, tandis que les deuxièmes gardent une trace du biais de leurs conditions initiales.

Classe 4

La même analyse a été réalisée pour la règle 110 et 54. Pour des C.I. aléatoires, le comportement de $H_{TN}(t)$ est montré sur la figure 5.7. La décroissance observée trouve son origine dans l'organisation de la suite par l'action de la règle 110 (resp. 54). Une analyse DFA du signal montre que les variations autour de la tendance générale peuvent être associées à un bruit du type mouvement brownien partionnaire, avec un exposant moyen $\alpha_H = 0,87$ pour la règle 110 (resp $\alpha_H = 0,77$ pour la règle 54), soit un bruit hautement corrélé. Plus que la décroissance lente, ce sont surtout ces variations qui doivent être interprétées comme la signature de la complexité de la règle.

Les résultats obtenus dans la section précédente concernant les règles additives montrent aussi une grande variabilité. Ceci est cohérent en ce sens que l'on peut interpréter les structures émanant de chaque point noir singulier des C.I. comme un mode de transport de l'information. Ces modes se

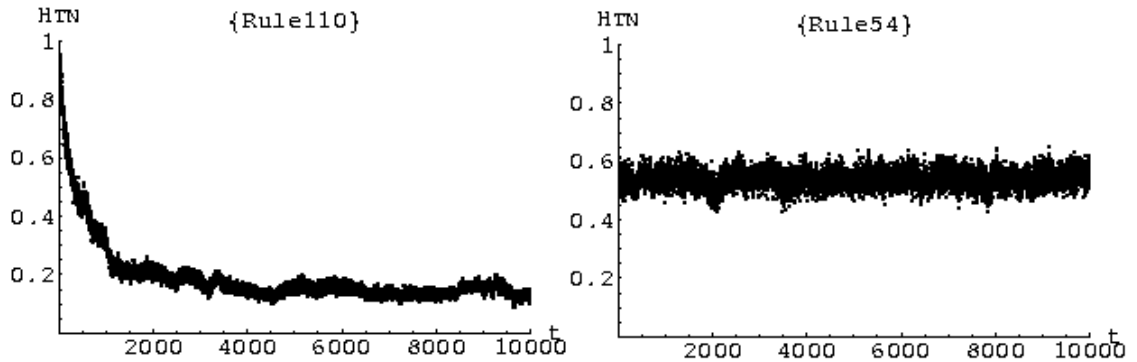


Fig. 5.7 – $H_{TN}(t)$ pour les règles 110 et 54.

distinguent toutefois des glisseurs propres aux règles de classe 4, en ce qu'ils ne sont pas localisés. En particulier, ils ne permettent pas une diminution de l'entropie lorsqu'ils sont lancés sur des C.I. aléatoires.

5.2.2 Conclusions et comparaisons avec des études antérieures

Bien que les études exposées ci-dessus soient loin d'être exhaustives, il semble se dégager de ces résultats que le comportement de H_{TN} obtenu sur des C.I. aléatoires peut servir à caractériser le type de règle auquel on a affaire. En particulier, l'exposant α_H , caractérisant les variations de H_{TN} autour d'une tendance générale, semble être un indicateur de la complexité des règles utilisées. Cette interprétation est motivée par le fait qu'elle est en accord avec des études préliminaires effectuées par A. Wuensche[26]. Wuensche utilisait H_{2r+1} , l'entropie de Shannon à l'ordre égal à la taille du voisinage, ce qui limite la portée des corrélations identifiées. Dans notre cas, comme expliqué plus haut, l'identification de corrélations à une portée donnée dépend de la taille N du système et peut donc être directement améliorée en considérant des systèmes de grande taille. Afin d'apprécier la capacité de calcul d'une règle, il utilisait la déviation standard de l'entropie mesurée autour de la valeur moyenne. Bien que plus laborieuse³ à exécuter de manière automatique, l'étude du coefficient α_H permet d'aller plus loin que la déviation standard, en caractérisant la nature propre des oscillations.

Wuensche utilisa ces résultats pour générer de manière automatique des règles présentant un caractère complexe. Nous montrons sur la figure 5.8, issue de [26], un résultat qu'il obtint en générant aléatoirement 17680 règles, pour des A.C. $d = 1, S = 2, r = 2$, en calculant systématiquement les valeurs moyennes de l'entropie ainsi que la déviation standard autour de cette moyenne sur un nombre arbitraire de pas (430, en ne considérant que les 400 derniers pas). La répartition obtenue s'interprète comme suit : La "tour" dans le coin arrière gauche correspond aux règles chaotiques, présentant une faible déviation autour d'une valeur de l'entropie maximum. les règles ordonnées (classe 2) se trouvent le long du bord gauche et présentent des valeurs variables de l'entropie moyenne mais s'en écartant très peu. Les règles plus complexes, elles, se trouvent le long du bord droit, où la déviation standard est maximum (les règles pour lesquelles cette déviation dépassait la valeur maximale du graphique ont été ramenées sur ce bord).

³C'est toutefois possible.

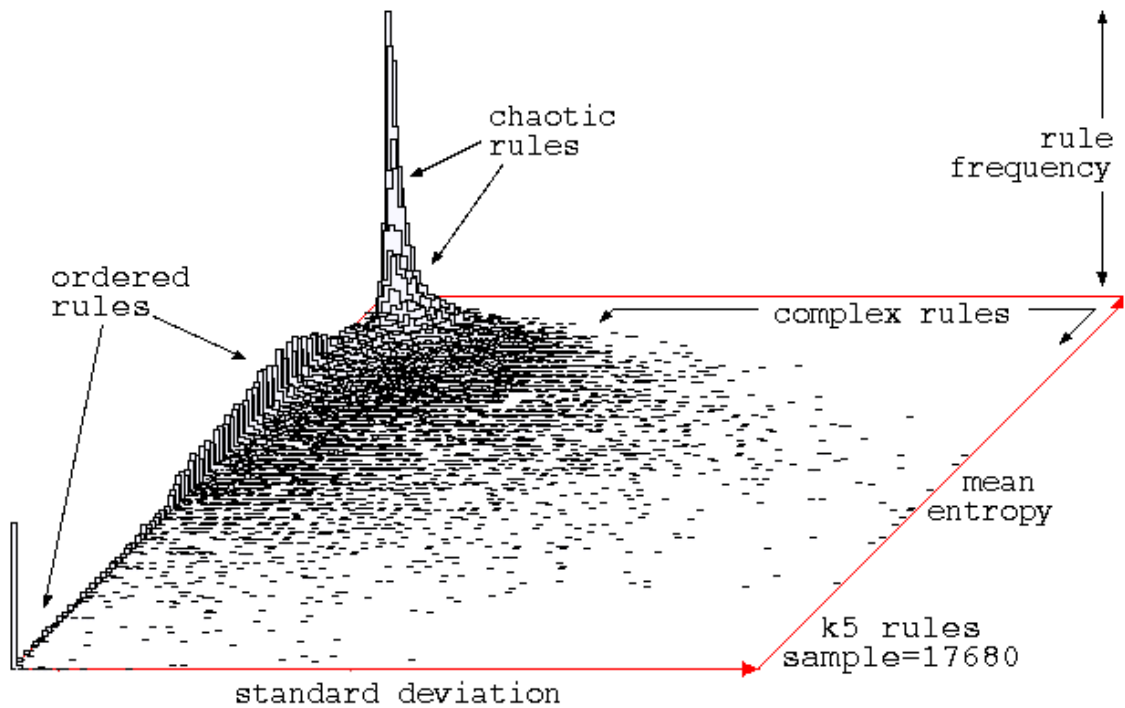


Fig. 5.8 – Classification topologique de Andy Wuensche[26]. 17680 règles, avec $d = 1, S = 2, r = 2$, sont triées en fonction de la valeur moyenne et de la déviation standard de H_k mesurée sur 400 pas.

En accord avec ce qui a été écrit dans la section 4.2.1, la hauteur de la "tour", soit la proportion de règles chaotiques, grandit lorsque l'on refait la même expérience avec des automates $r > 2$.

Si on génère aléatoirement des règles et qu'on les répartit sur ce graphique, on peut obtenir des règles d'un type particulier souhaité en les sélectionnant dans la partie correspondante du graphique.

Il est à remarquer que l'on n'observe pas, ici, de frontière nette entre les différentes zones et, en particulier, que les règles complexes ne se situent pas entre les règles ordonnées et les règles chaotiques. Il faut se rappeler, néanmoins, qu'il ne s'agit pas d'un paramétrage génotypique de l'espace des règles, du type de celui recherché dans la section 4.2.2, mais bien d'un paramétrage phénotypique, entaché donc par l'arbitraire du choix du nombre de pas et de la taille des automates considérés. En particulier, la sélection d'une règle dans le graphique ci-dessus dans un but précis doit encore être suivie d'une observation directe, afin de juger son comportement.

5.3 Explorations dans l'espace des phases

L'évolution d'un automate peut aussi être caractérisée en analysant sa dynamique dans l'espace de phase qui lui est propre. L'espace de phase, Φ , accessible pour un automate de N cellules et deux états, peut être mis en correspondance point par point avec l'ensemble des entiers allant de 0 à $2^N - 1$. De fait, à chacune des 2^N configurations que peuvent prendre les N bits, correspondent de manière unique un des 2^N premiers entiers écrit en binaire. Suivant sa dynamique propre, sa règle, l'automate passe de configuration en configuration, soit de point en point.

On peut réaliser un diagramme du comportement d'un automate donné, en affichant un point pour chaque état possible, et en reliant chacun de ces états à celui qui serait obtenu après 1 pas d'évolution. On a, au final, un résumé de la dynamique de l'automate puisqu'on peut partir d'un point particulier, correspondant aux C.I., et retracer l'évolution en suivant les flèches.

Dans cette section, nous discuterons de la forme générale des dits diagrammes selon le type de règle considéré. Nous illustrerons nos commentaires par des représentations graphiques en se limitant, pour ce faire, à des systèmes de petite taille. Enfin, nous mettrons nos résultats en relation avec d'autres, plus généraux, trouvés dans la littérature.

Si l'idée d'une représentation graphique est séduisante, il apparaît rapidement un léger problème : le nombre d'états à représenter devient très vite beaucoup trop grand (32.768 points pour des systèmes à 15 cellules, près de 10^{15} pour $N=50$). La représentation graphique des diagrammes ne pourra donc se faire que pour les systèmes de petite taille. Néanmoins, le nombre de points à afficher peut être grandement diminué sans perdre de généralité. En effet, l'utilisation de conditions au bords cycliques rend beaucoup de ces points équivalents. Par exemple, la situation "0,0,0,1,0,0,0" est équivalente à la situation "0,0,0,0,0,0,1", "0,0,0,0,0,1,0", etc... En ne considérant que les points non cycliquement équivalents, on supprimera une majeure partie des points à afficher tout en conservant les points essentiels, à savoir, un point par groupe d'états équivalents (cf. Fig 5.9). On notera Φ_r l'espace de phase ainsi réduit. La relation analytique $\#\Phi_r(N)$ n'est pas immédiate, mais à titre d'exemple, il n'y a plus que 2192 points non-cycliquement équivalents pour un système à 15 cellules.

A chaque point de l'espace de phase $c \in \Phi$ correspond une image $F(c)$ (et une seule, puisque le système est déterministe) et éventuellement une ou plusieurs préimages $\{d \in \Phi | F(d) = c\}$, soit les états qui amènent vers le point considéré. Sur la figure 5.9 le point 1 a pour image le point 0 et pour préimage les points 5,7,11 et 13.

Les graphiques de cette section ont été obtenus en créant l'ensemble des couples $\{c, F(c)\}$ pour tous les points de l'espace réduit Φ_r , puis en remplaçant les points images par leur équivalent cyclique présent dans Φ_r . Nous voulions mettre en évidence l'évolution de l'entropie, mais pour des suites si courtes l'algorithme de Titchener n'est pas valable, et en particulier, la T-entropie varie pour des points cycliquement équivalents, ce qui ne peut être accepté. A chaque point a donc été associé une couleur, sur $N/2$ niveaux de gris, représentant le premier ordre de l'entropie de Shannon, donc, grossièrement, le rapport de points blancs et de points noirs, le plus clair se rapportant à $H_1=0$, le plus foncé, à $H_1=1$.

La plupart des automates fondamentaux présentent des dynamiques contractives : pour un ensemble de conditions initiales possibles, il y a un nombre moindre d'états possibles après t pas.

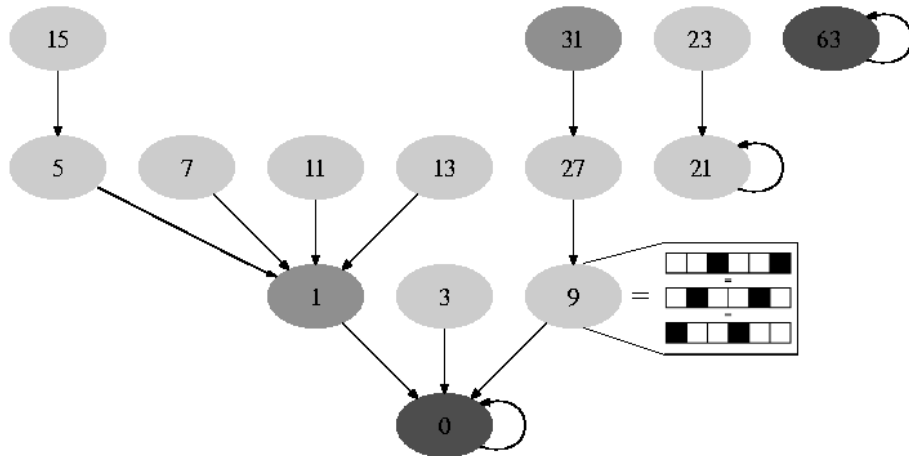


Fig. 5.9 – Diagramme d'évolution de la règle 160, pour $N = 6$. La configuration correspondant au point 9, ainsi que les configurations cycliquement équivalentes sont détaillées.

L'ensemble des configurations observables diminue donc le temps, d'où le terme de "contraction". C'est naturellement lié au fait que deux états peuvent renvoyer vers un seul et même état, ou, de manière équivalente, que tout les états implémentables en conditions initiales ne peuvent pas nécessairement être obtenus par évolution de l'automate. Dans l'étude des systèmes dynamiques la contraction d'un volume de l'espace de phase est la caractéristique des systèmes dissipatifs, condition *sine qua non* pour permettre l'apparition d'attracteurs.

Certaines configurations ne pourront jamais être obtenues par la dynamique de l'automate. Dans la littérature ces points inaccessibles sont appelés "Garden-Of-Eden States" (GOES). Ils correspondent à des états qui ne seront jamais visités, à moins que l'automate ne soit spécifiquement initialisé dans ces configurations précises. Sur les diagrammes ces points sont ceux qui forment les extrémités des branches. On notera G la proportion d'état GOE, soit $G = \frac{\#JdE}{\#\phi}$ avec $JdE = \{d \in \Phi | \exists c \in \Phi tq F(c) = d\}$. De manière plus complète, il est aussi intéressant d'étudier la I-distribution, soit la distribution du nombre de préimages par point[26].

A l'opposé des Jardins d'Eden, il existe des attracteurs. Ceux-ci peuvent être de type ponctuel, une configuration stable, ou bien cyclique, plusieurs configurations renvoyant indéfiniment les unes vers les autres en boucle fermée.

Classe 1

Les automates de classe 1 présentent systématiquement un attracteur ponctuel unique. La figure 5.10 montre le diagramme obtenu pour la règle 160 avec $N = 11$. Tout les points du diagramme sont reliés à un attracteur unique qui renvoie vers lui-même. Le même aspect général est observé pour d'autres N , et pour d'autre règles de classe 1.

Le système converge donc uniformément vers ce point. La hauteur du diagramme, soit le nombre de flèches séparant un GOES de l'attracteur, permet de se faire une idée du nombre de pas nécessaire pour arriver à l'attracteur final, soit la vitesse de convergence. La convergence est ici très rapide,

ce qui est évidemment lié à l'exposant de Liapounov négatif observé pour cette classe. Pour les automates de cette classe, G est grand et croît avec N et la I-distribution présente deux maximas locaux, dont un en 0[22].

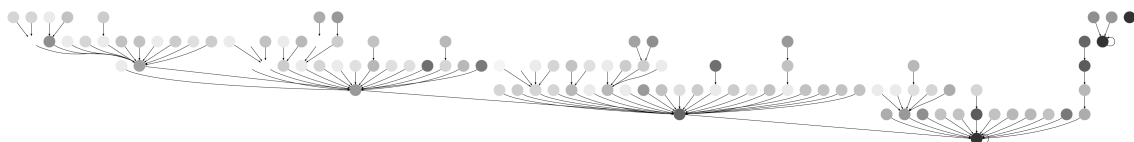


Fig. 5.10 – Bassin d'attraction de la règle 160 (classe 1), pour $N=10$.

Comme nous l'avons déjà dit, puisque le nombre d'états est limité (2^N), les systèmes de taille limitée repassent nécessairement par un état déjà visité, et dès lors, suivent la même évolution. Tous les automates finis finissent donc forcément par tomber dans des attracteurs. Seulement, ils n'y vont pas tous par le même chemin.

Classe 2

La figure 5.11 montre le diagramme de la règle 108, appartenant à la classe 2, pour un nombre de cellules $N = 11$. L'aspect général est très différent des diagrammes de classe 1, ci-dessus.

Premièrement, on observe que, très vite, l'automate est attiré dans un cycle de longueur courte (ici un ou deux points maximum). Attention, il est important, ici, de se rappeler que nous avons réduit l'espace de phase. Par conséquent, un point qui renvoie sur lui-même ne représente pas nécessairement un état immobile. Un tel point pourrait aussi représenter un cycle entre états cycliquement équivalents, par exemple, le déplacement régulier, à chaque pas, de toutes les cellules vers la droite (cf. règle 180, appendice B). À défaut de pouvoir visualiser la longueur réelle des cycles stables, on peut néanmoins en apprécier la forme générale et la topologie.

Deuxièmement, il faut remarquer que le diagramme n'est pas composé d'un seul groupe, mais, au contraire, est morcelé en plusieurs bassins distincts, dont le nombre augmente avec N [21]. L'évolution d'un tel automate restera très localisée, c'est à dire qu'il visitera une partie fort restreinte de l'espace de phase située autour de sa condition initiale, ce qui est en accord avec les observations de la section 5.1. La I-distribution typique est semblable à celle des classe 1, cette forme étant, en fait, générale pour les systèmes ordonnés[22].

Classe 3

Ces deux derniers cas sont évidemment assez clairs quant à l'interprétation qu'on peut en faire. Voyons, maintenant, comment on peut caractériser les diagrammes des automates de classe 3. Les figures 5.12, 5.13, et 5.14 montrent les diagrammes de la règle 30 pour N allant de 11 à 13.

Qu'observons-nous ? L'ensemble des points est réparti en peu de morceaux différents : généralement un gros groupe, ou deux, ou trois, mais pas une collection d'îlots comme dans le cas de la classe 2. De plus, ils se différencient de ceux de la classe 1 par deux traits importants. Premièrement, les branches sont beaucoup plus longues, ce qui signifie que la phase de transition pour arriver à l'état stationnaire est plus longue que dans le cas de la classe 1. Deuxièmement, ces états stationnaires ne sont pas ici des points fixes, mais bien des cycles dont la longueur augmente avec la taille du

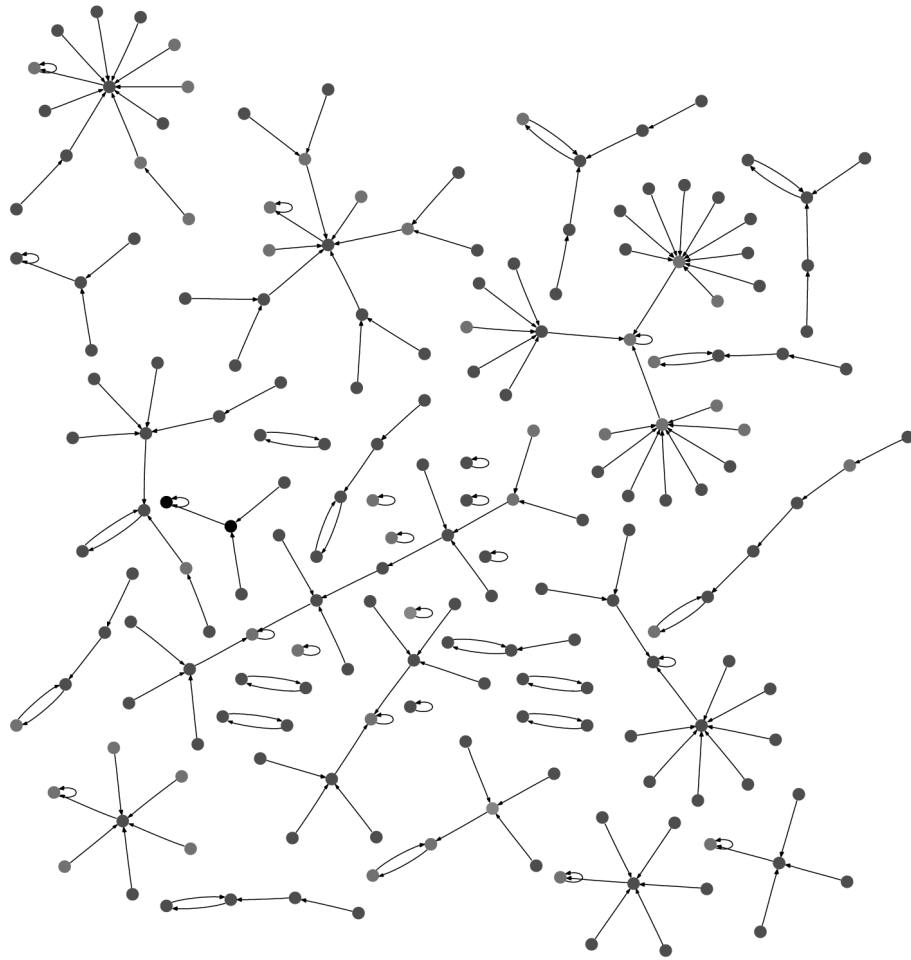


Fig. 5.11 – Les différents bassins de la règle 108 (classe 2), pour $N=11$.

système[21]. La I-distribution pour les règles chaotiques est caractérisée par un G petit et une grande fréquence pour les petits nombres de préimages [22].

Les diagrammes de la règle 30 présentent donc quelques bassins d'attraction, très étendus, dont l'attracteur final est un cycle. Peut-t-on concilier ces résultats avec ceux obtenus préalablement, à savoir que la règle 30 tend à présenter les aspects extérieurs d'une dynamique aléatoire?

La période de croissance, observée au chapitre 5.2 pour les règles intrinsèquement chaotiques sur des C.I. biaisées, correspond aux débuts des branches qui amènent jusqu'au cycle final, les états fortement biaisées étant, comme on l'a vu, des "GOES". Pour ce qui est du comportement stationnaire, il nous faut d'abord nous pencher d'un peu plus près sur les notions de surjectivité et d'injectivité avant de continuer la discussion.

Bien que, comme nous l'avons vu, la plupart des automates fondamentaux présentent de nombreux Jardins d'Eden, il existe pourtant plusieurs règles surjectives. Il s'agit des règles pour lesquelles $F : C \rightarrow C$ est surjective, soit pour lesquelles tous les points admettent une préimage. Des règles, donc, pour lesquelles il n'y a plus de Jardins d'Eden et où G tend vers 0.

Dans nos observations, limitées à des systèmes de petite taille, les diagrammes de la règle 30

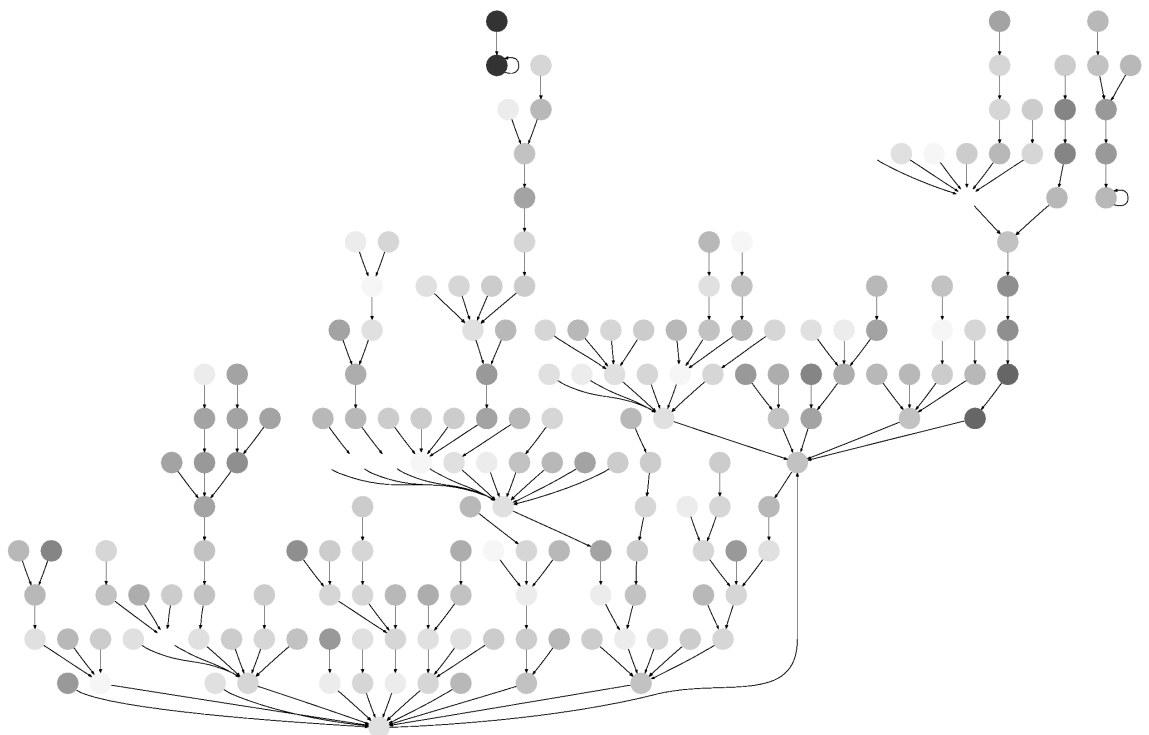


Fig. 5.12 – Diagramme pour la règle 30, pour $N = 11$.

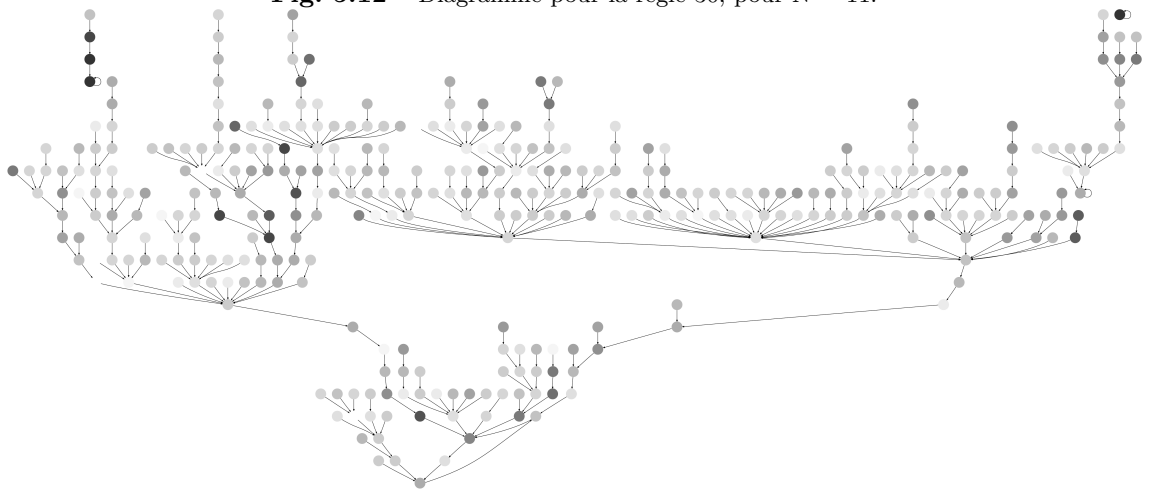


Fig. 5.13 – Diagramme pour la règle 30, pour $N = 12$.

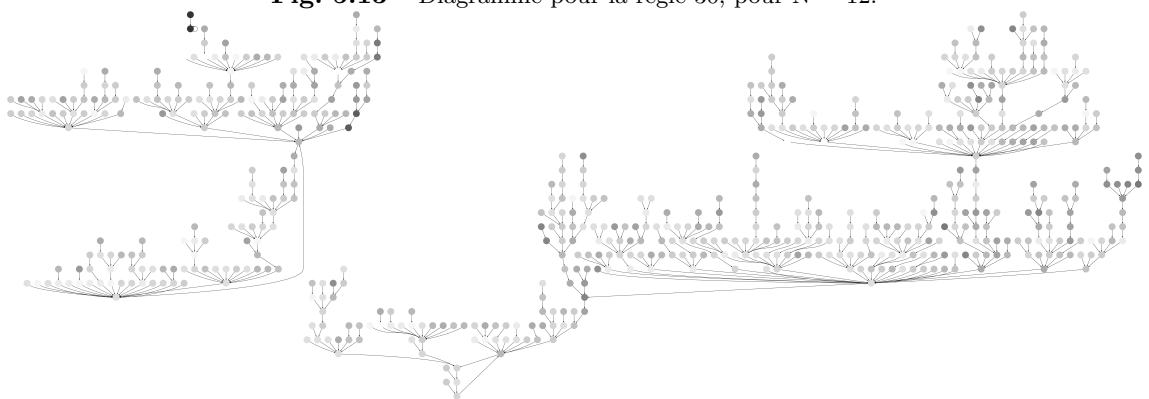


Fig. 5.14 – Diagramme pour la règle 30, pour $N = 13$.

contiennent toujours des GOES. Mais il a été montré[23] que, pour la règle 30, G tend exponentiellement vers 0 à la limite $N \rightarrow \infty$, tous les points se retrouvant alors dans des cycles ou sur de longues branches de transition⁴. Ceci à l'inverse des règles 110, 54, 18 et 4, pour lesquelles la fraction de GOES tend vers 1. Ceci est très important car la surjectivité est une condition inévitable pour qu'un système puisse prétendre à l'ergodicité.

Il ressort des résultats précédents que la règle 30 tend, quelle que soit les C.I., vers des configurations d'entropie maximum. En d'autres termes, les configurations de basse entropie doivent être confinées aux extrémités des branches. Le fait que la règle 30 tend à la surjectivité, signifie que les diagrammes tendent à présenter des branches de plus en plus longues. En fait, la longueur de ces branches croît plus vite avec N que leur nombre et l'état stationnaire correspond au parcours le long de ces branches.

Classe 4

Pour ce qui est de la classe 4, une fois de plus, la discussion est ambiguë. Il semblerait qu'il s'agisse d'un compromis entre les trois classes, à savoir que le nombre de bassins séparés va augmenter avec N comme pour la classe 2, qu'il y a un grand nombre de préimages par point comme pour la classe 1, mais aussi des branches relativement longues comme pour la classe 3. Mis à part le fait qu'elle soit à l'équilibre entre les classes deux et trois, les propriétés spéciales de la classe 4, et en particulier de la règle 110, n'apparaissent pas clairement ici (cf.fig. 5.15, 5.16 et 5.17). Wuensche note cependant[26] que la I-distribution suit des lois de puissances pour les règles complexes.

5.4 Règles inversibles

L'étude des règles inversibles est motivée par le caractère inversible que présentent la plupart des lois de la physique classique à petite échelle. L'inversibilité implique que l'on puisse retrouver la préimage d'une configuration donnée. Outre la prédictibilité, on exige donc ici la rétroprédictibilité, c'est à dire que le système doit rester déterministe lorsqu'on inverse la flèche du temps.

Un automate $\mathcal{A} = (d, \vec{S}, \vec{V}, f)$ de fonction de transition globale F est dit inversible s'il existe un automate $\mathcal{B} = (d, \vec{S}, \vec{V}', f')$ de fonction de transition globale F' tel que $F' = F^{-1}$. L'automate \mathcal{B} est alors dit automate inverse de \mathcal{A} .

Un automate unidimensionnel est inversible ssi il est injectif[28].

Ce résultat implique qu'un automate est inversible si sa fonction de transition globale est injective. Par conséquent, tout automate injectif est aussi surjectif⁵.

Ceci peut se comprendre facilement en suivant le petit raisonnement que voici. On peut implémenter en C.I. toutes les configurations possibles. Quelle que soit celle que l'on choisit, appelons-la c , elle aura nécessairement une et une seule image d . Maintenant, puisque l'automate est inversible, il doit exister un automate inverse qui réalise l'inverse du diagramme de ce premier automate; un automate inverse, donc, qui, à chacune de ces images d , associe maintenant comme image c , et ce de manière univoque. Or, il est impossible qu'une configuration de cet automate inverse n'admette

⁴C'est entre autre ce genre de propriétés qui a amené à utiliser la règle 30 en cryptographie

⁵Ceci dans le cas $N \rightarrow \infty$.

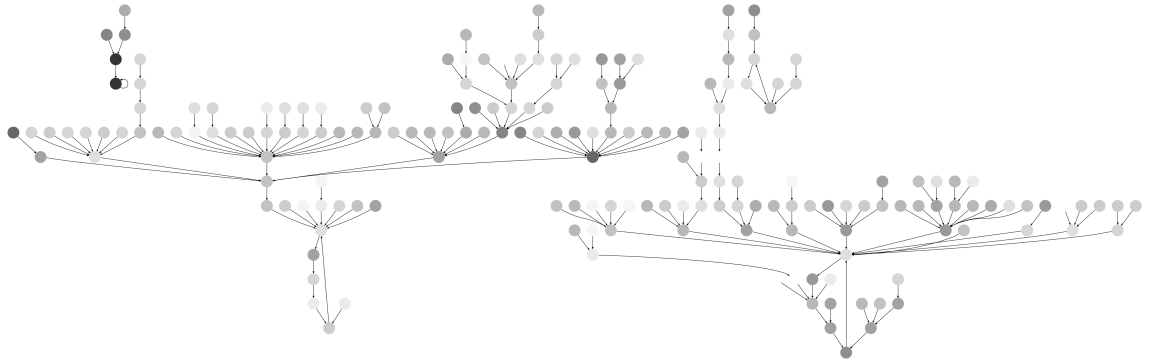


Fig. 5.15 – Diagramme pour la règle 110, pour $N = 11$.

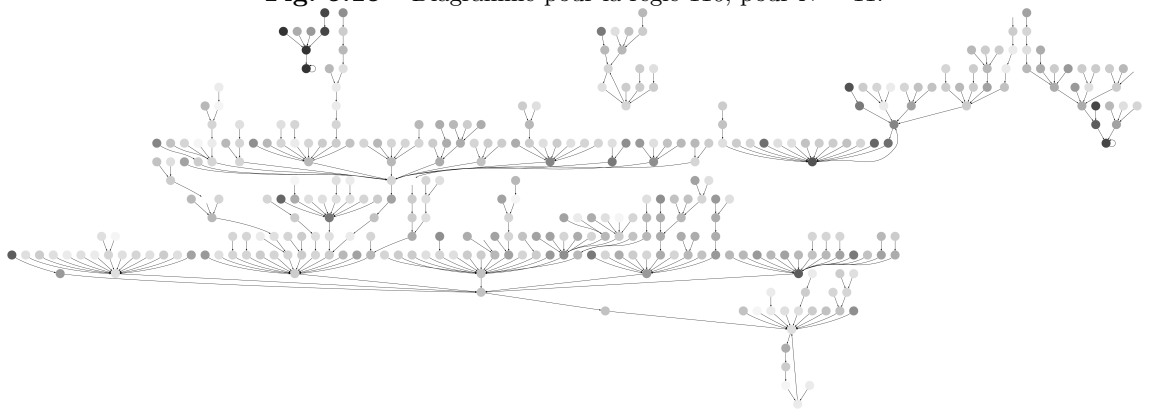


Fig. 5.16 – Diagramme pour la règle 110, pour $N = 12$.

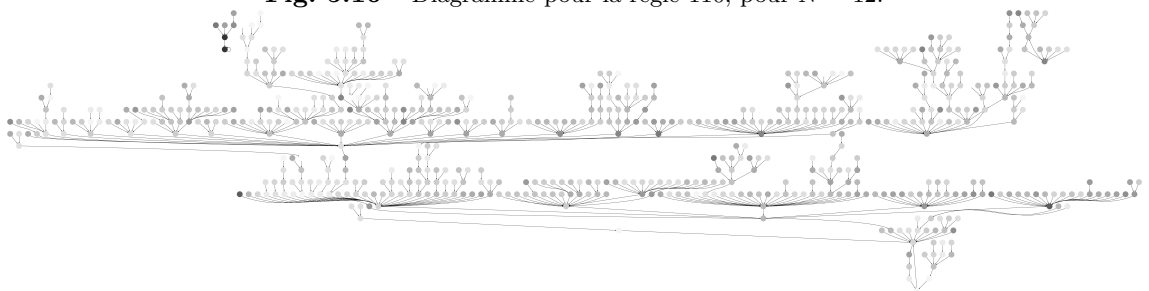


Fig. 5.17 – Diagramme pour la règle 110, pour $N = 13$.

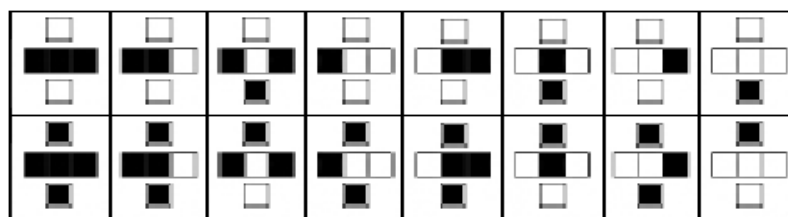


Fig. 5.18 – Illustration graphique de la règle inversible 37R.

pas d'image. On a donc, au final, pour tout automate inversible, que chaque configuration admet une préimage et qu'il n'existe pas de GOES.

Chaque point d'un automate inversible a donc une et une seule préimage, ainsi qu'une et une seule image. La seule topologie imaginable pour un diagramme correspondant à un tel automate est donc soit une boucle fermée, soit un ensemble de plusieurs boucles fermées, contenant, dans un cas comme dans l'autre, toutes les configurations possibles dans l'espace des phases.

Il existe, parmi les automates fondamentaux, 6 règles inversibles, présentant toutes des comportements de classe 2; mais Wolfram[12] montre qu'en considérant des A.C. de plus grande portée d'interaction ou d'un plus grand nombre d'états, on peut trouver des règles réversibles de classe 3 et même de classe 4.

Il est possible de créer expressément des règles inversibles en considérant un voisinage de deuxième ordre temporel. Soit, si x désigne la position d'une cellule,

$$c_{(x,t)} = f(c_{(x-1,t-1)}, c_{(x,t-1)}, c_{(x+1,t-1)}, c_{(x,t-2)}). \quad (5.5)$$

On peut, par exemple, partir des règles élémentaires, et imposer, comme dépendance au second ordre, que le nouvel état de la cellule soit inversé si la cellule était noire deux pas auparavant. On obtient automatiquement, de la sorte, des règles inversibles, puisqu'en inversant les tables selon l'axe vertical, on obtient l'automate inverse dont il était question plus haut. Ces règles ne sont donc pas simplement inversibles, elles ont, en plus, la propriétés d'être leur propre inverse. Nous noterons ces règles de la même manière que précédemment, en considérant les trois cellules du premier ordre, mais en affublant leur nom d'un R qui signifie réversible (voir l'exemple figure 5.18 pour la règle 37R).

Les exemples montrés sur la figure 5.19, pour la règle 214R de classe 3, illustrent une conséquence intéressante de cette inversibilité.

Si l'on part d'une condition simple, on observe une évolution directe vers des configurations de grande entropie mais, dans certains cas, il est possible que l'évolution passe par des situations simples. En particulier, puisque l'automate est son propre inverse, on peut préparer les C.I.⁶ dans une configuration obtenue à partir du premier cas, et on retrouvera la situation simple des premières C.I.. L'émergence de ces configurations de basse entropie est néanmoins extrêmement rare et éphémère. En fait, elle sont d'autant plus rare que le système est grand.

Ceci est en accord avec le résultat établi par Poincaré affirmant que, dans le cas, par exemple, d'un gaz dans une boîte, le système passera nécessairement par des situations de basse entropie,

⁶Qui comporte maintenant la donnée de deux suites.

éventuellement après un temps infiniment long. Ces événements sont toutefois infiniment rares et éphémères, et c'est ce qui sauve le deuxième principe si on exclut des fluctuations négligeables de ce type.

Qu'en est-il des classe 4 ? L'une d'elle est la règle 37R, que nous utilisons ici comme exemple (cf. figure 5.20). Nous mesurons, ici, l'évolution de la T-entropie, en prenant comme C.I. deux suites biaisées ($p=1/20$). Une première observation que l'on est tenté de faire est que ces règles ne donnent pas lieu, comme dans les cas précédent, à une décroissance de l'entropie. En effet, plusieurs expériences montrent le même résultat typique : H_{TN} croît jusqu'à une valeur situé légèrement au dessous du maximum ($H_{TN}(t) \sim 0.9$ pour $t > 2000$) et semble y être stable.

Tirer de cette observation un résultat général serait une erreur. En effet, rien ne nous empêche, comme ci-dessus, de sélectionner la dernière et l'avant dernière ligne du tableau de la figure 5.20 et de les imposer comme premières et deuxième lignes. De nouveau, on y observera sûrement une décroissance de la T-entropie, puisque la tableau obtenu sera exactement l'inverse de ce dernier.

Impossible, alors, de conclure un comportement général ? Il est intéressant, dans ce cas théorique particulier que sont les A.C. inversibles, de considérer l'approche de l'entropie algorithmique.

Pour simplifier les choses, nous reprenons notre discussion dans le cadre des automates de premier ordre.

Les suites issues d'une configuration initiale simple par le biais d'un automate inversible, celles, donc, qui mènent à une configuration simple dans le cas de l'automate inverse, sont tout-à-fait atypiques par rapport à l'ensemble des suites possibles. Qu'elles soient atypiques signifie que, malgré leur apparence normale, elles ont quelque chose qui les distingue de toutes les autres suites et qui fait d'elles des suites très particulières. En fait, malgré leur grande entropie statistique, il semble que ces suites possèdent une entropie algorithmique très basse. Eclaircissons quelque peu ce dernier point.

Pour décrire s , la n^{eme} ligne du tableau de la figure 5.20 à un ami éloigné, il existe deux méthodes.

La première consiste à en faire directement la description, éventuellement en utilisant un algorithme de compression statistique du style de la méthode de Huffman : c'est l'approche de l'information statistique, que nous avons considérée depuis le début en mesurant H_{TN} .

La deuxième méthode consiste à donner les C.I., la règles et le nombre de pas à effectuer : c'est l'approche de Kolmogoroff. Ces données définissent effectivement la suite de manière univoque et, dans le cas de C.I. simples, cette méthode pourrait être beaucoup plus économique. On aurait, dans ce cas, $K(s) < H(s)$.

Soit deux suites s_1 et s_2 , dont la deuxième est issue de la première par le biais d'un automate inversible \mathcal{A} après un temps t . La première peut donc être obtenue à partir de la deuxième par l'automate inverse $\mathcal{B} = \mathcal{A}^{-1}$ après le même temps t . Puisqu'on peut, pour chacune de ces suites, en faire la description à partir de l'autre, comme expliqué plus haut, on a les inégalités suivantes :

$$k(s_2) \leq k(s_1) + \log(t) + C_{\mathcal{A}} \quad (5.6)$$

$$k(s_1) \leq k(s_2) + \log(t) + C_{\mathcal{B}}, \quad (5.7)$$

où les termes $k(s)$ sont minorés par $\log(N)$ (il faut au minimum décrire la taille de la suite), et où

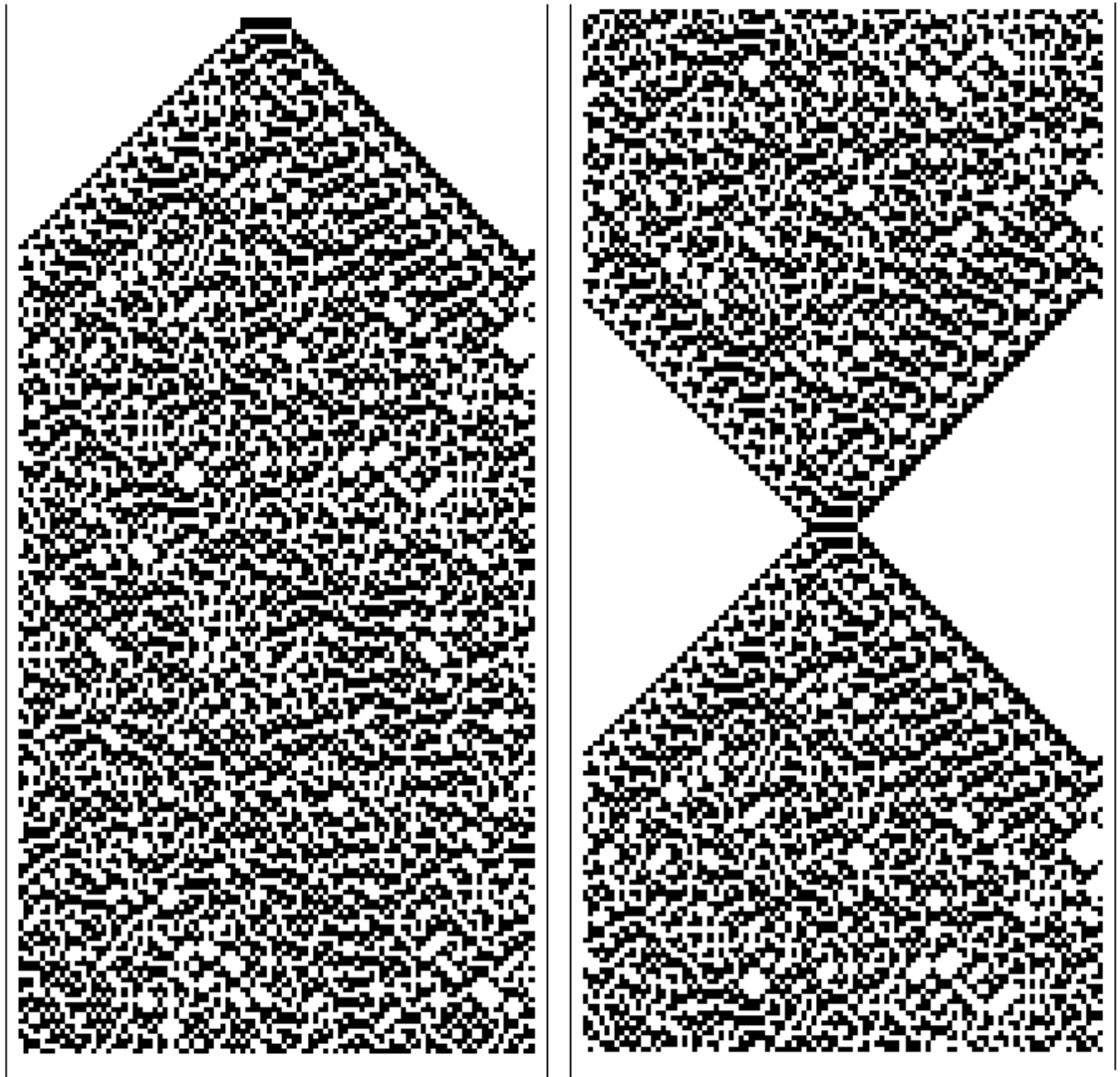


Fig. 5.19 – Automate réversible de classe 3, règle 214R. À gauche, évolution à partir de C.I. simples. À droite, évolution à partir des dernière et avant-dernière lignes du tableau de gauche.

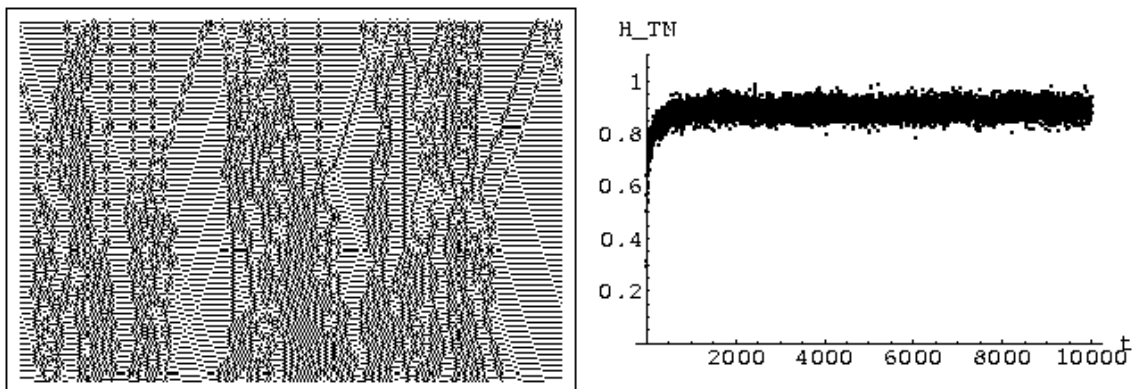


Fig. 5.20 – Automate réversible de classe 4, règle 37R. L'image présente le comportement typique, avec, pour C.I., deux suites biaisées de 400 bits ($p=1/20$). Le graphique correspond au même type de C.I. avec $N=2000$.

les constantes $C_{\mathcal{A}}$, $C_{\mathcal{B}}$ représentent les longueurs des règles des automates \mathcal{A} et \mathcal{B} .

De manière générale, en théorie de l'information, les suites considérées sont telles que les termes en N l'emportent sur tous les autres. Les deux derniers termes des inéquations 5.6 et 5.7 des membres de gauche doivent donc être négligés. En fait, puisqu'on a vu que l'évolution de ces automates se fait sur des boucles fermées, on pourrait avoir $s_1 = s_2$ si t est la longueur de la boucle. Dans ce cas, il est évident qu'il faille négliger les termes $\log(t)$. Il en résulte que $k(s_2) = k(s_1)$

Au final, nous montrons donc que l'entropie algorithmique est conservée par les automates inversibles. Ceci, qui n'est pas sans rappeler la conservation de l'entropie thermodynamique pour les transformations réversibles, peut s'interpréter comme suit : chaque ligne conserve, pour les automates inversibles, l'information de son passé, et se situe donc au même niveau d'un point de vue informationnel.

Les diagrammes correspondant aux automates inversibles, dont on a dit plus haut qu'il devaient avoir la forme d'un ou de plusieurs cercles, dessinent, dès lors, des iso-entropiques dans l'espace des phases. En particulier, le cas du cercle unique est à exclure, puisque cela signifierait que toutes les suites de longueur N ont la même entropie algorithmique.

Ce résultat n'aurait pu être atteint par le seul biais de l'entropie statistique et illustre particulièrement les limites de cette approche.

La considération, dans un cadre plus général, de l'entropie algorithmique et de son évolution dans les A.C. serait, assurément, une étude des plus intéressantes. Mais celle-ci demanderait, pour être complète, un investissement théorique dépassant largement le cadre de ce travail. Pour cette raison, nous nous sommes contenté, dans cet exemple concernant les règles inversibles, de souligner son intérêt et laissons à titre de question ouverte la généralisation de l'évolution de l'entropie algorithmique aux automates quelconques.

Chapitre 6

Discussion sur la notion de complexité

La complexité des règles est liée à leur capacité de calcul. Une grande capacité de calcul nécessite, de fait, des modes de propagation interne de l'information complexes et variés. Ces modes sont visibles, dans certains cas, sous la forme de glisseurs. Mais les liens entre ces deux grandeurs ne sont pas directs, et le sens qu'on leur accorde dans différents articles ne sont pas toujours équivalents.

Comme nous l'avons dit dans la section 4.2.2, il est deux intérêts distincts à l'étude des automates :

- La possibilité d'utiliser ceux-ci pour modéliser des systèmes physiques en exploitant la propriété d'évolution parallèle de tous les composants du système.
- L'étude générale des phénomènes d'auto-organisation, basée sur la recherche de relations liant les modes d'interactions locaux, imposés, aux comportements généraux, observés à grande échelle.

Ces deux objectifs distincts mènent à quelques confusions. Prenons l'exemple du débat qui oppose entre autre Mélanie Mitchell[24] et Chris Langton[13]. Dans les tentatives de classification, on tente d'apprécier la capacité de calcul des automates. L'expérience réalisée séparément par H. Packard [29], et M. Mitchell [24], consistait à générer des règles complexes - et ici nous commentons l'abus de langage à la base de ces confusions - par un algorithme génétique¹ en posant comme critère de sélection la capacité de résoudre un problème donné de tri². Cette approche est évidemment propre à la première des deux finalités citées ci-dessus, en ce sens que la complexité y est reliée à la capacité de résoudre effectivement un problème posé. Pour M. Mitchell c'est ce genre de critère qui doit être pris en compte pour apprécier la capacité de calcul d'un automate. De son point de vue, cette capacité de calcul ne présente d'intérêt que si elle permet d'effectuer concrètement des calculs, comme par exemple calculer les décimales de Pi ou effectuer une analyse

¹Un algorithme génétique est un algorithme qui fait évoluer un programme en générant à partir de celui-ci une série de variantes par le biais de modifications aléatoires, puis en sélectionnant parmi ces variantes celles qui répondent le mieux à un critère de sélection prédéfini. La plupart des recherches actuelles sur l'intelligence artificielle et notamment sur les "*self-learning algorithm*", utilisent des algorithmes génétiques.

²On y demandait à l'automate d'être capable de déterminer si les conditions initiales contenaient plus de 1 ou de 0, en convergeant vers un état stable correspondant.

des conditions initiales ou n'importe quel autre problème *posé à l'avance*. L'automate est vu ici comme un programme, dont les C.I. seraient l'*input*, et l'état stable vers lequel il convergerait, l'*output*.

Les critères comme ceux du temps de convergence, utilisé par Langton (voir section 4.2.2 et 5.1), ou des variations de l'entropie (voir section 5.2.1), concernent, eux, la deuxième finalité. On peut se demander quel est l'intérêt de cette deuxième considération, sans doute motivée plutôt par des questions philosophiques que par des soucis pratiques, mais dans le cadre de la théorie de l'information une telle étude peut apparaître comme fondamentale. Développons.

Si tous les systèmes sont vus comme des suites de chiffres, l'évolution d'un système va consister en des variations dans cette suite. Or, *la meilleure machine pour calculer une telle évolution, n'est autre que le système lui-même*, qui à partir de sa configuration initiale et de règles d'évolutions calcule le nouvel état qui en résulte. Considérons l'exemple d'une quantité d'eau placée dans une température plus basse que 0° . Nous savons que l'eau va se changer en glace. Mais nous savons aussi cette transition ne s'effectue pas instantanément : elle nécessite un processus de nucléation et se transmet de proche en proche. Dans notre façon de voir les choses cette transmission de proche en proche correspond au transfert d'information, comme dans les rouages d'une antique machine à calculer ou dans les circuits imprimés d'un ordinateur, qui mène en bout de parcours à la solution finale : le bloc de glace.

On essaye souvent, dans ce type d'étude, de rapprocher les notions de capacité de calcul de celles, mieux connues, de transitions de phases. En fait, il semblerait qu'il y ait plus de sens à expliquer les transitions de phase comme dues aux limites fondamentales des systèmes à calculer leur propre évolution.

On pourrait imaginer que les machines que sont les systèmes physiques peuvent être limitées par leur propre capacité de calcul. De par cette limite, elle conduiraient, lors du calcul de leur évolution, à des phases transitoires, s'étendant éventuellement à l'infini avant de converger vers la solution de leur état final.

Il semble que ces phases transitoires s'étendant à l'infini soient une bonne définition physique que l'on puisse faire de la vie. En fait, si les mathématiques et la physique permettent aujourd'hui une description étonnamment pertinente du monde qui nous entoure, elles semblent oublier d'y inclure l'évolution biologique, qui apparaît comme une singularité dans le cadre de la thermodynamique classique. Il n'existe à ce jour aucun modèle rendant compte de manière satisfaisante de ce phénomène, pourtant central dans notre existence, qu'est l'émergence et l'évolution de la vie[31]. C'est dans la construction d'un tel modèle que la deuxième des considérations citées plus haut trouve son intérêt.

Conclusion

Nous avons commencé ce mémoire en rappelant quelques notions essentielles de la théorie de l'information. La ligne conductrice suivie dans cette première partie aboutit à l'établissement d'une mesure déterministe de l'information contenue dans une suite finie. Cette mesure, la T-entropie est ensuite testée dans plusieurs cas connus, afin d'en apprécier le comportement.

La deuxième partie débute par quelques définitions et notations permettant de se familiariser avec les automates cellulaires. Il s'ensuit une discussion sur les divers types de classifications et les différents points de vue trouvés dans la littérature à leur sujet.

Ayant défini la mesure et l'objet à mesurer, nous avons poursuivi les investigations suivantes.

Une première étude sur les dépendances des A.C. aux conditions initiales met en relation les comportements globaux observés avec les modes de transport d'une perturbation dans le système. Il ressort de cette étude, outre quelques résultats essentiels sur les classes 1,2 et 3, que les règles complexes peuvent être distinguées par un grand temps de convergence lors du calcul du paramètre Λ qui y est défini.

Une deuxième étude a pour objet le comportement de la T-entropie d'une suite évoluant sous l'action d'un automate. Nous y caractérisons le comportement typique de la T-entropie pour les différentes classes et y montrons une manière intéressante de distinguer les deux types de classe 3, à savoir, intrinsèque et additif. Cette étude, complétée par une analyse DFA, montre que les règles complexes sont caractérisées par des oscillations de l'entropie correspondant à un bruit hautement corrélé. Ces résultats sont ensuite confrontés à d'autres résultats similaires, ressortant d'une analyse semblable trouvée dans la littérature.

Un troisième volet de ces investigations concerne les diagrammes représentant l'espace de phase propre aux automates. Nous y illustrons certains faits généraux, confirmés dans la littérature, concernant la topologie de ceux-ci et comment cette dernière offre un moyen supplémentaire de distinguer les différents types d'automates. Outre ces résultats, nous profitons de ce chapitre pour discuter des notions d'attracteurs, de Jardins d'Eden, de surjectivité, d'injectivité et de leur conséquence dans le domaine des automates.

Ces notions amènent directement à la considération d'un cas particulier : celui des règles présentant la propriété d'inversibilité. Dans ce pénultième chapitre, nous indiquons une méthode de construction de telles règles et en présentons quelques comportements typiques. Ces règles amènent un cas théorique intéressant, où les approches statistiques et algorithmiques de l'information sont directement confrontées. En particulier, nous y établissons un résultat concernant la conservation de l'entropie dans les automates inversibles, qui n'est accessible que par le biais du point de vue

algorithmique de la théorie de l'information. Ceci nous permet d'illustrer la richesse théorique de la considération du point de vue algorithmique. Néanmoins, une étude exhaustive de cette question dépassant largement le cadre de ce travail, nous nous contentons, à ce sujet, de cette illustration.

Parallèlement à ces investigations, une discussion est menée tout au long de ce travail sur les notions d'émergence de la complexité dans les systèmes simples. Cette discussion est précisée dans le dernier chapitre qui présente les différents sens que l'on peut lier à cette complexité et les amalgames qui en résultent. On y discute, de manière plus ouverte, de l'intérêt d'études portant sur de tels concepts et de leur portée philosophique.

Annexe A

Démonstration de l'expression de l'entropie de Shannon

Shannon exprime trois conditions pour que H corresponde à ce qu'il veut mesurer (cf. section 3.1.1) :

1. H doit être continue sur les p_i .
2. Si les p_i sont égaux pour $i = 1, \dots, M$, H doit être une fonction monotone croissante sur M.
3. Si on sépare un choix en plusieurs sous-choix, le H initial devrait être la somme pondérée des H propres à chacun des "sous-choix".

Soit $A(n) = H(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})$, l'entropie d'un choix entre n éléments équiprobables.

De 3, on peut décomposer un choix de s^m possibilités en m choix de s possibilités. On a alors

$$A(s^m) = m A(s) \quad (\text{A.1})$$

de même

$$A(t^n) = n A(t) \quad (\text{A.2})$$

Pour s et t donnés on peut choisir n très grand et trouver m tel que

$$s^m \leq t^n \leq s^{m+1} \quad (\text{A.3})$$

En prenant le logarithme et en divisant par $n \log s$, on a

$$\frac{m}{n} \leq \frac{\log t}{\log s} \leq \frac{m+1}{n} \Leftrightarrow \left| \frac{\log t}{\log s} - \frac{m}{n} \right| \leq \epsilon \quad (\text{A.4})$$

où ϵ est très petit. Vu la condition de monotonie 2, on peut écrire

$$A(s^m) \leq A(t^n) \leq A(s^{m+1}) \Leftrightarrow m A(s) \leq n A(t) \leq (m+1)A(s) \quad (\text{A.5})$$

Soit, en divisant par $n A(s)$

$$\frac{m}{n} \leq \frac{A(t)}{A(s)} \leq \frac{m+1}{n} \Leftrightarrow \left| \frac{A(t)}{A(s)} - \frac{m}{n} \right| \leq \epsilon \quad (\text{A.6})$$

$$\left| \frac{A(t)}{A(s)} - \frac{\log t}{\log s} \right| \leq 2\epsilon \quad (\text{A.7})$$

Il s'ensuit

$$A(t) = K \log t \quad (\text{A.8})$$

où K est une constante positive pour satisfaire 2.

Maintenant supposons que nous ayons un choix entre N possibilités dont les probabilités sont données par $p_i = \frac{n_i}{\sum n_i}$. On peut décomposer ce choix de $\sum n_i$ possibilités équiprobable en un choix de N possibilités de probabilités p_i suivi, si la $i^{\text{e}}me$ possibilité est choisi, d'un choix de n_i possibilités équiprobables. En écrivant les deux points de vue selon 3, on a :

$$K \log \sum n_i = H(p_1, \dots, p_N) + K \sum p_i \log n_i \quad (\text{A.9})$$

D'où

$$H = K[\log \sum n_i - \sum p_i \log n_i] \quad (\text{A.10})$$

$$\Leftrightarrow H = K[\sum p_i \log \sum n_i - \sum p_i \log n_i] \quad (\text{A.11})$$

$$= -K \sum p_i \log \frac{n_i}{\sum n_i} = -K \sum p_i \log p_i \quad (\text{A.12})$$

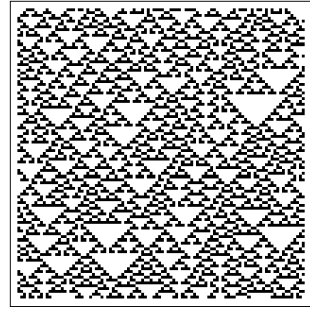
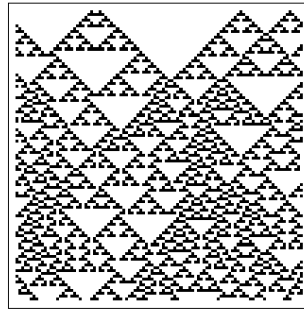
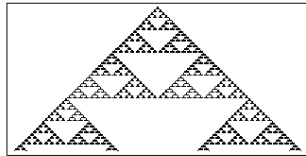
Pour des raison explicitées dans le texte, la constante K est fixée à $K = 1$.

Annexe B

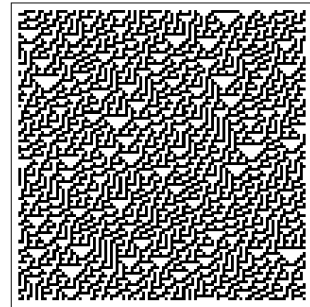
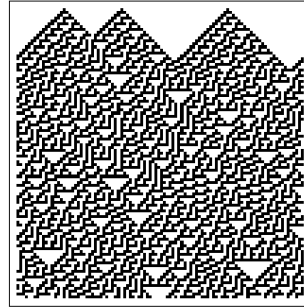
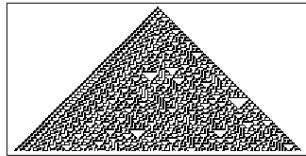
Illustration des règles utilisées

Pour chaque règle utilisée dans ce travail, nous montrons ici la représentation graphique de la règle, ainsi que son comportement sur des C.I. élémentaires, biaisées et aléatoires.

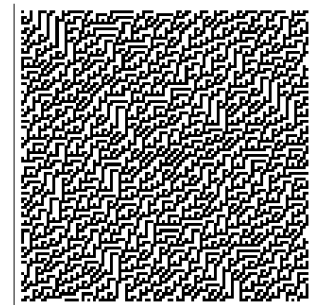
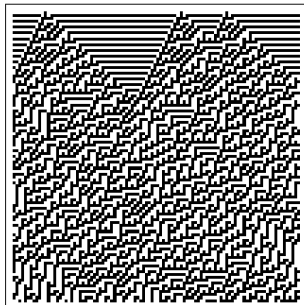
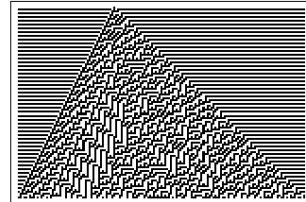
Règle 22



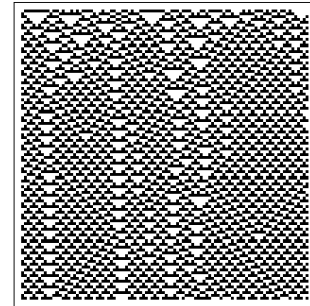
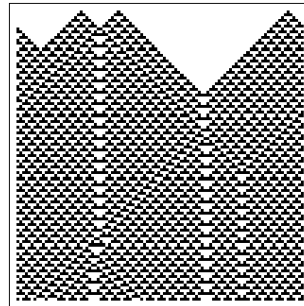
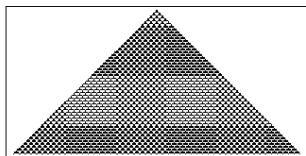
Règle 30



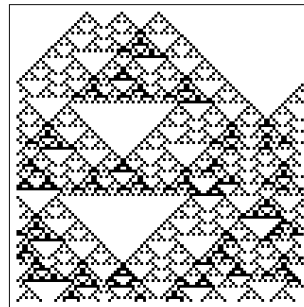
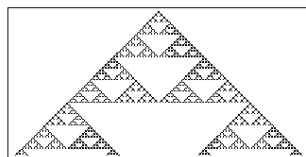
Règle 45



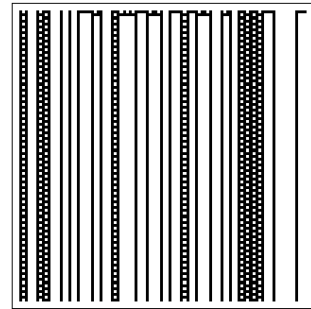
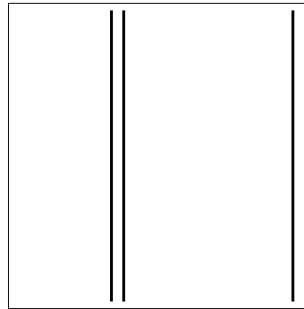
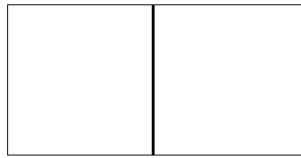
Règle 54



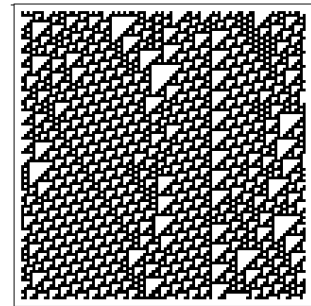
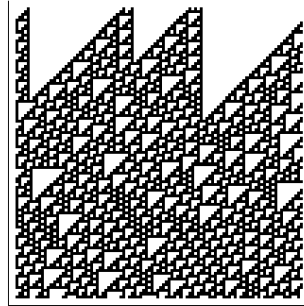
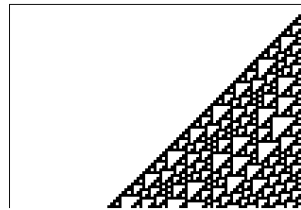
Règle 90



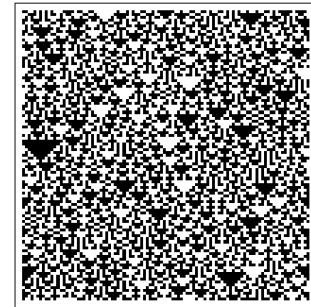
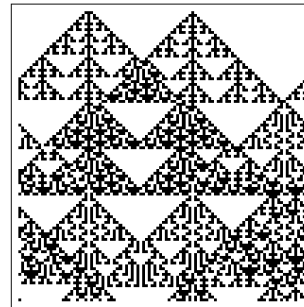
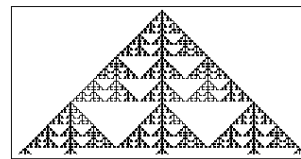
Règle 108



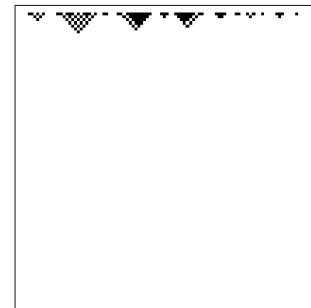
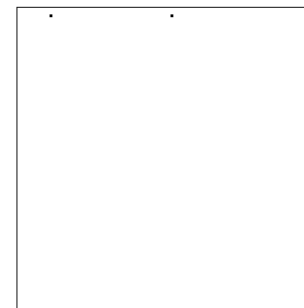
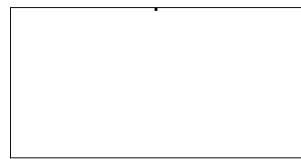
Règle 110



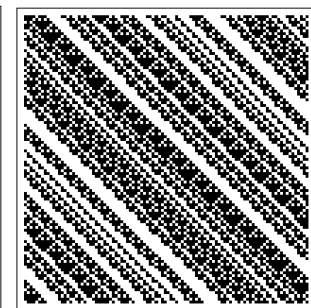
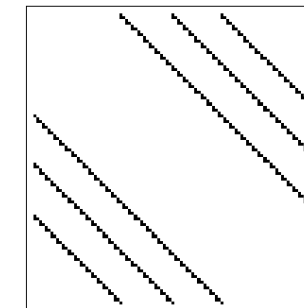
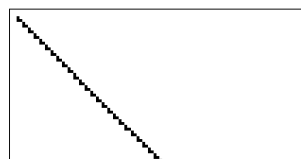
Règle 150



Règle 160



Règle 180



Annexe C

Analyse DFA

C.1 Introduction

L'analyse d'un signal temporel peut être effectuée par les méthodes de l'analyse fractale. Parmi ces techniques, la méthode de Hurst et la méthode DFA sont souvent utilisées pour caractériser le type de dynamique sous-jacente à un signal donné, en précisant le type de corrélations observées selon l'échelle ou la portée considérée. Ces méthodes sont utilisées notamment en éconophysique, ou, plus généralement, dans les domaines où l'on étudie des signaux émanant de systèmes complexes, difficilement abordables de manière analytique. Nous considérerons ici les automates cellulaires unidimensionnels de règle 110 et 30 ainsi qu'un signal de même forme mais de source aléatoire. Pour chacune de ces règles, nous avons considéré des automates de 2000 cellules qui, partant de conditions initiales aléatoires, ont évolué sur 10000 pas temporels. Des suites ayant évolué sur 20000 pas ont aussi été considérées pour comparaison. A chaque pas, nous avons calculé la T-Entropie et c'est le signal de l'évolution de la T-entropie que nous traitons ici par les méthodes de Hurst et méthode DFA.

C.2 Méthode de Hurst

C.2.1 Méthode

Dans les années 50' le physicien et hydrologue britannique H.E. Hurst se vit confier la tâche de calculer la capacité optimale du barrage d'Assouan en Egypte. Dans le cadre de cette étude il a été amené à mettre au point de nouvelles techniques statistiques qui lui ont permis de mettre en évidence des corrélations à long terme dans les reflux annuels du Nil. Nous présentons ici le détail de la méthode.

Soit X_t l'affluence d'eau dans le barrage pour l'année t ($t = 1, \dots, N$), et $\overline{X_N}$ sa moyenne sur les N années. Si on libère du barrage une quantité $\overline{X_N}$ chaque année, il restera dans le barrage, à l'instant t , une quantité :

$$X(t, N) = \sum_{i=1}^t (X_i - \overline{X_N}) = \left(\sum_{i=1}^t X_i \right) - t\overline{X_N} \quad (\text{C.1})$$

Afin d'éviter l'assèchement ou le débordement, la capacité du réservoir doit être supérieure à la différence $R(N)$ entre les maximum et le minimum des quantités d'eau possible, soit :

$$R(N) = \max_{1 \leq t \leq N} X(t, N) - \min_{1 \leq t \leq N} X(t, N) \quad (\text{C.2})$$

$R(N)$ est appelé rang de $X(t, N)$.

Cette grandeur dépend du nombre N d'années sur lesquelles on l'évalue. Afin de découpler cette valeur de l'influence du nombre d'années prises en compte, Hurst imagina de considérer le "Scaling Rank" :

$$R^*(N) = \frac{R(N)}{\sigma(N)} \quad (\text{C.3})$$

Soit le rapport du rang sur la déviation standard $\sigma(N)$, dépendante elle aussi de N .

$R^*(N)$ dépend toujours de N mais d'une manière bien plus intéressante, en effet Hurst constata que cette valeur suivait une loi de puissance du type $R^*(N) \propto N^H$ où H a été défini comme étant l'exposant de Hurst, compris entre 0 et 1.

Cet exposant permet de caractériser le mécanisme sous-jacent au signal analysé de la manière suivante :

- $0 < H < 1/2$: la tendance locale sur l'intervalle observé s'inverse. Comme pour un système avec "feed-back" négatif. C'est le cas de l'antipersistance. La meilleure approximation correspond à la valeur moyenne sur cet intervalle.
- $H = 1/2$: Les étapes succesives sont indépendantes. C'est le cas du mouvement brownien. La meilleure prédiction est la dernière valeur mesurée.
- $1/2 < H < 1$: la tendance locale sur l'intervalle observé va continuer. C'est le cas de la persistance. La meilleure approximation correspond à une extrapolation de la tendance.

Cette méthode, appelée aussi méthode R/S, peut être appliquée à n'importe quel signal temporel. En comparant les limites maximales et minimales de l'évolution du signal dans des petites boîtes temporelles avec l'écart-type local, elle fournit une mesure quantitative du caractère persistant de l'évolution du signal considéré. La méthode de Hurst s'applique sur l'entièreté du signal et donne un indice sur le comportement général. De manière complémentaire la méthode DFA, présentée dans la suite, permet de donner une appréciation du type de corrélation selon la portée temporelle d'interaction considérée.

C.2.2 Résultats obtenus et commentaires

Pour chacun des cas, l'analyse a été faite sur 10 séries de données, différant seulement par leurs conditions initiales.

Règle 30

La règle 30 a la particularité d'exhiber un comportement d'apparence chaotique, même en partant de conditions initiales élémentaire.

Pour cette règle, la méthode de Hurst ne donne pas de très bons résultats. Le comportement d'échelle du rang échellonné s'écarte assez bien d'une loi de puissance et semble montrer plusieurs zones différentes (cf exemple Fig.C.1). Or on ne peut tirer d'information, ici, de ces transitions car, comme nous l'avons dit, la méthode de Hurst doit être interprétée sur l'ensemble du signal. On constate seulement qu'en ajustant des tendances linéaires sur les diagrammes log-log, les meilleures exposants trouvés sont situés autour de la valeur 0,5 (cf Fig. C.2), mais les comportements sont trop éloignés de la tendance en loi de puissance pour que l'on puisse s'en servir pour caractériser

ici le signal : la dispersion autour de cette valeur est grande et l'intervalle de confiance accordé à chaque exposant est de l'ordre de 0,05. A défaut, nous avons comparé ce résultat avec celui obtenu pour une série de suites prises aléatoirement, afin de voir s'il nous serait donné de différencier les deux phénomènes.

Nous avons donc généré 10000 suites de 2000 bits de manière complètement aléatoire. Comme nous l'avons observé lors d'une précédente étude, la distribution obtenue des valeurs de la T-entropie est la même (même forme, même moyenne, même écart-type) que pour les séries obtenues grâce à la règle 30. Lorsqu'on applique la méthode de Hurst à cette nouvelle série aléatoire (cf exemple Fig. C.3), on obtient des résultats similaires, présentant aussi des variations par rapport à loi de puissance, mais cette fois les perturbations sont légèrement plus petites.

Règle 110

Pour le cas de la règle 110, la méthode semble être assez efficace. On retrouve effectivement des lois de puissances, avec des exposants estimés en moyenne à $H = 0.874 \pm 0.001$ (cf. Fig. C.5). Cette valeur a été obtenue en ajustant des fonctions $y = Ax^H$ sur les comportements des rangs échellonnés en fonction de l'échelle considérée (voir méthode). Un exemple est donné sur la figure C.4, l'erreur sur l'exposant étant généralement de l'ordre du centième. Ce résultat met en évidence une persistance forte dans le signal : les variations du signal semblent suivre des tendances. Le système a mémoire de son histoire et a tendance à suivre l'évolution générale.

Bien sûr, on pouvait s'attendre à ce résultat, en constatant que le choix du type de C.I. implique dans le cas 110 une grande variation, donc une tendance générale fortement marquée, là où le cas de la règle 30 n'évolue pas beaucoup. Néanmoins l'étude DFA va permettre de caractériser la dépendance de ces caractères persistants en fonction de la taille des boîtes considérées, et donc de la portée temporelle des interactions.

C.3 Méthode DFA

La méthode DFA (Detrended Fluctuations Analysis) est utilisée pour caractériser une série temporelle. Elle s'effectue comme suit : On commence par transformer le signal $u(t)$ en son homologue centré et intégré :

$$y(t) = \sum_{t'=1}^t (u(t') - \langle u \rangle) \quad (\text{C.4})$$

Le signal $y(t)$ est découpé en petites boîtes temporelles de longueur τ . Ensuite on réalise, sur chacune de ces petites boîtes, la tendance linéaire locale $z(t) = at + b$, et on calcule les fluctuations par rapport à cette tendance.

$$F^2(\tau) = \frac{1}{\tau} \sum_{t=k\tau+1}^{(k+1)\tau} [y(t) - z(t)]^2 \quad (\text{C.5})$$

En prenant la moyenne de ces fluctuations pour chacune des boîtes on a un nombre qui dépend de la longueur τ des boîtes considérées, comme

$$\sqrt{\langle F^2(\tau) \rangle} \propto \tau^{\alpha_H} \quad (\text{C.6})$$

où α_H est un exposant semblable à celui de Hurst et s'interprétant de la même manière.

On peut effectuer cette méthode à un ordre supérieur, en ajustant des polynômes de degré n à la place de droites, on parle alors d'analyse DFA- n . Nous n'avons effectué ici que des analyses DFA-1.

C.3.1 Résultats obtenus et commentaires

Règle 30

On voit directement que la méthode DFA est beaucoup plus efficace que la méthode de Hurst. Sur la figure C.6, on voit bien le comportement en loi de puissance. La valeur moyenne des exposants est de $\alpha_H = 0.511 \pm 0.002$ et les valeurs obtenues sont cette fois toutes très proches de $H = 1/2$. Le comportement est complètement analogue, bien que très légèrement plus précis pour les cas ayant évolué sur 20000 pas. Ce résultat confirme le fait déjà énoncé, que le comportement de la règle 30 est très difficile à distinguer d'un processus aléatoire et que la nature déterministe du système sous-jacent est difficile à mettre en évidence.

Effectivement, lorsqu'on analyse la série obtenue par tirage aléatoire, on obtient des résultats tout à fait similaires, avec $\alpha_H = 0.519 \pm 0.003$ (cf exemple Fig. C.7). Les différences supposées lors de l'analyse de Hurst ne trouvent pas ici leur confirmation.

Règle 110

L'étude de la règle 110 se révèle assez intéressante, en ceci qu'elle montre une transition entre deux types d'interactions (cf exemple Fig. C.8). Tout d'abord, à courte portée (jusqu'à $\Delta t \sim 30$), se dégage le premier type d'interaction. Avec un exposant de l'ordre de $\alpha_H \sim 0,8$, il correspond au résultat obtenu lors de l'analyse de Hurst. La valeur de l'exposant précise qu'il s'agit d'un bruit fortement corrélé, identifiable à un mouvement brownien fractionnaire. L'autre partie de la pente ($\Delta t \geq 50$) présente un exposant $\alpha_H = 1.65$ typique des lois de puissances [27]. On a donc que l'exposant $\alpha_H \sim 0,8$ observé lors de la méthode de Hurst est découplé de la tendance générale et montre bien le comportement du bruit superposé à la tendance générale.

C.4 Conclusion

Les analyses effectuées lors de ce travail montrent que le comportement de la règle 30 est très difficile à distinguer d'un processus entièrement aléatoire. du moins pour ce qui est de l'évolution de la valeur globale que représente l'entropie. Plus précisément, la méthode utilisée ici ne permet pas de distinguer les deux. Ceci montre comment des lois simples et déterministes peuvent être présentes derrière tous les phénomènes d'apparence chaotique que nous voyons dans la nature. Nous montrons ensuite que la règle 110 exhibe, outre une tendance générale due au choix spécifique des C.I., un bruit du type mouvement brownien fractionnaire fortement corrélé qui est propre à la règle 110 et n'est pas qu'une conséquence de l'algorithme utilisé pour calculer la complexité.

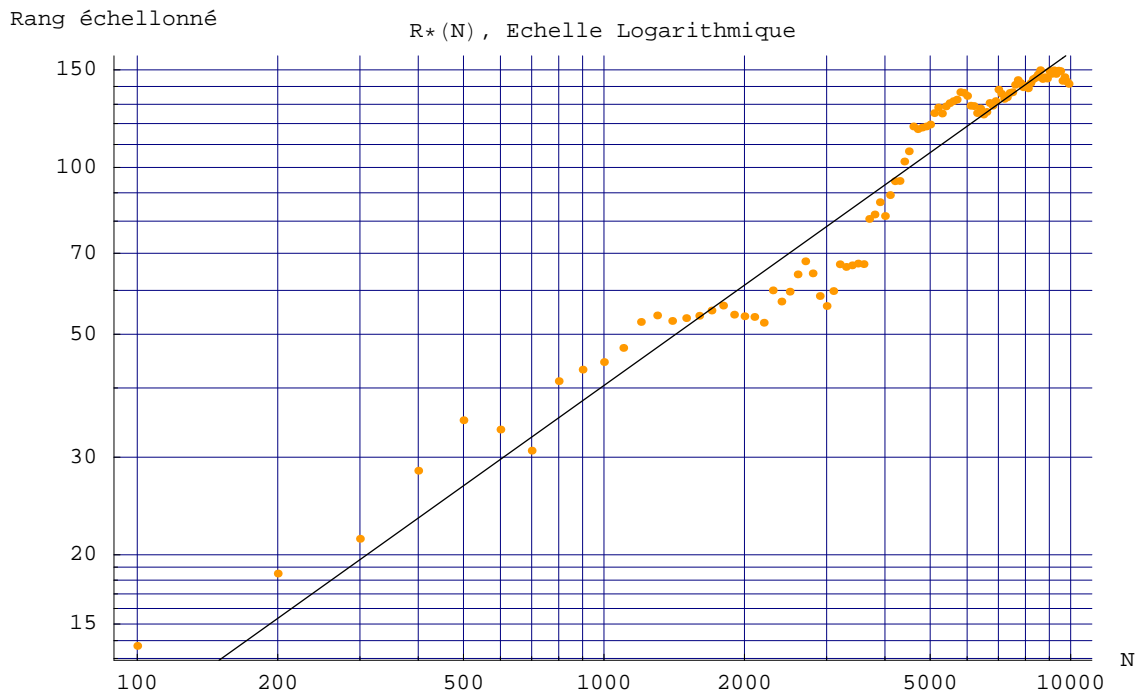


Fig. C.1 – Méthode de Hurst appliquée au signal de l'évolution la T-entropie d'une suite initialement aléatoire sous l'action de la règle 30. On voit que le comportement s'écarte fort de la loi de puissance attendue. Néanmoins la valeur des exposants approchant au mieux les comportements observés se situent autour de la valeur $H = 1/2$.

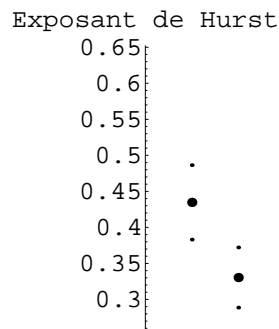


Fig. C.2 – Les exposants obtenus par la méthode de Hurst pour différentes séries obtenues par la règle 30, et leur intervalles de confiance.

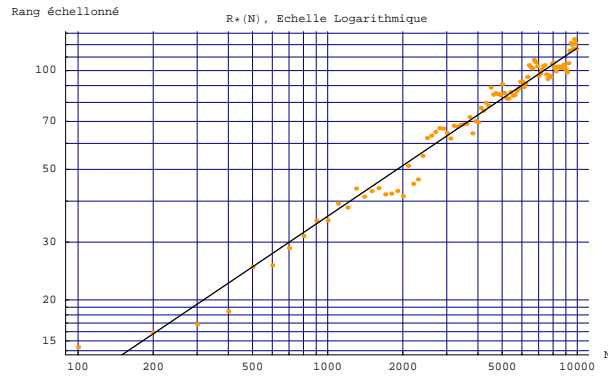


Fig. C.3 – Méthode de Hurst appliquée au signal des variations de la T-entropie pour une série de suite aléatoire. On voit que le comportement présente toujours des variations par rapport à la loi de puissance attendue mais nettement plus petite que dans la cas de la règle 30. L'exposant observé ici est de $0,51 \pm 0.01$.

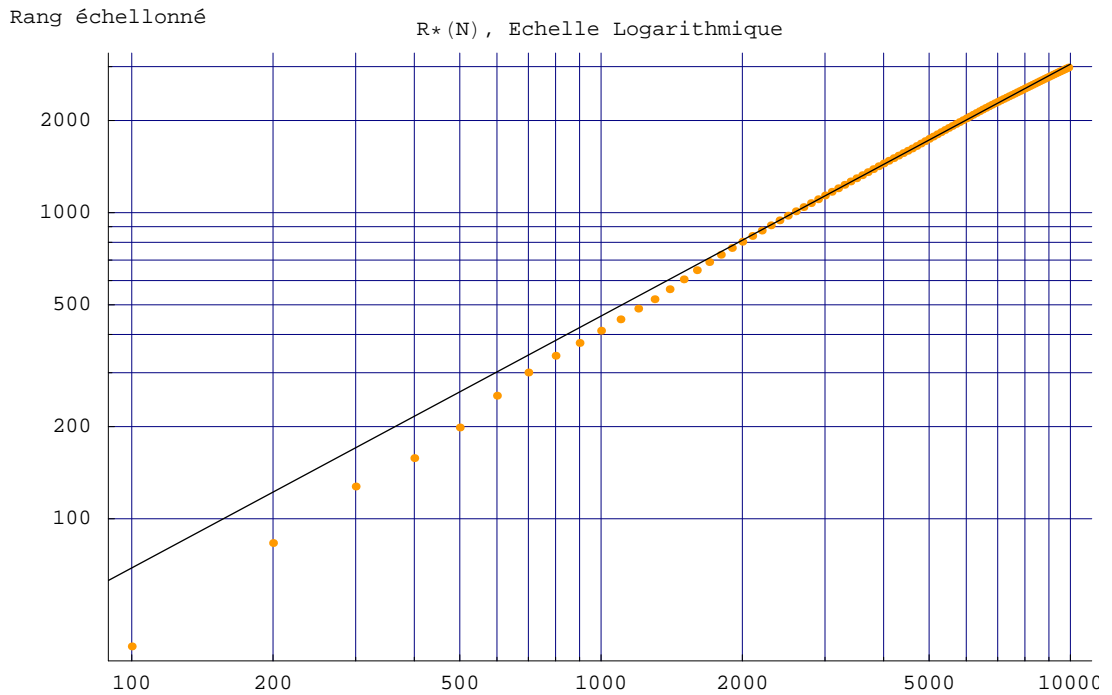


Fig. C.4 – Méthode de Hurst appliquée au signal de l'évolution la T-entropie d'une suite initialement aléatoire sous l'action de la règle 110. On voit que le comportement correspond bien à une loi de puissance. $\alpha = 1/2$.

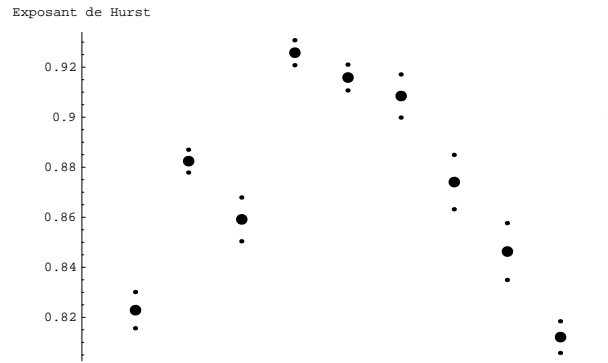


Fig. C.5 – Les exposants obtenus par la méthode de Hurst pour différentes séries obtenues par la règle 110, et leur intervalles de confiance.

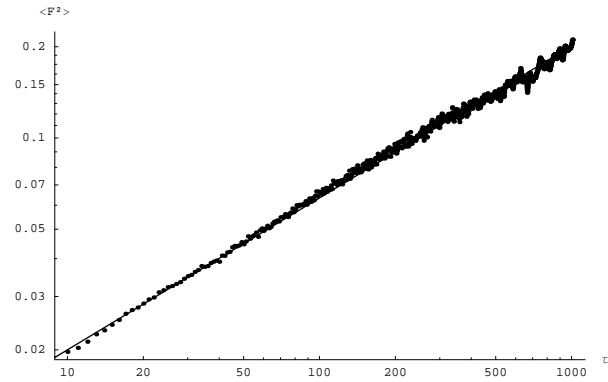


Fig. C.6 – Méthode DFA appliquée à l'évolution de la T-entropie d'un automate de règle 30. La pente est en moyenne de $\alpha_H = 0.511 \pm 0.002$.

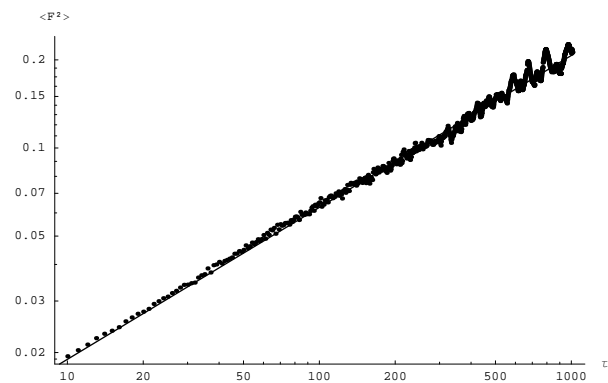


Fig. C.7 – Méthode DFA appliquée aux suites obtenues par tirages aléatoires. La pente est en moyenne de $\alpha_H = 0.519 \pm 0.003$.

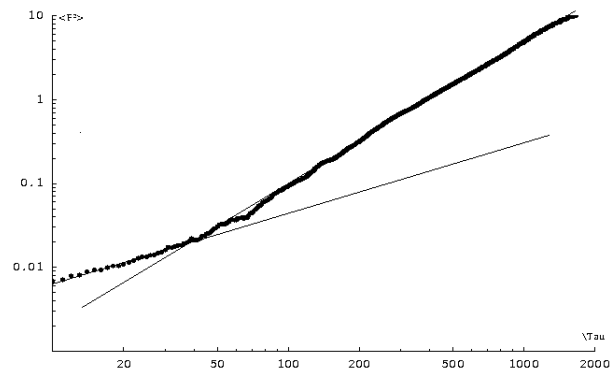


Fig. C.8 – Méthode DFA appliquée à l'évolution de la T-entropie d'un automate de règle 110. Les exposants sont respectivement de $\alpha_H \sim 0,87$ pour la première partie et de $\alpha_H \sim 1.65$ pour la deuxième partie.

Bibliographie

- [1] Massimo Falcioni, Vittorio Loreto, Angelo Vulpiani. "Entropie, Chaos et Complexité", in Roberto Livi, Angelo Vulpiani, *L'Héritage de Kolmogoroff en physique*, Belin, Echelles, 2003.
- [2] Shannon C. E., "A Mathematical Theory of Communication", *The Bell System Technical Journal*, **27**, p.379–423, 1948.
- [3] Kolmogoroff A. N., "Three Approaches to the Concept of the Amount of Information", *Prob. Info. trans.*, **1**, p.102, 1965.
- [4] MacKay D. J. C., "Information Theory, Inférence and Learning algorithm", Cambridge University Press, 2003.
- [5] Ebeling, W., Steuer, R., Titchener, M. R., "Partition-based Entropies of Deterministic and Stochastic maps", *Stochastics and Dynamics*, **1**(1), p.45-61, 2001.
- [6] Titchener, M. R., Gulliver, A., Nicolescu, R., Speidel, U., Staiger, L., "Deterministic Complexity and Entropy", *Fundamenta Informaticae*, **64**, p.443-461, 2005.
- [7] Titchener, "A measure of Information", Data Compression Conference 2000, p.353-362, 2000.
- [8] Hautot A., "Thermodynamique buissonnière", working paper, 2001.
- [9] Beck C., Schlögl F., "Thermodynamics of Chaotic Systems, an Introduction", *Cambridge Non Linear Science Series, 4*, Cambridge University Press, 1993.
- [10] Herzel H., Schmitt A. O., and Ebeling W., "Finite Sample Effects in Sequence Analysis", *Chaos, Solitons, Fractals*, **4**, 1994.
- [11] Delahaye J.P. "L'intelligence et le calcul", Belin-Pour La Science, 2002.
- [12] Wolfram S., "A New Kind of Science", Wolfram Media, 2002.
- [13] H. Gutowitz and C. Langton., "Mean Field Theory and the Edge of Chaos", In *Proc. of 3rd Europ. Conf on Art. Life*, 1995.
- [14] Cherry E. C., "Discussion on dr. Shannon's Papers", *IEEE Transactions on Information Theory*, vol. **1**, no.1, p.169, February 1953.
- [15] Lempel A. and Ziv J., "On the Complexity of Finite Sequences", *IEEE Transactions on Information Theory*, vol. **22**, p.75-81, 1976.
- [16] Wolfram S., "Computation Theory of Cellular Automata", *Comm. in Math. Phys.*, **96**, p.15-87, 1984.
- [17] Gardner M., "The Fantastic Combinations of John Conway's New Solitaire Game « life »", *Scientific American*, **223**, p.120-123, Octobre 1970.
- [18] J.-C. Dubacq, B. Durand and E. Formenti, "Kolmogorov Complexity and Cellular Automata Classification", *Theoretical Computer Science*, **259**, no. 1-2, p.271-285, 2001.

- [19] Grassberger P., "Long-Range Effects in an Elementary Cellular Automata", *Journal of Statistical Physics*, **45**, no. 1-2, p.27-39, 1986.
- [20] Correspondances privées avec M. R. Titchener, Senior Lecturer à l'Université de Auckland, New Zealand.
- [21] Wuensche, A., and M.J.Lesser, "The Global Dynamics of Cellular Automata ; An Atlas of Basin of Attraction Fields of One-Dimensional Cellular Automata", Santa Fe Institute Studies in the Sciences of Complexity, 1992.
- [22] Wuensche, A., "Complexity in One-D Cellular Automata : Gliders, Basins of Attraction and the Z parameter", Santa Fe Institute working paper, 1994.
- [23] C. Domain and H. Gutowitz, "The Topological Skeleton of Cellular Automaton Dynamics ", *Physica D*, **103**, nos 1-4, p.155-168, 1997.
- [24] Melanie Mitchell, Peter T. Hraber and James P. Crutchfield, "Revisiting the Edge of Chaos : Evolving Cellular Automata to Perform Computations", Santa Fe Institute Working Paper 93-03-014, 1993.
- [25] Howard A. Gutowitz, "A Hierarchical Classification of Cellular Automata", *Physica D*, **45**, 1989.
- [26] Wuensche A., "Classifying Cellular Automata Automatically", Santa Fe Institute working paper, 1998.
- [27] Kun Hu, Plamen Ch. Ivanov, Zhi Chen, Pedro Carpena, and H. Eugene Stanley, "Effect of trends on detrended fluctuation analysis", *Phys. Rev. E*, **64**, 2001.
- [28] Dubacq J.C., "How to Simulate Turing Machines by Invertible One-Dimensional Cellular Automata", *International Journal of Foundations of Computer Science*, **6**, no. 4, p.395-402, 1995.
- [29] N.H. Packard, "Adaptation towards the edges of chaos", in J.A.S Kelso, A.J. Mandell, and M.F Shlesinger, éditeurs, *Dynamics patterns in complex systems*, p.299-301, Singapore, World Scientific, 1988.
- [30] Chua L., "What's new in Wolfram's New Kind of Science", Symposium of the International Francqui Chair, Leuven, july 2006.
- [31] Chaitin G.J., "Speculation on Biology, Information and Complexity", Centre for Discrete Mathematics and Theoretical Computer Science Research Report Series, CDMTCS-282, July 2006.
- [32] Wikipédia : <http://en.wikipedia.org/>
- [33] MathWorld : <http://mathworld.wolfram.com/>
- [34] Andy Wuensche's homepage : http://www.cogs.susx.ac.uk/users/andywu/gallery/ddlab_gallery.html
- [35] CiteSeer : <http://citeseer.ist.psu.edu/>
- [36] Site de l'université de Santa Fe : <http://www.santafe.edu/>
- [37] Site du "Centre for Discrete Mathematics and Theoretical Computer Science" de l'université de Auckland : <http://www.cs.auckland.ac.nz/CDMTCS/>